

3

Splines

This chapter provides coverage of spline smoothers. Spline smoothers are another nonparametric regression technique used with scatterplots. One might question whether another smoothing technique is necessary given local polynomial nonparametric regression (LPR) models. In truth, for basic scatterplot smoothing, often one cannot tell much difference between spline and LPR fits. Splines, however, have several advantages over local polynomial regression. First, splines have an analytic foundation that is superior to that of local regression, as one can prove that a spline smoother will provide the best mean squared error fit. Second, one type of spline, the smoothing spline, is designed to prevent overfitting, a prominent concern with nonparametric smoothers. Third, there have been a number of advances in the methods used to estimate splines, while advances in local regression has been fairly static. As a result, the software for fitting spline models is typically superior to that for local regression. For example, most implementations of spline produce confidence bands, which may not be true for LPR smoothers. Moreover, other differences in the estimation algorithms will be obvious in the next chapter on automated smoothing. Finally, splines are easier to incorporate into semiparametric estimation, and they have become the smoothing method of choice for semiparametric regression models.

3.1 Simple Regression Splines

The term ‘spline’ originally referred to a tool used by draftsmen to draw curves. For our purposes, splines are piecewise regression functions we constrain to join at points called knots. In their simplest form, splines are regression models with a set of dummy variables on the right hand side of the model that we use to force the regression line to change direction at some point along the range of X . For the simplest regression splines, the piecewise functions are linear, a constraint that we will later relax. In essence, we fit separate regression lines within the regions between the knots, and the knots tie together the piecewise regression fits. Again, splines are a local model with local fits between the knots instead of within bins that allow us to estimate the functional form from the data.

Like local polynomial regression, the analyst must make several modeling decisions with splines. For LPR, we had to choose the degree of the polynomial, the span, and perhaps the weighting function. With splines, one must choose the degree of a polynomial for the piecewise regression functions, the number of knots, and the location of the knots. In Chapter 2, the choice of the span parameter, the percentage of data used in each local fit, proved to be critical since it affected the smoothness of the nonparametric estimate, while the other choices were less important. With splines, we again find that while the fit is invariant to some of the modeling choices, the analyst must focus on how smooth the fit should be. For some types of splines, the number of knots will control the amount of smoothing, while for other types of splines, a smoothing parameter controls the smoothing.

Perhaps the most confusing aspect of splines is that there are so many different types. For example, there are regression splines, cubic splines, B-splines, P-splines, natural splines, thin-plate splines, and smoothing splines to name a few. Moreover, there are often combinations such as natural cubic B-splines. The wide variety of splines partially stems from the progress in research on splines. Often a new type of spline either supplants an older type of spline or adds a refinement to existing methods. The splines most often used in statistics, smoothing splines, are more complex than regression splines, but they work on the same principle. Thus, in this next section, we focus on understanding the simplest type of splines before moving to more complicated types.

We start with the basic equation for a smooth fit between x and y :

$$y = f(x) + \varepsilon. \tag{3.1}$$

As before, we only assume that f is some smooth function, but we would like to estimate this model using a single regression model estimated with ordinary least squares as opposed to using a series of local models. To do this, we need to represent the model matrix so that we can estimate Equation (3.1)

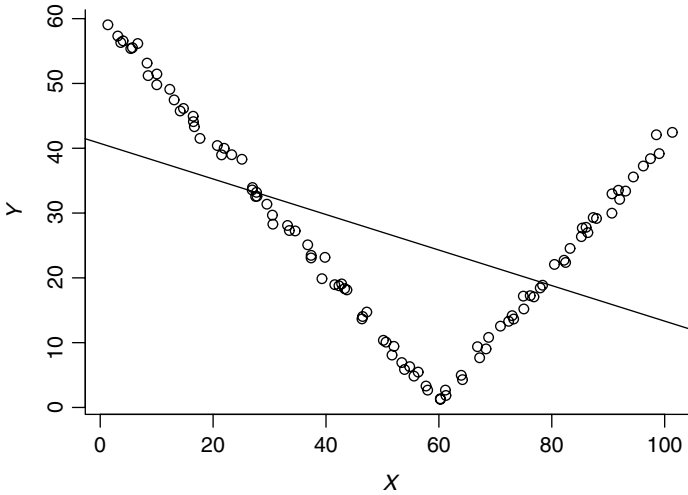


Figure 3.1 A simulated nonlinear functional form with OLS fit.

with least squares. This process is best demonstrated with a simple illustration. Figure 3.1 contains a scatterplot of the nonlinear relationship between two simulated x and y variables. The form of the nonlinear dependency between x and y is quite obvious. The solid line in Figure 3.1 represents the estimate from a global linear fit between the two variables. The global estimate clearly fails to capture the nonlinear dependency between x and y , as it only indicates that there is a fairly strong negative relationship between the two variables.

The logic behind a regression spline is to estimate two separate regression lines that will be joined at the kink in the data. The first regression line will approximate the negative dependency between the two variables, and the second regression line will approximate the upturn in the functional form. In this instance, we can clearly identify the change point in the relationship between x and y , and therefore, we can easily use two piecewise linear estimates joined at the change point. To estimate the spline model, we need to specify the point where the two separate OLS fits will be joined. Placement of the joinpoint, or knot, between the two regression lines here is easy since the kink is so obvious. We only need a single knot since there are just two piecewise linear fits to conjoin. Additional piecewise fits would require additional knots. We denote the single knot with c_1 . Using c_1 , we can write the following regression spline model:

$$y = \alpha + \beta_1 x + \beta_2(x)_+ + \varepsilon \quad (3.2)$$

where

$$(x)_+ = \begin{cases} x & \text{if } x > c_1 \\ 0 & \text{if } x \leq c_1. \end{cases} \quad (3.3)$$

The $(\cdot)_+$ function indicates that $(x)_+$ equals x if x is greater than the knot value and is equal to 0 otherwise. If we assume that the function is piecewise linear, we can rewrite Equation (3.2) as two separate but conjoined regression lines:

$$y = \begin{cases} \alpha + \beta_1 x & \text{if } x \leq c_1 \\ \alpha + \beta_1 x + \beta_2(x - c_1) & \text{if } x > c_1. \end{cases} \quad (3.4)$$

We have now rewritten a single linear model as two separate linear fits that describe each part of the nonlinear dependency between x and y . To jointly estimate the two piecewise regression fits requires a set of basis functions. Basis functions are an important concept in the estimation of splines, and we spend some time explaining basis functions for splines before estimating the current example.

3.1.1 Basis Functions

In linear algebra, the basis of a vector space (a set of vectors) is the number of columns or rows of that vector space that can be expressed as a linear combination. In the context of regression models, \mathbf{X} , the model matrix, is a vector space with a corresponding basis function. For example, if we have a regression model with a constant and a single covariate, the corresponding basis functions for this model would be a vector of 1's and the vector x_1 , the lone predictor variable. In fact, the right hand side of any regression model is a linear combination of basis functions and therefore forms a basis. We use additional basis functions to approximate nonlinearity for regression splines. The simplest way to add to the basis is to use additional predictor variables. With splines, we add another basis to the data matrix, but this basis is a transformation of the single predictor. These additional basis function will allow us to approximate the nonlinear relationship between x and y . To add an additional basis requires a set of basis functions, one basis function for each piecewise function. In the current example, we have two piecewise functions to estimate, so we must define two basis functions for the two piecewise linear fits, one for the left side knot and one for the right side of the knot:

$$B_L(X) = \begin{cases} c - x & \text{if } x < c \\ 0 & \text{otherwise} \end{cases} \quad (3.5)$$

$$B_R(X) = \begin{cases} x - c & \text{if } x > c \\ 0 & \text{otherwise.} \end{cases} \quad (3.6)$$

These basis functions are applied to the model matrix to form a new basis. In practical terms, application of the basis functions adds another vector to the data matrix. Before, the model matrix was comprised of a constant and x . The model matrix for the spline model will now consist of a constant and two data vectors. The first data vector has values of c_1 less x until the value of the knot equals the x values and then is zero for the rest of the values of x . The second data vector in the model matrix consists of zeros until we get to the value of x where we have placed a knot, and then the vector will take values of x less the knot value. Applying the basis functions adds an additional basis, before the model had a basis of dimension 2 and now will have a basis of dimension 3. In essence, by applying the basis functions, we have added an additional regressor to the model matrix. For the current example, to apply the basis functions, we must choose where to place the knot. We place a single knot at 60 since in the simulated data this is the exact x value where the kink occurs. This is the point in x where we conjoin the two piecewise linear fits. With the knot value chosen, we apply the basis functions to the simulated data. Practical implementation of the basis functions consists of writing two functions in a programming environment, applying those functions to the simulated x variable and then constructing a new model matrix comprised of a constant and the results from the application of each basis function. Below is an example of the model matrix, \mathbf{X} , constructed after we applied the two basis functions to the simulated x variable.

$$\mathbf{X} = \begin{bmatrix} 1 & (60 - x_1) & 0 \\ \vdots & \vdots & \vdots \\ 1 & (60 - x_{59}) & 0 \\ 1 & 0 & 0 \\ 1 & 0 & (x_{61} - 60) \\ \vdots & \vdots & \vdots \\ 1 & 0 & (x_n - 60) \end{bmatrix}. \quad (3.7)$$

Once we have formed a new model matrix, estimating a spline fit between x and y is simple. We use the new model matrix to construct the usual ‘hat’ matrix: $(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. Application of the hat matrix to the data vector for the outcome produces a set of predictions that form the nonparametric spline estimate of the relationship between x and y . Therefore, the spline estimate is

$$\hat{\mathbf{y}} = \mathbf{H}\mathbf{y} \quad (3.8)$$

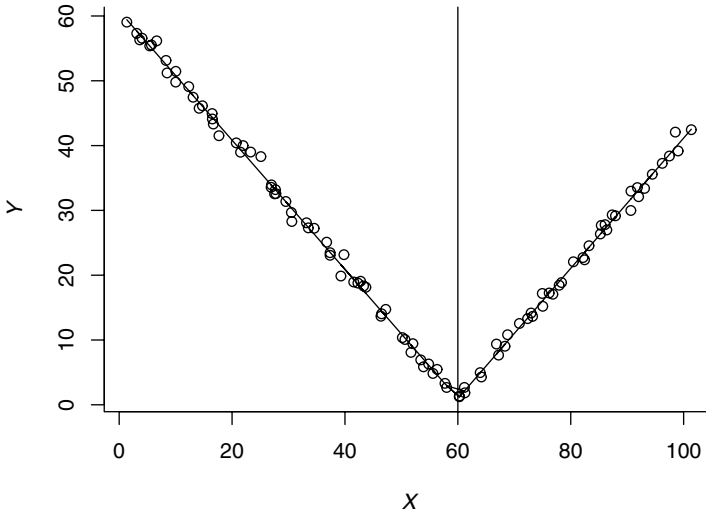


Figure 3.2 A piecewise linear regression spline fit.

where $\mathbf{H} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. We then plot the predictions from the model to view the spline estimate. Figure 3.2 contains a plot of the resulting estimated spline fit between x and y , which closely approximates the nonlinear relationship between x and y .

