CHAPTER THREE

3. Nonparametric regression analysis

3.1 Introduction

In parametric regression of the form $y = f(x) + e$, where f is some known, smooth function, the modeler must determine the appropriate form of f. In nonparametric regression, f is some unknown, smooth function and is unspecified by the modeler. A data- driven technique determines the shape of the curve. Similar to parametric regression, a weighted sum of the y observations is used to obtain the fitted values. Instead of using equal weights as in ordinary least squares (OLS) or weights proportional to the inverse of variance as is often the case in weighted least squares (WLS), a different rationale determines the choice of weights in nonparametric regression.

In the single regressor case, the observations with the most information about $f(x_0)$ should be those at locations x_i closest to x_0. Therefore, a decreasing function of the distances of their locations x_i from x_0 determine the weights assigned to y_i's. The points closest to x_0 receive more weight than those more remote from x_0. Often, points remote from x_0 receive little or no weight.

This chapter describes three forms of nonparametric regression, and follows closely the excellent development of these topics presented by Mays (1995). Only single regressor cases are presented and used in this project.

3.2 Kernel regression

The goal of kernel regression is to obtain and use appropriate weights $w_{ij(\ker)}$ to yield fitted values via

$$\hat{y}_{i(\ker)} = \sum_{j=1}^{n} w_{ij(\ker)} y_j \tag{3.1}$$

Note that equation (3.1) can also be thought of as calculating $\hat{f}(x_i)$ for $i = 1, 2, ..., n$. Each of the n data points is assigned a distinct weight $w_{ij(\ker)}$, $j=1,2,\ldots, n$, for any point of fit x_i (or distinct weight $w_{0j(\ker)}$ for a point of prediction x_0). In matrix notation, equation (3.1) can be expressed as

$$\underline{\hat{y}}_{(\ker)} = \mathbf{W}_{(\ker)} \underline{y} \tag{3.2}$$

where $\mathbf{W}_{(\ker)} = (w_{ij(\ker)})$ is denoted as the kernel "hat" matrix (Mays 1995). Similar to OLS where the hat matrix is used to transform the y's to the $\hat{y}'s$, the kernel "hat" matrix is used to tranform the y's to the $\hat{y}_{(\ker)}'s$.

Nadaraya (1964, 1965) and Watson (1964), proposed one of the more common methods of determining the weights by defining:

$$w_{ij(\ker)} = K[(x_i - x_j)/h] \Big/ \sum_{j=1}^{n} K[(x_i - x_j)/h] \tag{3.3}$$

where $K(u)$ is a decreasing function of $|u|$, and $h>0$ is called the bandwidth. $K(u)$, the kernel function, may be taken to be a probability density function (such as a standard

Gaussian), a function defined to be zero outside of a certain range of u, or any other convenient form (Mays 1995). The kernel function should be symmetric. The bandwidth, h, is a smoothing parameter and will be discussed in greater detail shortly.

A closer examination of the numerator of equation (3.3) lends some insight into the weighting scenario, that is, more weight is associated with the observations at locations close to x_i (the fit location) and less weight to observations farther removed. The denominator serves to scale the rows of $\mathbf{W}_{(\text{ker})}$ to unity sum, as in OLS (Mays 1995).

The estimated regression curve achieved by obtaining predictions over the entire range of data should provide insight into the form of the underlying, yet unknown, function f. Whether it be the kernel form of nonparametric regression described in this section or those whose descriptions follow, it must be noted that no explicit closed form expression for f exists (Mays 1995).

Härdle (1990) concluded that it is the choice of bandwidth, and not the choice of kernel function, that is critical to performance of the nonparametric fit. Therefore, the simplified normal kernel function will be used in this application:

$$K(u) = \exp(-u^2) \tag{3.4}$$

The bandwidth, h, determines how fast the weights decrease as the distance from x_0 increases. The rate at which the weights decrease relative to the locations of the x_i's in turn controls the smoothness of the resulting estimate of f (Mays 1995)

Consider first the case where h is small (close to zero). The point of prediction itself possesses most of the weight with only the closest observations to this point

receiving the remainder of the weight (recall the weights do sum to unity). Under such a scenario, the resulting fit would essentially "connect the dots" formed by the observed data points and is said to be undersmoothed, or overfit, and possess high variance (Mays 1995). In other words, instead of obtaining a robust underlying fit for the process, different samples would yield much different fits due to sampling variability and the over-dependence of the fits on the respective individual data sets.

