Chapter 4

Nonparametric Regression

Introduction

Nonparametric regression techniques were developed as an alternative to parametric regression in order that a user-specified model is not necessary. The theoretical basis for the methodology is to let the data form the estimated regression function without imposing any restrictions on that form. This is done by fitting functions in a local fashion (fitting some function to a small neighborhood of observations). The primary purpose of nonparametric regression is the flexibility of the model fit without specification of a form. Some common assumptions in nonparametric regression are constant variance of the error terms and that the mean function, $m(\mathbf{x}_i)$, is some smooth function, meaning that the first k derivatives of m exist and are finite.

4.A Kernel Regression

Kernel regression was one of the first forms of nonparametric regression, and it developed into a popular technique due to its theoretical and computational simplicities, and its straightforward extension to higher dimensions. It can be thought of as a moving weighted average of observations near the location of prediction. The idea is, at each point where a predicted value is desired, to assign large weights to observations close to the point of prediction, and gradually downweight observations based on the distance of those observations from the prediction location. The methodology owes its name to an integral part of kernel regression, the kernel function, K, which is a function that, when completely defined, determines the exact assignment of weights.

Several approaches have been proposed for using the kernel function to determine the weights, and a few of the most popular will be introduced in this chapter. The end result of kernel regression is the determination of weights that establish the final regression estimates, that is $\{ h_{ij}^{KER} \}$ such that

$$\hat{y}(x_i) = \hat{y}_i = \sum_{j=1}^{n} h_{ij}^{KER} y_j , \qquad (4.A.1)$$

where $\sum_{j=1}^{n} h_{ij}^{KER} = 1$. Note the similarity of these estimates to those of parametric regression, which are also a linear function of the observed response values. The vector of estimated values (that is, the n fitted values at the n points x_i, i=1,…,n) can be written in terms of a kernel regression Hat matrix, which, in general, has no closed form. This vector of predicted values can be written in the form $\hat{\mathbf{y}}^{KER} = \mathbf{H}^{KER}\mathbf{y}$, where the i^{th} row of $\mathbf{H}^{KER}$ is $\mathbf{h}_i' = (h_{i1}^{KER}\ h_{i2}^{KER}\ \ldots\ h_{in}^{KER})$. One main distinction of the kernel Hat matrix from the OLS Hat matrix is that it is not symmetric.

The kernel function that is used to define the Hat matrix is generally a symmetric, continuous, smooth, bounded, real-valued function K that integrates to 1. The kernel functions that will be considered here are all positive functions that are symmetric about 0, have a mode of 0, and are defined such that K(u) is a decreasing function of |u|. The latter property, when using some type of distance measure as the argument for K in determining the weights, implements the concept of giving the most weight to points near the location of prediction. Härdle (1990) states that the predicted values achieve nice asymptotic properties if K has compact support, but this discussion will not necessarily be limited to those types of functions. The discussion in this chapter will also be confined to the one regressor case.

4.A.1 Nadaraya-Watson (NW) Weights

The weighting scheme introduced by Nadaraya (1964) and Watson (1964) for the predicted value at x_i is given by

$$h_{ij}^{KER} = \frac{n^{-1}b_n^{-1}K\left(\frac{x_i - x_j}{b_n}\right)}{n^{-1}b_n^{-1}\sum_{k=1}^{n}K\left(\frac{x_i - x_k}{b_n}\right)} \qquad (4.A.2)$$

where b_n is the bandwidth which, indirectly, determines the support of the kernel function. Note the subscript "n" on the bandwidth, implying that the bandwidth is a function of the sample size. (Predictions at any non-data point can easily be found by inserting some value x_0 into Equation (4.A.1) in place of x_i and then calculating the corresponding weights $\{h_{0j}^{KER}\}$.) A necessary assumption for the kernel estimate to be a consistent estimator of m is for $b_n \to 0$ as $n \to \infty$. This is logical since as $n \to \infty$, we assume that the x-values gradually cover the range of possible x-values, so one would need a smaller and smaller neighborhood of values to obtain a predicted value.