This simple example nicely illustrates the basic procedure for using splines. It allows the reader to see that the regression spline estimate results from simply applying least squares to a model matrix altered with the basis functions. We can use a combination of additional knots and more complex basis functions to approximate more complex nonlinear relationships. Clearly much depends on knot placement, a topic we will take up shortly, but since splines models are essentially transformed regression models, it allows for an easy mixing of nonparametric estimation with more standard statistical models. One drawback to the simple regression splines we used in the current example is the restrictive assumption of piecewise linear functions, but we can easily relax this assumption.

3.2 Other Spline Models and Bases

Statisticians have developed a wide variety of basis functions for spline models. In general, a change of basis will not change the fit between x and y very much. Typically, different spline bases are used for better numerical stability and ease of implementation (Ruppert, Wand, and Carroll 2003). In this section, we review some of the more commonly used bases for spline models as well as a variety of improvements that have been proposed for splines.

3.2.1 Quadratic and Cubic Spline Bases

The simple regression splines we used in the last section to estimate the nonlinear dependence between the simulated x and y are not suitable for most applied smoothing problems. It is overly restrictive to only estimate piecewise functions that are linear between the knots since we wish to estimate more curvilinear functional forms. The solution is to combine piecewise regression functions with polynomial regression by representing each piecewise regression function as a piecewise polynomial regression function. Piecewise polynomials offer two advantages. First, piecewise polynomials allow for nonlinearity between the knots. Second, piecewise polynomial regression functions ensure that the first derivatives are defined at the knots, which guarantees that the spline estimate will not have sharp corners.¹

Altering the regression splines used thus far to accommodate piecewise polynomials is simple. For the spline model in the last section, we could estimate piecewise polynomial fits by adding x^2 to the basis and squaring the results from the basis functions. These alterations form a quadratic spline basis with a single knot at c_1 . Typically, cubic spline bases are used instead of quadratic bases to allow for more flexibility in fitting peaks and valleys in the data. A spline model with a cubic basis and two knots c_1 and c_2 is formed from the following linear regression model:

$$Y = \alpha + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 (x - c_1)_+^3 + \beta_5 (x - c_2)_+^3 + \varepsilon. \quad (3.9)$$

The spline estimate is again the predictions from the hat matrix applied to the outcome variable. To form the hat matrix, we must first construct a model matrix that contains the correct bases. For this example, the model will contain the following constructed data vectors:

$$\begin{aligned} x_1 &= x \\ x_2 &= x^2 \\ x_3 &= x^3 \\ x_4 &= (x - c_1)_+^3 \\ x_5 &= (x - c_2)_+^3 \end{aligned} \quad (3.10)$$

¹Recall that for a continuous function to be differentiable at a particular point, the function must not change dramatically at that point. Sharp corners in functions are places where the first derivative is not defined. For example, for the function $y = |x|$ the derivative does not exist at $x = 0$. For polynomial functions, such sharp corners, where the first derivative is undefined, do not exist.

where x represents the original predictor variable. The model matrix will consist of a constant and the above five variables. We use this model matrix to form a hat matrix that is applied to the outcome variable, and the predictions from this model serve as the spline estimate of the possibly nonlinear relationship between x and y . The number of parameters used to construct the spline estimate is controlled by the number of knots. If there are k knots, with a cubic basis, the function will require $k + 4$ regression coefficients (including the intercept). The cubic basis allows for flexible fits to nonlinearity between the knots and eliminates any sharp corners in the resulting estimate. The latter is true since the first derivative exists for $(x - c_1)_+^3$ and it follows that the first derivative will also exist for any linear combination of the terms in 3.10 (Ruppert, Wand, and Carroll 2003). For cubic regression splines, there are a number of equivalent ways to write the basis. For example, below we outline another way to write the cubic basis that is very convenient for estimation but looks rather daunting (Gu 2002). First, let the knot locations in x be denoted by x^* . We will have to select these knot locations, typically they are evenly spaced over the range of x . Now, we define $B(x, x^*)$ as the following function to represent the basis:

$$B(x, x^*) = [(x^* - 1/2)^2 - 1/12][(x - 1/2)^2 - 1/12]/4 \\ - [(|x - x^*| - 1/2)^4 - 1/2(|x - x^*| - 1/2)^2 + 7/240]/24. \quad (3.11)$$

Applying the above function to x does most of the work in the construction of a model matrix that allows for f to be estimated with a linear regression model. Application of the above function and appending the result to a constant and the x vector produces a model matrix where the i th row is

$$\mathbf{X}_i = [1, x_i, B(x_i, x_1^*), B(x_i, x_2^*), \dots, B(x_i, x_{q-2}^*)]. \quad (3.12)$$

The form of the model matrix above is identical to past discussions of the basis. The first column of the data matrix is for the constant, the next is the x variable, and the following columns form a set of piecewise cubic functions that are fit between each knot location. We can use a data matrix to form the hat matrix and estimate the spline fit for the relationship between x and y . The number of knots selected by the analyst will determine the dimension for the basis.

We now illustrate the use of cubic regression splines. For comparability, we return to the example from the 1992 House elections to again estimate the relationship between support for the challenger and support for Perot. In Chapter 2, we used local polynomial regression to fit a nonparametric regression model to these two variables. The result revealed a nonlinear dependence between these two variables due to a threshold effect. We now use cubic splines to estimate a nonparametric regression model for these same two variables.

For the analysis, we used a series of user-written functions instead of a pre-programmed function from a statistical software package. These functions are available on the book website and help make the basic operation of splines readily transparent. First, we rescale the Perot support variable to lie in the $[0, 1]$ interval. This rescaling of the variable allows for easier placement of the knots. Next, we select the number of knots; for this model, we use four knots, which implies a rank 6 basis or that there are six columns in the model matrix.² How does the number of knots affect the spline estimate? The number of knots determines the smoothness of the nonparametric estimate. Increasing the number of knots increases the number of piecewise functions to produce a more flexible fit. Using two knots produces a globally linear fit. We selected four knots on the basis of a visual trial and error method. We must also decide where to place the four knots. We place the knots at equal intervals over the range of the x variable starting at an x value of 0.2, which implies that the other three knots are placed at 0.4, 0.6, and 0.8. We now construct the model matrix. The first column in the model matrix is a vector of 1's, the second column is the vector x , the support for Perot variable. Given that there are four knots there will be four additional vectors in the model matrix. To create these data vectors, we apply the function in Equation (3.11) to x within each knot placement. Once the model matrix is completed, the process is simple. We form a hat matrix based on the constructed model matrix and apply it to the challenger's vote share to form the model predictions which make up the spline nonparametric estimate. We plot the spline estimate in Figure 3.3. The resulting spline looks very similar to the results using *lowess* in Chapter 2. We see the same nonlinear functional form where support for the challenger no longer increases once a threshold of support for Perot is reached.

This example demonstrates one advantage of spline smoothers: their simplicity. In this example, we estimated a nonlinear functional form with nothing more complicated than ordinary least squares and a transformed model matrix. While this simplicity matters less for nonparametric estimates, it will be valuable once we consider semiparametric estimation. The next type of spline smoother is not a change in basis but simply a convenient alteration of cubic splines to produce a better fit.

3.2.2 Natural Splines

While cubic splines are widely used, they are often altered slightly to improve the fit. One limitation of cubic splines is that the piecewise functions are only fit

²The number of vectors that span the basis is referred to as the rank.