Now consider the other case where *h* is very large (equal to or close to equal to the entire range in *x*-values). Instead of concentrating the weights on a single point or handful of data, the weight is fairly evenly distributed across all the observations. Such a fit is considered oversmoothed, or underfit (with high bias) because it essentially fits the value $\bar{y}$ at each data point. (Mays 1995).

A numeric example illustrating the effect of altering bandwidth is provided in Appendix A3.1. The delicate balance between underfitting and overfitting is determined via bandwidth selection and is discussed in Section 3.4.

There is one pitfall inherent to kernel regression. Consider what occurs when *x* approaches a boundary of the data (left or right). The kernel weights can no longer be symmetric. To illustrate, consider the right boundary of the data. Specifically, consider the process of obtaining a prediction $\hat{y}_0$ at $x_{0,}$ where x_0 is at or near this right boundary. Only points to the left of x_0 are capable of receiving kernel weights (other than x_0 itself). There are simply no points to the right of x_0 to receive any weight (Mays 1995).

Now, if the data (and the true function *f*) are decreasing toward the right boundary (as is the case with tree crowns when the tree top is equivalent to the right boundary),

then all y-values in the weighted sum used to obtain $\hat{y}_0$ are, most likely, greater than or equal to the value y_0 at x_0. The bias at the boundary will result in a prediction, $\hat{y}_0$, that will be too high. (Mays 1995). Local polynomial regression is a form of nonparametric regression that addresses this boundary problem and will be discussed in the next section.

3.3 Local polynomial regression

Local polynomial regression (see Cleveland [1979] among others) uses weighted least squares (WLS) regression to fit a d^{th} degree polynomial ($d \neq 0$) to data. An initial kernel regression fit to the data is used to determine the weights assigned to the observations. Kernel regression, as described previously, is just a special form of local polynomial regression with $d = 0$. Hastie and Loader (1993) showed that local polynomial regression addresses the boundary problem present in kernel regression. Additionally, Mays (1995) noted local polynomial regression addresses the problem of potentially inflated bias and variance in the interior of the x's if the x's are non-uniform or if substantial curvature is present in the underlying, though undefined, regression function (a problem common to simpler weighting schemes such as kernel regression).

Consider fitting y_i at the point x_i. First, a kernel fit is obtained for the entire data set in order to obtain the kernel hat matrix $\mathbf{W}_{(\text{ker})}$,

$$\mathbf{W}_{(\text{ker})} = \begin{bmatrix} \underline{w}'_{1(\text{ker})} \\ \underline{w}'_{2(\text{ker})} \\ \vdots \\ \underline{w}'_{n(\text{ker})} \end{bmatrix} \tag{3.5}$$

where $\underline{w}'_{i(\ker)}$ is the i^{th} row of $\mathbf{W}_{(\ker)}$. As before and if desired, the kernel hat matrix can be used to obtain the fitted value for y_i at location x_i via

$$\hat{y}_{i(\ker)} = \sum_{j=1}^{n} w_{ij(\ker)}\, y_j = \underline{w}'_{i(\ker)}\, \underline{y} \tag{3.6}$$

where the $w_{ij(\ker)}$, j=1, 2, . . ., n are the n elements of the i^{th} row of $\mathbf{W}_{(\ker)}$. The kernel weights $w_{ij(\ker)}$ give weight to y_j based on the location x_j from x_i. With local polynomial regression though, the $w_{ij(\ker)}$, for a fixed i, become the weights to be used in weighted least squares regression. It is important to note that these distinct weights vary with changing i (Mays 1995).