Kernel regression, in the sense that it is a moving local average, is equivalent to fitting a local location model. This is illustrated by noting that $\hat{m}(x_i)$ is the value of m that minimizes an objective function which resembles weighted least squares in the location model. That is, $\hat{y}_i^{KER} = \hat{m}_i$ where $\hat{m}_i$ minimizes

$$\sum_{j=1}^{n} h_{ij}^{KER}(y_j - m_i)^2 . \qquad (4.A.3)$$

Note that m_i is not a function of j and is thus a constant value when predicting at x_i. One interpretation of the kernel estimator at x_i is the estimate of location of the y-values for observations near x_i.

4.A.2 R-Nearest Neighbor (R-NN) Weights

R-Nearest Neighbor weights are based on a continuous uniform kernel function, the support of which is determined by the specification of r, the number of observations that the user desires to have an influence on prediction at any location. Thus, if predicting at x_i, the r nearest

neighbors of x_i receive equal weight (weight of 1/r) and all other observations receive zero weight.

In several proposed estimators, r-NN weights and the kernel weights are combined in such a way that exactly r observations receive positive weight when predicting at x_i, but the weights are a decreasing function of the observations' distances from x_i. The choice of r in this situation is analogous to the choice of bandwidth when calculating NW weights.

One major problem with kernel regression based on either NW and r-NN weights is the asymmetry of weights if the x-values are not equally spaced. In this situation, the estimate at x_i is biased toward those observations that are clustered together either to the right or to the left of x_i. Convolution weights were introduced to improve upon that bias.

4.A.3 Gasser-Müller (Convolution) Weights

Gasser and Müller (1979, 1984) introduced an integrated kernel estimate that incorporates a means of accounting for the density of the x-values. The weights are based on a sequence of numbers $s_0, s_1, \ldots, s_n$ such that $s_0 \le x_1 \le s_1 \le x_2 \le \ldots \le x_n \le s_n$. The weight for the j^{th} observation when predicting at x_i is defined as

$$h_{ij}^{(G-M)} = \int_{s_{j-1}}^{s_j} K\left(\frac{x_i - u}{b_n}\right) du . \qquad (4.A.4)$$

The choice of s_j is frequently the midpoint of x_j and x_{j+1}.

Even though the convolution weights improve upon the bias experienced by NW weights, the fit is much more variable. This is the result of the fact that points that are very near each other and contain similar amounts of information about $m(x_i)$ may receive drastically different weights (see Chu and Marron (1991) and Hastie and Loader (1993)).

One general problem with kernel regression is the bias of prediction at the boundaries. The lack of an equal number of points on either side of the location of prediction results in substantial bias of the predicted value at that point. In addition, kernel regression generally does not perform adequately in areas with substantial curvature. (For a more detailed discussion on the

shortcomings of kernel regression, see Fan (1992) and Hastie and Loader (1993). Also see Härdle (1990) for more discussion of kernel weighting schemes).

The bias problems of kernel regression led to the exploration of methods that improved upon that bias without losing too much efficiency with respect to variance. One such method is local polynomial regression.

4.B Local Polynomial Regression (LPR)

The fitting of a d^{th} degree local regression function (called local linear regression (LLR) for d = 1 and local polynomial regression (LPR) for d > 1) was formulated by Cleveland (1979) in the development of a robust nonparametric regression method. When fitting a polynomial of degree d to a data set, in the presence of *heterogeneous* variance, the OLS coefficient estimates are given by $\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y}$ where $\mathbf{W}$ = diagonal $(1/\hat{\sigma}_i^2)$ and the columns of $\mathbf{X}$ represent the terms in the polynomial function of the variable x. The idea of LPR is to fit a polynomial to a local neighborhood of points about x_i, then, using the estimated coefficients from that fit, obtain the fitted value at x_i. Instead of downweighting an observation because of a large variance, an observation is downweighted if it is far from x_i, the location of x where we are predicting. The method is a simple application of weighted least squares (wls), using the *kernel weights* in the weight matrix, resulting in the predicted value at the i^{th} location taking the form

$$\hat{y}(x_i) = \hat{m}(x_i) = \mathbf{x}_i^{NP\prime}(\mathbf{X}^{NP\prime}\mathbf{W}_i^{KER}\mathbf{X}^{NP})^{-1}\mathbf{X}^{NP\prime}\mathbf{W}_i^{KER}\mathbf{y} \qquad (4.B.1)$$

