Title

stata.com

npregress intro — Introduction to nonparametric regression

Description Remarks and examples References Also see

Description

Nonparametric regression models the mean of an outcome given the covariates without making assumptions about its functional form. This makes nonparametric regression estimates robust to functional form misspecification. `npregress` implements the two most common nonparametric regression estimators: series regression and kernel regression.

Nonparametric series estimation regresses the outcome on a function of the covariates. The function of the covariates is known as a basis function. A basis is a collection of terms that approximates smooth functions arbitrarily well. A basis function includes a subset of these terms. The bases used by `npregress series` are polynomials, piecewise polynomial splines, and B-splines.

Nonparametric kernel estimation computes a weighted average of the outcome. The weights are functions called kernels, which give rise to the name of the method. `npregress kernel` performs local–linear and local–constant kernel regression.

Whether we choose to approximate the mean of our outcome using series regression or kernel regression, we obtain estimates that are robust to assumptions about functional form. This robustness comes at a cost; we need many observations and perhaps a long computation time to estimate the elements of the approximating function.

This entry introduces the intuition behind the nonparametric regression estimators implemented in `npregress`. If you are familiar with these methods, you may want to skip to [R] **npregress kernel** or [R] **npregress series**.

Remarks and examples

stata.com

Remarks are presented under the following headings:

Overview
Nonparametric series regression
    Runge's phenomenon
    Piecewise polynomial splines and B-splines
Nonparametric kernel regression
Limitations of nonparametric methods

Overview

Nonparametric regression is used when we are uncertain about the functional form of the mean of the outcome given the covariates. For example, when we estimate a linear regression, we assume that the functional form for the mean of the outcome is a linear combination of the specified covariates. Both parametric (linear) regression and nonparametric regression provide an estimate of the mean for the different values of the covariates. Consider the simulated data in figure 1. The mean of the outcome for all values of `x` is overlaid on these points.

Figure 1.

Because the mean of the data in figure 1 is not linear in `x`, using a simple linear regression will not give us a correct picture about the effect of covariate `x` on the outcome. For example, if we perform a linear regression of the outcome on `x` for the data, we obtain the plot shown in figure 2.

Figure 2.

The change in the predicted outcome when `x` changes is positive and constant, yet the true mean is nonlinear. If the assumption about the functional form of the mean is incorrect, the estimates we obtain are inconsistent. If we instead fit the model using `npregress` and graph the estimates, we obtain figure 3.

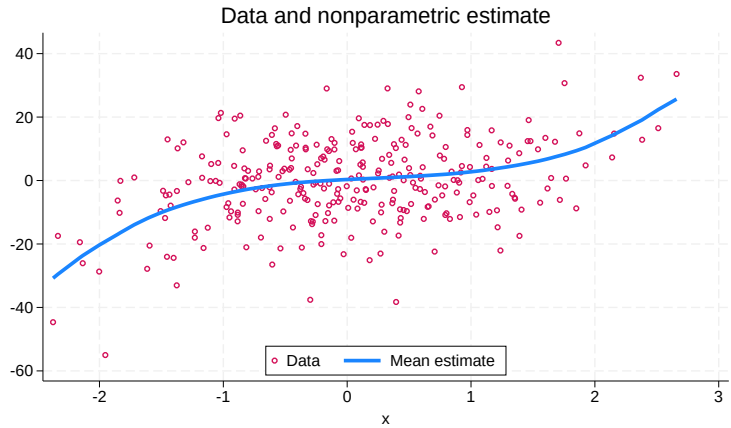

Figure 3.

`npregress` gives us the correct relationship between the outcome and the covariates. The nonparametric regression estimates are consistent as long as the true function is sufficiently smooth. If the linear regression assumptions are true, nonparametric regression is still consistent but less efficient.

Although nonparametric regression is a way to obtain estimates that are robust to functional form misspecification, this robustness comes at a cost. You need many observations and more time to compute the estimates. The cost increases with the number of covariates; this is referred to as the curse of dimensionality.

Nonparametric series regression

The basis and the basis function are concepts essential to understanding series regression. A basis is a collection of terms that can approximate a smooth function arbitrarily well. A basis function uses a subset of these terms to approximate the mean function. `npregress series` allows you to use a polynomial basis, a piecewise polynomial spline basis, or a B-spline basis. For each basis, `npregress series` selects the basis function for you.

We use an example to illustrate the use of a basis and a basis function. Suppose a researcher has data on the outcome y and a covariate x. We plot their relationship in the figure 4 below.

