

Lecture 8

Moving Beyond Linearity

CHUNG-MING KUAN

Department of Finance & CRETA

National Taiwan University

April 30, 2022

Lecture Outline

- 1 Introduction
- 2 Polynomial Regressions
- 3 Step Functions
- 4 Piecewise Polynomials
- 5 Regression Splines
 - Cubic Splines
 - Natural Splines
 - Number and Locations of Knots
- 6 Smoothing Splines

Introduction

- Linear models are easy to implement and interpret, but they may provide only a poor approximation to the true (conditional mean) function in many applications.
- A specific nonlinear model may be useful for fitting one sample but not the other. Moreover, fitting nonlinear models usually requires numerical optimization which may find only a local optimum. As such, nonlinear models do not necessarily outperform linear models.
- In this lecture we relax the linearity assumption and consider different classes of **flexible, nonlinear models**. We are particularly concerned with their implementability, interpretability, and generalizability.

- We begin with nonlinear models with only **one** regressor (predictor). We discuss polynomial regressions that provide **global** approximation to the true function. We then introduce methods for **local** approximation, including step functions, regression splines, and local regressions. A complete picture of the approximating function is obtained by piecing together these local functions.
- These nonlinear models are further extended to **generalized additive models** that admit **multiple** predictors, each transformed using some nonlinear function. The generalized additive models are useful alternatives to linear models in practice.

Polynomial Regressions

Consider fitting the data (y_i, x_i) with the **polynomial with degree k** :

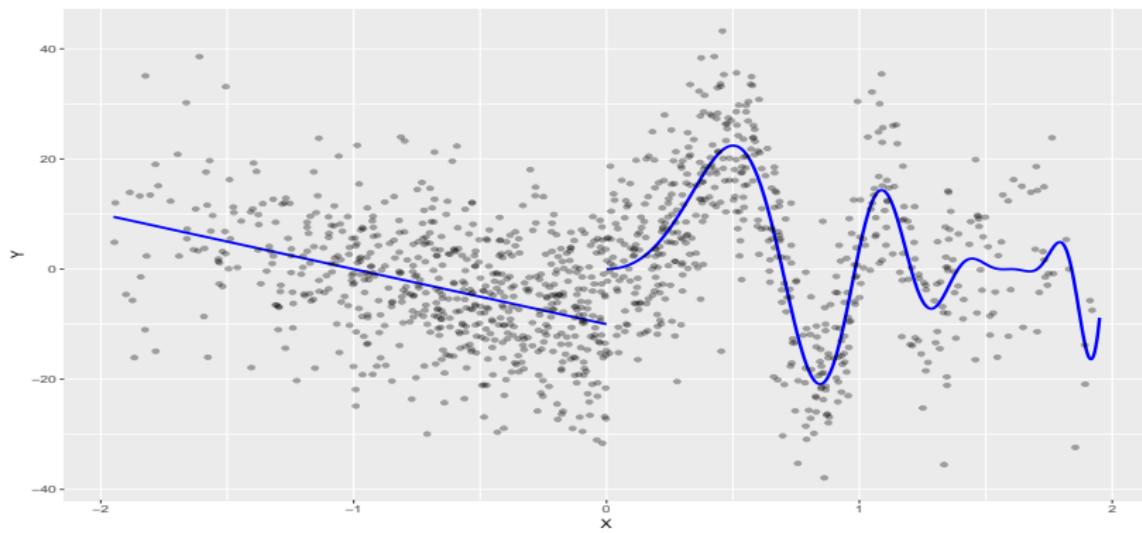
$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \cdots + \beta_k x_i^k + u_i.$$

- The class of polynomials is flexible and indexed by k . A polynomial is smooth when k is small (it is linear when $k = 1$), and it is able to represent complex nonlinearity when k is large. Finding a suitable k is always a challenging task.
- The polynomial regression function is **linear in parameters** and hence can be easily implemented using OLS. It is also easy to discuss the marginal effect of x on y .

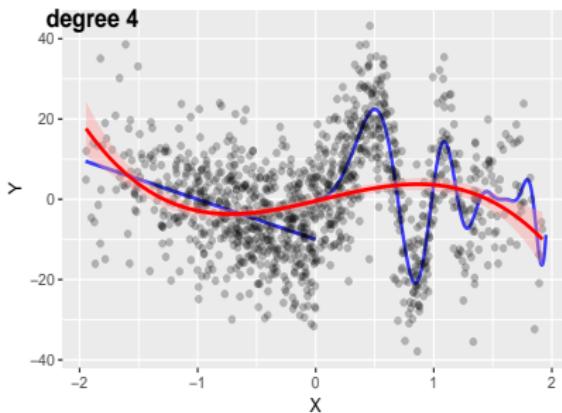
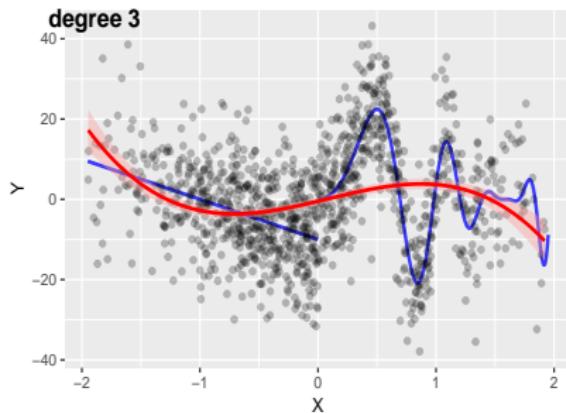
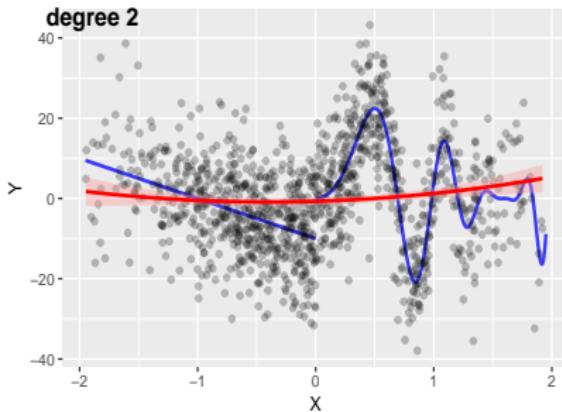
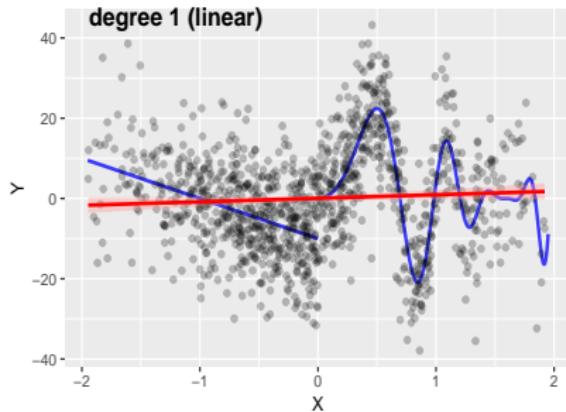
To examine the performance of polynomial regressions, we employ the DGP: $y_i = f(x_i) + u_i$, $i = 1, \dots, 1200$, where