Recall that the idea behind local polynomial regression is to use weighted least squares regression to fit a d^{th} order polynomial to observations close to x_i . The weights are defined to be the elements of the i^{th} row of $\mathbf{W}_{(\ker)}$, the initial kernel fit to the data. As described by Mays (1995), the diagonal weight matrix for local polynomial regression, used for fitting at x_i, can then be written as:

$$\mathbf{W}_{\text{diag}}(x_i) = \text{diag}(\underline{w}_{i(\ker)}) = \begin{bmatrix} w_{i1(\ker)} & 0 & \cdots & 0 \\ 0 & w_{i2(\ker)} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & w_{in(\ker)} \end{bmatrix} \tag{3.7}$$

Following the procedures of weighted least squares, the estimated coefficients for the local polynomial regression fit at x_i are then found via

$$\underline{\hat{\beta}}_{i(\text{LPR})} = (\mathbf{X}'\mathbf{W}_{\text{diag}}(x_i)\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}_{\text{diag}}(x_i)\underline{y} \tag{3.8}$$

where **X** is the **X** matrix from local polynomial regression determined by the degree *d* of the polynomial, with the i^{th} row defined as $\underline{x}'_i = (1, x_i, x_i^2, \ldots, x_i^d)$. Thus, provided $(\mathbf{X}'\mathbf{W}_{\text{diag}}(x_i)\mathbf{X})^{-1}$ exists, the fit at x_i is obtained as:

$$\hat{y}_{i(\text{LPR})} = \underline{x}'_i \underline{\hat{\beta}}_{i(\text{LPR})} = \underline{x}'_i (\mathbf{X}'\mathbf{W}_{(\text{diag})}(x_i)\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}_{(\text{diag})}(x_i)\underline{y} = \underline{w}'_{i(LPR)}\underline{y} \quad (3.9)$$

As Mays (1995) noted, the *n* fitted values can be expressed in matrix terms as $\underline{\hat{y}} = \mathbf{W}_{(\text{LPR})}\underline{y}$, where

$$\mathbf{W}_{(\text{LPR})} = \begin{bmatrix} \underline{w}'_{1(\text{LPR})} \\ \underline{w}'_{2(\text{LPR})} \\ \vdots \\ \underline{w}'_{3(\text{LPR})} \end{bmatrix} \quad (3.10)$$

Further development of this technique is provided in Cleveland (1979) and Hastie and Loader (1993). Authors generally agree that for the majority of cases, a first order fit (local linear regression) is an adequate choice for *d*. Local linear regression is suggested (Cleveland 1979) to balance computational ease with the flexibility to reproduce patterns that exist in the data. Nonetheless, local linear regression may fail to capture sharp curvature if present in the data structure (Mays 1995). In such cases, local quadratic regression (d=2) may be needed to provide an adequate fit.

Most authors agree there is usually no need for polynomials of order d>2 (Mays 1995). As a result, only first and second order local polynomial regression are used in this application.

3.4 Bandwidth choice

Selecting an appropriate bandwidth (smoothing parameter) is a key part of nonparametric regression fitting. In order to obtain a proper or "good" fit, the modeler must find a balance between the variance and bias (Mays 1995). Formulae for asymptotic bias and variance of a prediction when using the Nadaraya-Watson estimate of $w_{ij(\ker)}$, equation (3.3), are found in Chu and Marron (1991). These formulae illustrate why increasing the bandwidth increases bias and decreasing the bandwidth increases variance.

The need to balance bias and variance leads naturally to the minimization of a mean squared error criterion (or other global error criterion) as a logical starting point to determine what bandwidth to select for a given data set. For example, one may consider minimizing the prediction error:

$$SSE/n = (1/n)\sum_{j=1}^{n}[y_i - \hat{f}_h(x_j)]^2 \qquad (3.11)$$

Technically there is a weighting term, $\omega(x_i)$, included in the summation, but Härdle (1990) concluded this term had no significant effect on the choice of h. Therefore in this work it is assumed to take the constant value of one. Härdle (1990) demonstrated that using the criterion in equation (3.11) tends to overfit the data by choosing the smallest possible bandwidth.