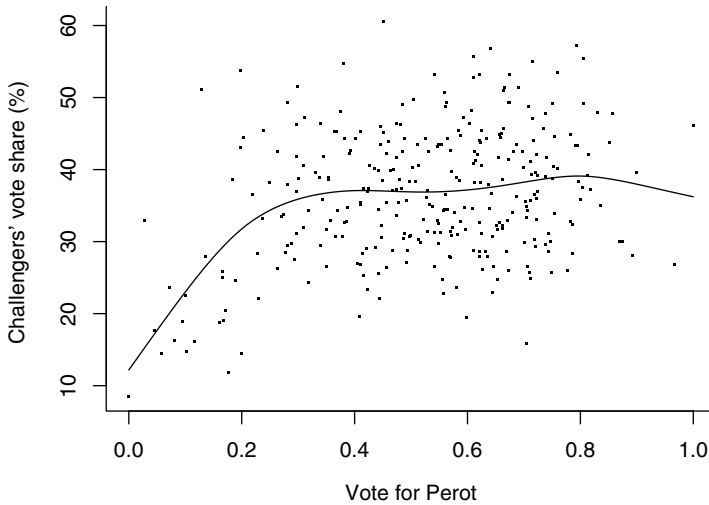


Figure 3.3 Support for the challenger as a function of support for Perot in the 1992 House elections.

between each knot. For data that falls before the first knot and beyond the last knot, we do fit not a piecewise function. Without fits to the boundary of the data, it is possible for the spline fit to behave erratically around the limits of x . Natural cubic splines add two knots to the fit at the minimum and maximum values of x and fit a linear function between the additional knots at the boundary and the interior knots. This constrains the spline fit to be linear before the first knot and after the last knot. Such enforced linearity at the boundaries avoids any wild behavior in the spline fit near the extremes of the data. Cubic splines may not display erratic fits at the boundaries, but natural splines can improve the overall spline fit should problems occur. Since little is lost by using natural splines while some gains in model fit are possible, natural cubic splines are generally preferred to cubic splines. Later in the chapter, we compare natural cubic splines to standard cubic splines.

3.2.3 B-splines

There is one further refinement that is also typically applied to cubic splines. For cubic splines (natural or otherwise), the columns of \mathbf{X} , the model matrix, tend to be highly correlated since each column is a transformed version of x , which can induce considerable collinearity. The collinearity may result in a nearly singular model matrix and imprecision in the spline fit (Ruppert, Wand, and Carroll 2003).

As a remedy, one can represent the cubic spline (and any other polynomial basis) as a B-spline basis. A k -knot cubic B-spline basis can be represented as:

$$f(x) = \sum_{i=1}^k B_i^2(x)\beta_i \quad (3.13)$$

where the B-spline basis functions are defined as:

$$B_i^2(x) = \frac{x - c_i}{c_{i+2+1} - c_i} B_i^{2-1}(x) + \frac{c_{i+2+1} - x}{c_{i+2+1} - c_{i+1}} B_{i+1}^{2-1}(x) \quad i = 1, \dots, k \quad (3.14)$$

and

$$B_i^{-1}(x) = \begin{cases} 1 & \text{if } c_i \leq x < c_{i+1} \\ 0 & \text{otherwise.} \end{cases} \quad (3.15)$$

The B-spline basis function is, in essence, a rescaling of each of the piecewise functions. The idea is similar to rescaling a set of X variables by mean subtraction to reduce collinearity. The rescaling in the B-spline basis reduces the collinearity in the basis functions of the model matrix. The resulting spline model is more numerically stable than the cubic spline. This is especially true if one is using a large number of knots and OLS is used to fit the spline model. See de Boor (1978) and Eilers and Marx (1996) for details. Many spline routines contained in statistical software are cubic B-splines or natural cubic B-splines. Before taking up another example to demonstrate the performance of the spline models outlined thus far, we take up the topic of knots.

3.2.4 Knot Placement and Numbers

In any spline model, the analyst must select the number of knots and decide where they should be placed along the range of x . In the example with simulated data, deciding on the placement of the single knot was trivial given the piecewise linearity of the function. Visual examination of the scatterplot usually, however, does little to aid the placement of knots. For example, the scatterplot between the challenger vote share and support for Perot reveals little about where one might place the knots. The knot placement we used for the spline estimate in Figure 3.3 probably appeared to be arbitrary. Knots tend to generate considerable confusion, so in this section we focus on both how to place knots and how many knots to use.

Knot placement is, however, not a complicated model choice. Stone (1986) found that where the knots are placed matters less than how many knots are used.

Standard practice is to place knots at evenly spaced intervals in the data. Equally spaced intervals ensure that there is enough data with each region of x to get a smooth fit, and most software packages place the knots at either quartiles or quintiles in the data by default. The analyst can usually override default placement of knots. If the data have an obvious feature, it may be useful to place the knots in a less automatic fashion.

But the question of how to select the number of knots remains, and the number of the knots has an important effect on the spline fit. The number of knots chosen affects the amount of smoothing applied to the data by controlling the number of piecewise fits. A spline with two knots will be linear and globally smooth since there is only one piecewise function. Increasing the number of knots increases the number of piecewise functions fit to the data allowing for greater flexibility. More piecewise functions results in an increased number of local fits. If one selects a large enough number of knots the spline model will interpolate between the data points, since more knots shrink the amount of data used for each piecewise function. The number of knots effectively acts as a span parameter for splines. Therefore, one is faced with the same tradeoffs embodied in the choice of the span parameter. If one uses a small number of knots, the spline estimate will be overly smooth with little variability but may be biased. Using a high number of knots, conversely implies little bias but increases variability in the fit and may result in overfitting.

Understanding how the number of knots affects the spline fit does not help in knowing how many to use. Fortunately, the spline fit is usually not overly sensitive to the number of knots selected, and two different methods exist for knot number selection. One method is to use a visual trial and error process as we did for span selection. Four knots is the standard starting point. If the fit appears rough, knots are added. If the fit appears overly nonlinear, knots are subtracted. Four or five knots is sufficient for most applications. The number of knots is also loosely dependent on sample size. For sample sizes above 100, five knots typically provides a good compromise between flexibility and overall fit. For smaller samples, say below 30, three knots is a good starting point.

The second method is less *ad hoc* than the visual method. Since each knot represents additional parameters being added to the model, Eilers and Marx (1996) recommend using Akaike's Information Criterion (AIC) to select the number of knots. One chooses the number of knots that returns the lowest AIC value. Using the AIC is less arbitrary than the visual method and produces reasonable results. As we will see later in the chapter, newer spline models dispense with choices about knots. Now, we return to the Congressional elections example to better understand the performance of the spline models discussed thus far and to explore how the number of knots affects the final estimate.

3.2.5 Comparing Spline Models

As before, we are interested in observing whether there is a nonlinear relationship between the challenger's vote share and support for Perot. Thus far, LPR and simple cubic spline estimates have revealed a nonlinear dependency between these two variables. We now use both cubic B-splines and natural cubic B-splines to estimate the nonparametric fit between these two variables. The software used in this example places the knots at the quantiles based on the distribution of the data by default. There do not appear to be obvious locations for the knots, so we rely on the software defaults. We use four knots for both spline models, and two knots are added at the minimum and maximum values of the Perot support variable for the natural spline model. We will evaluate whether this is the optimal number of knots in a moment. Both spline estimates are in Figure 3.4.

We see little difference between the cubic B-splines and the natural cubic B-splines estimates. In both, the by now familiar threshold effect is present, where the challenger's vote share levels off once support for Perot exceeds about 15% in the district. One might argue that the two spline models do differ slightly at the end points of each fit. The cubic fit curves up more for the higher values of support for Perot, while the natural spline fit displays less curvature since it has been constrained to be linear at the end points. As noted previously, with cubic splines strange behavior is possible at the extremes of x , so in general, natural splines are preferred.

In both of the spline models in Figure 3.4, we used the rule-of-thumb starting point of four knots. A useful exercise is to explore the effect of using different numbers of knots. We could simply add and subtract knots from the four knot baseline and observe whether the fit changes, but instead we calculate the AIC for a range of knot numbers. The lowest AIC value provides a summary measure of the best fit for the fewest number of knots. Using natural cubic B-splines, we

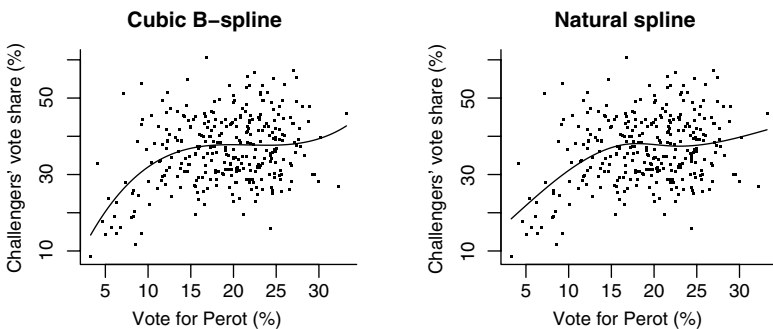


Figure 3.4 Cubic B-Spline and natural spline fit to challenger's vote share.

Table 3.1 AIC values for differing numbers of knots.

2 knots	2239.6
3 knots	2228.0
4 knots	2223.4
5 knots	2220.4
6 knots	2221.0
7 knots	2222.6
8 knots	2224.5
9 knots	2225.8

estimated models with two–nine knots and calculated the AIC for each model. The calculated AIC values are in Table 3.1.

The spline fit with five knots returns the lowest AIC value, which indicates that we would prefer a model with five knots. Using more than two knots always produces a substantially better fit than a model with two knots. This is preliminary evidence that a linear fit provides an inadequate description of the data. Figure 3.5 contains plots for the natural spline fits with four, five, six, and nine knots. While the AIC indicates that we should use five knots, there is little difference across the four estimates. The reader should notice that for the model with nine knots, we observe excess fit, as idiosyncratic variation is evident, which is undoubtedly caused by overfitting. Despite the overfitting, the same basic relationship is observed between the two variables. It is often the case that analysts worry that spline models are overly sensitive to knot selection, but as the example demonstrates, the fit is largely invariant to the number of knots used. While too many knots will cause overfitting and induce small and presumably random variation in the fit, the AIC provides a clear and familiar criterion for selecting the number of knots. We now turn to smoothing splines which are designed to further limit overfitting.