$$= \mathbf{x}_i^{NP\prime}\hat{\boldsymbol{\beta}}_i \ ,$$

where $\mathbf{x}_i^{NP\prime} = (1\ x_i\ x_i^2\ \ldots\ x_i^d)$, $\mathbf{X}^{NP}$ denotes the X-matrix that will be used in the nonparametric regression (to keep it distinct from the parametric X-matrix), and $\mathbf{W}_i^{KER}$ = diagonal $(h_{i1}^{KER}\ h_{i2}^{KER}\ \ldots\ h_{in}^{KER})$. For example, in LLR, $\mathbf{X}^{NP}$ would be an (nx2) matrix with a column of 1's and a column of the x-data. Note that $\hat{\boldsymbol{\beta}}_i$ is the vector that minimizes

$$\sum_{i=1}^{n} h_{ij}^{KER}(y_j - m_i(\mathbf{x}_j))^2, \qquad (4.B.2)$$

where $m_i(\mathbf{x}_j) = \mathbf{x}_j' \boldsymbol{\beta}_i$. The theoretical basis for LPR is derived from Taylor's Theorem, which states that a smooth function can be approximated fairly well at a given point by a polynomial function in some local neighborhood of that point.

In general, the more curvature that is present in the data, the higher the order of the local polynomial fit necessary. In most cases, however, a local linear regression or local quadratic regression (LQR) fit is adequate.

Hastie and Loader emphasize some of the advantages of LPR by examining an element from the vector of predicted values, which can be written in the form

$$\hat{y}_i = \mathbf{h}_i^{LPR\prime} \mathbf{y} = \sum_{j=1}^{n} h_{ij}^{LPR} y_j, \qquad (4.B.3)$$

where $\mathbf{h}_i^{LPR\prime} = (h_{i1}^{LPR}\ h_{i2}^{LPR} \ldots h_{in}^{LPR}) = \mathbf{x}_i^{NP\prime}(\mathbf{X}^{NP\prime}\mathbf{W}_i^{KER}\mathbf{X}^{NP})^{-1}\mathbf{X}^{NP\prime}\mathbf{W}_i^{KER}$. It can be shown that the LPR weights h_{ij}^{LPR} follow a relatively smooth trend, thus indicating a fit which is much less variable than one based on Gasser-Müller weights. The fitting of higher order local polynomials results in the reduction of bias in areas of curvature of the mean function. In addition, the bias problems at the boundaries discussed in the section on kernel regression do not exist for LPR - the estimated function fits very well at the boundaries of the data, with little loss in efficiency with respect to bias, making LPR superior to and the obvious choice over kernel regression.

4.C Bandwidth Selection

Three general categories that exist for bandwidth selection criteria are cross-validation, plug-in methods, and penalizing functions (for the sake of simplicity, only the first two will be discussed briefly in this section). The choice of the bandwidth is critical because of its effect on the mean squared error of the nonparametric regression estimator. A bandwidth that is too small will result in high variability, while a bandwidth that is too large will result in a fit that is highly

biased. Thus, a suitably chosen bandwidth is one that causes a moderate amount of both variability and bias.

The cross-validation criterion proposed by Härdle (Härdle and Marron (1985), Härdle, Hall, and Marron (1988), and Härdle (1990)) is given by

$$CV(b_n) = n^{-1}\sum_{i=1}^{n}(y_i - \hat{y}_{i,-i})^2 , \qquad (4.C.1)$$

where $\hat{y}_{i,-i}$ is the estimate of $m(x_i)$ with the kernel weight for observation (x_i,y_i) set to zero. Härdle states that this criterion is an asymptotically unbiased estimate (under certain assumptions) for the average squared error, which is defined as

$$ASE = n^{-1}\sum_{i=1}^{n}(\hat{y}_i - m(x_i))^2 . \qquad (4.C.2)$$

An additional mean squared error quantity is the integrated mean squared error, given by

$$IMSE = \int [\hat{y}(x) - m(x)]^2 f(x)dx . \qquad (4.C.3)$$

The cross-validation method is frequently used because of its computational ease and its standard form across a variety of regression models.

Plug-in methods are based on the asymptotic expansion of the mean squared error of the estimator. As noted in (4.C.2) and (4.C.3), mean squared error can be expressed in different ways. The ASE and IMSE are values that quantify the mean squared error of the estimator across a range of location. In the plug-in method, estimates are inserted for unknown quantities into the selected mean squared error quantity expansion, and the bandwidth that minimizes the estimated mean squared error is the one that is chosen.

