2

Regression Splines and Regression Smoothers

2.1 Introduction

This chapter launches a more detailed examination of statistical learning within a regression framework. Once again, the focus is on conditional distributions. But now, the mean function for a response variable is central. How does the mean vary with different predictor values? The intent is to begin with procedures that have much the same look and feel as conventional linear regression and gradually move toward procedures that do not.

2.2 Regression Splines

A "spline" is a thin strip of wood that can be easily bent to follow a curved line (Green and Silverman, 1994: 4). Historically, it was used in drafting for drawing smooth curves. Regression splines, a statistical translation of this idea, are a way to represent non-linear, but unknown, mean functions.

Regression splines are not used a great deal in empirical work. As we show, there are usually better ways to proceed. Nevertheless, it is important to consider them, at least briefly. They provide an instructive transition between conventional parametric regression and the kinds of smoothers commonly seen in statistical learning.

2.2.1 Applying a Piecewise Linear Basis

For a piecewise linear basis, the goal is to fit the data with a broken line (or hyperplane) such that at each break point the left-hand edge meets the right-hand edge. When there is a single predictor, for instance, the fit is a set of straight line segments, connected end to end, sometimes called "piecewise linear." Figure 2.1 is a simple illustration using three straight lines joined end to end. There is a response variable represented by y and a predictor represented by x. For now, only the fitted values are shown.

R.A. Berk, *Statistical Learning from a Regression Perspective*,
DOI: 10.1007/978-0-387-77501-2_2,

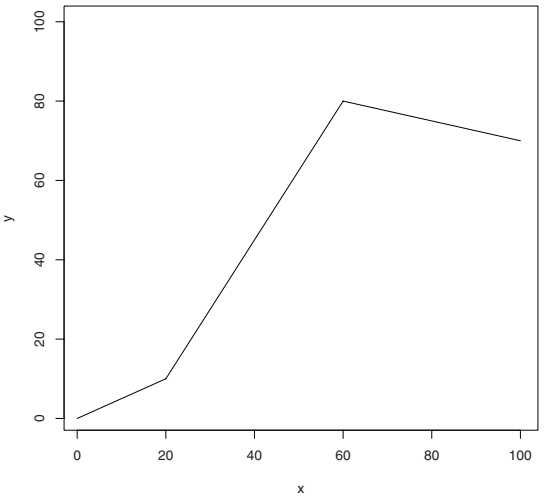

Fig. 2.1. An illustration of piecewise linear function with two knots.

Constructing such a function for the conditional means is straightforward in principle. First, one decides where the break points on x will be. If there is a single predictor, as in this illustration, the break points might be chosen after examining a scatter plot of y on x. When possible, subject matter expertise should also be used to help determine the break points. For example, x might be years, and then the break points might be determined by specific historical events. Thus, y might be a measure of a river's biodiversity, and x might be time in months, with one breakpoint representing the removal of a major dam and another breakpoint representing a toxic chemical spill. Let the break points here be defined at $x = a$ and $x = b$ (with $b > a$). In Figure 2.1, $a = 20$ and $b = 60$. Such break points are often called "knots."

The second step is to define two indicator variables to represent the break points. Here, the first (I_a) is equal to 1 if x is greater 20 and equal to 0 otherwise. The second (I_b) is equal to 1 if x is greater than 60 and equal to 0 otherwise. We let x_a be the value of x at the first break point, and x_b be the value of x at the second break point.

The third step is to define the mean function. Because at this point description is the primary goal, the conditional mean of y is represented by $\bar{y}|x$ rather than by $E(y|x)$. The latter implies that Y is a random variable. For now, it does not matter whether Y is a random variable. Then,

$$\bar{y}|x = \beta_0 + \beta_1 x + \beta_2(x - x_a)I_a + \beta_3(x - x_b)I_b. \tag{2.1}$$

Looking back at Equation 1.15, it is apparent that there are four transformations of X, $h_m(X)$s, in which the first function of x is a constant.

The mean function for x less than a is

$$\bar{y}|x = \beta_0 + \beta_1 x. \tag{2.2}$$

In Figure 2.1, β_0 is zero, and β_1 is positive.

For values of x greater than a but smaller than b, the mean function becomes

$$\bar{y}|x = (\beta_0 - \beta_2 x_a) + (\beta_1 + \beta_2)x. \tag{2.3}$$

For a positive β_1 and β_2, the line beyond $x = a$ is steeper because the slope is $(\beta_1 + \beta_2)$. The intercept is lower because of the second term in $(\beta_0 - \beta_2 x_a)$. This too is consistent with Figure 2.1. If β_2 is negative, the reverse would apply.

For values of x greater than b, the mean function becomes,

$$\bar{y}|x = (\beta_0 - \beta_2 x_a - \beta_3 x_b) + (\beta_1 + \beta_2 + \beta_3)x. \tag{2.4}$$

For these values of x, the slope is altered by adding β_3 to the slope of the previous line segment. The intercept is altered by subtracting $\beta_3 x_b$. The sign and magnitude of β_3 determine if the slope of the new line segment is positive or negative and how steep it is. The intercept will shift accordingly. In Figure 2.1, β_3 is negative and large enough to make the slope negative. The intercept is increased substantially.

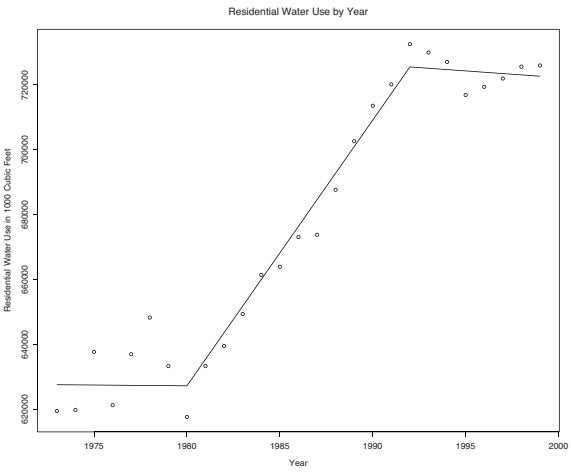

Fig. 2.2. A piecewise linear basis applied to water use by year.

Figure 2.2 shows a three-piece linear regression spline applied to water use data from Tokyo over a period of 27 years. Residential water use in 1000s of

cubic feet is on the vertical axis. Year is on the horizontal axis. The locations of the break points were chosen after inspecting the scatterplot, with some reliance on subject matter expertise about residential water use in Japan.

It is clear that water use was flat until about 1980, then increased linearly until about 1996, and then flattened out again. The first break point may correspond to a transition toward much faster economic and population growth. The second break point may correspond to the introduction of more water-efficient technology. But why the transitions are so sharp is mysterious. One possibility is that the break points correspond in part to changes in how the water use data were collected or reported.

It is perhaps most common to see regression splines fit to data in which time is used as the sole predictor. The end-to-end connections between line segments lead naturally to processes that unfold over time. The line segment on the right side of a knot begins where the line segment on the left side of the knot ends. But there is nothing about linear regression splines requiring that time be a predictor. For example, the response could be crop production per acre and the sole predictor could be the amount of phosphorus fertilizer applied to the soil. Crop production might increase in approximately a linear fashion until an excess of phosphorus caused other kinds of nutritional difficulties. At that point, crop yields might decline in roughly a linear manner.

Fitting line segments to data provides an example of "smoothing" a scatterplot, or applying a "smoother." The line segments are used in place of the data to characterize how x and y are related. The intent is to highlight key features of any association while removing unimportant details. This can often be accomplished by constructing fitted values in a manner that makes them more homogeneous than the set of conditional means of y computed for each unique value of x.

Imagine a scatterplot in which the number of observations was large enough so that for each value of x there were at least several values of y. One could compute the mean of y for each x-value. If one then drew a straight line between each of the adjacent conditional means, the resulting smoother would be an interpolation of the conditional means and as rough as possible. At the other extreme, imposing a single linear fit on all of the means at once would produce the smoothest smoother possible. Figure 2.2 falls somewhere in between. How to think about the degree of smoothness more formally is addressed later.

For a piecewise linear basis, one can simply compute functions such as Equation 2.1 with ordinary least squares. With the regression coefficients in hand, fitted values are easily constructed. Indeed, many software packages compute and store fitted values on a routine basis. Also widely available are procedures to construct the matrix of regressors, although it is not hard to do so one term at a time using common transformation capabilities. For example, the library *spline* has a procedure bs() that constructs a B-spline basis (discussed later) that can be easily used to represent the predictor matrix for piecewise linear regression.

In contrast to most applications of conventional linear regression, there would typically be little interest in the regression coefficients themselves; they are but a means to an end. The point of the exercise is to superimpose the fitted values on a scatterplot so that the relationship between y and x can be more effectively visualized. As we show later, and as was briefly anticipated in the last chapter, model selection will not necessarily be the same as regressor selection.

2.2.2 Polynomial Regression Splines

Smoothing a scatterplot using a piecewise linear basis has the great advantage of simplicity in concept and implementation. And by increasing the number of break points, very complicated relationships can be approximated. However, in most applications there are good reasons to believe that the underlying relationship is much smoother than can be easily represented with a set of straight line segments.

Greater continuity can be achieved by using polynomials in x for each segment. Cubic functions of x are a popular choice because they strike a nice balance between flexibility and complexity. When used to construct regression splines, the fit is sometimes called "piecewise cubic." The cubic polynomial serves as a "truncated power series basis" in x.

Unfortunately, simply joining polynomial segments end to end is unlikely to result in a visually appealing fit where the polynomial segments meet. The slopes of the two lines will often appear to change abruptly even when that is inconsistent with the data. Far better visual continuity usually can be achieved by constraining the first and second derivatives on either side of each break point to be the same.

Putting this all together, one can generalize the piecewise linear approach and impose the continuity requirements. Suppose there are K interior break points, usually called "interior knots." These are located at $\xi_1 < \cdots < \xi_K$ with two boundary knots added at ξ_0 and ξ_{K+1}. Then, one can use piecewise cubic polynomials in the following regression formulation,

$$\bar{y}|x = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \sum_{j=1}^{K} \theta_j (x - x_j)_+^3, \tag{2.5}$$

where the "+" indicates the positive values from the expression inside the parentheses, and there are $K + 4$ parameters whose values need to be computed. This leads to a conventional regression formulation with a matrix of predictor terms having $K + 4$ columns and N rows. Each row would have the corresponding values of the piecewise cubic polynomial function evaluated at the single value of x for that case. There is still only a single predictor, but now there are $K + 4$ basis functions.

The output for the far-right term in Equation 2.5 may not be apparent at first. Suppose the values of the predictor are arranged in order from low

to high. For example, $x = [1, 2, 4, 5, 7, 8]$. Suppose also that x_j is located at an x-value of 4. Then, $(x - x_j)_+^3 = [0, 0, 0, 1, 27, 64]$. The knot-value of 4 is subtracted from each value of x, the negative numbers set to 0, and the others cubed. All that changes from knot to knot is the value of x_j that is subtracted. There are K such knots and K such terms in the regression model.

Figure 2.3 shows the water use data again, but with a piecewise cubic polynomial overlaid that imposes the two continuity constraints. The fit looks quite good to the eye and captures about 95% of the variance in water use. But, in all fairness, the scatterplot did not present a great challenge. The point is to compare Figure 2.2 to Figure 2.3 and note the visual difference. The linear piecewise fit also accounted for about 95% of the variance. Which plot would be more instructive in practice would depend on the use to be made of the fitted values and on prior information about what a sensible $f(X)$ might be. The regression coefficients ranged widely and, as to be expected, did not by themselves add any useful information. The story was primarily in the fitted values.

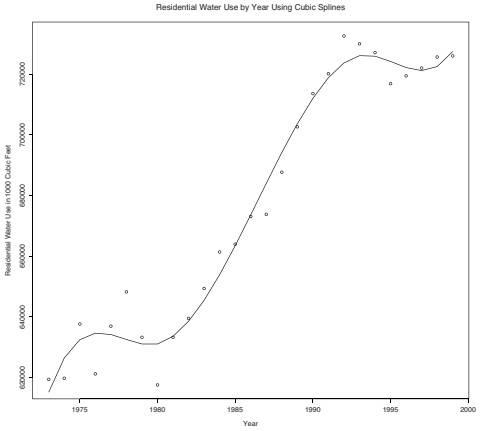

Fig. 2.3. A piecewise cubic polynomial applied to water use by year.

2.2.3 Natural Cubic Splines

Fitted values for piecewise cubic polynomials near the boundaries of x can be unstable because they fall at the ends of polynomial line segments where there are no continuity constraints, and where there may be little data. By "unstable" one means that a very few observations, which could vary over random samples from the same population, produce substantially different fitted values near the boundaries of x. As a result, the plot of the fitted values near the boundaries could look rather different from sample to sample.

Sometimes, constraints for behavior at the boundaries are added to increase stability. One common constraint imposes linearity on the fitted values beyond the boundaries of x. This introduces a bit of bias because it is very unlikely that if data beyond the current boundaries were available, their relationship with the response would be linear. However, the added stability is often worth it. When these constraints are added, the result is a "natural cubic spline."

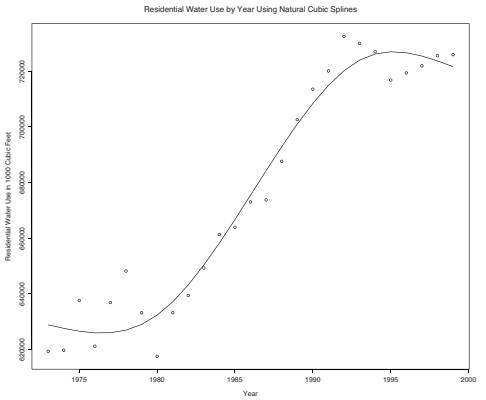

Fig. 2.4. Natural cubic regression splines applied to water use by year.

Figure 2.4 shows again a plot of the water use data on year, but now with a smoother constructed from natural cubic splines. One can see that the fitted values near the boundaries of x are somewhat different from the fitted values near the boundaries of x in Figure 2.3. The fitted values in Figure 2.4 are smoother, which is the desired result. There is one less bend near both boundaries. More generally, how one can formulate the boundary constraints is discussed in Hastie et al. (2001: Section 5.2.1).

The option of including extra constraints to help stabilize the fit provides an example of the bias–variance tradeoff. This is a topic to which we return many times in the pages ahead. For now, an informal overview for natural cubic splines may be useful.

The bias–variance tradeoff addresses some important properties of fitted values when the data are a random sample from a population or a realization of a stochastic process. Bias and variance refer to what can happen over a limitless number of hypothetical, independent, random samples or realizations; the context is the usual frequentist thought experiment. Therefore, the bias–variance tradeoff is only directly relevant when statistical inference is on the table and does not formally provide much insight when summary statistics are being used solely for description.