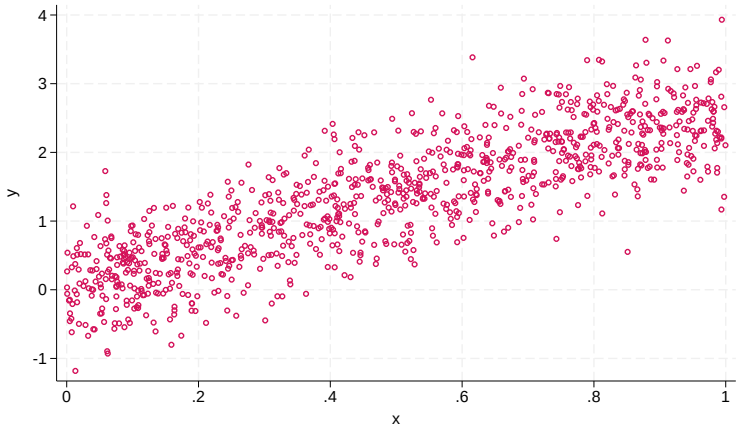

Figure 4.

In this case, a regression of y on x will do a good job of approximating the true function. If our data looked like the data in figure 5, however, a regression of y on x would be inadequate.

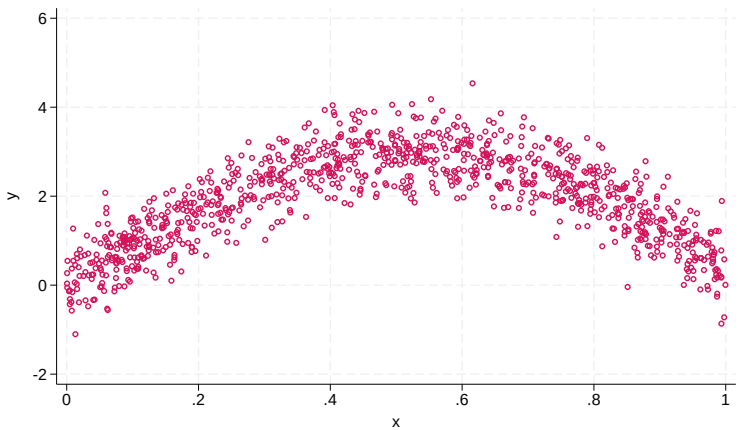

Figure 5.

In this case, a regression of y on x and x^2 is more appropriate.

In each case, we include terms from a polynomial basis. In the first case, we need a constant and the linear term x. In the second case, we need a constant, the linear term x, and the quadratic term x^2. A more complex function would require a basis function that includes more terms from the polynomial basis.

If we want to use a polynomial basis function, `npregress` will select a degree of the polynomial for us. Additional terms reduce bias but increase the variance of the estimator. `npregress` will select the terms that optimally tradeoff bias and variance. In other words, `npregress` selects a basis function that includes the terms that minimize the mean squared error. Our example above used a polynomial basis function, but `npregress` can also select terms from a piecewise polynomial spline or B-spline basis.

Runge's phenomenon

Polynomials are the most intuitive basis but not the preferred basis for nonparametric series estimation. The reason is that they are poor at interpolating. This problem shows up at the boundaries of the support of the covariates, where, as you increase the order of the polynomial, the polynomial approximation oscillates frequently, even when the true function does not behave this way.

Let us demonstrate. Below is an example for which we model a mean function using a third-order polynomial. We plot the data and the estimate of the mean function:

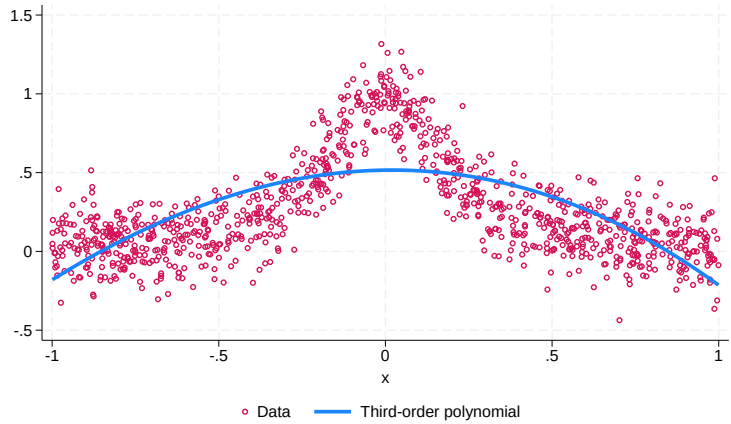

Figure 6.