Table 4.1 presents bias and variance calculations for both kernel and local linear regression using different weighting schemes. Note that these different weighting methods and different degrees of local polynomial fits result in different mean squared error expansions (these quantities are bias and variance quantities at a given value of x and are used to formulate ASE or IMSE). Notice that the derivatives of f(x) and m(x) are unknown and would require estimation for a plug-in procedure.

Table 4.1 Bias and variance calculations for kernel regression using Nadaraya-Watson weights and Gasser-Müller weights, and local linear regression using Nadaraya-Watson weights. Table from Fan (1992)

Method	Bias	Variance
Nadaraya-Watson Kernel Regression	$\frac{1}{2}m''(x) + \frac{m'(x)f'_X(x)}{f_X(x)}\int_{-\infty}^{\infty} u^2K(u)du \cdot b_n^2$	$\frac{\sigma^2(x)}{f_X(x)nb_n}\int_{-\infty}^{\infty} K^2(u)du$
Gasser-Müller Kernel Regression	$\frac{1}{2}m''(x)\int_{-\infty}^{\infty} u^2K(u)du \cdot b_n^2$	$\frac{3\sigma^2(x)}{2f_X(x)nb_n}\int_{-\infty}^{\infty} K^2(u)du$
Local Linear Smoother Using N-W Weights	$\frac{1}{2}m''(x)\int_{-\infty}^{\infty} u^2K(u)du \cdot b_n^2$	$\frac{\sigma^2(x)}{f_X(x)nb_n}\int_{-\infty}^{\infty} K^2(u)du$

The above techniques have been discussed in the context of a global bandwidth (one value of the bandwidth across the whole range of the data), but they can also be used for selection of a *variable* bandwidth (one that takes on different values in different locations within the data). Fan and Gijbels (1992) developed a variable bandwidth local linear regression estimator, with the bandwidth chosen to minimize the estimated expression for the IMSE. The optimal bandwidth that results varies indirectly with f(x) and m"(x), and directly with $\sigma^2(x)$. This is logical because one would want a less smooth (more local) fit in areas with dense x values or substantial curvature, and a more smooth fit (use more observations to obtain a predicted value) if the variability is high.

The estimator of Fan and Gijbels is only one of many variable bandwidth estimators. For other references on local bandwidth estimators and selection techniques, see Müller and Stadtmüller (1987) and Hall (1990).

The method of bandwidth selection in the current research has been a variation on the cross-validation criterion. Further discussion of bandwidth selection is in Section 8.C.

4.D Example of Local Polynomial Regression

Consider a simulated data set generated from a true underlying model of the form

$$E(y|x) = \frac{87}{18}x^4 - \frac{125}{18}x^3 + 2x^2, \qquad -.2 \le x \le 1.$$

Figure 4.1 illustrates a LLR fit to a data set of size 25 generated from the above mean function. The value of σ for the simulated normal random errors in this data set was 0.035, which is

relatively small given the range of the data in y-space. The kernel function that results from forming the predicted value at x = 0.75 is plotted along the horizontal axis, with the blocks superimposed on the curve representing the actual weights that observations near x = 0.75 receive for that prediction.