3.3 Splines and Overfitting

If we want a flexible estimate of the statistical relationship between two variables, both splines and local polynomial regression can provide such an estimate with few assumptions about the functional form. A common criticism of both of these nonparametric regression models in the social sciences is that they are too flexible. The concern with both methods is that it is easy to have a surfeit of (local) parameters, which produces overly nonlinear nonparametric estimates that overfit data. Critics argue that while nonparametric regression estimators make few

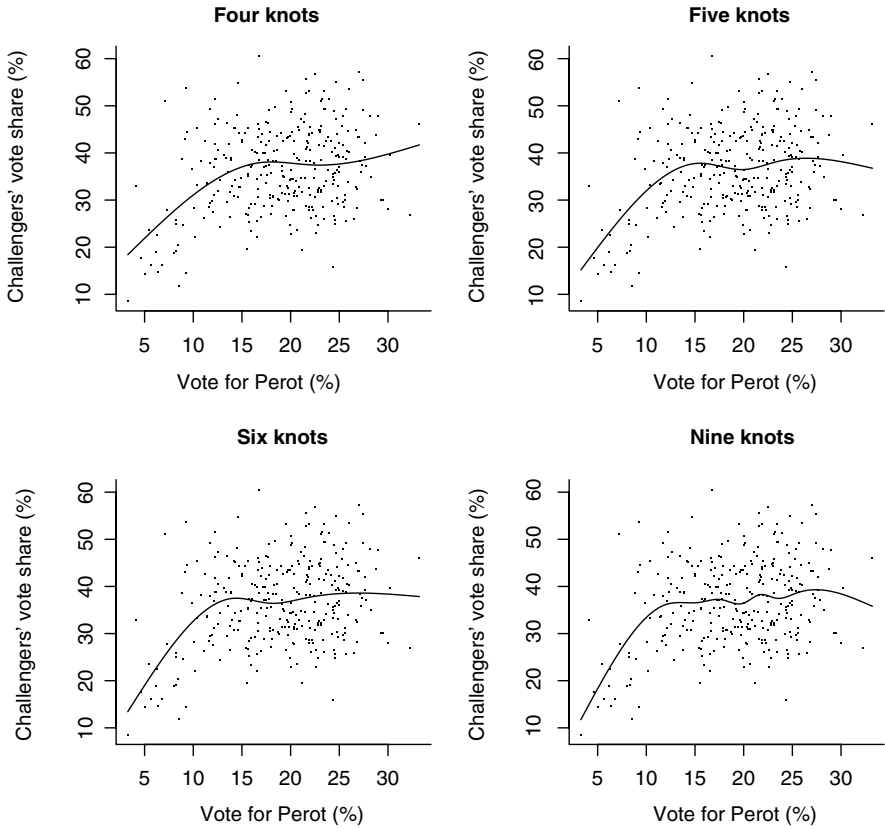


Figure 3.5 Differing numbers of knots with natural splines.

assumptions, they too easily display idiosyncratic local variation between x and y that is not of substantive interest. In short, instead of observing an estimate of the possibly nonlinear functional form, the analyst observes a nonlinear estimate due to overfitting.

While overfitting is possible with nonparametric regression, many of these criticisms are overstated. It is often the case that an analyst would have to use a very small span setting or a large number of knots to overfit the data. This has been the case in the Congressional election example as it required a span setting of 0.20, or nine knots to overfit the data. That said, it is possible to overfit relationships using nonparametric regression. The standard remedy for such overfitting has been a suggestion that analysts err on the side of undersmoothing when choosing either the span or the number of knots. While undersmoothing provides a simple

solution to the problem of overfitting, it is not a solution that appeals to any principles of statistical theory. Moreover, this is a solution that suggests erring on the side of increasing the bias in the fit. We would prefer a less *ad hoc* solution grounded in statistical theory. Penalized splines are a nonparametric regression technique that relies on principles of statistical theory to minimize the possibility of overfitting.

3.3.1 Smoothing Splines

Of course, it is possible to overfit both parametric and nonparametric regression models. Overfit statistical models have too many parameters relative to the amount of data and cause random variation in the data to appear as a systematic effects. The solution to overfitting is to reduce the number of parameters in the model, but the parameters cannot be dropped in an *ad hoc* fashion. The solution is to reduce the parameters subject to some criterion. The AIC is one criterion for reducing parameters, and as we saw in the last section can be applied to splines.

Another solution to overfitting is penalized estimation. Here, for each parameter used in the model, a penalty is added to the model. Penalized estimation is not widely used in the social sciences but is relatively common in statistics. Social scientists are, however, very familiar with one statistic that relies on the principles of penalized estimation: the adjusted R^2 . The adjusted R^2 measures the fit of a model but does so subject to a penalty for each additional parameter used in the model. Smoothing splines operate in a similar fashion by placing a penalty on the number of local parameters used to estimate the nonparametric fit.

Since spline models are estimated with least squares, they share the properties of linear regression models. This implies that the spline estimate, \hat{f} , minimizes the sum of squares between y and the nonparametric estimate, $f(x_i)$

$$SS(f) = \sum_{i=1}^n [y - f(x)]^2. \quad (3.16)$$

The concern, however, is that the estimate of f that minimizes Equation (3.16) may use too many parameters. The penalized estimation solution is to attach a penalty for the number of parameters used to estimate f . This suggests that we minimize $SS(f)$ but subject to a constraint or penalty term for the number of local parameters used. The penalty we use for spline models is

$$\lambda \int_{x_1}^{x_n} [f''(x)]^2 dx \quad (3.17)$$

which implies that the spline estimate is

$$\text{SS}(f, \lambda) = \sum_{i=1}^n [y_i - f(x_i)]^2 + \lambda \int_{x_1}^{x_n} [f''(x)]^2 dx. \quad (3.18)$$

In Equation (3.18), we minimize the sum of squares between y and the nonparametric estimate, $f(x)$ subject to the penalty in Equation (3.17).

The term in Equation (3.17) is the constraint known as a roughness penalty. This penalty has two parts. The first is λ , often referred to as the smoothing or tuning parameter, and the second is the integrated squared second derivative of $f(x)$. The logic behind the use of the integrated squared second derivative of $f(x)$ is fairly intuitive. The second derivative measures the rate of change of the slope for a function or curvature. A large value for the second derivative means high curvature and vice versa.³ Through the use of the squared integral, the term sums a measure of curvature along the entire range of the nonparametric estimate in essence giving us a measure of curvature along the range of the nonparametric estimate. When it is large, $f(x)$ is rougher, and when it is small, $f(x)$ is smoother.

The λ parameter, which is nonnegative, directly controls the amount of weight given to the second derivative measure of how smooth f is. Therefore, λ establishes a tradeoff between closeness of fit to the data and the penalty giving λ a function analogous to the bandwidth for a *lowess* smoother. As the value for λ decreases, then $\hat{f}(x)$ interpolates the data, and we get a rough fit. As $\lambda \rightarrow \infty$ the integrated second derivative is constrained to be zero, and the result is the smooth global least squares fit. More generally, as λ increases, we get a smoother fit to the data but perhaps a biased fit, and as the parameter decreases, the fit displays less bias but with increased variance.

While a very small value for λ comes close to interpolating the data and a large value of λ returns a least square fit, intermediate values of λ often do not have an interpretable effect on the amount of smoothing applied to the data. To solve this problem, we can transform λ to be an approximation of the degrees of freedom, which is logical since we might view λ as controlling the number of local parameters used for the nonparametric estimate. The degrees of freedom for smoothing splines are calculated in a manner similar to that for OLS models and local polynomial regression, but the penalty creates some complications. Recall from the last chapter that the degrees of freedom for a linear regression model is equal to the number of fitted parameters which is equal to $\text{tr}(\mathbf{H})$ where $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}$. For standard spline models, which rely on an altered model matrix

³As another example, the Hessian is a matrix of second derivatives that measures the amount of curvature around the likelihood maximum.

but are then fit with least squares, the degrees of freedom would also be calculated by $\text{tr}(\mathbf{H})$. To calculate the degrees of freedom for penalized splines, we must generalize \mathbf{H} .

First, we write the penalized spline model in Equation (3.18) in matrix form. Given the equivalence between splines models and linear regression models, we can write the first term in Equation (3.18) as linear regression model in matrix form. It can be shown that the penalty term from Equation (3.18) can be written as a quadratic form in β (Ruppert, Wand, and Carroll 2003). This allows us to write the penalty in matrix form as

$$\int_{x_1}^{x_n} f''(x)^2 dx = \beta' \mathbf{D} \beta \quad (3.19)$$

where \mathbf{D} is a matrix of the following form:

$$\mathbf{D} = \begin{bmatrix} \mathbf{0}_{2 \times 2} & \mathbf{0}_{2 \times k} \\ \mathbf{0}_{k \times 2} & \mathbf{I}_{k \times k} \end{bmatrix}. \quad (3.20)$$

In the above matrix k denotes the number of knots. With a matrix form for the penalty, we can write the penalized spline regression model in matrix form

$$\text{SS}(f, \lambda) = \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda\beta' \mathbf{D}\beta. \quad (3.21)$$

The hat matrix for the penalized spline has a form where the standard linear regression matrix is altered to accommodate the penalty term. Ruppert, Wand, and Carroll (2003) derive the following hat or smoother matrix for penalized splines:

$$\mathbf{S}_\lambda = \mathbf{X}(\mathbf{X}'\mathbf{X} + \lambda^{2p}\mathbf{D})^{-1}\mathbf{X}' \quad (3.22)$$

where p is the order of the polynomial of the basis functions. In this form, we see that the penalty term is a scalar λ^{2p} multiplied by the matrix operator \mathbf{D} . The trace of \mathbf{S}_λ , just as in linear regression, represents the degrees of freedom in the spline model and is nearly equivalent to the number of parameters in the spline fit. Due to shrinkage from the penalty term, the degrees of freedom for a penalized spline model will not be an integer. At this point, it should be clear how penalized splines are estimated. The user selects values for λ and p and constructs a model matrix with a set of basis functions identical to that used for a standard spline model. Using these values, we form the matrix \mathbf{S}_λ and apply it to the outcome vector to form a set of predictions

$$\hat{\mathbf{y}} = \mathbf{S}_\lambda \mathbf{y}. \quad (3.23)$$

The predictions from the above equation are the penalized spline estimate that we can then plot. The simplicity of the smoothing spline matrix notation makes programming smoothing splines only slightly more complicated than a cubic spline. The process is the same as for a regression spline model in that y is regressed on a constructed model matrix using least squares. However, while estimating the smoothing spline is feasible with least squares, this method is often not numerically stable. Orthogonal matrix methods, such as either the Cholesky decomposition or a spectral decomposition, are required for numerical stability. An example of basic smoothing spline code is available on the book website.