Looking at the data, it appears that a higher-order polynomial would be a better fit for the data. Below is the mean function we get using a sixth-order and a tenth-order polynomial:

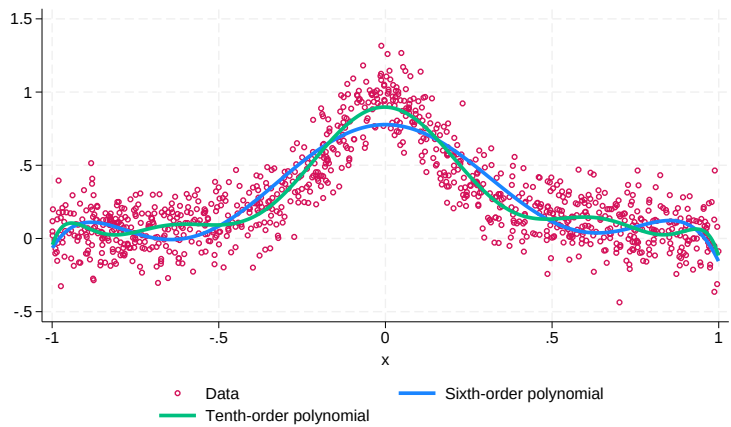

Figure 7.

The predictions improve at values near the middle of the range of x but become more variable at the edges of the parameter space.

What we illustrated above is referred to as Runge's phenomenon. Increasing the complexity of the polynomial order did not improve our approximation. In fact, as we increased the polynomial order, the behavior at the edges of the parameter space became more variable. The way to address this is to use a basis that does a better job of interpolating: piecewise polynomial splines or B-splines.

Piecewise polynomial splines and B-splines

Piecewise polynomial splines and B-splines are preferred to a polynomial basis because they are better at approximation. We discuss piecewise polynomial splines to provide intuition for both the piecewise polynomial spline basis and the B-spline basis.

Low-order polynomials do a great job of approximating functions in regions where the true function does not change too much. Splines continuously connect a set of low-order polynomials to create a basis to approximate a smooth function. The graph below illustrates what this definition means. We show in maroon a piecewise polynomial spline estimate of the mean function for the data in the example above.

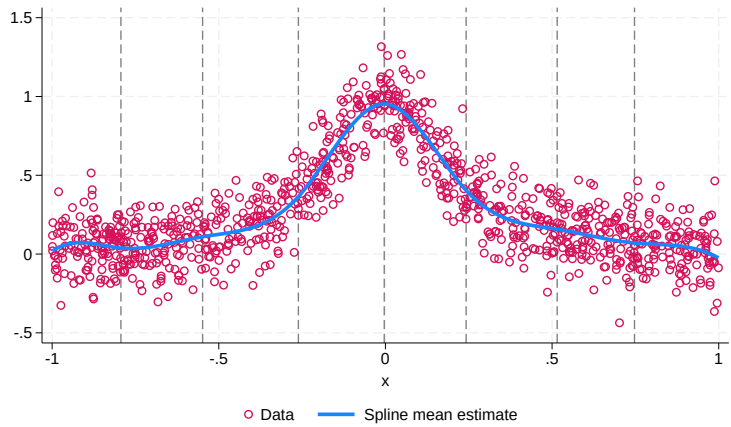

Figure 8.

To see that splines are better than polynomials, note that the spline approximation of the mean function fits the data well and that there are no regions where the approximation wiggles wildly.

Now, we delve into the definition above. The vertical lines in the graph partition the support of x into subregions. The piecewise polynomial spline basis allows for a different low-order polynomial in each subregion, and it forces the polynomials in neighboring regions to be continuously connected. In figure 8 above, the basis used is a third-order polynomial in each subregion. The graph illustrates that the polynomials are smoothly connected at the subregion boundaries. The subregion boundaries are known as the knot points, or just the knots, because they are where the different polynomials are tied together.

By default, `npregress` selects the number of knots for you. Alternatively, you may specify the number of knots yourself.

We now look at how the mean function at each region was computed. We show this mathematically and graphically.

Defining the seven knots as $t_1, \ldots, t_7$, where $t_1 < t_2 < \cdots < t_6 < t_7$, the third-order piecewise polynomial spline estimate is given by

$$\widehat{E}\left(y_i|x_i\right) = \widehat{\beta}_0 + \widehat{\beta}_1 x_i + \widehat{\beta}_2 x_i^2 + \widehat{\beta}_3 x_i^3 + \sum_{j=1}^{7} \beta_{j+3} \max(x_i - t_j, 0)^3$$

Thus, for all x_i that are less than the smallest knot, t_1, the mean estimate is given by the third-order polynomial

$$\widehat{E}\left(y_i|x_i{\leq}t_1\right) = \widehat{\beta}_0 + \widehat{\beta}_1 x_i + \widehat{\beta}_2 x_i^2 + \widehat{\beta}_3 x_i^3$$

Here it is graphically:

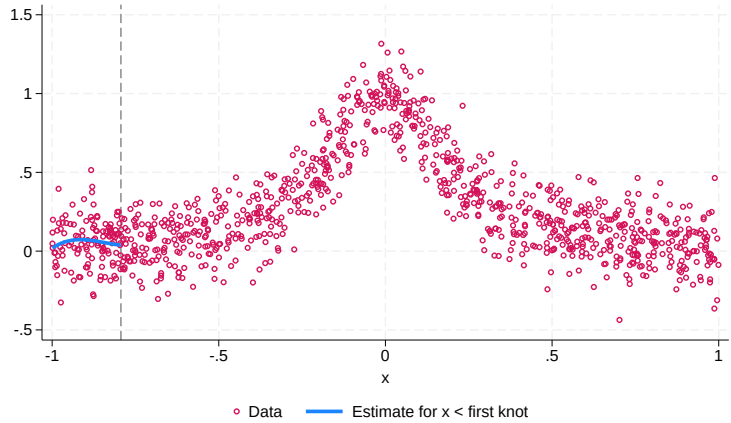

Figure 9.

Likewise, if x is less than the second knot, t_2, then the mean estimate for that region is different if $x_i > t_1$ than if $x_i \leq t_1$, and is given by

$$\widehat{E}\left(y_i|x_i{\leq}t_2\right) = \widehat{\beta}_0 + \widehat{\beta}_1 x_i + \widehat{\beta}_2 x_i^2 + \widehat{\beta}_3 x_i^3 + \beta_4 \left(x_i - t_1\right)^3 \left(x_i > t_1\right)$$

Here it is graphically:

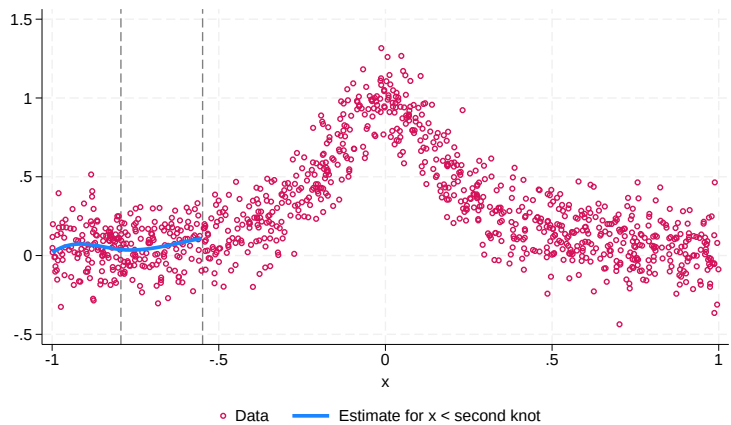

Figure 10.

As x increases, there are additional contributions from each subregion. If we continue plotting the resulting mean estimates, the following graphs would be what we would obtain:

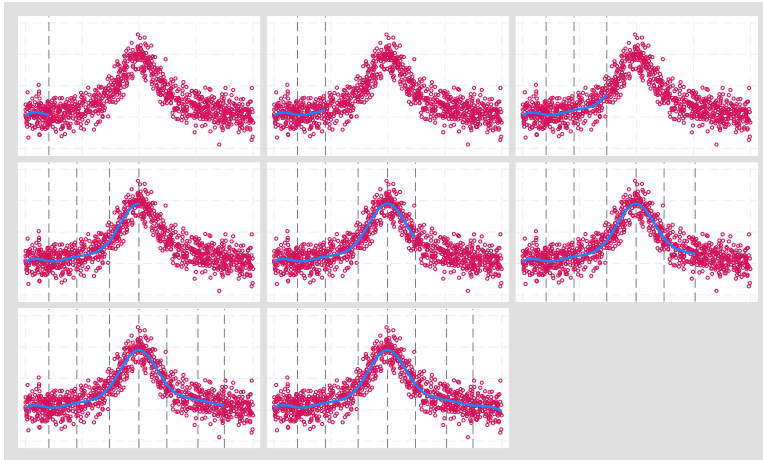

Figure 11.

This example illustrates how the terms in the spline basis approximate the mean function. Both the graph of the estimated function and the intuition in the example illustrate why the spline basis is better than the polynomial basis.

In the examples above, we used a piecewise polynomial spline basis. Specifically, we used a third-order piecewise polynomial spline basis function to obtain our estimates of the conditional mean. We could have also used second-order or first-order piecewise polynomial splines, where the order of the splines is defined by the order of the polynomial terms in the covariates used in each subregion.

As mentioned before, piecewise polynomial splines are preferred to a polynomial basis because they are better at approximation. However, piecewise polynomial splines also have some issues. In particular, they can be highly collinear and therefore numerically unstable. You can see this in the regions delineated in figure 11, which are defined by terms of the form $\max(x_i - t_j, 0)$ that may overlap.