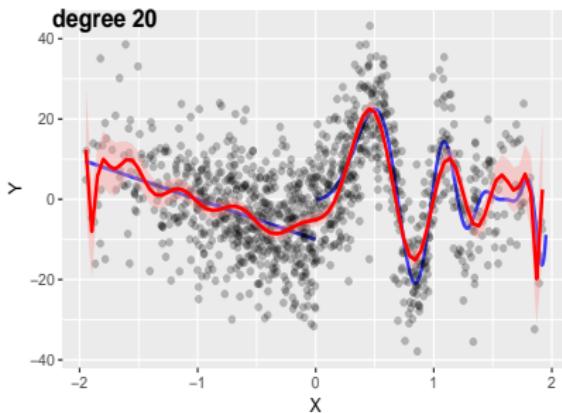
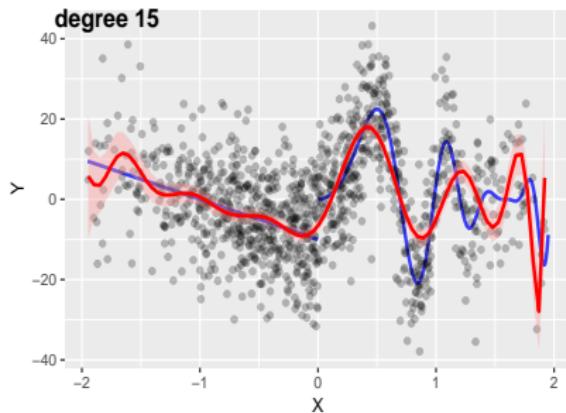
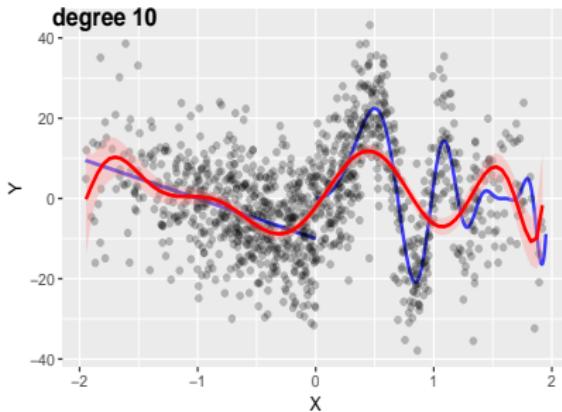
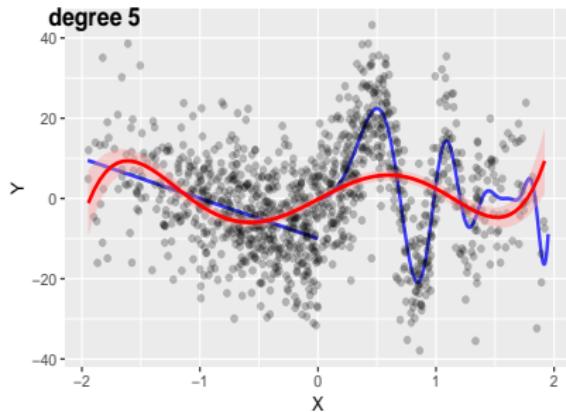
$$f(x) = \begin{cases} -10 - 5x, & -2 \leq x \leq 0, \\ (x+5)(x-1.6)^2(x+1)^3 \sin(6.5x) + 10, & 0 < x \leq 2, \end{cases}$$

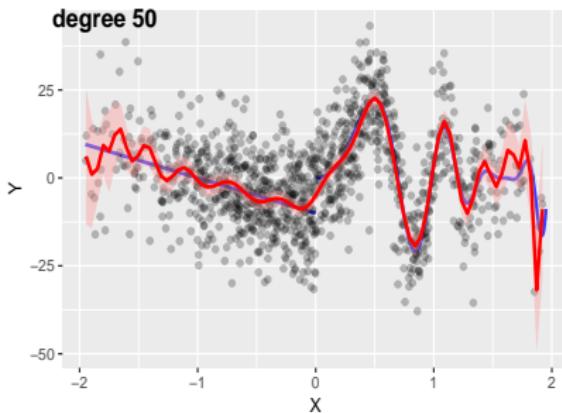
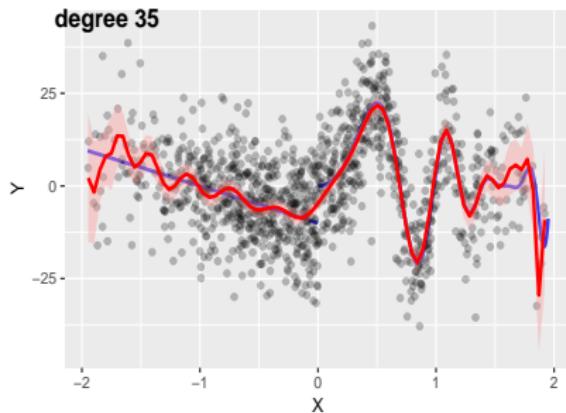
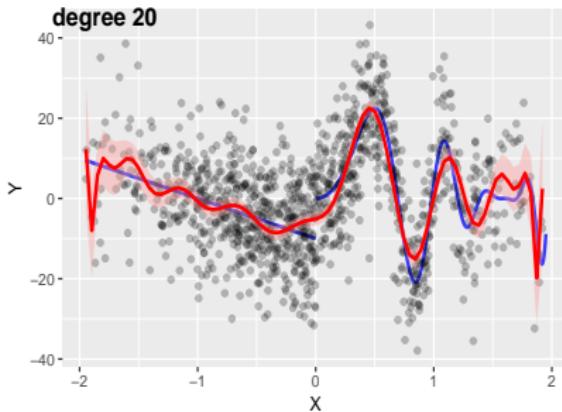
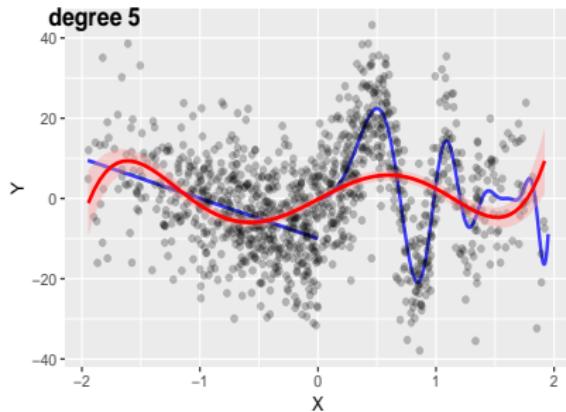
with x_i and u_i i.i.d. $\mathcal{N}(0, 0.81)$ and $\mathcal{N}(0, 100)$, respectively.



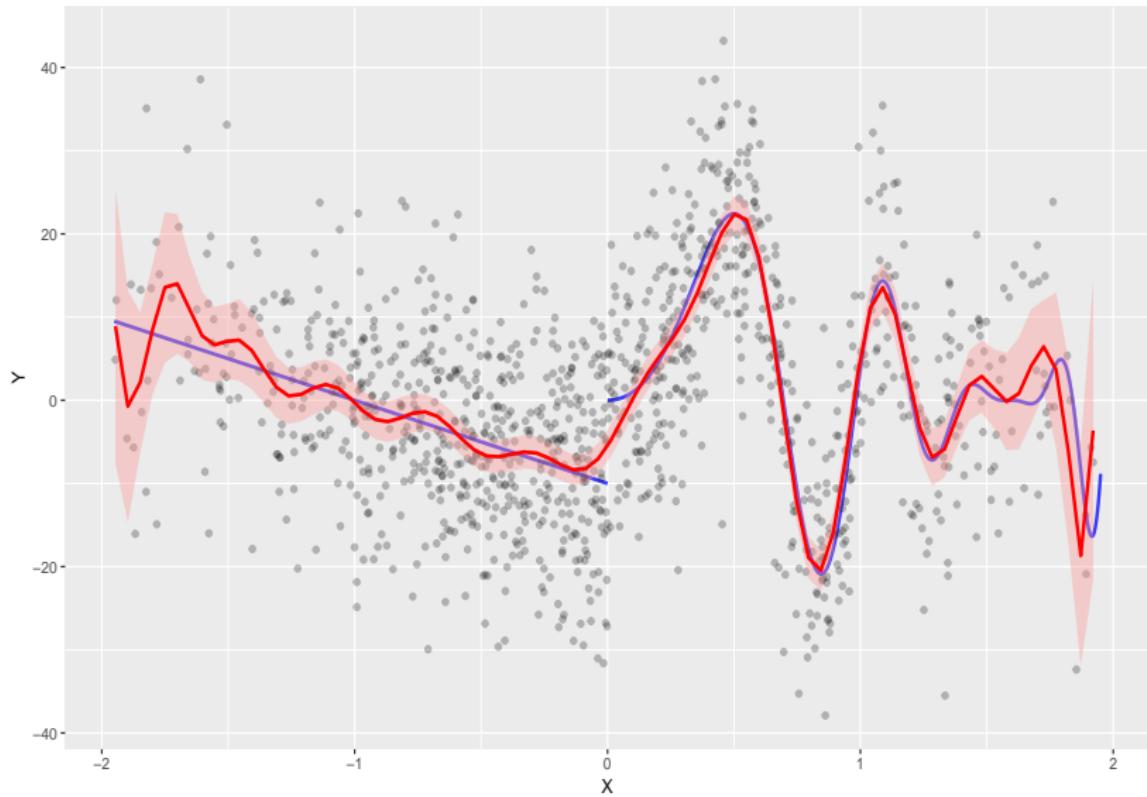
- In the figures below, we estimate the polynomial with different degrees using OLS; each red solid line is an estimated polynomial, and the shaded area is its 95% confidence interval.
- It can be seen that a polynomial with a low degree provides a good approximation to the linear part of f but under fits the other part of data (too simple for nonlinearity of f), while a polynomial with a high degree approximates the nonlinear part of f better but tends to over fits the data on the left (too complex for the linear function).
- In practice, the polynomial degree k may be determined by cross validation. This does not solve the problem; see the figure of polynomial with degree 31.
- A major problem with polynomial regression is that it tries to use a single (global) structure to approximate the true function that involves different local structures.







The degree of polynomial, 31, is chosen by 10-fold CV.