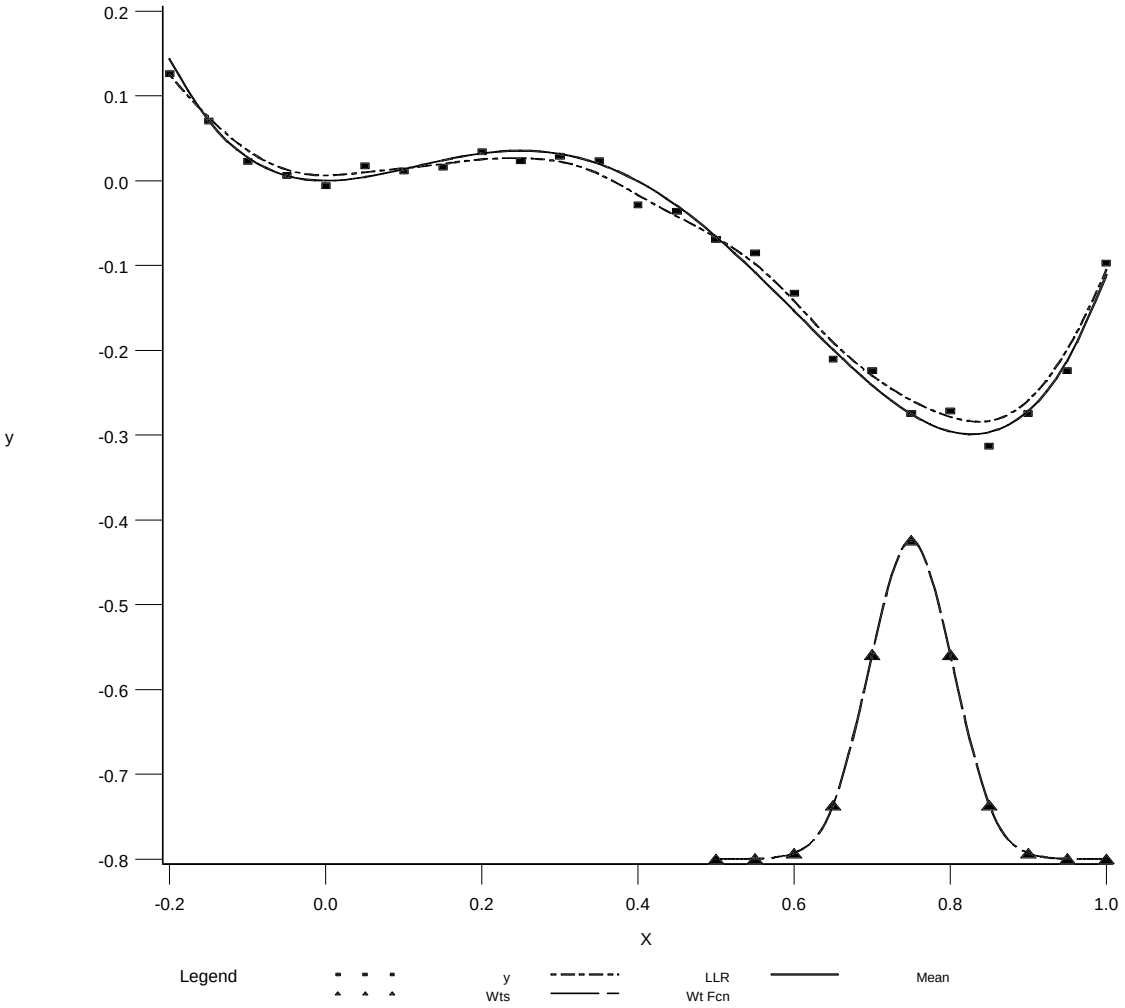

Figure 4.1 Plot of LLR fit to a simulated data set. The weights calculated for the fitted value at x = 0.75 are plotted on the horizontal axis (the curve represents the kernel function). The value of σ is 0.015 and the bandwidth is $b_n = 0.075$. The data were generated from the model

$$E(y|x) = \frac{87}{18}x^4 - \frac{125}{18}x^3 + 2x^2.$$

Note the excellent fit given by local linear regression, as it follows the true mean function almost exactly.

Consider another simulated data set from the same mean function, but with a larger value of σ than the one used in the previous example. This example is meant to illustrate the variability of the nonparametric fitting procedures and how the local aspect causes the fits to follow the data. Figure 4.2 is a plot of the raw data, the true mean function, and the fitted curve using local linear regression.