B-splines avoid this problem, so each term that goes into the conditional mean approximation is orthogonal. It is for this reason that B-splines are the default basis for `npregress series`. However, the intuition we obtain from piecewise polynomial splines and B-splines is equivalent. In fact, B-spline and piecewise polynomial spline bases can approximate the same functions. For a more detailed explanation of B-splines, see *Methods and formulas* in [R] **npregress series**.

In this section, we provided an intuitive and brief introduction to nonparametric series estimation. For detailed introductions to series estimators and the methods implemented by `npregress series`, see de Boor (2001), Schumaker (2007), Eubank (1999), Schoenberg (1969), Newey (1997), and Chen (2007).

Nonparametric kernel regression

`npregress kernel` approximates the mean by using a kernel function. In *Overview*, we plotted the following data and nonparametric estimate of the mean function:

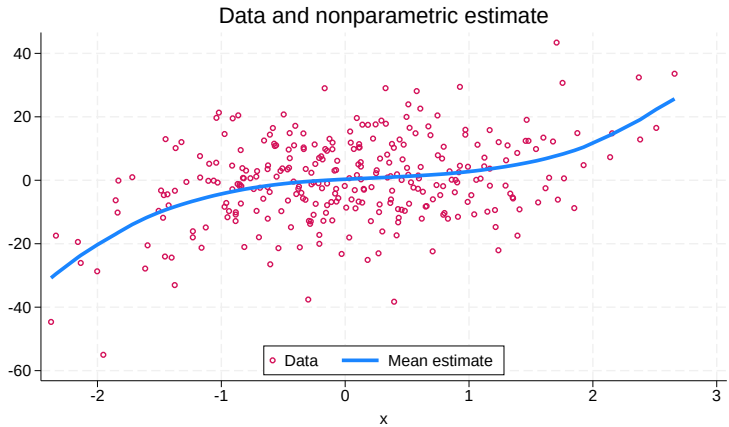

Figure 12.

We used kernel regression to estimate this mean function. With this method, we estimate the mean of the outcome at different values of the covariate `x`. In this section, we build our intuition for how kernel regression estimates these means, and we demonstrate this graphically.

Suppose covariate `x` is discrete. In this case, a consistent estimator of the mean of outcome `y` given that $\texttt{x} = a$ is the average of the values of `y` for which `x` is equal to a given value a. For instance, the sample average of the yearly income for married individuals is a consistent estimator for the population mean yearly income for married individuals.

Now, consider estimating the mean of `y` given that $\texttt{x} = a$ when `x` is continuous and a is a value observed for `x`. Because `x` is continuous, the probability of any observed value being exactly equal to a is 0. Therefore, we cannot compute an average for the values of `y` for which `x` is equal to a given value a. We use the average of `y` for the observations in which `x` is close to a to estimate the mean of `y` given that $\texttt{x} = a$. Specifically, we use the observations for which $|\texttt{x} - a| < h$, where h is small. The parameter h is called a bandwidth. In nonparametric kernel regression, a bandwidth determines the amount of information we use to estimate the conditional mean at each point a. We demonstrate how this works graphically.

For the simulated data in our example, we choose $h = 0.25$ and $a = -0.19$. The vertical lines in figure 13 delimit the values of `x` around a for which we are computing the mean of `y`. The light blue square is our estimate of the conditional mean using the observations between the vertical lines.

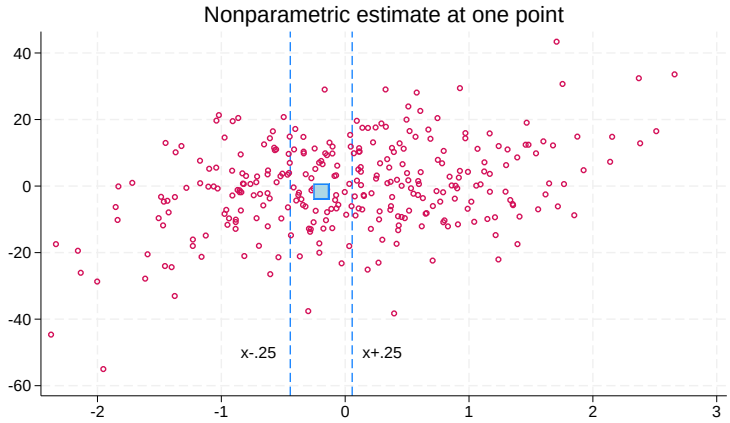

Figure 13.

Repeating this estimation when $a = 2.66$ produces figure 14.