Since the degrees of freedom are a mathematical transformation of λ , they provide control over the amount of smoothing applied to the data. By selecting the degrees of freedom, the analyst chooses the number of effective local parameters used in the spline estimate. Controlling the penalty using the the degrees of freedom is also more practical instead of selecting a value for λ , since often large differences in λ will translate into fairly small differences in the degrees of freedom. One important difference between penalized splines and standard splines is that the number of knots used now exerts very little influence over how smooth the fit is as the value of λ now controls how smooth the fit is. These penalized splines are commonly referred to as ‘smoothing splines’. In the text, we refer to these splines interchangeably as either penalized splines or smoothing splines. At this point, an illustration with smoothing splines is useful to show how they differ from standard splines.

We return to the Congressional elections example to better understand the operation of smoothing splines. Thus far, we have consistently seen a nonlinear relationship between challenger’s vote share and support for Perot in the 1992 election. We estimate the relationship between these two variables using smoothing splines to demonstrate how selecting different values for the degrees of freedom affects the fit.

Figure 3.6 contains four smoothing spline estimates where we set the degrees of freedom to 2, 4, 8, and 12 using a cubic spline basis. We hold the number of knots constant across all four fits. For smoothing splines, a large number of knots are placed evenly throughout the range of x such that 10 to 20 x values fall between each knot placement. There are also some simple selection rules to ensure an optimal number of knots. See Ruppert, Wand, and Carroll (2003) for details on knot selection algorithms. Different methods are implemented with different smoothing spline software, and most allow the user to control the number of knots directly as well. The software used in this example simply places knots so that there are 10 to 20 x observations between each set of knots. This implies that for the estimates in Figure 3.6 the models are fit with 31 knots. Given that nine knots produced noticeable overfitting with cubic B-splines, we might expect extreme

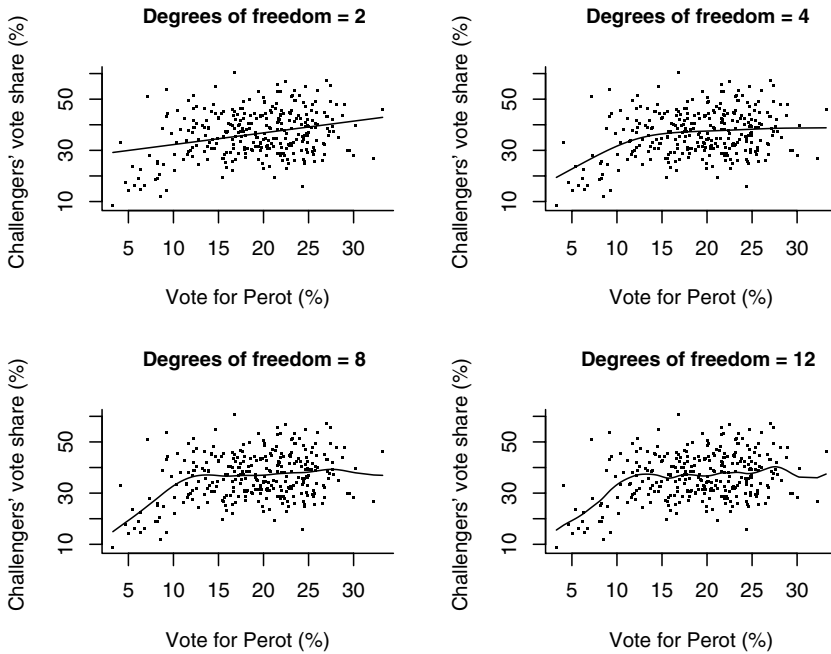


Figure 3.6 Smoothing spline fit to challenger vote share data with four different degrees of freedom values.

overfitting in this illustration. As the reader can see, however, this is not the case. The fit with 2 degrees of freedom is identical to a linear regression fit since there are only two effective parameters: one for the slope and one for the intercept. In the next model, with 4 degrees of freedom, we see the same pattern of nonlinearity found with other spline fits. There are only minor differences between the fit with 4 and 8 degrees of freedom. For the fit with 12 degrees of freedom, we see that considerable variability is caused by using too many parameters, and we overfit the data.

We next demonstrate that the number of knots used in the spline model matters little with smoothing splines. We fit two smoothing spline models to the Congressional election data. In both models, we hold the degrees of freedom constant at 4, but for one model we use four knots, and for the second model, we use 16 knots. The resulting fits are in Figure 3.7. For a cubic spline model, nine knots caused noticeable overfitting, but with the smoothing spline models, however, the results are invariant to the number of knots used. There is very little difference between four and 16 knots for the smoothing spline estimates. Despite

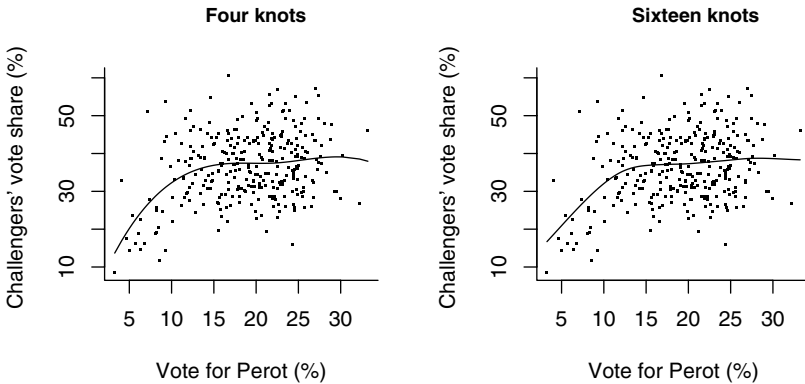


Figure 3.7 Smoothing spline fit to challenger vote share data with constant degrees of freedom and different knot numbers.

the additional parameters that are used for the model with 16 knots, the shrinkage imposed by the penalty term controls the amount of smoothing.

With smoothing splines, we have shifted control over the amount of smoothing from the knots to the tuning parameter, λ , or equivalently the degrees of freedom. The advantage is that the smoothing spline model will provide the same fit with fewer effective parameters, thus reducing the likelihood of overfitting regardless of the amount of smoothing imposed by the analyst. As such, while smoothing splines cannot eliminate the possibility of overfitting, smoothing splines reduce that possibility. Moreover, as we will see in the next section, smoothing splines provide an optimal tradeoff between fit and the number of parameters used. One might also ask: is there a formal method for selection of λ ? In fact, there are a number of different methods for choosing λ , which we consider in the next chapter.

3.3.2 Splines as Mixed Models

Random effects are a common solution to unobserved heterogeneity in statistical models. Panel data models, for example, are often susceptible to such heterogeneity. In panel data models, the model intercept varies across the units in the data. Random effects can be used to model the variation across the unit intercepts. In the random effects model, a random shock, drawn from a normal distribution with mean 0 and standard deviation σ^2 , is added to the intercept for each observation. This random shock is then integrated out of the likelihood, which removes the heterogeneity.

In statistics, models that include random effects are often referred to as *mixed models* (Pinero and Bates 2000). In the social sciences, a special case of mixed models called multilevel or hierarchical models are used with some frequency. We can use mixed models to estimate splines. In fact, smoothing splines are exactly represented as the optimal predictor in a mixed model framework (Ruppert, Wand, and Carroll 2003). While the mixed model framework has little effect on how we use splines in applied data analysis, the mixed model framework for splines provides two insights. First, it provides an analytic framework for understanding why smoothing splines are the optimal smoother. Second, the mixed model framework represents nonlinearity as unobserved heterogeneity, which helps to clarify the need for nonparametric regression techniques. Next, we (very briefly) outline the mixed model framework and demonstrate how to represent smoothing splines as mixed models. For a full account of splines as mixed models see Ruppert, Wand, and Carroll (2003). For readers unfamiliar with mixed models, this section can be skipped and little will be lost in terms of the practical usage of splines.

In the mixed model framework, we write a linear regression model for Y as:

$$Y_{ij} = \beta_{0j} + \beta_{1j}X_{ij} + e_{ij}. \quad (3.24)$$

For a mixed model, typically each term has two subscripts (more are possible). In this notation, we have i observations nested in j units. This might be i students nested in j schools or i voters nested in j counties. We believe there is heterogeneity across the j units; that is the intercepts and slopes may vary across the j units. If we suspect such variation across the slopes and intercepts, a random set of draws from a normal distribution is added to the intercept and slope parameters in the following manner:

$$\begin{aligned} \beta_{0j} &= \gamma_{00} + u_{0j} \\ \beta_{1j} &= \gamma_{01} + u_{1j} \end{aligned} \quad (3.25)$$

where u_{0j} and $u_{1j} \sim N(0, \sigma^2)$ are the random effects, γ_{00} is the mean outcome across the j units, and γ_{01} is the average slope across the j units. Using either MLE or restricted MLE, we integrate over the random effects to remove their effect from the log-likelihood.