Step Functions

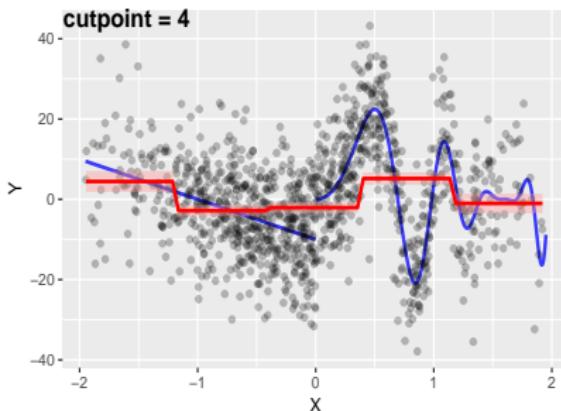
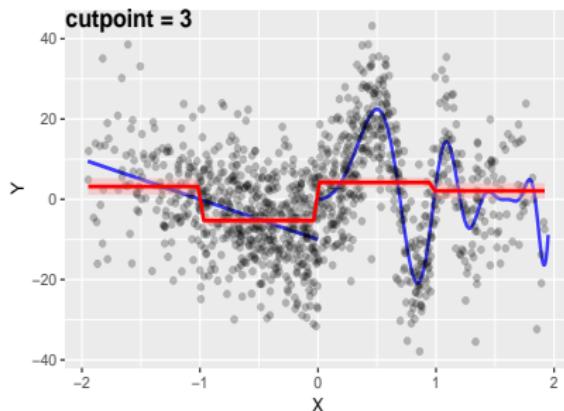
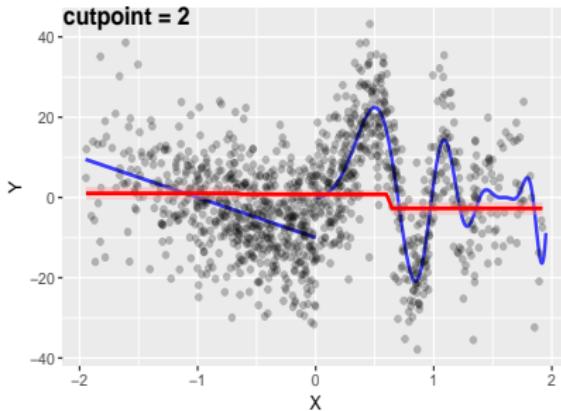
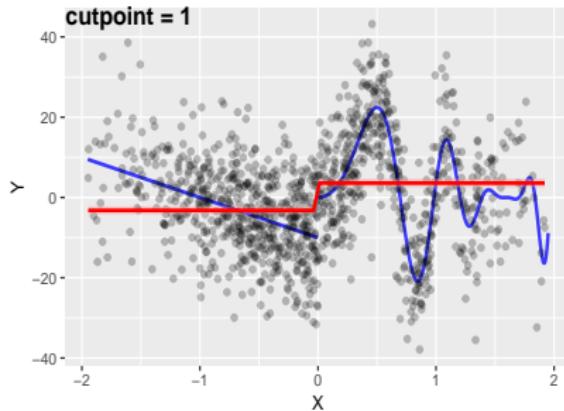
To avoid the aforementioned problem with polynomial regression, we may use step functions to **locally approximate** the true function. To this end, we choose the **cutpoints** c_1, c_2, \dots, c_k that partition the range of x into $k + 1$ disjoint intervals, and use the constant function to fit the data in each interval. This leads to the following **piecewise constant** (step) function:

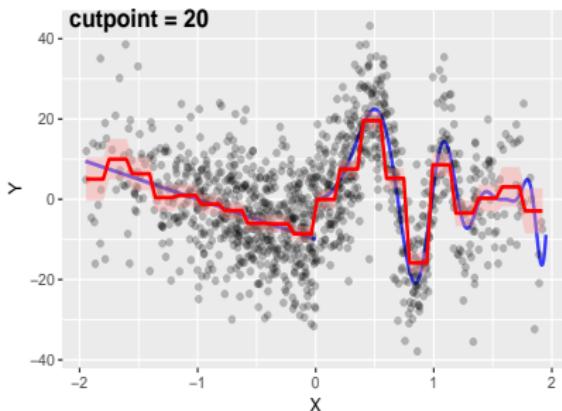
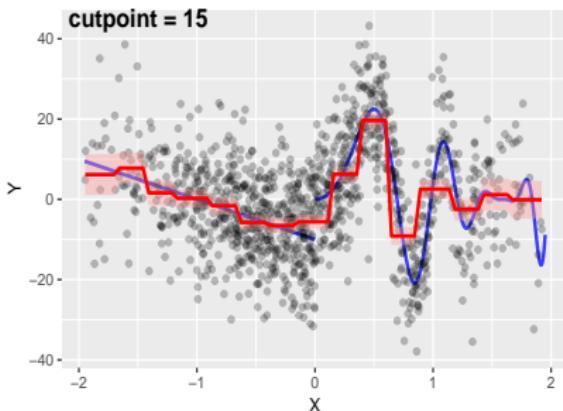
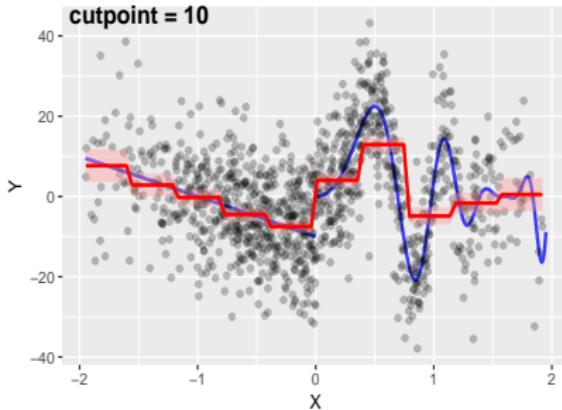
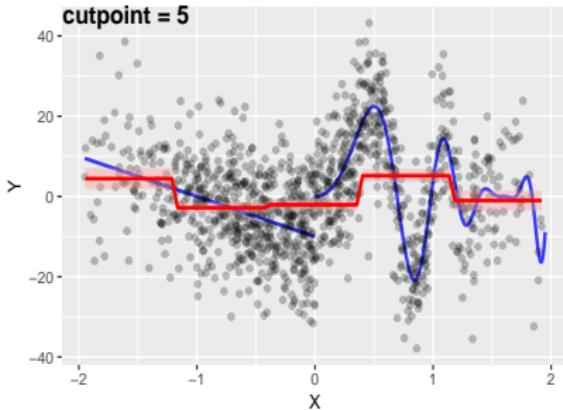
$$y_i = \beta_1 \mathbf{1}(x_i < c_1) + \beta_2 \mathbf{1}(c_1 \leq x_i < c_2) + \cdots \\ + \beta_k \mathbf{1}(c_{k-1} \leq x_i < c_k) + \beta_{k+1} \mathbf{1}(c_k \leq x_i) + u_i,$$

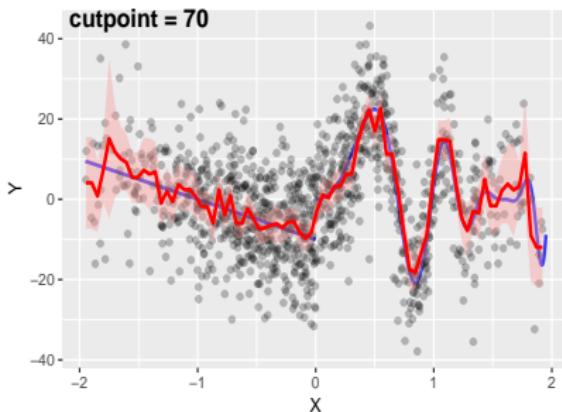
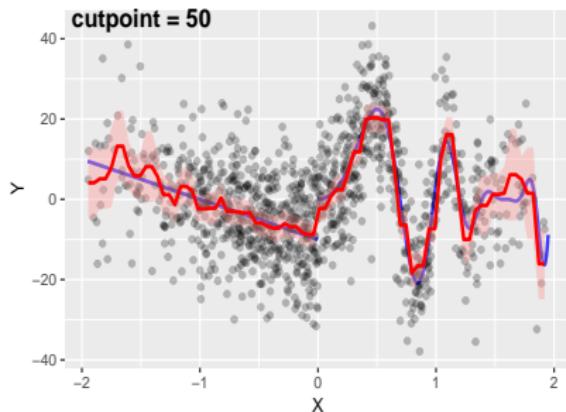
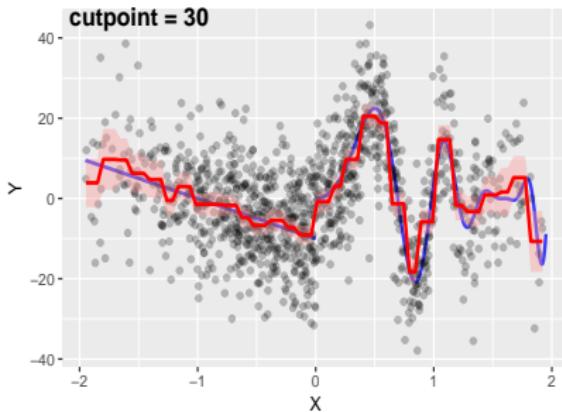
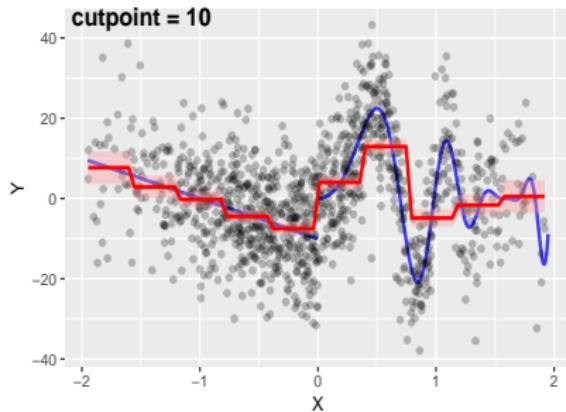
where $\mathbf{1}(\cdot)$ is the indicator function (a dummy variable).