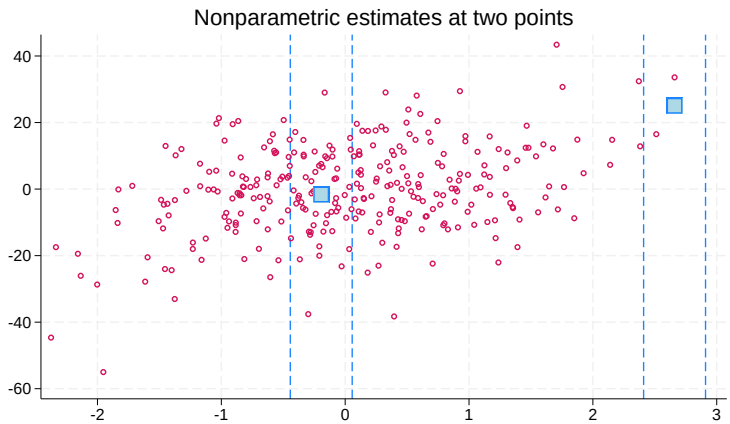

Figure 14.

Doing this estimation for each point in our data produces a nonparametric estimate of the mean for a given value of the covariates (see figure 15).

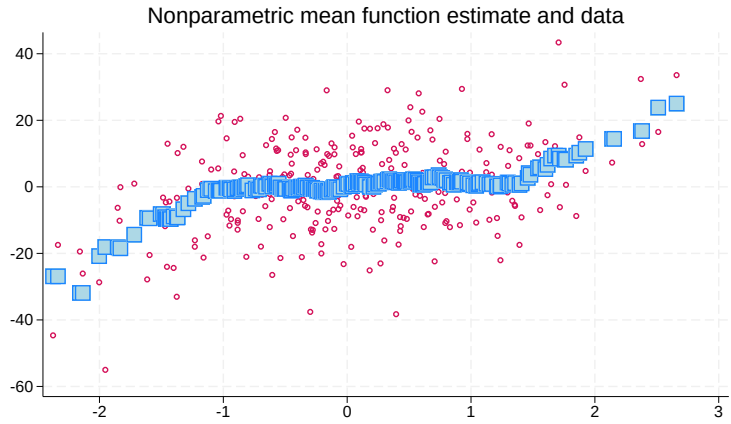

Figure 15.

The plotted blue squares in figure 15 form what is known as the conditional mean function. Because these are simulated data, we can compare our estimate with the true conditional mean function, a comparison we show in figure 16.

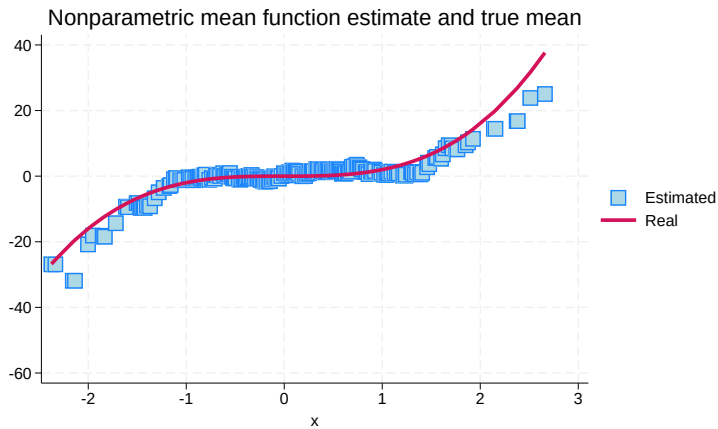

Figure 16.

We see that the estimate is a bit less smooth than the true function. The size of the bandwidth h determines the shape and smoothness of the estimated conditional mean function, because the bandwidth defines how many observations around each point are used. For example, if h is arbitrarily large—say, $h = 300$—then we get figure 17.

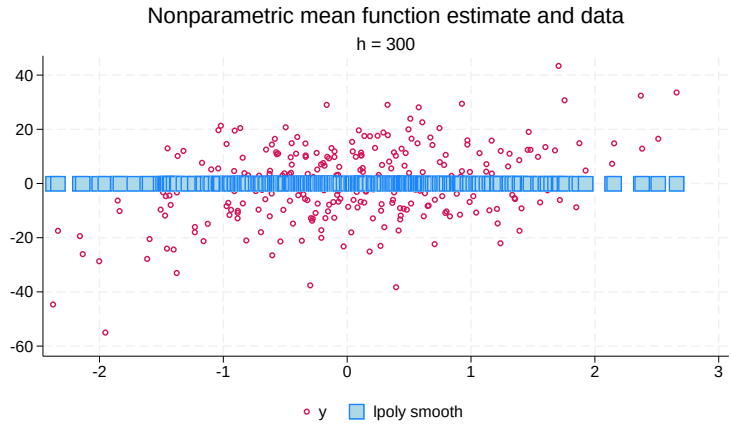

Figure 17.