Importantly, a mixed model represents a penalized approach to grouped data as it is a compromise between a parsimonious model and one with too many parameters. With data clustered in units, the analyst can estimate three different statistical models. In the first model, the analyst ignores the clustering within the j units and estimates a regression with the data pooled. The estimates from

this ‘pooled’ model will be biased if the j units differ much, but with the pooled data, there will be little variability in the fit as it pools all the observations. The model estimated with pooled data is the most parsimonious in terms of the number of parameters estimated. At the other extreme, the analyst could estimate one regression model for each of the j units. This option produces too many parameters, and the estimates of the β parameters will be highly variable if there are not very many observations within each j cluster. However, these estimates will be unbiased. The mixed model approach represents a middle ground between these two extremes. The mixed model estimate of the β 's is a compromise between ignoring the structure of the data and fully taking it into account by estimating j different models. The mixed model estimate shrinks the estimate from the pooled estimate toward the individual j estimates. The resulting estimates from the mixed model are weighted averages that ‘borrow strength’ across the units. This shrinkage estimate is superior in mean square error terms to both the pooled and unpooled alternatives. It can be shown that the mixed model estimates are the Best Linear Unbiased Predictors (BLUPs) (Pinero and Bates 2000; Ruppert, Wand, and Carroll 2003). Therefore, if there are significant differences across the j units, the mixed model provides the best fit according to the mean squared error criterion.

Nonlinearity is a similar form of heterogeneity across groups. The data within a set of knots forms each group. A linear fit to the data is identical to a pooled model that ignores any local variation that might exist. This pooled model ignores the underlying structure and uses a single parameter to summarize the relationship. This fit is the most parsimonious but ignores difference in the fit between x and y across the groups of data between knots. If the variation is substantial the linear fit is biased. A standard spline model with knots for each x value represents model with no pooling. Here, we use a parameter for each x and y value. Such a model produces an estimate that is very faithful to the local variation, but the estimates can be highly variable if there is too little data between each set of knots. The smoothing spline has the same structure as a mixed model as it takes the local variation into account but also borrows strength across all the knot segments. This implies that the smoothing spline should have the best fit according to the mean squared error criterion. The smoothing spline estimate is the BLUP for a model of heterogeneity due to nonlinearity as it shrinks the global estimate toward a model with highly local fits.

The regression spline model (with a linear basis) for f is:

$$f(x_i) = \beta_0 + \beta_1 x_i + \sum_{k=1}^K \beta_k^* (x_i - c_k)_+ + \varepsilon \quad (3.26)$$

where $(x_i - c_k)_+$:

$$(x_i - c_k)_+ = \begin{cases} 0 & x \leq c_k \\ x - c_k & x > c_k \end{cases} \quad (3.27)$$

represents a piecewise linear fit with knots at c_k . Now, rewrite each term in Equation (3.26) in matrix form:

$$\mathbf{X} = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \quad (3.28)$$

$$\mathbf{Z} = \begin{bmatrix} (x_1 - c_1)_+ & \cdots & (x_1 - c_k)_+ \\ \vdots & \ddots & \vdots \\ (x_n - c_1)_+ & \cdots & (x_n - c_k)_+ \end{bmatrix} \quad (3.29)$$

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} \quad (3.30)$$

$$\boldsymbol{\beta}^* = \begin{bmatrix} \beta_1^* \\ \vdots \\ \beta_k^* \end{bmatrix} \quad (3.31)$$

and

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix}. \quad (3.32)$$

The vector, $\boldsymbol{\beta}^*$, represents the coefficients that define the piecewise functions. We combine these terms to write Equation (3.26) in matrix form:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon}. \quad (3.33)$$

To write the linear spline model as a mixed model requires:

$$\mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_K \end{bmatrix}, \quad (3.34)$$

a vector of random effects, with each element drawn from $N(0, \sigma_{u_k}^2)$. We rewrite Equation (3.33) as the following mixed model:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \boldsymbol{\varepsilon}. \quad (3.35)$$

The difference in the models is that a random effect is placed on the location of each knot. The solution to Equation (3.35) is:

$$\hat{\mathbf{f}} = \mathbf{C}(\mathbf{C}'\mathbf{C} + \lambda^2\mathbf{D})^{-1}\mathbf{C}'\mathbf{y} \quad (3.36)$$

where $\mathbf{C} = [\mathbf{X} \ \mathbf{Z}]$, $\mathbf{D} = \text{diag}(0, 0, 1, \dots, 1)$, and $\lambda^2 = \sigma_{\varepsilon}^2 / \sigma_u^2$. Ruppert, Wand, and Carroll (2003) show that Equation (3.35) is equivalent to

$$\text{SS}(f, \lambda) = \sum_{i=1}^n y_i - f(x_i)^2 + \lambda \int_{x_1}^{x_n} f''(x)^2 dx \quad (3.37)$$

which is the smoothing spline representation.

What are the implications of this equivalence between smoothing splines and mixed models? First, this implies that the smoothing spline fit is the BLUP for the nonlinear estimate between x and y . As such, the smoothing spline fit should have the lowest mean squared error if nonlinearity is present. Therefore, the smoothing spline provides the best tradeoff between fit and the number of parameters used. Second, the mixed model framework allows us to conceptualize the nonlinearity between x and y as unobserved heterogeneity. If such unobserved heterogeneity is present and goes unmodeled, specification error results. Nonparametric regression provides us with a tool to model this heterogeneity. The logic behind nonparametric regression then is no different from that behind mixed models. Nonlinearity is a form of heterogeneity, and for such heterogeneity, the smoothing spline will provide the best estimate that balances fit with the number of parameters used. The mixed model representation of smoothing splines also allows smoothing splines to be easily translated into a Bayesian estimation framework, and it allows for the incorporation of smoothing splines into standard mixed models. We provide examples of both in Chapter 7.

3.3.3 Final Notes on Smoothing Splines

In general, analytic proofs for the properties of nonparametric regression are unavailable. Wood (2006), however, proves that cubic smoothing splines provide the fit with the least amount of error. The full proof is not overly complex but is lengthy, so we omit it here and refer the reader to Green and Silverman (1994) and Wood (2006). Below, we provide a sketch of the proof. The goal for any nonparametric regression model is to select the estimate of f that minimizes the

squared error between the fit and the data. We must select an estimate of f that is continuous on $[x_1, x_n]$ and has continuous first derivatives and is smoothest in the sense of minimizing

$$\int_{x_1}^{x_n} f''(x)^2 dx.$$

If $f(x)$ is the smoothing spline fit, it is the function that minimizes:

$$\sum_{i=1}^n y_i - f(x_i)^2 + \lambda \int_{x_1}^{x_n} f''(x)^2 dx. \quad (3.38)$$

Why? In short, the interpolant properties of smoothing splines ensure that no other function will have a lower integrated squared second derivative. Smoothing splines have the best properties in that no other smoothing function will have a lower mean squared error. Therefore, on analytic grounds, smoothing splines are to be preferred to other smoothers.

Finally, it is possible to use splines to smooth in more than two dimensions. The same caveats apply. The curse of dimensionality remains a problem, and it remains impossible to interpret fits in more than two dimensions. In general, however, most work in statistics uses splines to smooth in two dimensions rather than local polynomial regression.

3.3.4 Thin Plate Splines

We outline one last type of spline model. This form of spline, thin plate splines, is generally unnecessary for estimating simple nonparametric models or even most semiparametric models. Thin plate splines are primarily used for smoothing in more than one dimension, but they are also useful for estimating spline models with Markov Chain Monte Carlo (MCMC) methods.

With the cubic smoothing spline models considered thus far, the smoothing penalty is placed on the second derivative of the smoothed fit $f(x)$. It is possible to penalize any derivative of the smoothed fit subject to constraints imposed by the dimension of x . Thus, thin plate splines are a more general version of smoothing splines where the penalty placed in the spline fit can be placed on any order of derivative. The formal representation of a thin plate spline is as follows:

$$SS(f, \lambda) = \sum_{i=1}^n y_i - f(x_i)^2 + \lambda \int_{x_i}^{x_n} f^m(x_i)^2 dx. \quad (3.39)$$

Notice that the penalty on the derivative is now generalized. Here, m is the m th derivative such that $2m > d$, where d is the dimension of x_i . This implies that

moving beyond the second derivative requires smoothing in more than one dimension. The full representation of penalties for thin plate spline is fairly complex, but see Wood (2006) for an introduction. Thin plate splines come at a heavy computational cost as they have as many unknown parameters as there are data points. Typically some low rank approximation is used to avoid this problem. Unfortunately, even low rank thin plate splines are difficult to estimate with data sets over 1000 cases. While there are software packages that implement low rank thin plate regression splines, they offer few advantages for most estimation problems. However, low rank thin plate splines are useful when used with MCMC methods. Thin plate splines also work well when smoothing in more than one dimension.

3.4 Inference for Splines

Inference for splines is identical to inference for local polynomial regression smoothing. For confidence bands, we calculate pointwise standard errors from \mathbf{S} , the smoothing matrix, and plug these standard errors into the usual 95% confidence interval formula and plot the result along with the spline fit. We can also test spline models against linear models with or without transformations using F-tests. In this section, we develop the details for inference with splines and provide two illustrations.