- This function is **linear in parameters** and can be estimated using OLS.
- Each estimated coefficient is simply the average of y_i on the corresponding sub-interval.

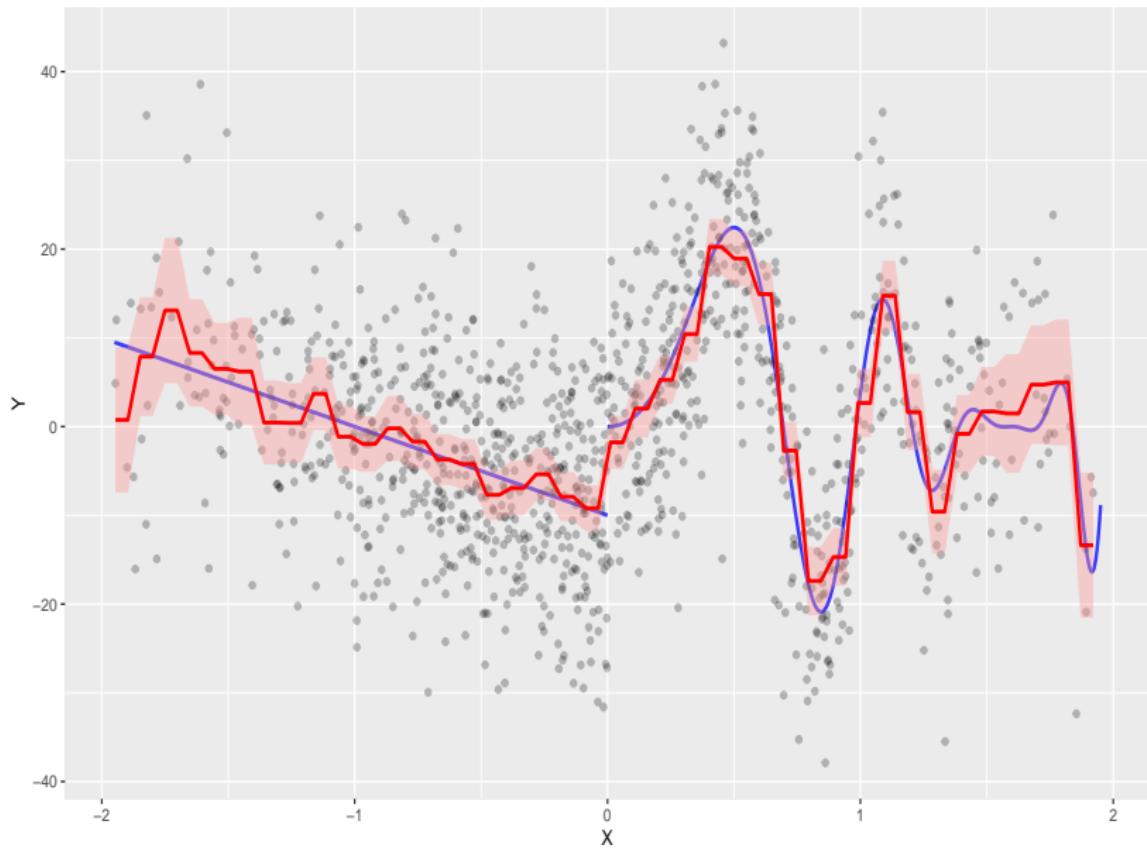
- The figures below illustrate the estimated step functions with different numbers of cutpoints, with the cutpoints evenly distributed on x .
- It can be seen that a large number of cutpoints may be helpful for approximating the nonlinear part while over fitting the linear part, but a small number of cutpoints may under fit the nonlinear part. In practice, one may choose k by cross validation.
- This step function does not seem to work well, because its local approximation (constant function) is far too simple. More generally, one may consider using linear (quadratic, cubic) function for local approximation.





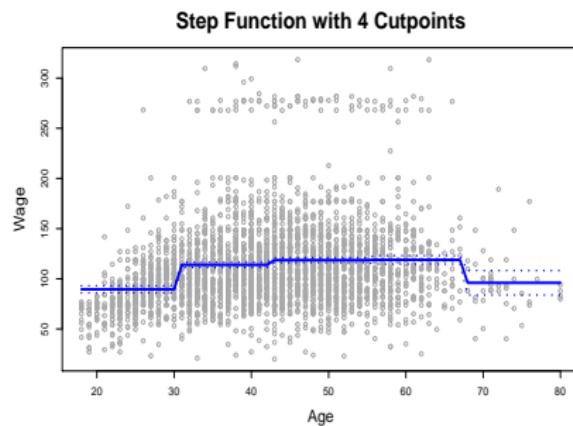
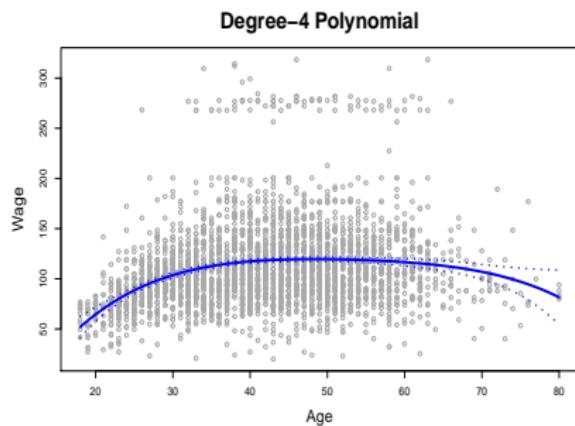


The number of cutpoints, 40, is chosen by 10-fold CV.



Example: Wage in the Mid-Atlantic region

The `Wage` dataset in `R` contains income and demographic information of 3,000 male workers in the Mid-Atlantic region. For the variable of interest `wage` and the predictor `age`, the estimated polynomial with degree 4 and the estimated step functions with 4 cutpoints are plotted below; the dotted curves are the estimated 95% confidence interval.



Piecewise Polynomials

The polynomial regression and piecewise constant regression are special cases of the **basis function approach**. To see this, let b_1, b_2, \dots, b_k be some basis functions of x . We now consider a model with $b_j(x)$ as predictors:

$$y_i = \beta_0 + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + \cdots + \beta_k b_k(x_i) + u_i.$$

This model is linear in parameters and can be estimated using OLS.

- For polynomial regression, $b_j(x) = x^j$.
- For step function, $b_j(x) = \mathbf{1}(c_{j-1} \leq x < c_j)$.
- More generally, one may consider the **piecewise polynomial** on each sub-interval of x .