In this case, all observations are used to estimate the conditional mean at each point, and the estimate is therefore a constant. On the other hand, a too-small bandwidth produces a jagged function with high variability, as illustrated in figure 18.

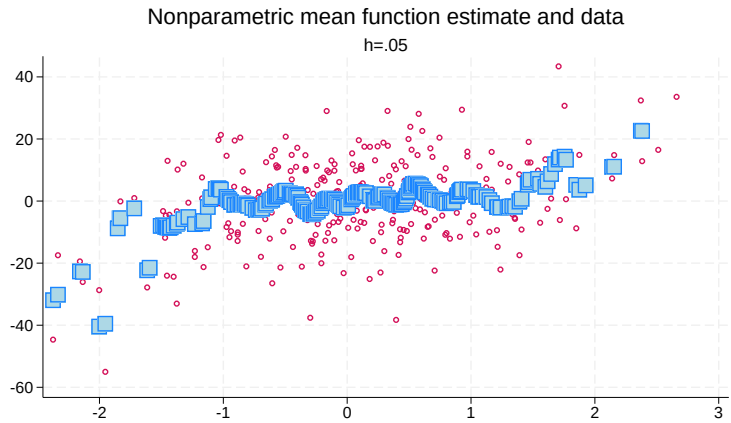

Figure 18.

The optimal bandwidth is somewhere in between. A too-large bandwidth includes too many observations, so the estimate is biased but it has a low variance. A too-small bandwidth includes too few observations, so the estimate has little bias but the variance is large. In other words, the optimal bandwidth trades off bias and variance. In the case of `npregress kernel`, the bandwidth is chosen to minimize the cost of this tradeoff by using either cross-validation, as suggested by Li and Racine (2004), or an improved Akaike information criterion proposed by Hurvich, Simonoff, and Tsai (1998).

How we average the observations around a point is also important. In the examples above, we gave the same weight to each observation for which $|\mathtt{x} - a| < h$. However, we might weight each observation differently. The weights that observations receive are determined by functions called kernels. We could have used any of the weights in [R] **kdensity**. For a nice introduction to kernel weighting, see Silverman (1986).

The estimator described above uses only nearby observations and is thus a local estimator. It uses a sample average, which is a regression on a constant, and is thus a locally constant estimator. For these reasons, the estimator described above fits what is known as a local-constant regression.

The generalization that uses the prediction from a local-linear regression on covariates is known as local-linear regression. Local-linear regression estimates the derivative of the conditional mean function in addition to the function itself. Understanding how the conditional mean changes when covariates change is sometimes the research question of interest, for example, how income changes for different levels of taxes. Local-linear regression provides an estimate for these changes for continuous and discrete variables.

See Fan and Gijbels (1996) and Li and Racine (2007) for detailed introductions to the kernel estimators implemented in `npregress kernel`.

Limitations of nonparametric methods

As discussed above, series regression and kernel regression approximate an unknown mean function. Series regression uses least squares on the basis function. Kernel regression uses a kernel-weighted average of nearby observations.

Series estimators are considered to be global estimators because they approximate the mean function at each point using the value of one overall approximating function. Kernel regression is considered a local estimator because it only uses nearby observations to approximate the mean for a given covariate pattern.

Although piecewise polynomial splines and B-splines are considered to be global estimators, in fact, they are local estimators. They are local because they fit a polynomial in each region defined by the knots. Like kernel estimators, piecewise polynomial spline and B-spline estimators require that there are enough data in each region. Suppose our data look like the data below.

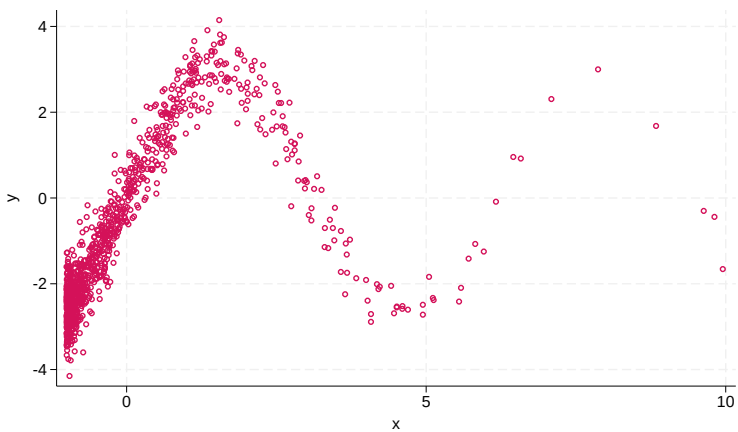

Figure 19.

Using a method to select knots optimally at percentiles of the data will give us figure 20.