When constraints are imposed on a fitting process to make the fitted values less variable, bias in the fitted values can be introduced. The means of the fitted values over many samples or realizations will often be farther from the true conditional means of the response variable, which are the values one wants to estimate. However, in repeated independent random samples, or independent realizations of the data, the fitted values will vary less. When the fit is smoother, each fitted value is constructed, in effect, from a larger number of y-values. This increases stability in the same way that larger samples in general provide estimates with a smaller variance. Conversely, but using the same reasoning, a rougher fit can imply less bias but more variance over repeated samples or realizations. A tradeoff naturally follows.

Ideally, just the right amount of bias can be combined with just the right amount of variance so that over repeated random samples or realizations, the fitted values would be on the average as close to true response variable values as possible. "Close" can be operationalized in several ways, but it is often desirable to work with a test sample and then try to minimize the mean of the squared deviations between the fitted values and the observed values of the response variable (i.e., the mean squared error in a test sample).

For piecewise cubic polynomials and natural cubic splines, the degree of smoothness is primarily a function of the number of interior knots. In practice, the smaller the number of knots, the smoother are the fitted values. A smaller number of knots means that there are more constraints on the pattern of fitted values because there are fewer end-to-end, cubic line segments used in the fitting process. Consequently, less provision is made for potential twists and turns.

But placement matters too. Ideally, knots should be located where it is thought that the $f(X)$ is changing most rapidly. In some cases, inspection of the data, coupled with subject matter knowledge, can be used to determine the number and placement of knots. The water use data just considered were analyzed in this manner.

Alternatively, the number and placement of knots can be approached as a model selection problem. Any of the fit statistics discussed in the last chapter, such as the GCV, can be used to determine the number of knots, given a set of candidate locations. The number of knots translates into a penalty for the number of regression parameters whose values are being estimated from the data. The penalty increases with the number of knots, just as the penalty would normally increase with the number of regression parameters whose values were not known a priori. Then, the goal is to choose the knot number that minimizes the fit statistic. Knot selection is essentially regressor selection. In other words, a set of potential knots is specified, and fit statistics are used to determine which knots are really needed.

The fit statistics are largely silent on where to place the knots. Two models with the same number of knots can produce very different fitted values if the placement of the knots substantially differs. Two models with a very different number of knots may fit the data about the same, depending on

where the knots are placed. Moreover, absent subject matter information, knot placement has been long known to be a difficult technical problem, especially when there is more than one predictor (de Boors, 2001). The fitted values are related to where the knots are placed in a very complicated manner. Fortunately, methods discussed later sidestep the knot location problem.

Even if a good case for candidate knot locations can be made, one must be careful about taking any of the fit measures too literally. First, there will often be several models with rather similar values, whatever the kind of fit statistic used. Then, selecting a single model as "best" using the fit measure alone may amplify a small numerical superiority into a large difference in the results, especially if the goal is to interpret how the predictors are related to the response. Some jokingly call this "specious specificity." Second, the same issues can arise when comparing the models selected by the different kinds of fit statistics. The impact of very small differences in fit can lead to very large difference in the results. Third, one must be a very careful to not let small differences in the fit statistics automatically trump subject matter knowledge. The risk is arriving at a model that may be difficult to interpret, or effectively worthless.

In summary, for regression splines of the sort just discussed, there is no straightforward way to arrive at the best tradeoff between the bias and the variance because there is no straightforward way to determine knot location. A key implication is that it is very difficult to arrive at a model that is demonstrably the best. Fortunately, there are other approaches to smoothing that are more promising.

2.2.4 *B*-Splines

In practice, data analyses using piecewise cubic polynominals and natural cubic splines are rarely constructed directly from polynomials of x. They are commonly constructed using a B-spline basis, largely because of computational convenience. A serious discussion of B-splines would take us far afield and accessible summaries can be found in Gifi (1990) and Hastie et al. (2001). Nevertheless several observations are worth making.

The goal is to construct transformations of x that allow for a cubic piecewise fit but that have nice numerical properties and are easy to manipulate. B-splines do well by all three criteria. They are computed in a recursive manner from very simple functions to more complex ones, and consistent with the approach to basis functions taken here, can be represented as a linear basis expansion.

For a series of knots, which usually include several beyond the upper and lower boundaries of x, indicator variables are defined for each region marked off by the knots. If a value of x falls within a given region, the indicator variable for that region is coded 1, and coded 0 otherwise. For example, if there is a knot at an x-value of 2 and the next knot at an x-value of 3, the x-values between them form a region with its own indicator variable coded 1

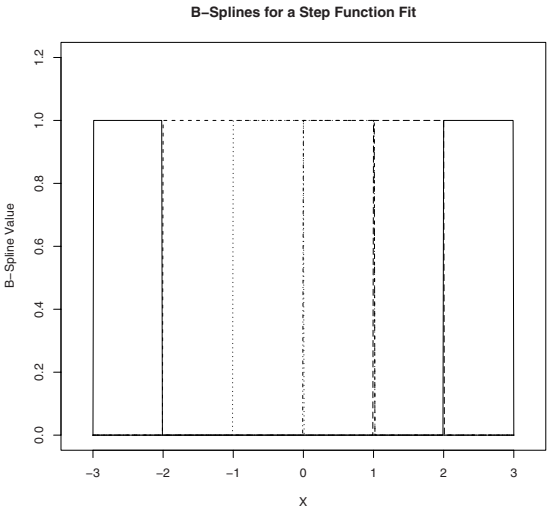

Fig. 2.5. Degree zero B-splines.

if the value of x falls in that region (e.g., $x = 2.3$), and coded 0 otherwise. The result is a set of indicator variables, with values of 1 or 0, for each region. These indicator variables define a set of degree zero B-splines.

Figure 2.5 is an illustration with interior knots at -2, -1, 0, 1, and 2. Using the indicator variables as regressors will produce a step function when y is regressed on x; they are the basis for a step function fit. The steps will be located at the knots.

Next a transformation can be applied to the degree zero B-splines. (See Hastie et al., 2001: 160–163). The result is a set of degree one B-splines. Figure 2.6 shows the set of degree one B-splines derived from the indicator variables shown in Figure 2.5. The triangular shape is characteristic of degree one B-splines, and implies that the values for each spline are no longer just 0 or 1, but proportions in between as well.

Degree one B-splines are the basis for linear piecewise fits. Here, the regressor matrix includes eight columns whose values appear in Figure 2.6. The content of each column is the B-spline values for each value of x. Regressing a response on that matrix will produce a linear piecewise fit with knots at -2, -1, 0, 1, and 2.

A transformation of the same form can now be applied to the degree one B-splines. This leads to a set of degree two B-splines that are the basis for a quadratic piecewise fit. For this illustration, there is now a matrix with nine columns that can serve as a regressor matrix. The set of such B-splines is shown in Figure 2.7 and as before, the shapes are characteristic.

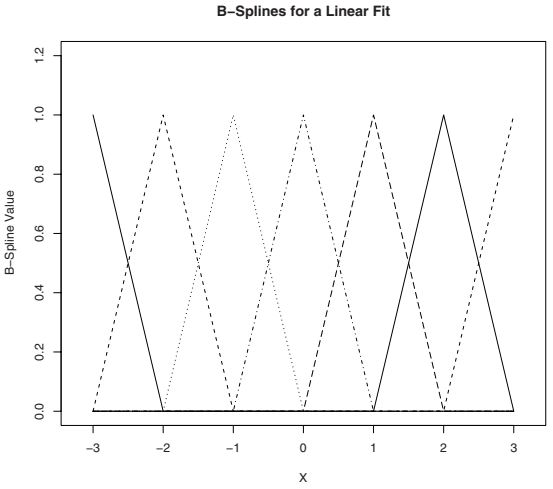

Fig. 2.6. Degree one B-splines.

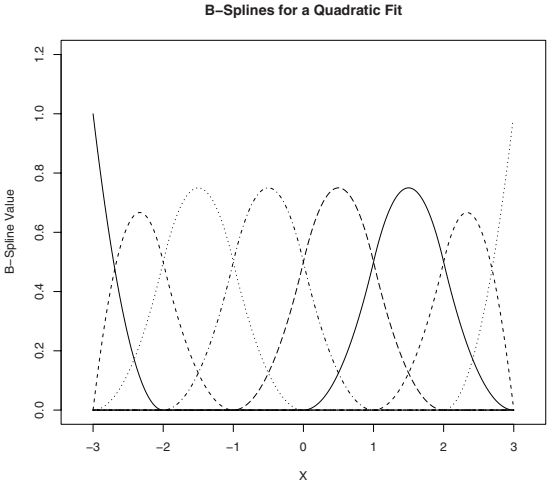

Fig. 2.7. Degree two B-splines.

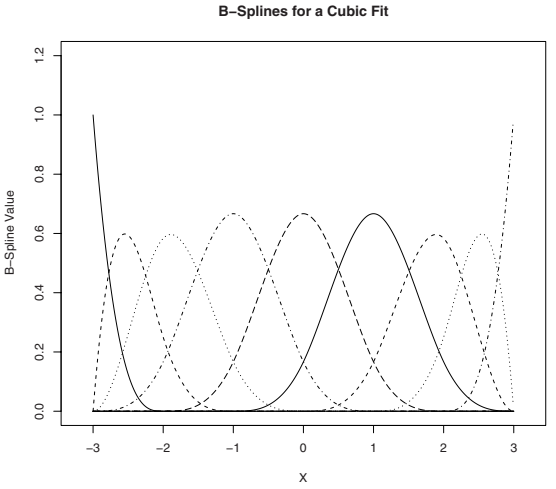

Fig. 2.8. Degree three B-splines.

The same kind of transformation can then be applied to the degree two B-splines. The result is a set of degree three B-splines that are the basis for a cubic piecewise fit. Figure 2.8 shows the set of degree three splines, whose shapes are, once again, characteristic. They can be used to construct a regressor matrix with nine columns.

All splines are a linear combinations of B-splines; B-splines are a basis for the space of all splines. They are also a well-conditioned basis because they are fairly close to orthogonal, and they can be computed in a stable and efficient manner. For our purposes, the main point is that B-splines are a computational device used to construct cubic piecewise fitted values. When such smoothers are employed, B-splines are doing the work behind the scenes.

2.3 Penalized Smoothing

The placement of knots, the number of knots, and the degree of the polynomial are subject to manipulation by a data analyst. All three can be used to construct a highly flexible fitting function that will track the data well. Because a good fit is typically considered desirable, there is sufficient reason in practice to worry about overfitting. The pull toward constructing a good fit can be very strong.

The fit statistics considered earlier can provide some protection against overfitting. They can help compensate for the amount of flexibility built into a given fitting function. However, they function indirectly. They are applied

after a model has been constructed to obtain a more honest measure of fit quality that can sometimes inform future fitting attempts.

A useful alternative is to alter the fitting process itself so that potential overfitting of a given model comes at a price. In particular, a penalty can be introduced into the loss function to be optimized that imposes increasing losses with increasing flexibility, regardless of how well the model is otherwise doing. In part because this approach has wide applicability, it is worth our attention now. Penalized fitting procedures figure significantly in this and later chapters.

2.3.1 Shrinkage

All of the procedures discussed in this chapter can be formulated as a conventional regression analysis. The procedures vary in the regressor matrix employed and how that matrix is determined. Whatever the regressor matrix used, there will be a set of regression coefficients. The larger the absolute value of these coefficients, other things being equal, the more the fitted values can vary.

To get some feel for this, consider a conventional regression analysis with an indicator variable as the sole regressor. If its regression coefficient equals zero, the fitted values will be a straight line, parallel to the x-axis, located at the unconditional mean of the response. As the regression coefficient increases in absolute value, the resulting step function will have a step of increasing size. The fit becomes more rough. More generally, the potential for rougher fit is greater with larger regression coefficients. Insofar as the roughness results from fitting idiosyncratic features of the data, there is overfitting. There are situations, therefore, in which it can be useful to control how large the regression coefficients are allowed to become.

A number of proposals have been offered for how to control the magnitude of regression coefficients. (See Ruppert et al., 2003: Section 3.5 for a very accessible discussion. Two popular suggestions are

1. Constrain the sum of the absolute values of the regression coefficients to be less than some constant C (sometimes called an L_1–penalty).
2. Constrain the sum of the squared regression coefficients to be less than some constant C (sometimes called an L_2–penalty).

The smaller the value of C is, the smaller the sum. The smaller the sum, the smaller is the typical magnitude of the regression coefficients. In part because the units in which the regressors are measured will affect how much each regressor contributes to the sum, it can make good sense to work with standardized regressors. The intercept does not figure in either constraint and is usually addressed separately.

Both constraints lead to "shrinkage methods." The regression coefficients can be "shrunk" toward zero, making the fitted values more homogeneous.

The goal is to introduce a small amount of bias into the computed regression coefficients in trade for a substantial reduction in their variance. There may also be subject matter reasons for preferring a smoother set of fitted values. Subject matter theory and/or past research may suggest that the response is a relatively smooth function of the predictors.

Shrinkage methods can be applied with the usual regressor matrix or with regressor matrices of the sorts we have considered in this chapter. With our focus on statistical learning, the latter is emphasized shortly. We start, however, within a conventional multiple regression framework and p predictors. We show that there can be two somewhat different goals: to construct more stable fitted values and to determine which regressors can be included as predictors. Shrinkage methods can be viewed as a form of "regularization," which figures significantly in later chapters.

One also can recast some measures of fit discussed in the last chapter within a shrinkage framework. The total number of regression coefficients to be estimated can serve as a constraint and is sometimes called an L_0–penalty. Maximizing the adjusted R^2, for example, can be seen as maximizing the usual error sum of squares subject to a penalty for the number of regression coefficients in the model (Fan and Li, 2006).

Ridge Regression

Suppose one adopts the constraint that the sum of the p squared regression coefficients is less than C. The L_2 constraint leads directly to ridge regression. The task is to obtain values for the regression coefficients so that

$$\hat{\boldsymbol{\beta}} = \min_{\beta} \left[\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right]. \tag{2.6}$$

In Equation 2.6, the usual expression for the error sum of squares has a new component. That component is the sum of the squared regression coefficients multiplied by a constant λ. When Equation 2.6 is minimized in order to obtain $\hat{\boldsymbol{\beta}}$, the sizes of the squared regression coefficients are taken into account.

For a given value of λ, the larger the $\sum_{j=1}^{p} \beta_j^2$ is, the larger the increment to the error sum of squares. The $\sum_{j=1}^{p} \beta_j^2$ can be thought of as the penalty function. For a given value of $\sum_{j=1}^{p} \beta_j^2$, the larger the value of λ is, the larger the increment to the error sum of squares; λ determines how much weight is given to the penalty. In short, $\sum_{j=1}^{p} \beta_j^2$ is what is being constrained, and λ imposes the constraint. C is inversely related to λ. The smaller the value of C, the larger is the value of λ.