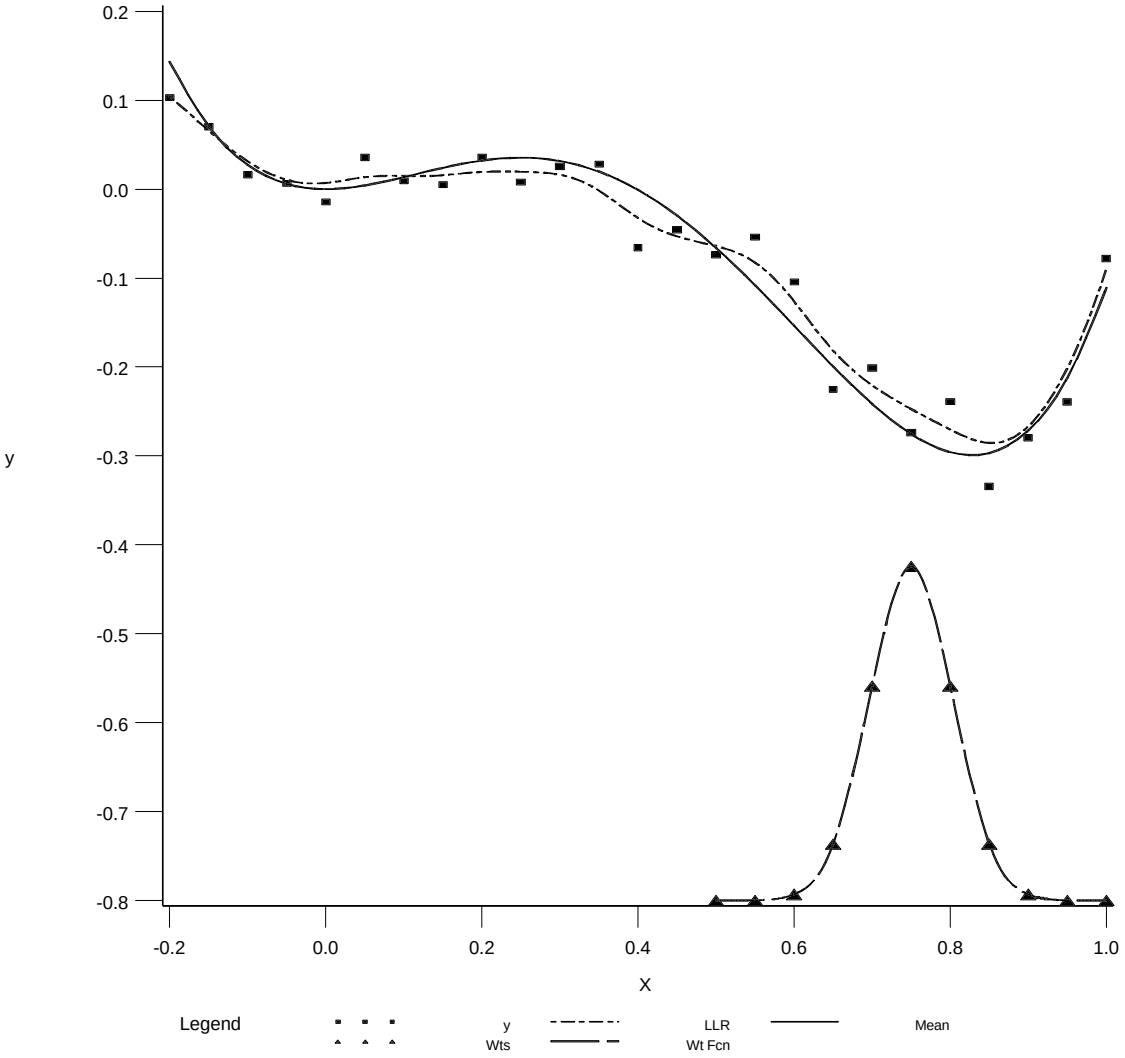

Figure 4.2 Plot of the local linear regression fit to data generated from the model $E(y|x) = \frac{87}{18}x^4 - \frac{125}{18}x^3 + 2x^2$. The value of σ is 0.035 and the bandwidth is $b_n = 0.075$.

Note that the fit is still very good, but it is lacking in a few areas where the data deviate somewhat from the mean function due to the increased random error term. This phenomenon is especially true when several data points fall some distance above or below the true mean function

consecutively - the nonparametric procedures cannot discriminate between these observations and good observations that fall on or near the mean function.

As a final illustration of the local fitting procedure discussed previously, consider the above data set with the injection of an outlier. Because of the local characteristic of the nonparametric regression methods, the outlier affects only the predicted values with a location in x-space that is near the location of the outlier. Here $\sigma = 0.035$ and $b_n = 0.075$. Figure 4.3 illustrates this effect.