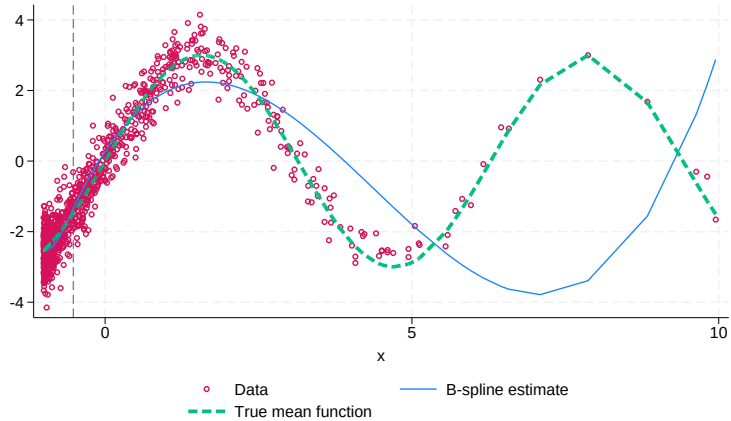

Figure 20.

The vertical line denotes the point at which the knot is placed. The blue line is the B-spline estimate, and the dotted green line is the true mean function. We see that our estimate of the mean function is not good, especially for higher positive values of the covariate. The reason is that data are sparse for these values. An alternative is to place the knots uniformly over the values of `x`. In this case, our estimate of the mean function improves. However, this does not change the fact that we have regions with insufficient data to make reliable inferences.

Thus, for kernel, piecewise polynomial spline, and B-spline estimators, we must have enough data points for all ranges of the data. In particular, piecewise polynomial spline and B-spline estimates should not be used to predict outside the support of the data.

Another important consideration is model selection. `npregress` selects the number of terms from a basis for series estimation and the bandwidth for kernel estimation. After model selection, the models are taken as given without accounting for model-selection error. You can find an in-depth discussion and references of some of the issues that arise when performing model selection in [LASSO] **Lasso intro**.

References

Cattaneo, M. D., M. Jansson, and X. Ma. 2018. Manipulation testing based on density discontinuity. *Stata Journal* 18: 234–261.

Chen, X. 2007. Large sample sieve estimation of semi-nonparametric models. In Vol. 6B of *Handbook of Econometrics*, ed. J. J. Heckman and E. E. Leamer, 5549–5632. Amsterdam: Elsevier. https://doi.org/10.1016/S1573-4412(07)06076-X.

de Boor, C. 2001. *A Practical Guide to Splines*. Rev. ed. New York: Springer.

Eubank, R. L. 1999. *Nonparametric Regression and Spline Smoothing*. 2nd ed. New York: Dekker.

Fan, J., and I. Gijbels. 1996. *Local Polynomial Modelling and Its Applications*. London: Chapman and Hall.

Hansen, B. E. 2009. University of Wisconsin–Madison, ECON 718, NonParametric Econometrics, Spring 2009, course notes. Last visited on 2019/01/15. https://www.ssc.wisc.edu/~bhansen/718/718.htm.

——. 2018. Econometrics. https://www.ssc.wisc.edu/~bhansen/econometrics/Econometrics.pdf.

Hurvich, C. M., J. S. Simonoff, and C.-L. Tsai. 1998. Smoothing parameter selection in nonparametric regression using an improved Akaike information criterion. *Journal of the Royal Statistical Society, Series B* 60: 271–293. https://doi.org/10.1111/1467-9868.00125.

Li, Q., and J. S. Racine. 2004. Cross-validated local linear nonparametric regression. *Statistica Sinica* 14: 485–512.

——. 2007. *Nonparametric Econometrics: Theory and Practice*. Princeton, NJ: Princeton University Press.

Newey, W. K. 1997. Convergence rates and asymptotic normality for series estimators. *Journal of Econometrics* 79: 147–168. https://doi.org/10.1016/S0304-4076(97)00011-0.

Pinzon, E. 2017. Nonparametric regression: Like parametric regression, but not. *The Stata Blog: Not Elsewhere Classified*. https://blog.stata.com/2017/06/27/nonparametric-regression-like-parametric-regression-but-not/.

Schoenberg, I. J., ed. 1969. *Approximations with Special Emphasis on Spline Functions*. New York: Academic Press.

Schumaker, L. L. 2007. *Spline Functions: Basic Theory*. 3rd ed. Cambridge: Cambridge University Press.

Silverman, B. W. 1986. *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall.

Also see

[R] **npregress kernel** — Nonparametric kernel regression

[R] **npregress series** — Nonparametric series regression

[R] **lpoly** — Kernel-weighted local polynomial smoothing

[R] **kdensity** — Univariate kernel density estimation

[R] **regress** — Linear regression