It follows that the ridge regression estimator is

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}, \tag{2.7}$$

where $\mathbf{I}$ is a $p \times p$ identity matrix. The column of 1s for the intercept is dropped from $\mathbf{X}$.

In Equation 2.7, λ plays same role as in Equation 2.6, but can now be seen as a tuning parameter. It is not an estimate of some feature of a population or a stochastic process. Its role is to help provide an appropriate fit to the data and can be altered directly by the data analyst. As such, it has a different status from the regression coefficients, whose values are determined through the minimization process itself, conditional upon the value of λ.

The value of λ is added to the main diagonal of the cross-product matrix $\mathbf{X}^T\mathbf{X}$, which determines how much the estimated regression coefficients are "shrunk" toward zero (and hence, each other). A λ of zero produces the usual least squares result. As λ increases in size, the least squares regression coefficients approach zero, and the fitted values are smoother. In effect, the variances of the predictors are being increased with no change in the covariances between predictors or with the response variable. This is easy to appreciate in the case of a single predictor. For a single predictor, the regression coefficient is the covariance of the predictor with the response divided by the variance of the predictor. So, if the covariance is unchanged and the variance is increased, the absolute value of the regression coefficient is smaller.

The results are not invariant to the scales used for the predictors; the regression coefficients obtained will differ in a complicated manner depending on the units in which the predictors are measured. It is common, therefore, to standardize the predictors before the estimation begins. However, standardization is just a convention and does not solve the problem of the results being scale-dependent. Knowing how much the average response changes in standard deviation units for a one standard deviation change in a predictor conveys little unless one also knows the size of the two standard deviations. And those standard deviations are scale-dependent.

A key issue is how the value of λ is chosen. One option is trial and error. Different values of λ are tried until the desirable amount of smoothness is achieved. Alternatively, the value of λ is selected by some measure of prediction error such as the cross-validation statistic. The value of λ is chosen to maximize prediction accuracy. Both methods can lead to overfitting insofar as many different models are applied to the training data.

What one makes of output from a ridge regression depends substantially on the usual issues. If estimation is an important goal, one must be able to credibly argue that for each configuration of x-values, one can treat the data on hand as a random sample or realization, as discussed earlier. Then, one must meet the usual regression assumptions. If, for example, there are omitted predictors, whether the resulting biases are likely to be large enough to matter in practice would need to be addressed on a case-by-case basis.

However, ridge regression introduces some additional complications. The estimates of the regression coefficients and hence, the fitted values, are biased by design. If hypothesis tests are undertaken and conventional regression output used, the reported p-values are no longer accurate. And if conventional confidence intervals are constructed, they do not have their usual coverage. The regression estimates are necessarily offset by a systematic but unknown

amount. We return to this matter a little later after other shrinkage procedures are discussed.

Ridge regression was first developed to address the instability of estimated regression coefficients when regressors are highly correlated. It is now appreciated that the applications are broader. Here, we are interested at least as much in description as in estimation, and ridge regression provides one means to alter the smoothness of the fitted values. Also, shrinkage methods can be given a Bayesian interpretation in which the regression coefficients are shrunk toward a prior joint distribution of the regression coefficients. Some researchers find this instructive.

The Lasso

Suppose that one now adopts the constraint that the sum of the absolute values of the regression coefficients is less than some constant. The L_1 constraint leads to a regression procedure known as the lasso (Tibshirani, 1996) whose estimated regression coefficients are defined by

$$\hat{\beta} = \min_{\beta} \left[\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right]. \tag{2.8}$$

Unlike the ridge penalty, the lasso penalty leads to a nonlinear estimator, and a quadratic programming solution is needed. As before, the value of λ is a tuning parameter, determined empirically, usually through some measure of prediction error. Just as with ridge regression, a λ of zero yields the usual least squares results. As the value of λ increases, the regression coefficients are shrunk toward zero.

Hastie and his colleagues (2001: Section 3.4.5) place ridge regression and the lasso in a larger context in order to compare them to each other and to other procedures. A major interest is the patterns of shrinkage as the λ changes. Ridge regression tends to shrink the coefficients so that they all reach zero together as λ gets large. The lasso shrinks the coefficients so that some reach zero well before others as λ gets large. Thus, the lasso performs in a manner that has some important commonalities with model selection procedures used to choose a subset of regressors. Rosset and Zhu (2007) consider the path that the regression coefficients take as the value of λ changes, place the lasso in a class of regularization processes in which the solution path is piecewise linear, and then develop a robust version of the lasso. Wang et al. (2007) combine quantile regression with the lasso to derive another robust variable selection approach. We show later that the lasso has some interesting connections to boosting. In short, the lasso is more than a regularization procedure. It can help to provide useful insights about a wide variety of statistical tools.

Of late, there has been a lot of interest in the theoretical properties of the lasso and related procedures (Fan and Li, 2006; Meinshausen and Bühlmann,

2006). In particular, assuming that one has a set of predictors that includes all of those that belong in the model, as well as some number of irrelevant predictors, a key question is whether the lasso selects the correct predictors, at least as the sample size increases without limit. At this point, the answer seems to be sometimes it does and sometimes it does not. In particular, certain patterns of moderate to high correlations between predictors can lead to inappropriate predictors being selected along with the correct ones. Moreover, it can be very difficult to know with a real dataset whether or the problematic relationships between the predictors exist. In part as a response, Zou (2006) has proposed the adaptive lasso, which in theory is an improvement. The wrinkle is to employ "cleverly chosen" weights for the regression coefficients in the L_1 penalty function (Zou, 2006: section 3.1). The weights, in turn, are determined by another tuning parameter (in addition to λ). Finally, concerns have been raised about how well the lasso performs when there are heavy-tailed disturbance distributions or outliers. One response is to combine the lasso with quantile regression so that larger residuals are given relatively less weight in the fitting process (Wang et al., 2007).

In practice, the overriding problem with the lasso is the usual one: the underlying regression formulation has to be effectively correct. The data were in fact generated by a process represented with sufficient accuracy by a particular linear regression model. It is just that one has a dataset that includes not only the correct regressors but some incorrect ones, and the data analyst does not know which is which. The proper kind of shrinkage will reveal which regressors belong in the model. Alternatively, one has precisely the correct predictors in the dataset, but better performing estimates might be obtained through regularization. In either case, however, statistical inference for the lasso suffers from the same complications as ridge regression. Conventional expressions for confidence intervals and hypothesis tests do not apply.

The Elastic Net

If an important goal of a regression data analysis is to reduce the complexity of the model, the lasso has some advantages over ridge regression. But the lasso can also run into problems (Zou and Hastie, 2005). For example, when the number of predictors is larger than the number of observations (which is common with microarray data), the number of predictors selected cannot exceed the number of observations. In addition, there are the problems already noted with the selection of some inappropriate predictors.

In response to these difficulties, Zou and Hastie (2005) combine the penalties from ridge regression and the lasso. The result, called the elastic net, is

$$\hat{\beta} = \min_{\beta} \left[\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2 \right]. \qquad (2.9)$$

Minimization of Equation 2.9 produces what Zou and Hastie (2005) call "naive" coefficients that need to be adjusted further. The adjustment is simple, and some initial applications and simulations suggest that the elastic net can improve on the lasso. Of course, the regression model still needs to be credible, and even if it is, conventional expressions for confidence intervals and hypothesis tests are inappropriate.

The Dantzig Selector

The Dantzig Selector is another shrinkage estimator that can be used for variable selection (Candes and Tao, 2007). It seems to perform especially well when the number of predictors is large relative to the number of observations and even allows for the number of predictors to be larger than the number of observations. One must assume that the true set of regression coefficients is "sufficiently sparse" so that a substantial number of predictors actually have regression coefficients of zero. In effect, this guarantees identifiability. If in practice the linear regression model specified satisfies all of the usual assumptions, save for including a relatively large number of unnecessary predictors, the Dantzig Selector can find the predictors with regression coefficients equal to zero.

The Dantzig Selector has the following formulation.

$$\hat{\beta} = \min_{\beta} \sum_{j=1}^{p} |\beta_j| \text{ subject to } \sum_{i=1}^{n} |x_{ij} r_i| < \lambda \tag{2.10}$$

for $j = 1, 2, \ldots, p$, where the predictors have been standardized to z-scores, r_i is the usual regression residual, λ is a tuning parameter, and p is the number of predictors.

The Dantzig Selector, like the lasso, uses the sum of the absolute values of the regression coefficients as an argument. Minimizing the sum of the absolute values of the regression coefficients can produce regression coefficients that are exactly zero, and therefore, the associated predictors are removed from the analysis. But the key idea is that $\sum_{i=1}^{n} |x_{ij} r_i|$ captures any association between the residuals and each predictor in turn. When for each predictor $\sum_{i=1}^{n} |x_{ij} r_i| = 0$, one has the usual least squares solution in which by construction, the predictors are unrelated to the residuals. With $\sum_{i=1}^{n} |x_{ij} r_i| > 0$, bias is introduced because one or more predictors is associated with the residuals. By setting the value of λ, one can introduce varying degrees of association between each predictor and the residuals, and varying degrees of bias in the estimated regression parameters.

Work by Gareth and Radchenko (2007) extends applications of the Dantzig Selector to the entire generalized linear model. It may also be a useful tool when applied to functional linear regression (Gareth and Zhu, 2007). An important insight is that the Dantzig Selector can be formulated within a maximum likelihood framework such that the tuning parameter allows the partial

derivatives of the likelihood function with respect to the regression coefficients to be nonzero. Consequently, the solution is moved away from the maximum likelihood result. As before, some bias is introduced that can shrink the appropriate regression coefficients to zero.

To date, hands-on experience with the Dantzig Selector is very limited and it is not clear how the Dantzig Selector performs compared to obvious competitors such as the lasso (Efron et al., 2007; Meinshausen, 2007) In addition, potential insights into statistical learning have yet to be well explored (Cai and Lv, 2007). However, the ideas built into the Dantzig Selector are provocative, and it may have a bright future.

Regularization and Derivative Expectation Operator: Rodeo

Rodeo is perhaps the most recent entry into the shrinkage sweepstakes (Lafferty and Wasserman, 2008). It is related to adaptive smoothing, which is discussed later in this chapter, as a result, and to the lasso. The goal is to apply shrinkage to nonparametric regression, also discussed shortly, so that irrelevant predictors can be identified and removed. Rodeo assumes, as before, that one has in the data all of the correct regressors and some additional ones. The irrelevant predictors make the full set of predictors "sparse."

It is difficult to be very specific before nonparametric regression is more fully discussed, but the basic approach can be easily described. Suppose there is a single predictor. The degree of smoothness for the computed $f(X)$ is varied starting with a very smooth $f(X)$ and gradually making it more rough. If on the average the fitted values are much the same regardless of the degree of smoothing, that predictor is not meaningfully related to the response. Smooth, rough or in between, the $f(X)$ does not change significantly. Conversely, if on the average the fitted values vary substantially as the degree of smoothness changes, the predictor is meaningfully related to the response. The degree of smoothness matters for the $f(X)$.

Now imagine having p predictors. If changing the degree of smoothing has little impact on the average fitted value for a given predictor, one can conclude that that predictor is not relevant. If changing the degree of smoothing has a large impact on the average fitted value for a given predictor, one can conclude that that predictor is relevant.

As a practical matter, rodeo begins with a very smooth version of the $f(X)$. Gradually, the $f(X)$ is made less smooth for each predictor. For any predictor and given amount of smoothness, there is an aggregate derivative over observations representing how much the $f(X)$ changes with infinitesimal changes in the amount of smoothing. When for any predictor the derivative is smaller than some threshold for that predictor, the predictor is deleted from the model. Ideally, the irrelevant predictors are deleted first leaving behind the relevant predictors.

It is far too early to know how effective rodeo will be in practice. More important for now is its conceptual structure. All of the shrinkage proce-

dures considered thus far place constraints on regression coefficients, which as derivatives represent how the average response changes as a function of an infinitesimal change in predictor values. Rodeo places constraints on how much the average response changes with infinitesimal changes in the amount of smoothing. More generally, rodeo addresses shrinkage for nonparametric regression. This provides a useful transition to smoothing splines, which are addressed shortly.

2.3.2 Shrinkage and Statistical Inference

If the data used in a shrinkage procedure have been generated by random sampling or by a known stochastic process, statistical inference may be called for. As mentioned earlier, however, shrinkage estimates present special problems. Even if the regression model meets the requisite assumptions, shrinkage introduces bias by design. If the regression estimates are biased, conventional confidence intervals will not have their advertised coverage. For example, the 95% confidence interval for a particular regressioncoefficient will not contain the true value for that regression coefficient 95% of the time. The estimate is offset by some unknown amount so that the actual coverage will be less than 95%. Similar problems exist for hypothesis tests. The disparity between the null hypothesis and the sample statistic will be either too large or too small because of the offset caused by the bias. As a result, the computed p-values will be too large or too small as well.

Recall that the traditional goal of shrinkage is to construct sample estimates as close as possible to their population counterparts by the mean squared error criterion. The uncertainty estimates, therefore, risk confounding the variance with the bias. This can mean that a sensible confidence interval needs to take both into account if the usual coverage is to be represented. Likewise, sensible tests need to produce p-values that respond to both.

Because the nature of the bias is unknown, there is no easy fix. All one can know for sure is that the conventional procedures by which one constructs confidence intervals or performs hypothesis tests will be incorrect and that statistical inference reported by the regression software is likely to be incorrect as well.

In some settings, it can be prudent to reduce aspirations. One can focus on the variance alone. If the question solely is how much instability there is in the estimates of the regression coefficients or the fitted values, the bias is not longer formally relevant (Buja and Rolke, 2007). It follows that bootstrap samples of the observations (i.e., of Y and of X) can be used, much as with the simple percentile method, to construct useful intervals, which are in theory covering properly the values of the population parameters shifted up or down by the shrinkage. The target is no longer the "truth."

There also seems to be the prospect of useful alternative procedures based on Stein estimators and empirical Bayes methods (Carlin and Louis, 1996). The basic idea is that if one computes the conditional mean for a small region

defined by predictor values, that estimate will likely be reasonably unbiased with respect to the true conditional mean in that region. The difference between that value and the average of the shrunken fitted values in the region can provide a useful approximation of the direction and size of the bias. That approximation can then be combined with an estimate of the variance to construct improved confidence intervals and tests (Brown et al., 2005). To date, this approach has only been developed for single predictors, but extensions to multiple predictors seem possible.