Another possible bandwidth choice criterion, after removing dependence on n, would logically be cross validation suggested by Rudemo (1982) or PRESS (prediction sum of squares, Allen [1974]):

$$PRESS = \sum_{i=1}^{n} (y_i - \hat{y}_{i,-i})^2 \qquad (3.12)$$

where $\hat{y}_{i,-i}$ is the fit at x_i when ignoring the observation y_i in obtaining the fit. Again the weight in the summation is set to one. This too, tends to overfit the data by selecting a bandwidth too small (Mays 1995), suggesting the possible need for a penalizing function to protect against such small bandwidths. Härdle (1990) examined several penalizing functions to adjust the prediction error in equation (3.11). Each penalty function took the form $\Xi(n^{-1}h^{-1})$, resulting in bandwidth selection criteria of the form $(\text{SSE}/n)\Xi(n^{-1}h^{-1})$. He found that such functions worked well for guarding against overfitting, but did not protect against underfitting.

The criterion chosen for this application of nonparametric regression was developed by Einsporn (1987), and is called "penalized PRESS", denoted PRESS*. In this case, a penalizing function is applied to PRESS and not the usual prediction error. By merging the two procedures, Einsporn maintained the versatility of cross-validation (via the use of PRESS), while also introducing extra protection against overfitting (by using a penalizing function). PRESS* penalizes for small bandwidths by dividing PRESS by $[n - \text{tr}(\mathbf{W}_{(\text{ker})})]$, a different penalizing scheme than the traditional form described above:

$$PRESS^* = [n - \text{tr}(\mathbf{W}_{(\text{ker})})]^{-1} \sum_{i=1}^{n} (y_i - \hat{y}_{i,-i})^2 \qquad (3.13)$$

The $\omega(x_i)$ in the summation is again equated to one. Note that the kernel weight matrix is shown in the formulation of equation (3.13) and is used in the description to follow.

However, the weight matrix of any of the three nonparametric methods (kernel, local linear, and local quadratic) described herein, or any other weight matrix for that matter, could be used.

In order to appreciate this criterion, consider fitting at x_i. As the bandwidth gets smaller, individual weights on x_i (the $w_{ii(\text{ker})}$'s) get larger. As a result, tr($\mathbf{W}_{(\text{ker})}$) increases. As tr($\mathbf{W}_{(\text{ker})}$) increases, the denominator of equation (3.13), namely, [n-tr($\mathbf{W}_{(\text{ker})}$)] decreases, and thus penalizes the criterion for small bandwidths.

It should be noted that the term tr($\mathbf{W}_{(\text{ker})}$) in and of itself might be considered a measure of model adequacy (Mays 1995). Recall that in ordinary least squares (OLS) regression, tr($\mathbf{W}_{(\text{OLS})}$)=$p$, the number of parameters that need to be estimated. Cleveland (1979) pointed out that tr($\mathbf{W}_{(\text{ker})}$) might be thought of as the "equivalent" model degrees of freedom for nonparametric regression. The “equivalent” model degrees of freedom is representative of the number of parameters that would be needed to obtain a similar OLS fit. It is desirable to have tr($\mathbf{W}_{(\text{ker})}$) small, denoting a "simple" fit (Mays 1995).

It may be desirable to have an estimate of the variance (σ^2) in the underlying regression model. The nonparametric estimate of variance:

$$s^2_{(\text{NP})} = SSE / [n - \text{tr}(\mathbf{W}_{(\text{ker})})] \qquad (3.14)$$

which resembles the usual mean squared error from ordinary least squares regression, may be used for this purpose. Note again that $\mathbf{W}_{(\text{ker})}$ may be replaced by the weight matrix of any fitting technique.

The three forms of nonparametric regression and the global error criterion described in this chapter will be used extensively throughout this application.

3.5 Variance of Prediction

For all three forms of nonparametric regression used herein, variance of prediction can be expressed as $\sigma^2 \mathbf{WW}'$ (Müller 1988), where σ^2 is the variance inherent in the underlying regression model. Most estimates for σ^2 in the literature are asymptotic in nature, or involve bootstrapping techniques (Hall and Marron 1990; Buckley *et al*. 1988; Härdle and Bowman 1988). These techniques are best suited for large samples, which is not the case in this endeavor. For purposes herein, $s^2_{(\text{NP})}$ from equation (3.14) could be used to estimate σ^2, and thus to give an estimate for variance of prediction, $\hat{\sigma}^2 \mathbf{WW}'$ (Müller 1988). Several techniques for calculating prediction confidence intervals and confidence bands are provided in Härdle (1990) and Eubank and Speckman (1993).