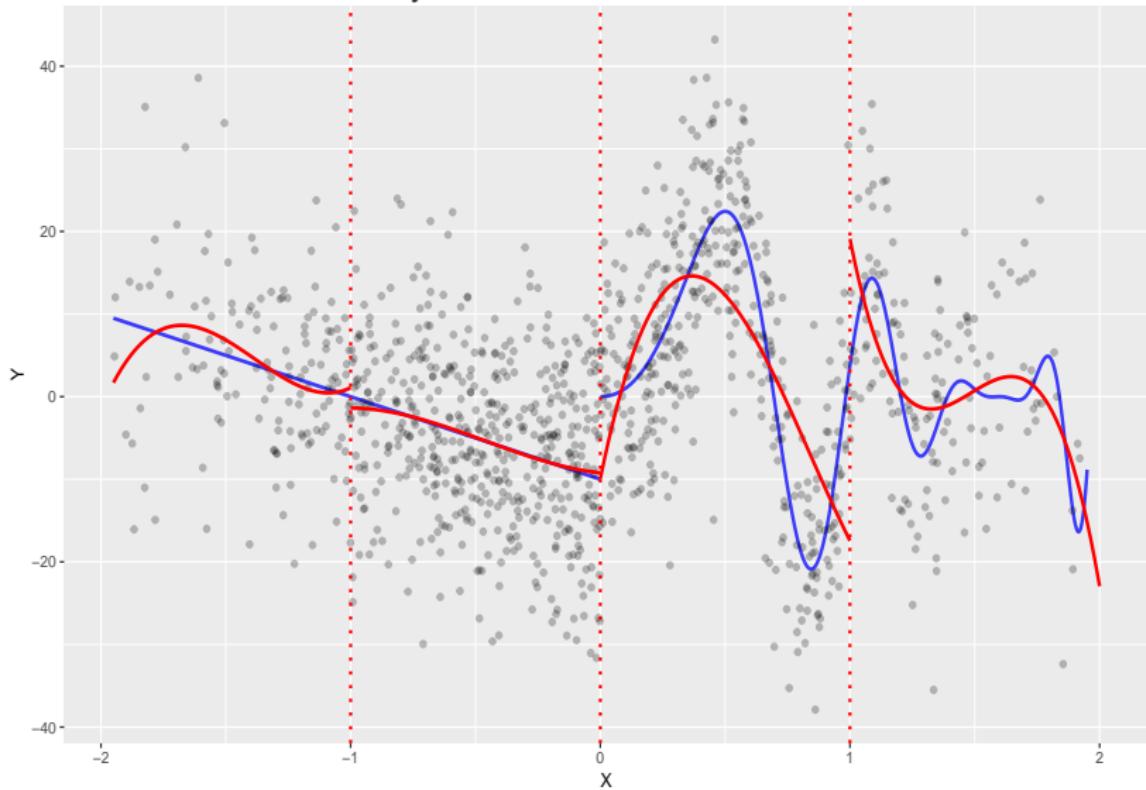
The function below is a **piecewise cubic polynomial** with 3 uniformly distributed cutpoints:

$$f(x) = \begin{cases} \beta_{00} + \beta_{01}x_i + \beta_{02}x_i^2 + \beta_{03}x_i^3, & -2 < x \leq -1, \\ \beta_{10} + \beta_{11}x_i + \beta_{12}x_i^2 + \beta_{13}x_i^3, & -1 < x \leq 0, \\ \beta_{20} + \beta_{21}x_i + \beta_{22}x_i^2 + \beta_{23}x_i^3, & 0 < x \leq 1, \\ \beta_{30} + \beta_{31}x_i + \beta_{32}x_i^2 + \beta_{33}x_i^3, & 1 < x \leq 2, \end{cases}$$

which involves 4 cubic polynomials on 4 sub-intervals of x . Letting $b_{ij}(x) = x^j \mathbf{1}(c_i < x \leq c_{i+1})$, $i, j = 0, 1, 2, 3$, with $c_0 = -2$, $c_1 = -1$, $c_2 = 0$, $c_3 = 1$ and $c_4 = 2$, we can write

$$y_i = \sum_{i=0}^3 \sum_{j=0}^3 \beta_{ij} b_{ij}(x) + u_i.$$

A piecewise cubic polynomial with 3 uniformly distributed knots



Continuous Piecewise Polynomials

A drawback of the piecewise polynomial is that it is **discontinuous** at each cutpoint. Discontinuity is not natural for approximating the true function because it causes very different approximations for data in the neighborhood of each cutpoint. To avoid this problem, we need a **constraint** that makes the piecewise polynomial continuous.

Suppose we want to connect two lines $\beta_0 + \beta_1 x$ and $\gamma_0 + \gamma_1 x$ at the point $x = \xi$, so that

$$\gamma_0 + \gamma_1 \xi = \beta_0 + \beta_1 \xi.$$

Letting $\gamma_1 = \beta_1 + \beta_2$, this implies the intercept of the second line must satisfy the constraint: $\gamma_0 = \beta_0 - \beta_2 \xi$.

We now introduce the following function with a given ξ :

$$h(x, \xi) = (x - \xi)_+ = \begin{cases} 0, & x \leq \xi, \\ (x - \xi), & x > \xi. \end{cases}$$

Then, $f(x) = \beta_0 + \beta_1 x + \beta_2 h(x, \xi)$ leads to the piecewise linear function:

$$f(x) = \begin{cases} \beta_0 + \beta_1 x, & x \leq \xi, \\ (\beta_0 - \beta_2 \xi) + (\beta_1 + \beta_2)x, & x > \xi, \end{cases}$$

where two line segments are connected at the cutpoint (**knot**) $x = \xi$.

Similarly, setting

$$h^2(x, \xi) = (x - \xi)_+^2 = \begin{cases} 0, & x \leq \xi, \\ (x - \xi)^2, & x > \xi, \end{cases}$$

the quadratic polynomial with $h^2(x, \xi)$ is

$$\begin{aligned} f(x) &= \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 h^2(x, \xi) \\ &= \begin{cases} \beta_0 + \beta_1 x + \beta_2 x^2, & x \leq \xi, \\ (\beta_0 + \beta_3 \xi^2) + (\beta_1 - 2\beta_3 \xi)x + (\beta_2 + \beta_3)x^2, & x > \xi. \end{cases} \end{aligned}$$

It can be verified that the two quadratic curves are connected at the knot: $x = \xi$, and $f(x)$ is a continuous, piecewise quadratic function. Moreover, the first derivative, $f'(x)$, is also continuous, because the derivatives of these two curves, $\beta_1 + 2\beta_2 x$ and $(\beta_1 - 2\beta_3 \xi) + 2(\beta_2 + \beta_3)x$, are connected at $x = \xi$.

Cubic Splines

Define $h^3(x, \xi)$ as

$$h^3(x, \xi) = (x - \xi)_+^3 = \begin{cases} 0, & x \leq \xi, \\ (x - \xi)^3, & x > \xi, \end{cases}$$

which connect two cubic curves at $x = \xi$. The functions $h(x, \xi)$, $h^2(x, \xi)$, and $h^3(x, \xi)$ are known as the **truncated power basis function** at ξ . The **cubic spline** with k knots ξ_1, \dots, ξ_k is the cubic polynomial with k truncated power basis functions:

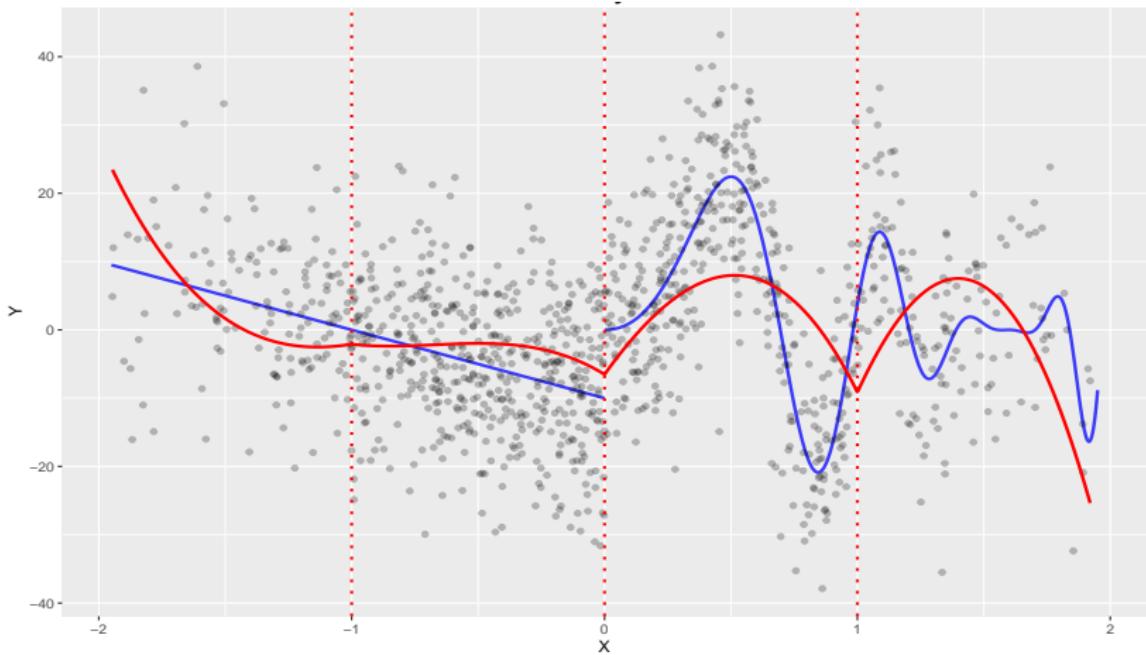
$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 h^3(x, \xi_1) + \cdots + \beta_{3+k} h^3(x, \xi_k),$$

which has continuous first and second derivatives. More generally, a **degree- d spline** has continuous derivatives up to order $d - 1$.