In summary, statistical inference for shrinkage estimates is largely an unsolved problem. At the very least, there seems to be no consensus on how best to undertake statistical inference on shrinkage estimates when there is a need to consider the impact of the bias. Similar issues can arise for a number of the procedures considered in later chapters.

2.3.3 Shrinkage: So What?

When shrinkage is applied to conventional regression estimates there can be, as noted earlier, two goals. First, one might be interested in model selection. The lasso and the elastic net can provide useful alternatives to conventional model selection procedures, such as nested statistical tests, if their assumptions are approximately met. Shrinkage is used to select the regressors and then a conventional regression equation is estimated. However, problems discussed earlier about postmodel selection statistical inference remain, and there is never any guarantee, regardless of the method, that the model selected will make scientific or policy sense. There is no necessary correspondence between the statistical criteria and good science or good policy. The models that result should be seen as highly provisional.

Second, one might be interested in striking a good balance between the bias and the variance; the problem is not model selection in the usual sense. Then, whether ridge regression, the lasso, or the elastic net (or some other penalty formulation) should be strongly preferred is less clear. A lot depends on the properties of the data on hand (Zou and Hastie, 2005).

In short, shrinkage procedures at this point look to be primarily niche players in routine data analysis. They have some promise for model selection and for addressing the bias–variance tradeoff in conventional regression. The major reason why shrinkage has been discussed here is that imposing penalties on the fitting process to smooth the fitted values is more generally useful. In addition, the issues that shrinkage raises, and the concepts shrinkage introduces, play an important role in more advanced smoothers, and in procedures considered in later chapters. There are also some interesting applications that are beyond the scope of this book. For example, Zou, Hastie, and Tibshirani (2006) apply the elastic net to principal components analysis.

2.4 Smoothing Splines

For the spline-based procedures considered thus far, the number and location of knots had to be determined a priori or in the case of the number of knots, by some measure of fit. We now consider an alternative that does not require a priori knots. A key feature of this approach is to effectively saturate the predictor space with knots and then protect against overfitting by constraining the impact the knots can have on the fitted values. The influence that knots can have can be diluted; the initial number of knots does not have to change but the impact of some can be shrunk to zero. The key is a somewhat different kind of penalty for undue complexity.

We begin by requiring that for our single predictor and response variable, there is a function $f(X)$ with two derivatives over its entire surface. This is a common assumption in the statistical learning literature and in practice does not seem to be particularly restrictive. The goal is to minimize a "penalized" error sum of squares of the form

$$\text{RSS}(f, \lambda) = \sum_{i=1}^{N} [y_i - f(x_i)]^2 + \lambda \int [f''(t)]^2 dt, \tag{2.11}$$

where λ is, as before, a tuning parameter. The first term on the righ-hand side captures how close the fitted values are to the actual values of y. It is just the usual error sum of squares. The second imposes a cost for the complexity of the fit. The integral quantifies the roughness penalty, and λ determines the weight given to that penalty in the fitting process. At one extreme, as λ increases without limit, the fitted values approach the least squares line. Because no second derivatives are allowed, the fitted values are as smooth as they can be. At the other extreme, as λ decreases toward zero, the fitted values approach an interpolation of the values of the response variable.

Equation 2.11 addresses the bias–variance tradeoff head-on. When λ is larger, the fitted values are smoother, with the likely consequence of more bias and less variance. When λ is smaller, the fitted values are rougher with the likely consequence of less bias and more variance. Thus, the value of λ can be used in place of the number of knots to tune the bias–variance tradeoff.

For a given value of λ, Equation 2.11 can be minimized. Hastie et al. (2001: Section 5.4) explain that a unique solution results, based on a set of natural cubic splines with N knots. This assumes that there are N distinct values of x. There will be fewer knots if there are less than N distinct values of x.

It follows that

$$f(x) = \sum_{j=1}^{N} N_j(x)\theta_j, \tag{2.12}$$

where θ_j is a set of weights, $N_j(x)$ is an N-dimensional set of basis functions for the natural cubic splines being used, and j stands for the number of knots, of which there can be a maximum of N.

Consider the following toy example, in which x takes on values 0 to 1 in steps of .20. In this case, $j = 6$, and Equation 2.12, written as $f(x) = \mathbf{N}\theta$, takes the form of

$$f(x) = \begin{pmatrix} -.267 & 0 & 0 & -.214 & .652 & -.429 \\ .591 & .167 & 0 & -.061 & .182 & -.121 \\ .158 & .667 & .167 & -.006 & .019 & -.012 \\ 0 & .167 & 0.667 & .155 & .036 & -.024 \\ 0 & 0 & .167 & .596 & .214 & .024 \\ 0 & 0 & 0 & -.143 & .429 & .714 \end{pmatrix} \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_2 \\ \theta_4 \\ \theta_5 \\ \theta_6 \end{pmatrix}. \tag{2.13}$$

Equation 2.11 can be rewritten using a natural cubic spline basis and then the solution becomes

$$\hat{\theta} = (\mathbf{N}^T\mathbf{N} + \lambda\boldsymbol{\Omega}_N)^{-1}\mathbf{N}^T\mathbf{y}, \tag{2.14}$$

with $[\boldsymbol{\Omega}_N]_{ij} = \int N_j''(t)N_k''(t)dt$, where the second derivatives are for the function that transforms x into its natural cubic spline basis. $[\boldsymbol{\Omega}_N]$ has larger values where the predictor is rougher, and given the linear estimator, this is where the fitted values are rougher as well. The penalty is the same as in Equation 2.11.

Equation 2.14 can be seen as a generalized form of ridge regression. With ridge regression, for instance, $[\boldsymbol{\Omega}_N]$ is an identity matrix. In practice, N is replaced by a basis of B-splines that is used to compute the natural cubic splines.

The requirement of N knots may seem odd because it appears to imply that N degrees of freedom are used up. However, for values of λ greater than zero, the resulting smoother is shrunk toward a linear fit. In other words, whenever the penalty for complexity comes into play, it makes the fitted values more smooth, and in so doing, reduces the number of degrees of freedom actually used up. Larger values of λ mean that fewer degrees of freedom are lost.

As with the number of knots, the value of λ can be determined a priori or through model selection procedures. One common approach is based on N-fold (drop-one) cross-validation, briefly discussed in the last chapter. The value of λ is chosen so that

$$\mathrm{CV}(\hat{f}_\lambda) = \sum_{i=1}^{N}[y_i - \hat{f}_i^{(-i)}(x_i)]^2 \tag{2.15}$$

is as small as possible. Recall that $\hat{f}_i^{(-i)}(x_i)$ is the fitted value with case i removed. Using the CV to select λ is one automated way to find a promising balance between the bias and the variance in the fitted values. However, all of the earlier caveats apply.

2.4.1 An Illustration

To help fix all these ideas, we turn to an application of smoothing splines. Figure 2.9 shows a smoothed scatterplot based on equations 2.11 and 2.15. The data come from seven Japanese cities from 1973 through 1999. The response variable is residential water use in 1000s of cubic feet. The predictor is population size. The standard thinking about water consumption is that it increases linearly with population.

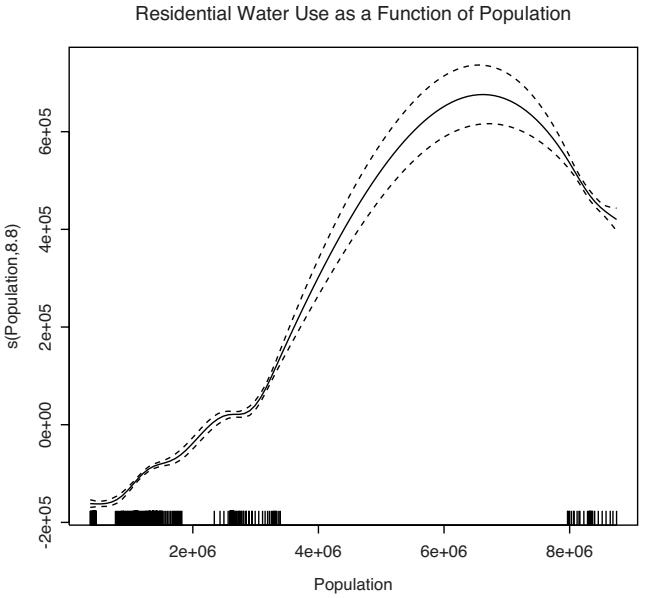

Fig. 2.9. An application of penalized regression splines.

In Figure 2.9, population is on the horizontal axis, and the fitted values are on the vertical axis. The smoother is represented by the solid line, and a point-by-point 95% confidence interval by the broken line, assuming that estimation is at least in principle justified. If $\mathbf{S}$ is the smoother matrix, the covariance of $\hat{f}(x) = \mathbf{S}\mathbf{S}^T\sigma^2$. With σ^2 estimated by the error sum of squares divided by $N - tr(\mathbf{S})$, the main diagonal of $\text{cov}[\hat{f}(x)]$ contains point-by-point estimates of the error variance. Then, with Gaussian errors and negligible bias, plus or minus twice the square root of the variances can be viewed as a point-by-point 95% confidence interval. (Hastie and Tibshirani, 1990: Section 3.8). We consider statistical inference for smoothers in more depth shortly.

The rug plot at the bottom of the plot shows where the population data tend to be located. One implication is that there are no data over the range where the curve starts to bend downward. As one would expect, the confidence

interval widens substantially around the large bend in the fitted values because there are very little data providing support. Although this makes good sense, we consider below why it would be risky to treat the band plotted in Figure 2.9 as a 95% confidence interval.

Figure 2.9 shows a positive relationship for the smaller population centers that is approximately linear, and a negative relationship for larger population centers that is also approximately linear. The latter results from factors in the biggest cities, such as affluence and the use of water-efficient technology, that are not considered when population is the sole predictor (Berk and Rothenberg, 2004).

Figure 2.9 was constructed with the gam() procedure in the *mgcv* library. The symbol "s" in the label on the vertical axis means that a smoother has been applied. In this case, the smoother is based on penalized regression splines of the sort just discussed with the value of λ determined by the GCV statistic. The "8.8" in the label is the effective degrees of freedom (or the equivalent number of parameters) used up by the smoother. Clearly, 8.8 is a lot smaller than the number of observations, but some distance from 1.0. The result is a rather smooth function that is substantially nonlinear. One degree of freedom would have been used up had a linear smooth materialized. With a greater effective degrees of freedom, the fitted values are less smooth.

2.5 Locally Weighted Regression as a Smoother

2.5.1 Nearest Neighbor Methods

Thus far, the discussion of smoothing has been built upon a foundation of conventional linear regression. Another approach to smoothing is from the perspective of nearest neighbor methods. Consider Figure 2.10 in which the shaded ellipse represents a scatter of points for values for x and y.

There is a target value of x, labeled x_0, for which a conditional mean $\bar{y}_0$ is to be computed. There may be only one such value of x or a relatively small number of such values. As a result, a conditional mean computed from those values alone risks being very unstable. One possible solution is to compute $\bar{y}_0$ from observations with values of x close to x_0. The rectangle overlaid on the scatterplot illustrates a region of "nearest neighbors" that might be used. Insofar as the conditional means for x are not changing systematically within that region, a useful value for $\bar{y}_0$ can be obtained. If that conditional mean is to be used as an estimate, it will be unbiased and likely be more stable than the conditional mean estimated only for the observations with $x = x_0$. In practice, however, some bias is often introduced. As before, the hope is that the increase in the bias is small compared to the decrease in the variance.

A key issue is how the nearest neighbors are defined. One option is to take the k closest observations using the metric of x. For example, if x is age, x_0 is 24 years old, and k is 10, the ten closest x-values may range from 23

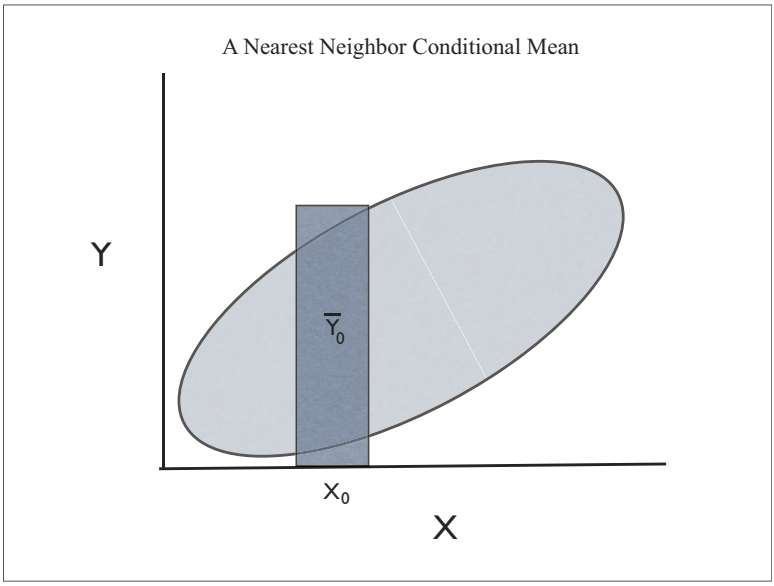

Fig. 2.10. A conditional mean for a target value of X.

to 27 years of age. Another option is take the some fixed fraction f of the observations that are closest to x_0. For example, if the closest 25% of the observations were taken, k might turn out to be 31, and the age-values might range between 21 and 29. Yet another option is to vary either k or f depending on the variability in y within a neighborhood. If there is more heterogeneity that is likely to be noise, larger values of k or f can be desirable to improve stability. Note that for any of these approaches, the neighborhoods will likely overlap. For another target value near x_0, some near neighbors will likely be in both neighborhoods. There also is no requirement that the neighborhood be symmetric around x_0.

Suppose now that for each unique value of x a nearest neighbor conditional mean for y is computed using one of the approaches just summarized. Figure 2.11 shows a set of such means connected by straight lines. The pattern provides a visualization of how the means of y vary with x. As such, the nearest neighbor methods can been seen as a smoother.

Figure 2.11 will change depending on the size of the neighborhood. Larger neighborhoods will tend to make the smoothed values less variable. If the smoothed values are to be treated as estimates, they will likely be more biased and more stable. Smaller neighborhoods will tend to make the smoothed values more variable. If the smoothed values are to be treated as estimates, they will likely be less biased and less stable.