Confidence Bands

For splines models, how we define \mathbf{S} depends on whether one is using a smoothing spline or not. Recall that cubic and natural spline models are simply regression models with an expanded model matrix. In the last chapter we defined \mathbf{S} as similar to the \mathbf{H} or hat matrix from a regression model. Since cubic and spline models *are* regression models, then \mathbf{S} is equivalent to the hat matrix: $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. The only difference is that we have altered the model matrix \mathbf{X} using a set of basis functions. For smoothing splines, \mathbf{S} is more complex. We defined the \mathbf{S} matrix for smoothing splines as $\mathbf{X}(\mathbf{X}'\mathbf{X} + \lambda^2\mathbf{D})^{-1}\mathbf{X}'$. To calculate confidence bands, either form of the \mathbf{S} matrix is used in an identical manner. In the discussion that follows, we assume that λ is fixed and \mathbf{S} represents either formulation of the smoothing matrix. Under these assumptions, the covariance matrix for the fitted vector $\hat{\mathbf{f}} = \mathbf{S}\mathbf{y}$ is

$$\text{cov}(\hat{\mathbf{f}}) = \mathbf{S}\mathbf{S}'\sigma^2. \quad (3.40)$$

If we have an unbiased estimate of σ^2 and the sample size is large enough, we form pointwise confidence interval bands using ± 2 times the square root of the diagonal elements of $\mathbf{S}\mathbf{S}'\sigma^2$. We still need an estimate for σ^2 , and we use the same method we used with local polynomial regression. If the estimate of f is

unbiased, an unbiased estimator of σ^2 is

$$\text{RSS}/df_{\text{res}} \quad (3.41)$$

where RSS is the residual sum of squares which is defined as: $\sum_{i=1}^n [y_i - \hat{f}(x_i)]^2$ and df_{res} is the residual degrees of freedom:

$$df_{\text{res}} = n - 2\text{tr}[\mathbf{S} + \text{tr}(\mathbf{S}\mathbf{S}')]. \quad (3.42)$$

In practice, these pointwise bands provide reasonable estimates of variability for the spline fit. In terms of statistical theory, however, these variability estimates have some limitations. First, it would be desirable to account for variability in $\hat{\sigma}$ as an estimate of σ , though this variability should be small so long as the sample size is large. If n is small, one should replace the Normal critical value of 2 with a t -distribution critical value with k degrees of freedom, where k is the number of parameters in the model (the trace of \mathbf{S}). More importantly, we must assume that the estimate of $f(x)$ is approximately unbiased. This assumption is difficult to test, and given what we know about nonparametric regression, we must assume that there is some bias in the estimate of f . The nonparametric fit is a tradeoff between bias and variance, and a fit with very little bias will be highly variable, so we must accept some bias to reduce the variance. Since we never observe the true smooth function, it is difficult to judge the level of bias in the fit. For these pointwise variability bands, it is often not unreasonable to assume the fit is approximately unbiased. If so, these variability bands can be interpreted as confidence bands. It would be better, however, if we could account for the possible bias in the estimate of f in the estimation of the variability bands. Again, smoothing splines are superior in this respect as they allow for bias adjustment to the confidence bands. There are broadly two different methods for the estimation of bias adjusted variability bands for splines. One is based on Bayesian principles, and the other relies on the mixed model framework. We, first, discuss the Bayesian variability bands.

Splines can be derived from a Bayesian smoothing model that assumes f is a sum of underlying random linear functions and an integrated Weiner process. Wahba (1983) and Nychka (1988) provide full details on the Bayesian derivation of splines. Importantly, they demonstrate that the variability bands from the Bayesian model take into account possible bias in the estimate of f . Using $\hat{\sigma}\mathbf{S}_\lambda$ to estimate the variance for f produces bias adjusted variability bands that are equivalent to the Bayesian confidence bands (Hastie and Tibshirani 1990). Not all smoothing software estimates bias corrected variability bands in this fashion. For example, Wood (2006) suggests several refinements and implements them in

his software for smoothing. In general, these refinements alter the estimates of the variability bands little.

When splines are estimated in a mixed model framework, the bias adjusted variance estimate, $\hat{\sigma}_{adj}^2$ is

$$\hat{\sigma}_{adj}^2 = \hat{\sigma}_\varepsilon^2 \sqrt{\mathbf{C}_x \left(\mathbf{C}'\mathbf{C} + \frac{\sigma_\varepsilon^2}{\sigma_u^2} \mathbf{D} \right)^{-1} \mathbf{C}'_x} \tag{3.43}$$

where $\mathbf{C} = [\mathbf{X} \mathbf{Z}]$, $\mathbf{D} = \text{diag}(0,0,1, \dots ,1)$. In the mixed model framework, the predictions are conditioned on the random effects to account for possible bias in the estimate of f (Ruppert, Wand, and Carroll 2003). The mixed model variability bands typically differ little from those estimated under the Bayesian framework.

We return to our example of challenger’s vote share and support for Perot to illustrate the estimation of confidence bands for spline models. First, we estimate the spline fit between the two variables using a natural cubic spline fit and then estimate pairwise standard errors and calculate 95% confidence bands for the spline fit. The result is in Figure 3.8. The estimated confidence bands indicate that the spline estimate is fairly precise. The confidence bands here differ little from those for local polynomial regression. Importantly, these confidence bands are not bias adjusted.

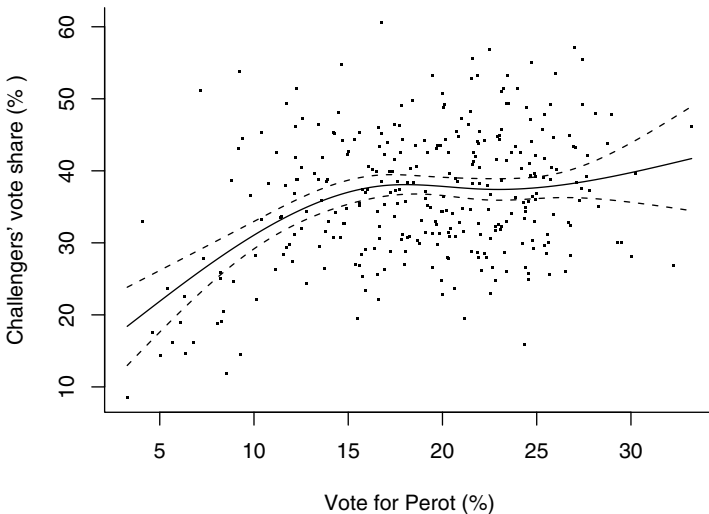


Figure 3.8 Confidence bands for natural cubic spline fit to challenger’s vote share.

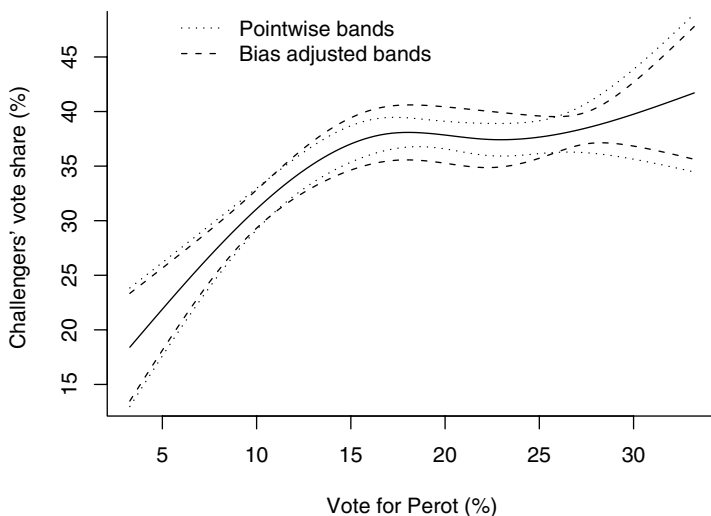


Figure 3.9 A comparison of pointwise variability bands and bias adjusted confidence intervals.

In Figure 3.9, we plot the Bayesian bias adjusted confidence bands along with the standard pointwise variability bands for the fit between challengers' vote share and support for Perot. The difference is negligible. The chief difference is that the bias adjusted bands provide a confidence interval for the fit while the variability bands are a pointwise estimate.

Hypothesis Tests

We can also perform hypothesis tests. Since the spline model nests models with global fits, we can rely on approximate F-tests for hypothesis tests (Hastie and Tibshirani 1990). Again to test for a statistically significant effect, we test the spline model against a model with only a constant. As before, we often test the spline model against a model with a global linear fit or against models with power transformations on the right hand side. If RSS_1 and RSS_2 are the residual sum of squares from a restricted model and the spline model respectively, an approximate F-test is:

$$\frac{(RSS_1 - RSS_2)/(df_{res2} - df_{res1})}{RSS_2/(n - df_{res2})} \sim F_{df_{res2} - df_{res1}, n - df_{res2}}. \quad (3.44)$$

This is most useful as a means of testing for nonlinearity and assessing the adequacy of power transformations. If there is no difference between a spline fit

and a simpler model, the simpler model is preferred for reasons of parsimony. Consequently, there is no need to use nonparametric methods unless the evidence from the data suggests that they are necessary.⁴

Next, we test various hypotheses about the nature of the relationship between the challengers' vote share and support for Perot. We compare the smoothing spline model to a model with only a constant to test whether the effect of support for Perot is significantly different from zero. Not surprisingly, we find that the relationship is highly significant as the test statistic is 17.14 on 3 and 311 degrees of freedom.⁵ We also test the spline model against a global linear fit, and the F-test test statistic of 10.31 on 2 and 310 degrees of freedom is highly statistically significant ($p < 0.001$). The results from this test indicate that the relationship between challengers' vote share and support for Perot is sufficiently nonlinear that a global linear model is inadequate. Finally, we test the spline fit against quadratic and logarithmic transformations of the support for Perot variable. In both instances, the F-test indicates that the spline fit is superior. As a result, a spline fit is recommended over these transformations, which do not adequately model the nonlinearity.

Derivative Plots

While nonparametric fits do not allow for the same form of concrete interpretation as parametric models, there are other methods of interpretation available. For example, we can plot the first derivative for the nonparametric estimate. The first derivative, in general, tells us about the rate of change between x and y , and we can calculate the first derivative along the entire range of the nonparametric fit and plot the result to further understand the dependency between the two variables. For example, a plot of the first derivatives provides insight into the following questions for the Congressional elections example:

- How fast does challengers' vote share increase?
- When does the effect of Perot support start to level off?
- Does the effect of support for Perot level off or does it have a significant negative effect on challengers' votes share for higher values?