Below is an estimated piecewise cubic polynomial with $h(x; \xi)$ at 3 knots:

$$\beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \beta_4 h(x_i, -1) + \beta_5 h(x_i, 0) + \beta_6 h(x_i, 1).$$

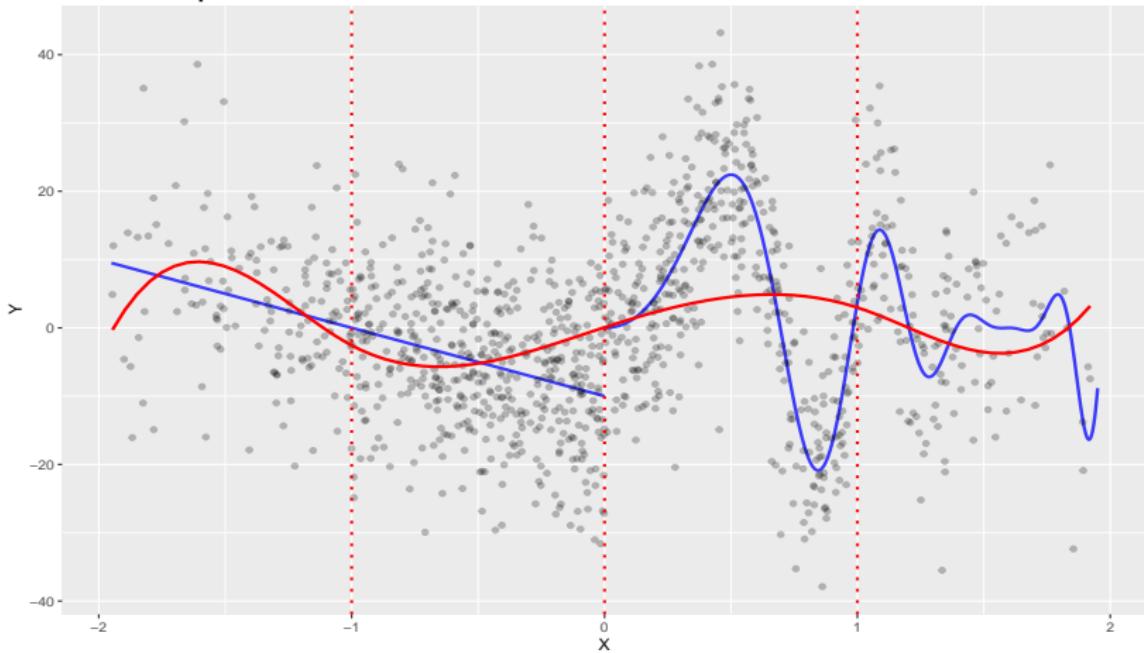
This curve is continuous but with sharp transition at each knot.



Below is an estimated cubic spline with $h^3(x, \xi)$ at 3 knots:

$$\beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \beta_4 h^3(x_i, -1) + \beta_5 h^3(x_i, 0) + \beta_6 h^3(x_i, 1).$$

Note that this curve is very smooth at each knot.

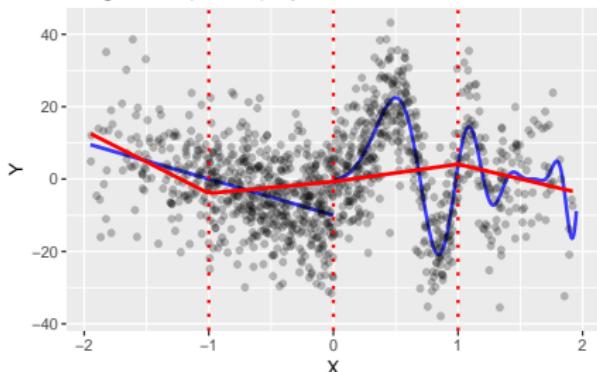


Why Cubic Splines?

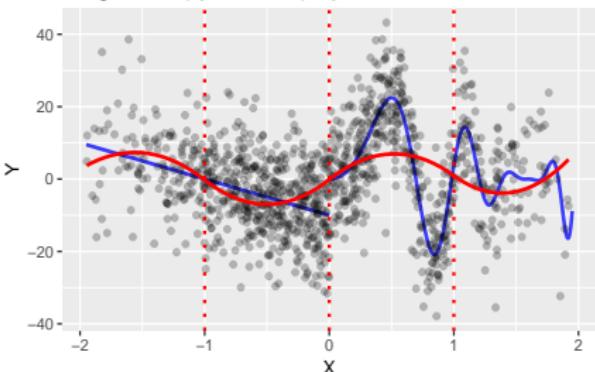
Cubic spline is a common choice in applications because of its flexibility. Moreover, a cubic spline is very smooth at each knot, and 2 segments have the **same curvature** (characterized by their continuous second derivatives) at the knot they meet. This is in contrast with quadratic splines whose curvature may change at each knot. Beyond curvature, it is hard to visualize the difference between high-order derivatives of 2 segments.

In the following pages, we plot the linear, quadratic, cubic and degree-4 splines with 3, 10, and 20 knots.

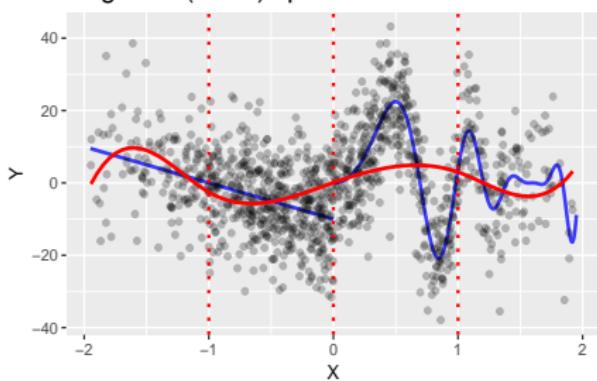
Degree 1 (linear) spline with 3 knots



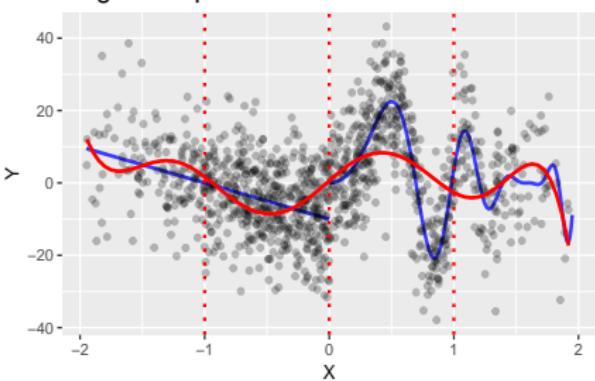
Degree 2 (quadratic) spline with 3 knots