Nearest neighbor methods can be very effective in practice and have been elaborated in many ways. There can be more than one predictor, for example,

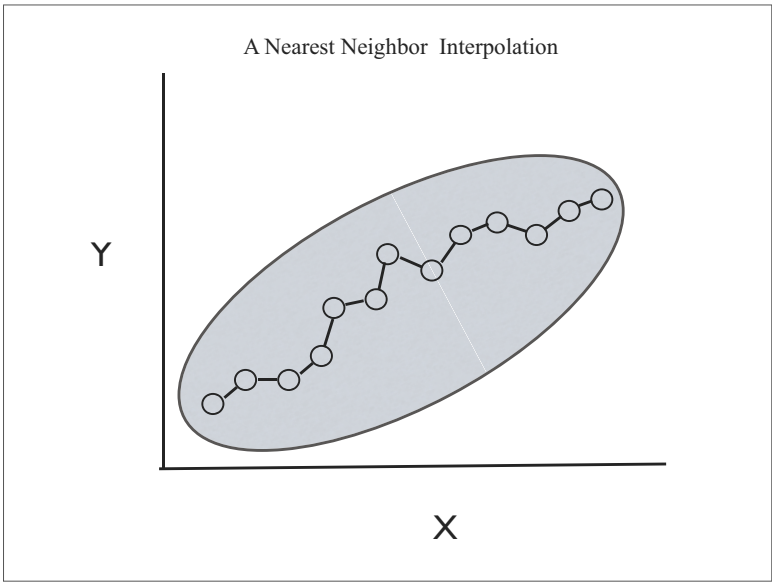

Fig. 2.11. Interpolated conditional means.

which raises some difficult issues about how to best define the neighborhood (e.g., Hastie and Tibshirani, 1996). This is a matter to which we return.

For our purposes, perhaps the major weakness of nearest neighbor methods is they are not derived as a way to represent how Y is related to X; they are not explicitly linked to some $f(X)$. One consequence is that when there are more than two predictors, there is little guidance on how to represent the manner in which the predictors are related to the response.

Nevertheless, nearest neighbor methods introduce some very important issues and procedures that figure significantly in this and later chapters. Indeed, the line between nearest neighbor methods and a number of other techniques can be pretty fuzzy. Readers interested in learning more about nearest neighbor methods should consult Ripley (1996) and Shakhnarovich (2006).

What if within each neighborhood the conditional means of y vary systematically? At the very least, there is information being ignored that could improve the estimate of $\bar{y}_0$. Just as in conventional linear regression, if y is related to x in a systematic fashion, there can be less variation in the regression residuals than around the neighborhood mean of y. More stable estimates can follow. The idea of applying linear regression within each neighborhood leads directly to a smoothing procedure known as locally weighted regression.

2.5.2 Locally Weighted Regression

Although spline smoothers are widely used, lowess (Cleveland, 1979) is a useful alternative that was developed before penalized regression smoothers. It is

comparatively easy to understand and remains a very handy tool. Lowess also has a more "algorithmic" feel than penalized regression smoothers and is, therefore, a useful didactic device for the material that follows. Lowess stands for "Locally Weighted Scatterplot Smoothing," although there seem to be a number of translations of "lowess."

We stick with the one predictor case a bit longer. For any given value of the predictor x_0, a polynomial regression is constructed only from observations with x-values that are nearest neighbors of x_0. Among these, observations with x-values closer to x_0 are weighted more heavily. Then, $\hat{y}_0$ is computed from the fitted regression and used as the smoothed value of the response y at x_0. The process is repeated for all other values of x. Although the polynomial is often of degree one (linear), quadratic and cubic polynomials are also used. It is not clear that much is gained in practice using the quadratic or cubic form. In some implementations, one can also employ a degree zero polynomial, in which case no regression is computed, and the conditional mean of y in the neighborhood is used as $\hat{y}_0$. This is the nearest neighbor approach discussed above except for the use of distance weighting.

The precise weight given to each observation depends on the weighting function employed. The normal distribution is one option. That is, the weights form a bell-shaped curve centered on x_0 that declines with distance from x_0. The tricube is another option. Differences between x_0 and each value of x in the window are divided by the length of the window along x. This standardizes the differences. Then the differences are transformed as $(1 - |z|^3)^3$, where z is the standardized difference. Values of x outside the window are given weights of 0.0. As an empirical matter, most of the common weighting functions give about the same results.

As discussed for nearest neighbor methods, the amount of smoothing depends on the proportion of the total number of observations used when each local regression line is constructed. Proportions between .25 and .75 are common. The proportion has been given various names in the smoothing literature; "window" or "span" or "bandwidth" are all used. The larger the proportion of observations included, the smoother are the fitted values. The bandwidth plays the same role as the number of knots in regression splines or λ in smoothing splines. Some software also permits the bandwidth to be chosen in the units of the regressor. For example, if the predictor is population size, the span might be defined as 10,000 people wide.

More formally, each local regression at each x_0 is constructed by minimizing the weighted sum of squares with respect to the intercept and slope for the $M \leq N$ observations included in the window. Thus,

$$\text{RSS}^*(\boldsymbol{\beta}) = (\mathbf{y}^* - \mathbf{X}^*\boldsymbol{\beta})^T \mathbf{W}^* (\mathbf{y}^* - \mathbf{X}^*\boldsymbol{\beta}). \tag{2.16}$$

The asterisk indicates that only the observations in the window are included. The regressor matrix $\mathbf{X}^*$ can contain polynomial terms for the predictor should that be desired. $\mathbf{W}^*$ is a diagonal matrix conforming to $\mathbf{X}^*$, with

diagonal elements w_i^*, which are a function of distance from x_0. This is where the weighting-by-distance gets done. The algorithm then operates as follows.

1. Choose the smoothing parameter such as bandwidth, f, which is a proportion between 0 and 1.
2. Choose a point x_0 and from that the $(f \times N = M)$ nearest points on x.
3. For these M nearest neighbor points, compute a weighted least squares regression line for y on x.
4. Construct the fitted value $\hat{y}_0$ for that single x_0.
5. Repeat Steps 2 through 4 for each value of x. Near the boundary values of x, constraints are sometimes imposed much like those imposed on cubic splines and for the same reasons.
6. Connect these $\hat{y}$s with a line.

There is also a robust version of lowess. After the entire fitting process is completed, residuals are computed in the usual way. Weights are constructed from these residuals. Larger residuals are given smaller weights and smaller residuals larger weights. Using these weights, the fitting process is repeated. This, in turn, can be iterated until the fitted values do not change much (Cleveland, 1979) or until some predetermined number of iterations is reached (e.g., three). The basic idea is to make observations with very large residuals less important in the fitting.

Whether the "robustification" of lowess is useful will be application-specific and depend heavily on the window size chosen. Larger windows will tend to smooth the impact of outlier residuals. Equally important, because the scatterplot being smoothed is easily plotted and examined, it is usually easy to spot the possible impact of outlier residuals and if necessary, take them into account when the results are reported. In short, there is no automatic need for the robust version of lowess when there seem to be a few values of the response that perhaps distort the fit.

An Illustration

Figure 2.12 shows for a set of Japanese cities over 21 years a (nonrobust) lowess smooth of residential water consumption on the average price of water. Economic theory says the slope should be negative, other things being equal. It is difficult to make much of Figure 2.12. The window is set at .10 (10% of the data) so the fitted values are highly variable.

In Figure 2.13, the span is increased to .50 (50% of the data). Clearly, the result is a much smoother fit. In Figure 2.14, the span is still .50, but the fitting is based on an M-estimator (to "robustify" the fitted values), not conventional least squares. The change of the fitting function makes little difference in this example, and that seems to be a common outcome.

Figure 2.15 uses a span of .90 (90% of the data) and returns to the Gaussian weighting function. Clearly, this produces by far the smoothest fit. But which

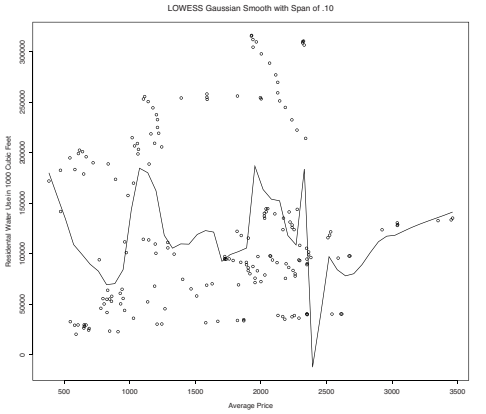

Fig. 2.12. Lowess Gaussian smooth of water consumption on average price: span = .10.

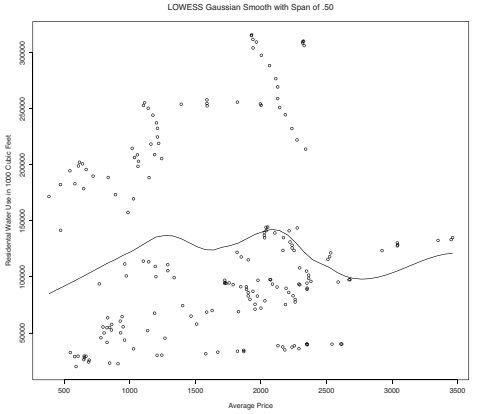

Fig. 2.13. Lowess Gaussian smooth of water consumption on average price: span = .50.

fit is best? The answer depends heavily on subject matter knowledge. In this case, one would anticipate a rather smooth, monotonically declining curve. All of the fitted values but the final set seem unduly variable and inconsistent with the way consumers should respond to price. Figure 2.15 is, therefore, probably the most informative.

However, the smoothed values are quite flat, with a slight upward trend followed by a slight downward trend. When water is relatively cheap, higher prices lead to more water consumption. When water is relatively expensive, higher prices lead to less water consumption. It is difficult to think of an explanation consistent with economic theory and more likely the positive seg-

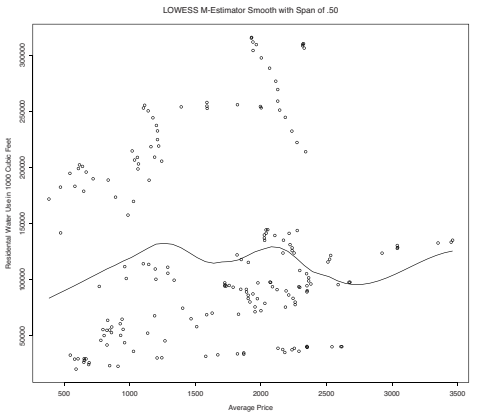

Fig. 2.14. Lowess M-estimator smooth of water consumption on average price: span = .50.

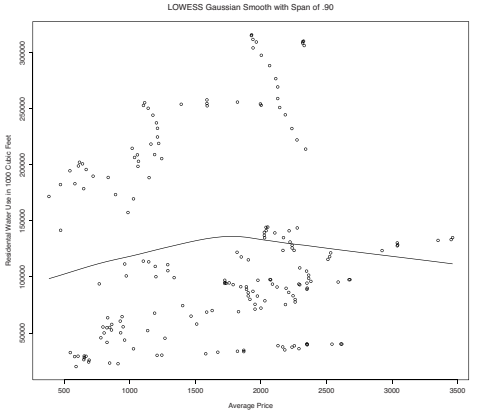

Fig. 2.15. Lowess Gaussian smooth of water consumption on average price: span = .90.

ment (at least) of the curve is an artifact caused by omitted predictors such as income. (For further discussion see Berk and Rothman, 2004.)

It may be important to underscore that even though the smoothed values in Figures 2.12 through 2.15 do not represent causal models, any interpretations resting on cause-and-effect claims need to consider many of the same issues that arise in conventional causal modeling. Omitted variables are surely one key concern. If the goal is description alone, then it is not even clear what an "omitted variable" is. The statistical definition requires that for a potential predictor to be an "omitted variable," it must be correlated with the response variable and any predictors already included in the analysis. But it is difficult to attach much import to the word "omitted" except in a causal context.

Perhaps the strongest statement that could be made is that the description is not complete.

In any case, the main message here is not meant to be substantive. The main message is that the bandwidth specified for lowess can make a big difference. The choice of bandwidth really matters. Because of this, there have been, in much the same spirit as the choice of λ in penalized regression, many attempts to automate and rationalize the selection of bandwidth size. For example, the generalized cross-validation statistic can be used to select the bandwidth (Loader, 2004: Section 4).

Such procedures can work well as a place to start. But once again, automation takes no notice of subject matter knowledge, and more useful visualizations are often produced when the choice of bandwidth is informed, at least in part, by information brought to the analysis from outside the data. It is doubtful that an automated procedure would have selected Figure 2.15. More likely, something close to Figure 2.13 would have been chosen. There is also the risk of overfitting, especially if a large number of bandwidths is tried.

2.6 Smoothers for Multiple Predictors

The last set of figures is only the most recent example in which the limitations of a single predictor were apparent. Many more things could be related to water consumption than price alone. The time has come to consider smoothers when there is more than one predictor.

In principle, it is a simple matter to include many predictors and then smooth a multidimensional space. However, there are three significant complications in practice. The first problem is the "curse of dimensionality." As the number of predictors increases, the space the data need to populate increases as a power function. Consequently, the demand for data increases very rapidly, and one risks data that are far too sparse to produce a meaningful fit. There are too few observations, or those observations are not spread around sufficiently to provide the support needed. One must, in effect, extrapolate into regions where there is little or no information. To be sensible, such extrapolations would depend on knowing the $f(X)$ quite well. But it is precisely because the $f(X)$ is unknown that smoothing is undertaken to begin with.

The second problem is that there are often conceptual complications associated with multiple predictors. In the case of lowess, for example, how is the neighborhood near x_0 to be defined (Fan and Gijbels, 1996: 299-300)? One option is to use Euclidian distance. But then the neighborhood will depend on the units in which predictors happen to be measured. The common practice of transforming the variables into standard deviation units does not really seem to solve the problem, especially when coupled with the need to weight observations by proximity to x_0.

Consider the case of two predictors. Suppose the standard deviation for one predictor is five years of age, and the standard deviation for the other predictor is two years of education. Now suppose one observation falls at x_0's value of education, but is five years of age higher than x_0. Suppose another observation falls at x_0's value for age, but is two years higher in eduction than x_0. Both are one standard deviation unit away from x_0 in Euclidian distance. But do we really want to say they are equally close?

Another approach to neighborhood definition is to use the same span for both predictors, but apply it separately in each direction. Why this is a better definition of a neighborhood is not clear. And one must still define a distance metric by which the observation in the neighborhood will be weighted.

Yet another alternative is to define a neighborhood by the importance of each dimension of the predictor space or a transformed predictor space. Where in that space the response is changing more rapidly, the neighborhood should be smaller. That way, significant variation in the fitted values is not smoothed away. We show later in this chapter that locally adaptive smoothers take a related approach. We learn that significant computation problems can follow.

The third problem is that gaining meaningful access to the results is no longer straightforward. When there are more than two predictors, one can no longer graph the fitted surface in the usual way. How does one make sense of a surface in more than three dimensions?