⁴The F-test is only considered to be approximate for smoothing splines. However, as Hastie and Tibshirani (1990) note general investigation indicates that the approximation is fairly good. Bootstrapping techniques can be used to replace the parametric F-test assumption. See Chapter 8 for details on bootstrapping nonparametric models.

⁵I set the smoothing parameter, λ , to 4 and used a cubic basis function. This was based on earlier analyses that indicated that a fit with 4 degrees of freedom provided a reasonable fit.

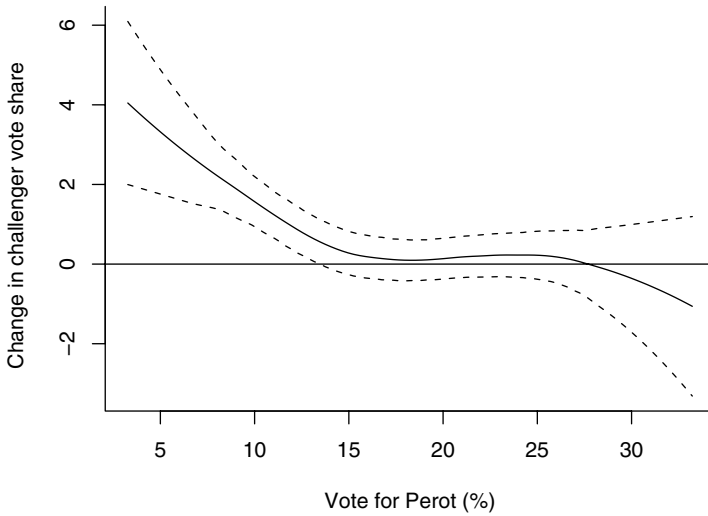


Figure 3.10 Plot of first derivative for the effect of support for Perot on the challenger’s vote share.

Figure 3.10 contains a plot of the first derivative for the spline estimate. In the plot, challengers gain vote share quite rapidly until support for Perot reaches approximately 15%. After that, there is no significant change in the vote share for challengers until support for Perot exceeded 25% in the district. However, at this point the confidence bands are so wide, one cannot place much emphasis on the deviation from 0. This is clearly due to the fact that there were very few Congressional districts where support for Perot exceeded 25%. Derivative plots are useful since they often allow for a richer interpretation of nonparametric regression models.

3.5 Comparisons and Conclusions

Thus far, we have presented the reader with several different nonparametric regression models. In this section, we provide a brief comparison of these models. It is logical to ask whether one form of nonparametric regression is preferable to another. For many applied problems, there aren’t strong reasons to prefer one smoother over another, but the smoothing spline does have some advantages. It is the only smoother that places a penalty on the number of parameters used in the fit. While the critique of nonparametric regression as prone to overfitting is perhaps overstated, the smoothing spline is designed to meet this criticism directly.

Moreover, the mixed model representation of smoothing splines provides an analytical foundation for these smoothers.

We present some basic simulation evidence to demonstrate how well different smoothers approximate a highly nonlinear functional form. For the simulation, we generated a function for f and fit four different smoothers, two local polynomial smoothers and two spline smoothers, to the simulated data. The simulated functional form is

$$y = \sin^3(2\pi x^2) + \varepsilon. \quad (3.45)$$

The result will be a cyclical pattern between x and y . Such functional forms are rare in the social sciences but make difficult estimation problems for nonparametric regression models. The four nonparametric fits are plotted in Figure 3.11.

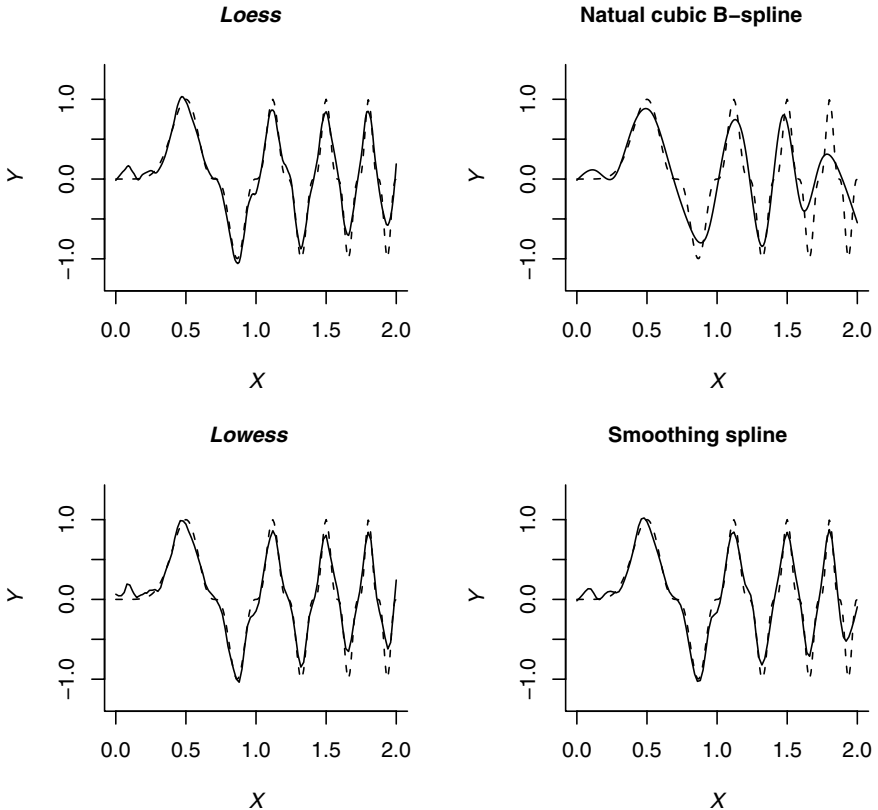


Figure 3.11 Comparison of smoother fits. Dotted line is true functional form, solid line is nonparametric estimate.

The first nonparametric model is the *loess* smoother. Using the visual trial and error method, we selected a value of 0.1 for the span. The *loess* fit is good though some minor undersmoothing occurs as it misses some of the peaks and valleys. In the upper right hand panel is a natural cubic B-spline with the knots chosen by AIC values. This model displays noticeable undersmoothing and provides a poor estimate for the upper range of x . In the lower left panel is a *lowess* estimate. The *lowess* estimate is almost identical to that from *loess*, so again we see that the additional weights used in *lowess* add little to the estimate. Finally, we estimate a smoothing spline using 31 degrees of freedom also selected through visual trial and error. The smoothing spline closely matches the nonlinearity and has only minor undersmoothing. The performance for the smoothing spline is quite similar to both the LPR smoothers. The simulation evidence suggests that both the LPR models and the smoothing spline are good choices. For basic smoothing of scatterplots, there is little reason to prefer one smoother over another. Only in rare instances should we find that different smoothers provide different estimates. As we will see in the next chapter, however, there are additional reasons to prefer the smoothing splines. Moreover, for semiparametric regression models, smoothing splines are preferable.

Smoothers are, by themselves, little more than powerful diagnostic tools. It is hard to imagine conducting an analysis with nonparametric regression alone. Smoothers, however, are particularly useful for deciding whether the relationship between two variables is linear or not. A visual examination of a scatterplot without the use of a smoother is a poor second option when smoothers allow you to easily see whether a relationship is linear or not. As we will see, smoothers are more powerful when used in conjunction with standard parametric models.

3.6 Exercises

Data sets for exercises may be found at the following website: http://www.wiley.com/go/keele_semiparametric.

1. The data in `demo2.dta` describe political demonstrations across 76 nations. The dependent variable is the percentage of activity in the country that can be classified as political demonstrations. The data set also contains two independent variables. One which records the level of unionization in the nation and a second which records the level of inflation. (NB: Don't name your data frame `demo`, R will confuse the `demo` variable with the data frame.)
 - (a) First, use cubic splines to study the relationship between the demonstrations and unionization. Compare the cubic spline model to a natural cubic spline

- model. Do you see in differences in the fit near the boundaries of the data?
- (b) Using natural cubic splines experiment with both the number of knots and the placement of the knots. How many knots are required before the fit appears to be clearly overfit? Select the number of knots using AIC.
 - (c) Compare a spline fit to a *lowess* fit. Are the results any different? Next, compare a smoothing spline model to a natural cubic spline model.
 - (d) Calculate 95% confidence interval bands and plot the relationship with CI bands.
 - (e) Test whether the relationship is statistically significant and whether there is a significant amount of nonlinearity. Does the spline model improve upon linear fits with quadratic or logarithmic transformations?
 - (f) Add inflation to the model and plot the joint nonparametric effect. Does inflation improve the fit of the model?
 - (g) Finally, develop computer code for your own spline smoother for this data. Start with piecewise linear functions. Can you roughly approximate the spline models with piecewise linear functions? Try piecewise quadratic fits.
2. The `forest-sub.dta` data on environmental degradation across countries. The data set contains a measure of deforestation, a measure of democracy (−10 to 10) and a measure of GDP per capita. Scholars in international relations often debate whether the effect of this democracy scale is constant. As a result, some analysts recode the scale to dummy variables for democracy (6 to 10) and autocracy (−6 to −10). Investigate whether splines can capture the effect of democracy on deforestation. First, use a scatterplot to examine the relationship between democracy and deforestation. Then fit a spline model to the data. Add confidence bands to the plot. Test this fit against a linear, quadratic and logarithmic models. Which model do you prefer? Does the spline model reveal anything substantive about the effect of democracy? What does it imply if the effect is linear as opposed to nonlinear?