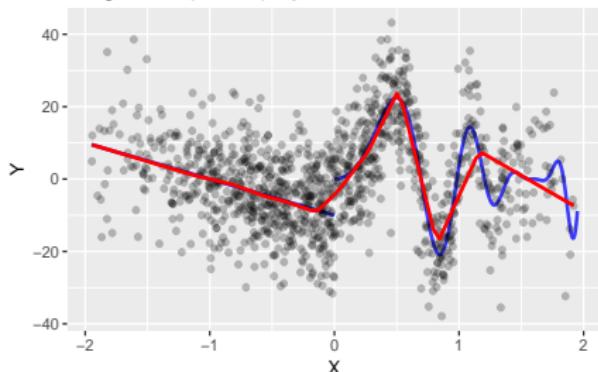
Degree 3 (cubic) spline with 3 knots



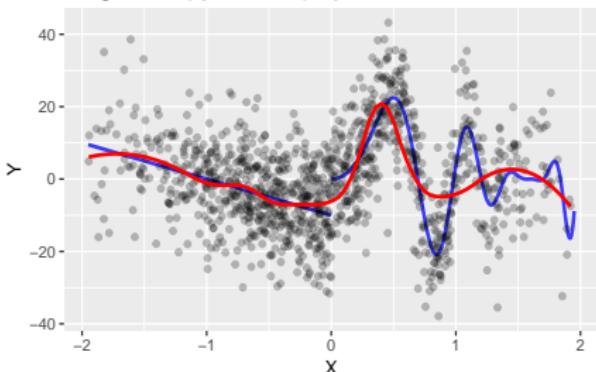
Degree 4 spline with 3 knots



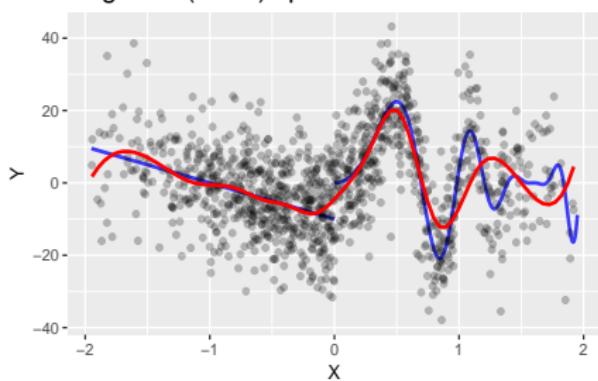
Degree 1 (linear) spline with 10 knots



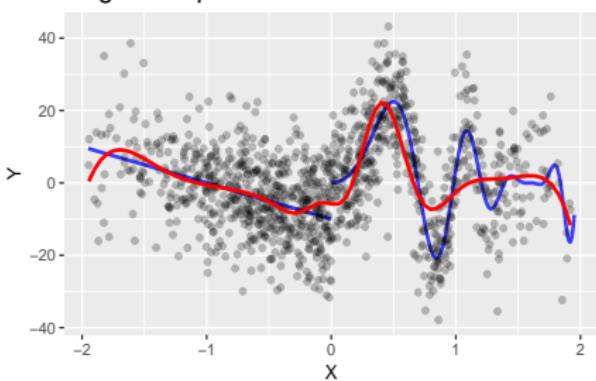
Degree 2 (quadratic) spline with 10 knots



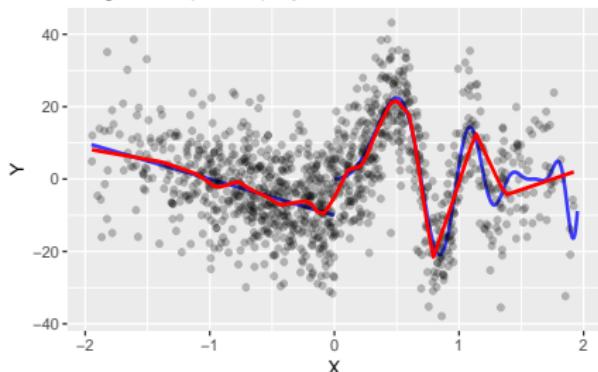
Degree 3 (cubic) spline with 10 knots



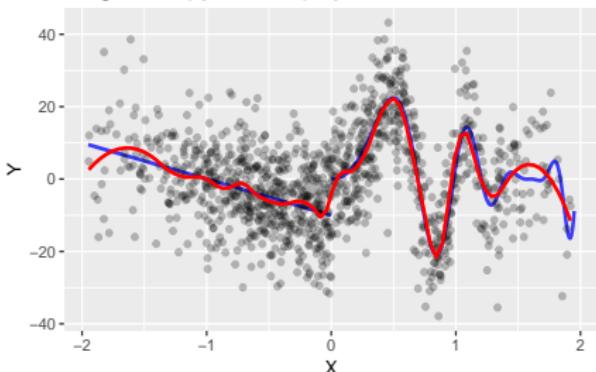
Degree 4 spline with 10 knots



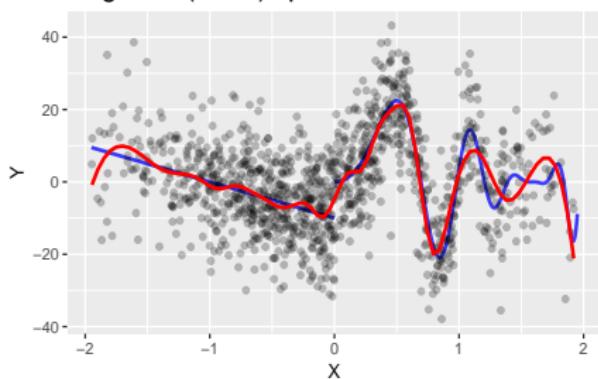
Degree 1 (linear) spline with 20 knots



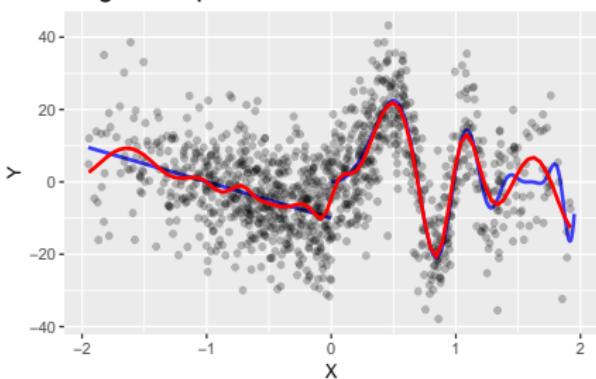
Degree 2 (quadratic) spline with 20 knots



Degree 3 (cubic) spline with 20 knots



Degree 4 spline with 20 knots



Natural Splines

- A drawback of the cubic spline is that, because the data are usually scarce near the boundaries of the predictor x , it has large variances and may exhibit erratic patterns at two ends.
- To improve the boundary behaviors of a cubic spline, one may add additional **boundary constraints** such that the cubic spline is **linear** beyond the boundary knots of x . This leads to the **natural cubic spline** (or simply the natural spline). Due to the linearity constraints, a natural spline typically has smaller variance near the boundaries of x than does a cubic spline.
- When fitting a natural spline, one must first determine the (interior) knots ξ_1, \dots, ξ_k , as in fitting a cubic spline.

Boundary Constraints

Consider the cubic spline with k knots:

$$\beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 h^3(x, \xi_1) + \cdots + \beta_{3+k} h^3(x, \xi_k).$$

For $x < \xi_1$, the linearity requirement implies $\beta_2 = \beta_3 = 0$. For $x > \xi_k$,

$$\beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 (x - \xi_1)^3 + \cdots + \beta_{3+k} (x - \xi_k)^3,$$

where $(x - \xi_j)^3 = x^3 - 3\xi_j x^2 + 3\xi_j^2 x - \xi_j^3$. The linearity requirement then implies $\beta_2 - 3 \sum_{j=1}^k \beta_{3+j} \xi_j = 0$ and $\beta_3 + \sum_{j=1}^k \beta_{3+j} = 0$. When $\beta_2 = \beta_3 = 0$ is imposed, we have the following constraints:

$$\sum_{j=1}^k \beta_{3+j} \xi_j = \sum_{j=1}^k \beta_{3+j} = 0.$$

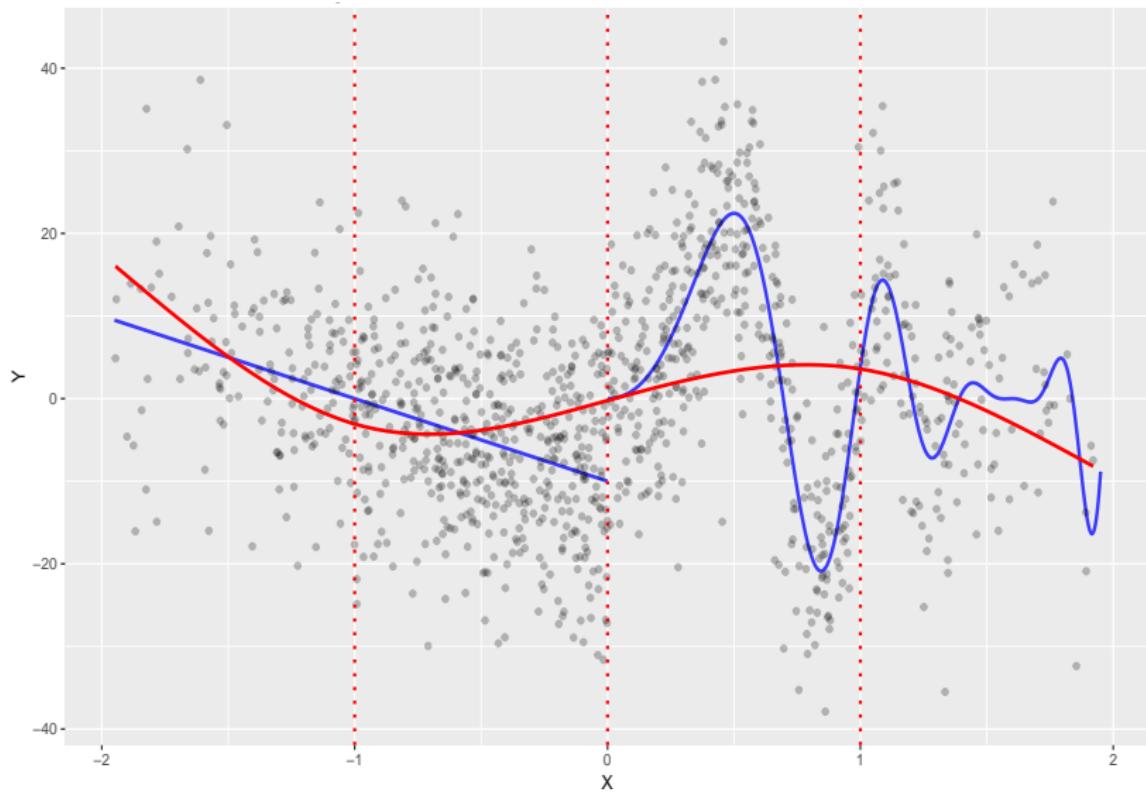
With the four constraints above, it can be shown that a natural cubic spline with k knots can be represented using $k + 2$ basis functions:

$$\sum_{j=1}^{k+2} N_j(x) \theta_j,$$

where N_j are the basis functions with $N_1(x) = 1$ and $N_2(x) = x$; for other basis functions, see pp. 145–146 of HTF (2009). (We omit the details.) Hence, a natural cubic spline can also be estimated using OLS.