2.6.1 Smoothing in Two Dimensions

With only two predictors, there are some fairly straightforward extensions of conventional smoothers that can be instructive, even in the face of the three problems just discussed. For example, with smoothing splines, the penalized sum of squares in Equation 2.11 can be generalized. The solution is a set of "thin plate splines," and the results can be plotted. Thin plate splines are a two-dimensional generalization of the one-dimension cubic splines considered earlier. More specifically, Equation 2.11 can be generalized as

$$\min_f \sum_{i=1}^{N} \{y_i - f(x_i)\}^2 + \lambda J[f], \tag{2.17}$$

where J is an appropriate penalty functional of f. For the two-dimensional case,

$$J[f] = \int\int_{\Re^2} \left[\left(\frac{\partial^2 f(x)}{\partial x_1^2} \right)^2 + 2 \left(\frac{\partial^2 f(x)}{\partial x_1 x_2} \right)^2 + \left(\frac{\partial^2 f(x)}{\partial x_2^2} \right)^2 \right] dx_1 dx_2. \tag{2.18}$$

Equation 2.18 captures the roughness of the fitted values in a two-dimensional predictor space. The fitted values are rougher when the two second derivatives are larger. As before, the weight of this penalty is determined by the value of λ.

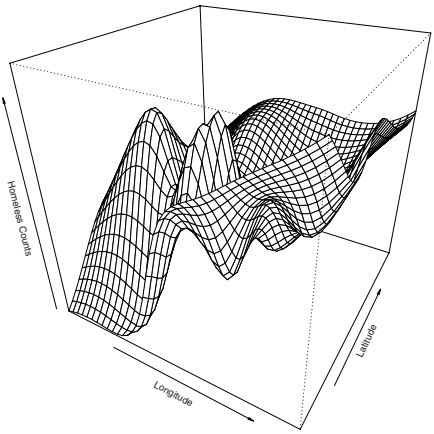

Fig. 2.16. Perspective plot of smoothed values of homelessness constructed from smoothing splines.

An Illustration

Figure 2.16 shows for a particular urban area a two predictor smooth of the homeless counts for census tracts by longitude and latitude. The value of λ was determined by the generalized cross-validation statistic. One can see that homelessness varies substantially by census tract. For example, the peak toward the middle of the plot is the downtown skid row area. The immediately surrounding areas have relatively low numbers of homeless individuals.

Figure 2.17 repeats the analysis with a two-predictor lowess smoother. The extension of lowess from one predictor to two proceeds as one would expect. A neighborhood and the within-neighborhood weighting are defined by Euclidian distance. Each neighborhood is now a solid rather than a plane so the local regression has two predictors rather than one. In this application, both predictors are in the same units, which makes the use of Euclidian distance far less controversial.

There is again a concentration of homeless in the skid row area, but now the spike is far more pronounced. It is difficult to determine precisely why the two plots differ. One possible explanation involves the manner in which the degree of smoothing is determined. For Figure 2.16, the value of λ was computed as part of the fitting algorithm. For Figure 2.17, the size of the span was determined in part by subject matter knowledge that made some results more credible than others. It is hard to know if the amount of smoothing in

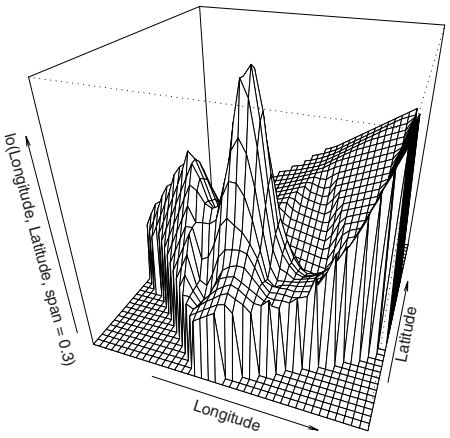

Fig. 2.17. Perspective plot of smoothed values of homelessness constructed from lowess.

the two figures is the same, but plots with somewhat different span values never eliminated the clear difference in the skid row effect.

The two smoothing procedures also differ substantially in their internal machinery. Smoothing splines, as do natural cubic splines, place a premium on fitted values that possess great continuity. Lowess does not build in such continuity so that sharp changes in direction can appear, especially when the span is small. Therefore, it would not be surprising to find that Figure 2.17 has a more jagged appearance. In short, a likely reason for the lower peak in homelessness for skid row in Figure 2.16 is that the sharp spike is rounded off.

Is there a way to choose between Figure 2.16 and Figure 2.17? The spatial patterns of homelessness show sharp differences over a distance of just a few blocks. Skid row, for example, is only two blocks from a cluster of modern, high rise office buildings where the number of homeless on the streets is very low. As a result, it is misleading to round off most of the transitions between areas with many homeless individuals and areas with few. Reality is closer to a two-dimensional step function. On these grounds, Figure 2.17 is probably a more accurate (if less elegant) visualization.

With more than two predictors, one generally needs another strategy. The data are often too sparse, and visualization is a major obstacle. The generalized additive model is one popular approach that meshes well with an

emphasis on regression and the use of a linear combination of basis functions.

2.6.2 The Generalized Additive Model

The Generalized Additive Model (GAM) is superficially an easy extension of the Generalized Linear Model (GLM). GAM tries to defeat the curse of dimensionality by assuming that the conditional mean of the response is a linear combination of functions of each predictor. Thus, the mean function for the generalized additive model with p predictors can be written as

$$\bar{Y}|X = \alpha + \sum_{j=1}^{p} f_j(X_j), \tag{2.19}$$

where α is a conventional intercept term.

In the same manner as the generalized linear model, the generalized additive model permits several different "link functions" and disturbance distributions. For example, with a binary response, the link function can be the log of the odds (the "logit") of the response, and the disturbance distribution can be logistic. This is analogous to logistic regression within the generalized linear model. But, there are no regression coefficients associated with the predictors. Regression coefficients would just scale up or scale down the functions of predictors, and so they are unnecessary. Whatever impact they would have is absorbed in the function itself. The role of the regression coefficients cannot be distinguished from the role of the transformation and therefore, the regression coefficients are not identified.

Each predictor can have its own functional relationship to the response. These functions can be estimated using single-predictor smoothers of the sort addressed above. Hence, the term nonparametric is usually applied despite the a priori commitment to an additive formulation. Alternatively, all of the functions may be specified in advance with the usual linear model as a special case. All of the common regression options are available, including the wide range of transformations one sees in practice: logs, polynomials, roots, product variables (for interaction effects), and indicator variables. As a result, GAM can be parametric as well and in this form is really no different from the generalized linear model. The parametric and nonparametric forms can be mixed so that some of the functions are derived empirically from the data, and some are specified in advance. Then the model is often called semiparametric.

With the additive form, one can use for GAM the same conception of "holding constant" that applies to conventional linear regression. The relationship between a given predictor and the response is constructed with (1) the linear dependence between the response and all other predictors removed, and (2) with the linear dependence between the given predictor and all other predictors removed. It is important to recall that the linear dependence removed is between the variables in whatever their transformed states happen to

be. Thus, there is no requirement of linear relationships between the variables in their original units.

More formally, consider for now the case of predictors x and z. Let

$$(\bar{y}|x, z) = \alpha + f_1(x) + f_2(z). \tag{2.20}$$

For now, assume that the $f_1(x)$ and $f_2(z)$ have been determined. For notational convenience $f_1(x)$ is denoted by x^*, and $f_2(z)$ is denoted by z^*. We focus first on $f_1(x)$.

Suppose that we estimate the regression parameters (i.e., intercepts and slopes) of the following two equations,

$$(\bar{y}|z^*) = \beta_0 + \alpha_1 z^*, \tag{2.21}$$

$$(\bar{x}^*|z^*) = \gamma_0 + \gamma_1 z^*. \tag{2.22}$$

For each, we compute the residuals $e_{y|z^*}$ and $e_{x^*|z^*}$. Finally, we estimate δ and $f_3(e_{x^*|z^*})$ in

$$(\bar{e}_{y|z^*}|e_{x^*|z^*}) = \delta + f_3(e_{x^*|z^*}). \tag{2.23}$$

The function $f_3(e_{x^*|z^*})$ should be identical to the function $f_1(x)$. A similar logic applies to $f_2(z)$. In other words, with the two functions determined, the usual covariance adjustments apply. "Holding constant" means to residualize precisely as Equations 2.20 through 2.23 specify. This is exactly the same logic that lies beneath added variable plots, sometimes called "partial plots" (Cook and Weisberg, 1999: Section 10.5).

But what if the transformations of all of the predictors are not known in advance? What if at least one of the functions (and usually several) is to be constructed empirically from the data? How does one estimate the function when the function needs to take the covariance adjustments into account? And one cannot apply the covariance adjustments unless the functions are known. To solve this problem, we turn to a computational algorithm called "backfitting."

A GAM Fitting Algorithm

The backfitting algorithm is a common way to estimate the functions and coefficients in Equation 2.19 (Hastie and Tibshirani, 1990: Section 4.4). The basic idea is to cycle through one function at a time in the following manner.

1. Initialize with $\alpha = \bar{y}, f_j = f_j^0, j = 1, \ldots, p$. Each predictor is given an initial functional relationship to the response such as a linear one. The intercept is given an initial value of the mean of y.
2. Repeat for $j = 1, \ldots, p, 1, \ldots, p, \ldots,$

$$f_k = S_j(y - \alpha - \sum_{j \neq k} f_j(x_k)). \tag{2.24}$$

A single predictor j is selected. Fitted values are constructed using all of the other predictors. These fitted values are subtracted from the response. A smoother S_j is applied to the resulting "residuals," taken to be a function of the single excluded predictor. The smoother updates the function for that predictor. Each of the other predictors is, in turn, subjected to the same process.

3. Continue Step 2 until the individual functions do not change.

Within any backfitting algorithm, a wide variety of smoothers can be applied and in the past have been. For example, both lowess and penalized regression splines have been available in R. Some procedures also permit the use of functions of two predictors at a time, so that the smoothed values represent a surface rather than a line, just as in Figures 2.16 and 2.17. That is, one can work with a linear combination of bivariate smoothed values.

In recent work (Wood, 2000, 2003, 2004), a somewhat different algorithm has been developed. The basic idea is to represent each of the functions to be determined empirically by a set of B-splines so that there is a single matrix of regressors for all of the unknown functions. This can then be combined with a regressor matrix for any terms whose functions are taken to be known a priori. The result is a multivariate generalization of penalized regression splines considered earlier when Equation 2.11 was discussed. Claims have been made that this approach has several advantages including more stable estimates, direct links to penalized fitting, and more straightforward extensions to conventional statistical inference. Whether any of these advantages would make much difference in practice is still to be determined.

The procedure gam() in R from the *mgcv* library is now implemented using this new algorithm. There are two GAM procedures in R, both called gam(). GAM using the traditional backfitting algorithm can be found in the R library *gam*.

An Illustration

We return now to the data used earlier on the possible deterrence impact of the death penalty. Recall that the data are a pooled cross-section time series of 50 states over 20 years. As before, the homicide rate is the response and the number of executions lagged by one year is a predictor. To control for the average differences between states, the homicide rate in each state, just before the beginning of each time series (1977), is used as a control variable. The multivariate penalized regression smoother just described is employed with the size of the penalty for each (λ) determined by the generalized cross-validation statistic.

The fit is excellent. About 90% of the variance is accounted for. Nearly all of this can be attributed to the values of the homicide rate when used as a control. Figure 2.18 shows the fitted values as a function of each predictor. If one ignores the very few values for the homicide rate that represent a very

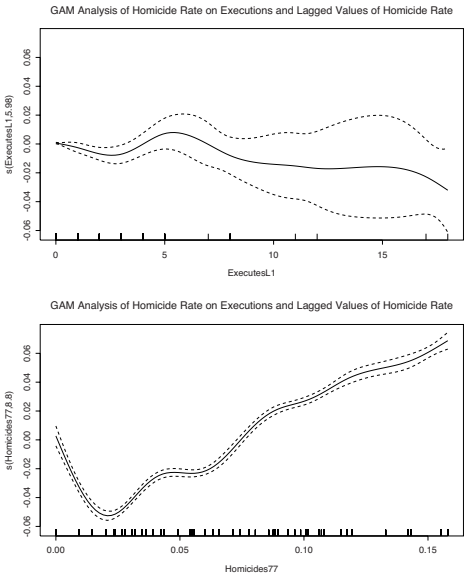

Fig. 2.18. GAM homicide results for executions controlling for the homicide rate in 1977.

few states over a very few years, the homicide rate over time is strongly and positively related to the homicide rate just before the beginning of the time series (i.e., Homicides77). Whatever the social processes in states that caused variation in the homicide rate in 1977, those same processes appear to persist over the next 20 years. The negative slope at the far left of the curve is much more difficult to understand and would need to be examined further. The likely explanation is the role of one or more predictors not included in the analysis.

The relationship between the number of executions lagged by one year (i.e., ExecutesL1) and the homicide rate is not strong overall. When there are five or fewer executions, which reflects 99% of the data, the relationship starts out being slightly negative and then turns more strongly positive. Net, the relationship is positive: more executions, more homicides a year later. The relationship when there are more than five executions, which reflects 1% of the data, is moderately negative. However, in part because there are so few observations, the nominal 95% point-by-point confidence interval (more on that shortly) is very wide and encloses a region that would easily allow for a flat or even positive slope. In short, there is no evidence whatsoever for deterrence for most of the states in most of the years, and evidence in favor of deterrence for the few outliers is not much stronger. (For more details see Berk, 2005a.)

One might wonder why there was no discussion of any regression coefficients. Recall that there are none. The fitted values for each predictor capture the average change in the response variable Y for small changes in a predictor. Because the fitted values have a nonlinear relationship with the response, there is not a single slope. In a sense, the graphs are the "slope." Or more formally, the derivative at any value of a predictor is the slope at that point.

Finally, both graphs in Figure 2.18 plot the fitted values centered on zero. This follows from the residualizing process described earlier. Recall that when there is an intercept, residuals have a mean of zero. Note also that the vertical axes have the same scales. This facilitates making comparisons between the response functions for different predictors.

2.7 Smoothers with Categorical Variables

As discussed in Chapter 1, smoothers can be used with categorical variables. When a predictor is categorical, however, there is really nothing to smooth. A binary predictor can take on only two values. The smoother is then just a straight line connecting the two conditional means of the response. For a predictor with more than two categories, there is no way to order the categories along the predictor axis. Any imposed order would imply assigning numbers to the categories. How the numbers were assigned could make an enormous difference in the resulting fitting values, and these assigned numbers would necessarily be arbitrary.

When the response is categorical and binary, smoothing can be a very useful procedure. All of the earlier benefits apply. In addition, because it is very difficult to see much in a scatterplot with a categorical response, a smoother may be the only way to gain some visual leverage on what may be going on.