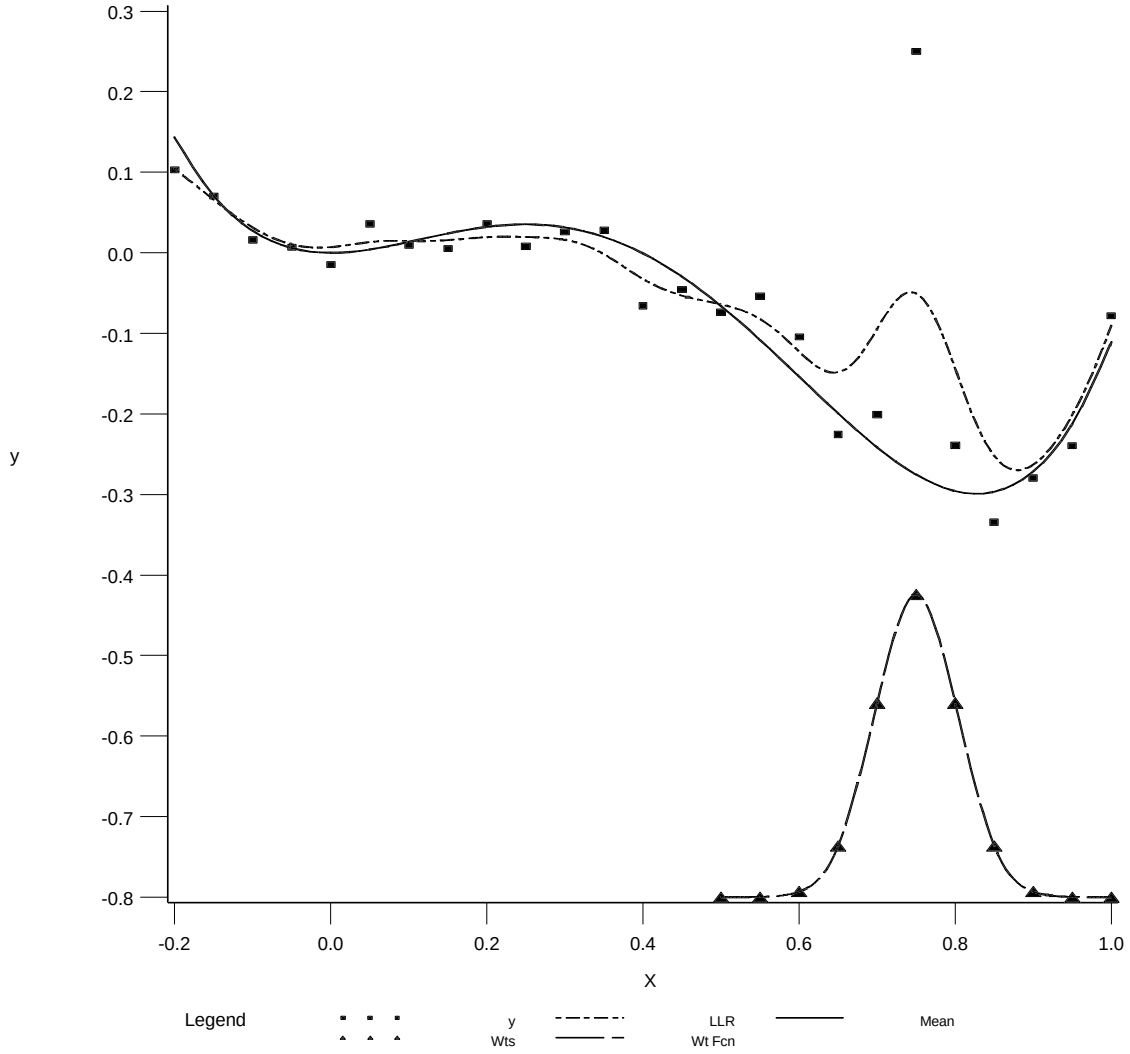

Figure 4.3 Plot of the local linear regression fit to simulated data with an extreme observation injected at x = 0.75. The value of σ is 0.035 and the bandwidth is $b_n = 0.075$. The outlier has only a local effect on the fitted values obtained from local linear regression. The data were generated from the model

$$E(y|x) = \frac{87}{18}x^4 - \frac{125}{18}x^3 + 2x^2$$

Again, the kernel function is plotted on the horizontal axis as a reference curve to illustrate the weights for observations when predicting at the location where the outlier occurs. Note the obvious non-robustness of LPR to outliers from the defining equations in expression (4.B.2). Since LPR is a local least squares type of fit, it is not resistant to outliers. Thus, even though LLR is robust to model misspecification in that it does not depend on some global model form, it *is* adversely affected by outliers, albeit only locally. This shortcoming of traditional nonparametric regression techniques led to the development of robust methods not dependent on global model forms.