2.7.1 An Illustration

We return now to the admissions data from a large public university. We apply GAM with admitted or not as the response. The predictors for each applicant are (1) mathematics SAT score, (2) verbal SAT score, (3) grade point average in high school, and (4) self-identified ethnic background. Figures 2.19 through 2.21 show the plots for the first three predictors. The residualized data are also shown. In each case, the vertical axis is in logits, just as it would be with logistic regression.

Figure 2.19 shows that beginning with SAT math scores of about 600 or higher, SAT math scores are positively, but modestly, related to the log-odds of admission. For lower math scores, there seems to be no relationship with the log-odds of admission.

Figure 2.20 shows that beginning with SAT verbal scores of about 450, SAT verbal scores are positively, but modestly, related to the log-odds of admission.

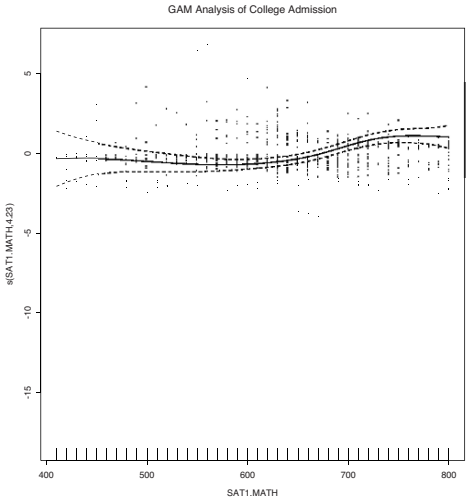

Fig. 2.19. Admission as a function of SAT math score.

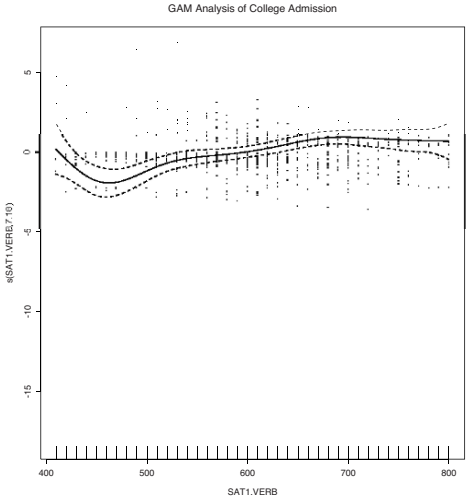

Fig. 2.20. Admisson as a function of SAT verbal score.

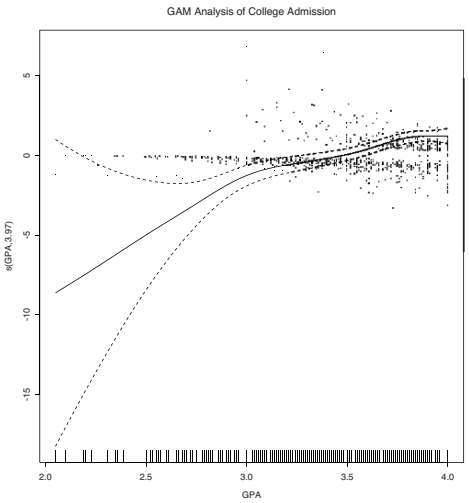

Fig. 2.21. Admission as a function of high school GPA.

The relationship seems to flatten out around a score of about 700 and even turn slightly negative. The negative relationship for SAT verbal scores below 450 is difficult to understand, but the data in that region are very sparse.

Figure 2.21 shows that over its full range, high school GPA is positively and strongly related to the log-odds of admission. It seems that the admissions process is weighting high school GPA far more heavily than SAT scores.

There are no plots for the categorical variable ethnicity. But for GAM, conventional regression coefficients are provided for all predictors whose functional forms are determined a priori. And that includes categorical variables. In this case, the regression coefficients reveal that holding constant SAT scores and high school GPA, the odds that an Anglo or Asian student will be admitted are substantially lower than for Hispanic and African-American students.

In short, there is certainly no lockstep relationship between earlier academic performance, as measured by SAT scores and high school GPA, and admission. Other factors are taken into account. This means that the apparent impact of ethnicity needs to be unpacked. Are there other predictors that would eliminate ethnicity as a useful explanatory variable? And if not, one cannot know without further study how an applicant's ethnicity comes into play. Is it directly used in the admission decision and/or is the impact really explained by characteristics of the applicant that are associated with ethnicity but not part of the official record?

As noted earlier, if the point is to explain why response functions come out as they do, causal thinking is often unavoidable. But there is nothing in any of the results that conveys what would happen if, for example, a given applicant's reported SAT scores were altered. To learn that, one would have to

actually alter the scores used in the admissions process. Such an experiment could perhaps be done if humansubject concerns could be overcome. What the analysis indicates thus far is that SAT scores, high school GPA, and ethnicity would probably need to be among the factors manipulated. And there would surely be others. The analysis also implies that if the purpose was to forecast admissions, SAT scores, high school GPA, and ethnicity might well provide considerable forecasting skill.

The statistical message is much the same as before. Allowing the data to determine how predictors are related to the response can be very instructive, even when (or perhaps especially when) the response variable is categorical. A natural question, however, is how restrictive the GAM's additive form is in practice. Experience to date suggests that the additive restriction is often not a serious obstacle. For instance, if there is an interest in interaction effects, these can be represented by a two-dimensional smoother (for two-way interactions) or by including product variables. This comes up often with spatial data, for example, where location is measured by variables such as longitude and latitude. If the response surface is substantially torqued, the additive terms are insufficient. One needs either a two-dimensional smoother or a product of the two spatial dimensions as another term in the model.

2.8 Locally Adaptive Smoothers

Under some circumstances, regression splines and regression smoothers can stumble when relationships with the response have sharp inflection points or steps. If all of the sharp inflection points or steps are about the same size, smoothing parameters can be set to either remove them all or to show them all. But when they are substantially different sizes, the risk is that some will be removed and some will not, even if all are equally informative.

One potential solution is to allow the smoothing parameters to vary locally so that they can adapt to sharp changes in the response function. For example, the bandwidth can be made smaller where the mean function seems to be changing most rapidly. This is very hard to do by hand without an impractical amount of trial and error. But there are a number of "locally adaptive" procedures that can automate the process. (Fan and Gijbels, 2006; Loader, 1999).

Figure 2.22 shows an example based on simulated data taken from Loader's instructive book (Loader, 1999). The simulated data have several telling features. First, the signal-to-noise ratio is very high. Second, the apparent pattern is far more variable in some regions than others. Third, the number of observations is relatively large. Finally, the mean function is extremely nonlinear. Taken together, these four features make it relatively easy to see what the mean function looks like without the aid of any smoother whatsoever.

The adaptive smoother that is overlaid clearly performs very well. A smoother with a single smoothing parameter would likely wash out the cycles

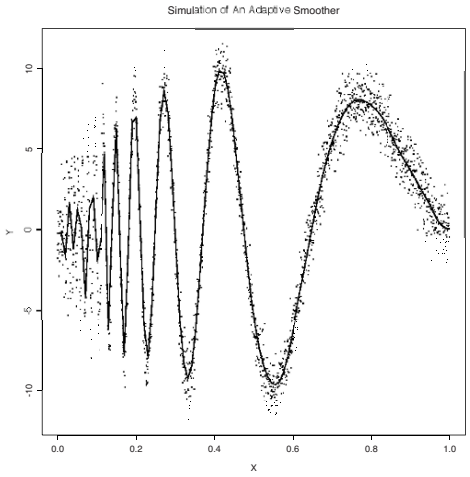

Fig. 2.22. Loader's illustration of a locally adaptive smoother.

at the far left while retaining those in the middle and far right. Important information would be lost. But how often will real data have the four properties that characterize these data? Many disciplines such as engineering have data in which such features may be common. But in the social and life sciences, these kinds of data are rare.

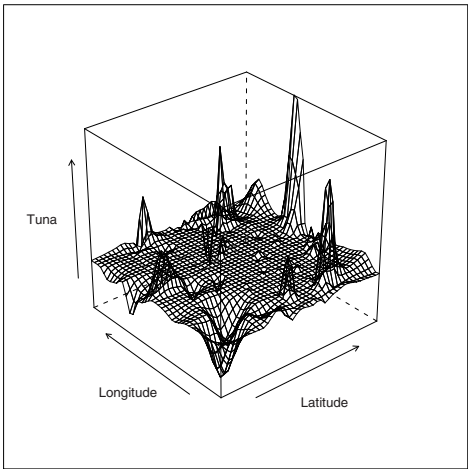

Fig. 2.23. Adaptive smoothed values of tuna caught in the southeastern Pacific.

Figure 2.23 presents some real data. The response is the number of tuna netted at various locations in the south eastern Pacific Ocean. The predictors

are longitude and latitude. Because of the flow of ocean currents and the clustering of smaller fish that tuna eat, the good fishing grounds have relatively sharp, nonlinear boundaries. Moreover the data are quite good. Because of international concern about the risks to dolphin when nets are used to catch tuna, there are official observers on tuna boats to record the size and locations of all catches. In short, the data themsleves are probably not seriously distorted by measurement error.

Once again the adaptive smoother does a good job. The productive fishing grounds are dramatically shown with some being far better than others. The smoothing process does not seem to be sacrificing the smaller spikes.

When the data are of just the sort required, adaptive smoothers can be very effective. Using them when they are not needed may not cause any scientific harm because adaptive procedures will, in effect, revert to a single smoothing parameter approach. But there can be significant computational costs. Existing software can seriously challenge the capacity of desktop computers, will usually require that several tuning parameters be specified, and are typically limited to no more than two predictors.

2.9 The Role of Statistical Inference

Many of the smoothers we have considered in this chapter rest upon a set of regression equations constructed for partitions of the data. The partitions are defined as functions of predictor values. Then, for any given partition, the fitted value is determined by a conventional parametric regression equation (sometimes with weights). Alternatively, the smoother results from a regression equation with a penalty attached for complexity. In either case, it might seem that conventional expressions for the standard error of fitted values would follow as usual. So, let's pursue that for a bit.

2.9.1 Some Apparent Prerequisites

A key issue that must be addressed before statistical inference with smoothers is undertaken is whether estimation itself is a reasonable activity. There are three scenarios.

1. There is an assumed $f(X)$, and the data are a random sample from a well-defined population or a random realization from a well-defined stochastic process. Estimation is at least on the table accompanied by assessments of uncertainty.
2. No $f(X)$ is assumed, but a goal is to arrive at a best guess of the values of a set of conditional means or proportions in a population or as features of a stochastic process. Estimation is again on the table along with assessment of uncertainty.

3. The sole goal is description of the data on hand. Estimation is taken off the table and with it, assessments of uncertainty.

We begin with the first scenario. We show that superficially all looks well. We also show, as the saying goes, that looks can be deceiving.

2.9.2 Confidence Intervals

As before we let

$$Y = f(X) + \varepsilon, \tag{2.25}$$

but where $\varepsilon \sim \text{NIID}(0, \sigma^2)$. One estimates $f(X)$ with $\hat{f}(X)$, which for the smoothers we have considered is $\mathbf{Sy}$. Then

$$\text{cov}(\hat{\text{f}}(\text{X})) = \mathbf{S}\,\text{cov}(\mathbf{y})\mathbf{S}^{\text{T}} = \mathbf{S}\mathbf{S}^{\text{T}}\sigma^2. \tag{2.26}$$

The square root of the diagonal elements of $\mathbf{S}\mathbf{S}^T\sigma^2$ are the standard errors for each fitted value. To make it operational, one needs $\hat{\sigma}^2$.

The error sum of squares can be computed as the sum of the squared differences between the fitted values and the observed values. The denominator is where there can be complications: what is one to use as the degrees of freedom lost to the fitting function? One popular definition, noted earlier, is the trace of the smoother matrix $\mathbf{S}$, which is related to the number of basis functions and to the number of parameters in the model (Hastie et al., 2001: 130). This definition is intuitively pleasing, broadly applicable to a variety of smoothers, and works well with hypothesis tests (considered shortly).

The larger the trace, the less smooth are the fitted values. This is because more relative weight is given to the values of the response variable actually being fitted and less relative weight is given to other (usually nearby) values of the response. Consider again the toy example from Chapter 1. Because the rows sum to 1.0, making the elements in the main diagonal larger makes the weights off the main diagonal smaller. The result is that the weighted average more heavily counts the value of the response being smoothed. The fitted values are relatively less smooth.

$$\begin{pmatrix} 1.0 & 0 & 0 & 0 & 0 \\ .25 & .50 & .25 & 0 & 0 \\ 0 & .25 & .50 & .25 & 0 \\ 0 & 0 & .25 & .50 & .25 \\ 0 & 0 & 0 & 0 & 1.0 \end{pmatrix} \begin{pmatrix} 3.0 \\ 5.0 \\ 6.0 \\ 9.0 \\ 10.0 \end{pmatrix} = \begin{pmatrix} 3.00 \\ 4.75 \\ 6.50 \\ 8.50 \\ 10.00 \end{pmatrix}. \tag{2.27}$$

With the effective degrees of freedom defined, $\hat{\sigma}^2$ is computed by dividing the error sum of squares by $N - \text{trace}(\mathbf{S})$. The denominator, in the same spirit as the usual regression estimate of $\hat{\sigma}^2$, represents the degrees of freedom "left over" by the model. Then $\hat{\sigma}^2$ is used in place of σ^2, making Equation 2.26 operational. Adding ± 1.96 times the square root of the diagonal to the fitted

value results leads to what looks like a 95% confidence interval at that point. All other points are treated in a similar fashion.

Because σ^2 is assumed to be constant, and because $N - \text{trace}(\mathbf{S})$ is a constant for any given dataset and model, the size of the standard error depends substantially upon the diagonal elements of $\mathbf{SS}^T$. These, in turn, depend on the diagonal elements in $\mathbf{S}$. So, larger estimated standard errors for a given dataset are found in regions where the fitted values are less smooth. And it is here that the fitted values may be a less effective stand-in for the values of the response variable. Taken at face value, this would seem to make sense.

However, there are several problems. First, the value of λ (or the analogous tuning parameter) is assumed to be known. When it needs to be determined from the data, there is an additional source of uncertainty that is not taken into account. Second, trying different possible values for λ is a form of data snooping and will often lead to estimates of uncertainty that are too optimistic. Third, unless the data were generated by probability sampling, the usual confidence intervals depend on model-based sampling, here, centered on how the values of ε are supposed to be generated (Thompson, 2002; Berk, 2003). Constructing a plausible story is usually very difficult, especially when the fitting function is to be inductively determined. Finally, the smoother tuned by λ is assumed to provide unbiased estimates of the true conditional means. In practice, this is very unlikely to be true. In particular, it will often be desirable to introduce bias to reduce the variance. And if there is bias, a 95% confidence interval will not cover the true value 95% of the time. The interval will be shifted higher or lower by the unknown value of the bias.

Recalling our earlier discussion of statistical inference for shrinkage estimators, one response can be to settle for estimates of the instability of the fitted values; the impact of the bias is ignored. Then to address instability, a bootstrap resampling of cases can lead to helpful results (Buja and Rolke, 2007) for such instability. But any intervals constructed in this manner are unlikely to be defensible as true confidence intervals.

If potential bias is to be addressed as well, there are some recent advances that have promise (Goldenshluger and Tsybakov, 2001; Zhang, 2005; Brown et al., 2005). Just as in the shrinkage case briefly addressed earlier, one can often obtain reasonably unbiased estimates of the true conditional means using the estimated conditional means for small regions of the predictor space. The disparity between those estimated conditional means and the conditional means produced by the smoother can provide important information on the direction and size of the bias in each region. When this information is combined with estimates of the variance, approximately correct confidence intervals can follow. There is not yet much formal mathematics behind these approaches and it is not clear at this point how well the procedures will perform in practice. There is also the current limitation to a single predictor.

The prospects might seem somewhat brighter under the second scenario: there is no $f(X)$, but there are population or stochastic process parameters, and the data were generated in a manner allowing for statistical inference, such

as random sampling. With no $f(X)$ to steer the analysis, interest centers only on a set of conditional means. If one treats the predictor values as fixed, this may seem like business as usual. However, the smoothing will likely introduce bias in the fitted values and the same problems surface. Confidence intervals risk being seriously misleading. In short, we are back to the previous scenario.

In practice, the third scenario is likely to be the operational one. There may be no credible $f(X)$, no population or stochastic process, or the data may be of insufficient quality (e.g., key predictors are missing). Then, the goal is likely to be description, and estimation is inappropriate. Whether any of these obstacles are recognized is often for subjectmatter experts to determine.

2.9.3 Statistical Tests

The statistical tests associated with conventional parametric regression have a structure that can be ported to the smoothers we have been considering. Consider the usual F-test used with the conventional linear regression model. Recall that the F-ratio is constructed in part from the error sum of squares under the null hypothesis and the error sum of squares from the alternative hypothesis, with their difference adjusted for the difference degrees of freedom. The ratio is meant to capture how much worse the fit becomes under the null hypothesis. The same kind of formulation can be applied with regression splines and regression smoothers.

Assume that Equation 2.25 holds. Then, drawing on Loader's discussion (2004: 17–18) — see also Hastie and Tibshirani (1990: 65–67) — suppose before looking at the data one decides that the null hypothesis is a conventional linear fit, and the alternative hypothesis is any smoother-based fit. Is the fit produced by the smoother "statistically significant" compared to the linear fit? This may not be a very interesting or instructive comparison, however, it comports well with conventional regression practice.

Let $\mathbf{H}$ be the hat matrix for a linear regression fit of the data, and $\mathbf{S}$ be the smoother matrix for some alternative fit. From this, one can construct the usual sort of test statistic as follows,

$$F = \frac{(\mathbf{Sy} - \mathbf{Hy})^2/df}{\hat{\sigma}^2}, \tag{2.28}$$

where $df = \text{trace}((\mathbf{S}-\mathbf{H})^T(\mathbf{S}-\mathbf{H}))$, and $\hat{\sigma}^2$ is usually estimated from the larger model. Loader points out that the F-ratio in Equation 2.28 does not quite have an F distribution, although there are ways to make the approximation better. Such tests are approximate and insofar as the assumed normality is incorrect, the test may not live up to its billing. Alternatively, the bootstrap can be applied. The idea is to work with the residuals in much the same manner as done for parametric regression (Efron and Tibshirani, 1993: 111–112).

1. Apply a smoother to the data.

2. Compute the residuals as the differences between the fitted values and observed values of the response variable (i.e., $\mathbf{y} - \hat{\mathbf{y}}$).
3. Draw with replacement a random sample of residuals the same size as the number of observations in the data.
4. Construct new values for the response by adding to each fitted value from Step 1, a sampled value from the residuals.
5. Compute the F-statistic of interest as in Equation 2.28. This will mean applying the null model and the alternative model to the reconstituted data.
6. Repeat Steps 2–5 a large number of times (e.g., 1000).
7. The histogram of the F-statistics provides an estimate of the true distribution of the F ratio.

But in the end, all such tests must be treated with caution. All of the concerns noted about confidence intervals apply. In particular, the smoothing process will typically lead to bias. If there is bias in the fitted values, the p-values computed will capture not just the variance but the bias. The distance between the null hypothesis and the estimated fitted values will be too large or too small, depending on the nature of the bias, and the p-values will be either too large or too small as well.

2.9.4 Can Asymptotics Help?

The asymptotics for the smoothers we have considered require that the number of observations must increase without limit and the number of unique values of the predictors (i.e., "design points") must increase without limit. That is, in order to obtain consistent estimates of the conditional means, both these conditions must apply. The number of design points must increase without limit so there are no "holes" in the fitted values. If there are holes, some form of interpolation or averaging is necessary, which means that the true conditional means in that hole will probably not be accurately represented.

In some very large datasets with relatively few predictors, these requirements may be approximately met. But for many datasets, the approximation to the requisite thought experiment is poor so that it is very difficult to rely on the asymptotic results. Equally important, if any smoothing is undertaken, there is the risk of nonnegligible bias that remains even asymptotically. To take an extreme case, if a linear fit is forced on a nonlinear $f(X)$, increasing the sample size does not overcome the bias introduced.

In short, even if estimation is a worthy goal, the associated statistical inference can be highly problematic. If one cares only about the stability of the fitted values, resampling procedures can be instructive. But if one cares about taking the bias into account, it is currently not clear how best to proceed. The good news is that statistical inference for smoothers is being addressed by some very talented statisticians. There may be some useful procedures soon.

2.10 Software Issues

All of the computing done is this chapter was implemented in R. Within R, the following smoothing and regression procedures were used.

1. Linear Regression: lm()—a very flexible and very powerful procedure for implementing the general linear model.
2. Generalized Linear Model: glm()—a very flexible and very powerful procedure for implementing the generalized linear model. Its structure is much like lm().
3. Scatterplot Smoothing: scatter.smooth()—a very flexibile and rich implementation of a two-dimensional lowess smoother of a scatter plot. The output is the scatter plot with the fitted values overlaid.
4. Local Adaptive Smoothing: locfit()—a generalization of lowess to allow for up to two predictors with local adaptation for bandwidth. The code is powerful and sophisticated, but the documentation is spotty. It can be found in the R library *locfit*.
5. Generalized Additive Model: gam()—it comes in two implementations. One can be found in the library (gam) and uses the backfitting algorithm. A second implementation uses penalized regression and can be found in the R library *mgcv*. They perform broadly the same, but differ a bit in the options offered to users.
6. Spline Basis Construction: bs(), ns()—two procedures that are used to construct b-spline bases for smoothers, bs() for B-splines and ns() for natural cubic splines. These are automatically called by some smoothing procedures or can be used as an intermediate step for more hand-tailored smoothing. They can be found in the R library *splines*.
7. Two-Dimensional Plotting: plot()—this can be used as a standalone or when paired with an R object produced by procedures such as lm(), locfit(), or gam().
8. Three-dimensional plotting: contour(), persp() for contour plotting and perspective plotting, respectively—both are slick and powerful, but a bit tricky to use. There is a need to construct the plotting grid before points and any fitted values are overlaid. Alternatively one can work with the graphing procedures in the library *lattice*. For example, wireframe() is a very elegant improvement over contour().

Perhaps the major operational problem for smoothing is sparse data. In the simplest case, there may be only a few distinct values for a predictor so that there is really nothing to smooth. For example, if a predictor only has observations at three of its values, there is not much that can be done. The choice is between no smoothing at all (i.e., just connecting the three conditional means of the response), or a single straight line. There is no clear lower limit to the number of predictor values for which there must be data, but smoothing when there are fewer than about ten values is not likely to be instructive. There can be few unique values for a predictor either because the

data are lumpy or because of how the predictor is defined and measured. For example, the number of children in a household for a given sample may have only five distinct integer values.

In addition, the curse of dimensionality can rapidly turn an adequate dataset into an inadequate one. The data may become far too thin overall, so that the large variance associated with the fitted values will negate any possibility of seeing what the mean function is likely to be. Or, important partitions of the data may suffer from the same problem. Most frustrating of all, some procedures will abort with sparse data, sometimes taking down the statistical procedures being used and even the entire operating system.

Many of the smoothing procedures have tuning parameters that can be used for taking relatively large bites of the data. For data that are potentially sparse, it will often be helpful to begin an analysis with large bites so that within each window there are a sufficient number of observations. If the fitted values seem stable, smaller bites may be tried.

A good sense of how stable the fitted values are can sometimes be obtained from a point-by-point confidence interval, as long as one does not take the attached probability very seriously. As noted earlier, bias will offset the intervals so that the coverage is unknown. But, if the point-by-point intervals are so large that the fitted values could plausibly range very widely, the fitted values do not provide a useful fix on the mean function. This is very important to take into account when the fitted values are interpreted.

2.11 Summary and Conclusions

Regression splines and regression smoothers can be very useful tools for describing relationships between a response variable and one or more predictors. As long as one is content to "merely" describe, these methods are consistent with the goals of an exploratory data analysis.

Experience suggests that for most datasets, it does not make a great difference which brand of smoother one uses. The dominant factor is usually bandwidth or other parameters that determine the bias–variance tradeoff. Likewise, all of the measures of fit that take model complexity into account lead to largely the same substantive results, especially when data are noisy.

There are also several important caveats that need to be kept in mind. First, as with any regression analysis, there is no necessary connection between the computer output and how the data were generated. There is, therefore, no necessary connection to causal inference. Although the output can be very helpful when considering matters of cause and effect, regression splines and regression smoothers are usually not meant to represent how manipulating one or more predictors will change the response.

Second, statistical inference should be approached with great care. Smoothers are often meant to be exploratory and as such can easily jeopardize formal tests and confidence intervals. Moreover, they typically introduce bias into

the fitted values with the goal of reducing their variance. It is also important to look beneath the computer output and understand how the statistical inference was undertaken.

Third, overfitting can be a serious problem. The results from the data examined may not generalize well to other random samples from the same population. We consider overfitting in depth in later chapters. For now, *caveat emptor*.

Finally, for a wide range of problems, there are statistical learning techniques that arguably perform better than the procedures discussed in this chapter. They can fit the data better, are less subject to overfitting, and permit a wider range of information to be brought to bear. One price, however, is that the links to conventional regression analysis become far more tenuous. In the next chapter, we start down this path.

Exercises

Problem Set 1: Smoothers with a Single Predictor

1. Load the dataset called airquality using the command data(airquality). Attach the data with the command attach(airquality). Use gam() from the *gam* library with Ozone as the response and Temp as the sole predictor. Estimate the following three models assigning the output of each to its own name (e.g., output1 for the first model).

   ```
   gam(Ozone ~ Temp)
   gam(Ozone ~ as.factor(Temp) )
   gam(Ozone ~ s(Temp) )
   ```

   The first model is the smoothest model possible. Why is that? The second model is the roughest model possible. Why is that? The third model is a compromise between the two in which the degree of smoothing is determined by the GCV statistic. (See the gam() documentation followed by the smoothing spline documentation.)

   For each model, examine the numerical output and plot the fitted values against the predictor. For example, if the results of the first model are assigned to the name "output1," use plot.gam (output1, residuals=TRUE).

   Which model has the best fit judging by the residual deviance? Which model has the best fit judging by the AIC? Why might the choice of the best model differ depending on which measure of fit is used? Which model seems to be most useful judging by the plots? Why is that?

2. Overlay a lowess smooth on a scatterplot with the variable Ozone on the vertical axis and the variable Temp on the horizontal axis. Vary three tuning parameters: span: .25, .50, .75; degree: 0, 1, 2; family as Gaussian or symmetric. Describe what happens to the fitted values as each tuning parameter is varied. Which tuning parameter seems to matter most?

3. The relationship between temperature and ozone concentrations should be positive and monotonic. From the question above, select a single set of tuning parameter values that produces a fit you like best. Explain why you like that fit best. If there are several sets of fitted values you like about equally, explain what it is about these fitted values that you like.

4. For the overlay of the fitted values you like best (or select a set from among those you like best) describe how temperature is related to ozone concentrations.

5. One can address the stability of the fitted values using the bootstrap percentile method. Load the library *simpleboot*. The procedure first requires that you run lowess and then you apply the bootstrap. For example: assign loess(Ozone $\sim$ Temp) to a name such as "smooth." Then assign loess.boot (smooth) to a name such as"bo." Finally use plot(bo). The point-by-point interval is constructed by taking the standard deviations of the fitted values for each point over bootstrap samples, multiplying each by two, and adding that product to the fitted values at each point and subtracting that product from the fitted values at each point.

   For what values of temperature does the instability appear to be about the largest? For what values of temperature does the instability appear to be the smallest? What in the data accounts for these differences?

Problem Set 2: Smoothers with Two Predictors

1. From the library *assist* load the dataset TXtemp. Load the library *gam*. With mmtemp as the response and longitude and latitude as the predictors, apply gam(). Construct the fitted values using the sum of a 1-D lowess smooth of longitude and a 1-D smooth of latitude. Try several different values for the degrees of freedom of each. Try different values for the degree of the polynomial. You can learn how to vary these tuning parameters with help(gam) and help(lo). Use the summary() command to examine the output and the plot.gam() to plot the two partial response functions. To get both plots on the same page use par(mfrow=c(1,1)). How are longitude and latitude related to temperature?

2. Repeat the analysis in 1, but now construct the fitted values using a single 2-D smoother of longitude and latitude together. Again, try several different values for the span and degree of the polynomial. Examine the tabular output with summary() and the plot using plot.gam(). How do these results compare to those using two 1-D predictor smooths?

Problem Set 3: Smoothers with More Than Two Predictors

1. Now build an additive model for mmtemp with the predictors longitude, latitude, year, and month. Use a lowess smooth for each. Try different spans and polynomial degrees. Again use the summary() and plot.gam() command. To get all four graphs on the same page use par(mfrow=c(2,2)). How is temperature related to each of the four predictors?

2. Repeat the analysis done for 1, but with penalized smoothing splines. The operator in front of each predictor is now s and not lo. Read the help documentation for gam(), and s(). How is temperature related to each of the four predictors? How do the conclusions from 1 compare with the conclusions drawn here? Why?

Problem Set 4: Smoothers with a Binary Response Variable

1. From the *car* library, load the dataset Mroz. Using the glm(), regress labor force participation on age, income, and the log of wages. From the library *gam*, use gam() to repeat the analysis, smoothing each of the predictors. Note that labor force participation is a binary variable. Compare and contrast your conclusions from the two sets of results. Which procedure seems more appropriate here? Why?