Remark: To estimate a natural cubic spline with k (interior) knots, `ns` in **R** adds two **boundary knots** at the minimum and maximum values of x as its default. One may also specify the locations of the boundary knots. The fitted natural spline in **R** is **linear outside the boundary knots** of x .

Compared with the cubic spline plotted in p. 27, the natural cubic spline behaves very differently at two ends.

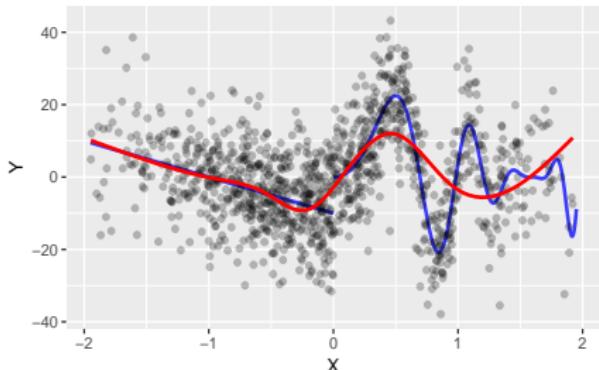


Number and Locations of Knots

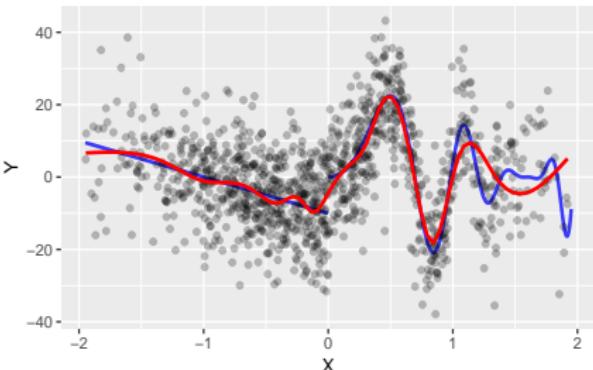
To capture complex nonlinearity, a cubic spline can be made more flexible by increasing the number of knots and/or placing more (less) knots in the regions where the data appear to be volatile (stable).

- We plot in the next page the natural cubic splines with 5, 15, 25, and 35 knots, all uniformly distributed on data quantiles. It can be observed that the curve with 5 knots misses much variation of the true function on the right. When there are more knots, the resulting spline captures more nonlinearity and fits the data on the right better. Yet, they also result in unnecessary fluctuation for the linear function part.
- We also compare the splines with knots evenly distributed on x and those uniformly distributed on certain quantiles; for the latter, there are more (less) knots in the region where the data are dense (sparse).

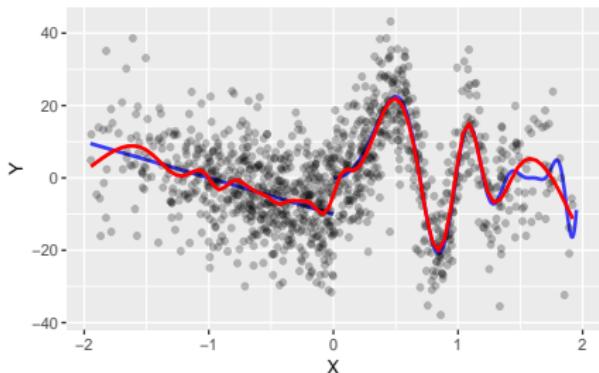
Natural Cubic Spline with 5 knots



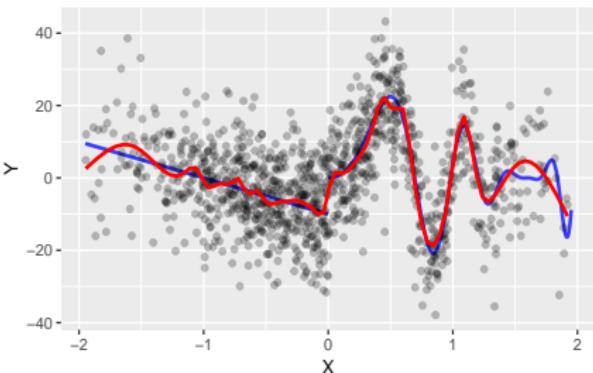
Natural Cubic Spline with 15 knots



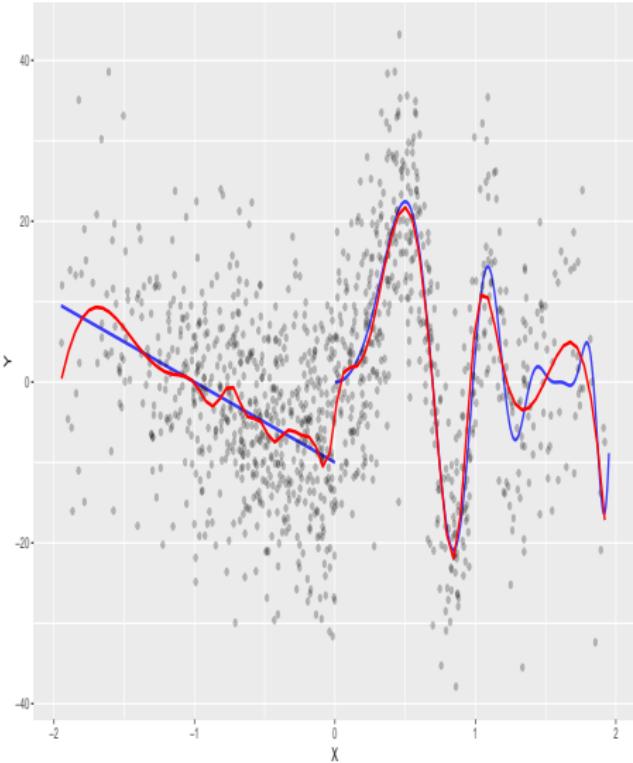
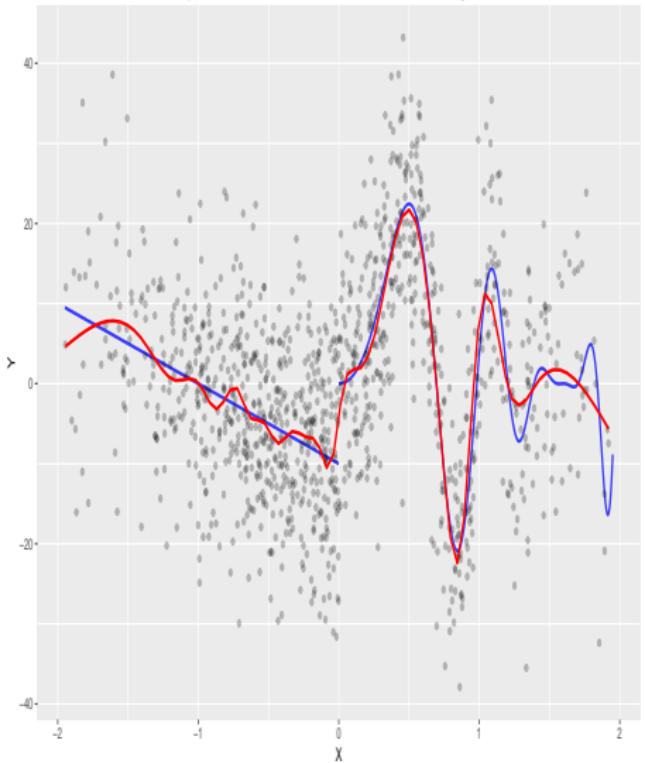
Natural Cubic Spline with 25 knots



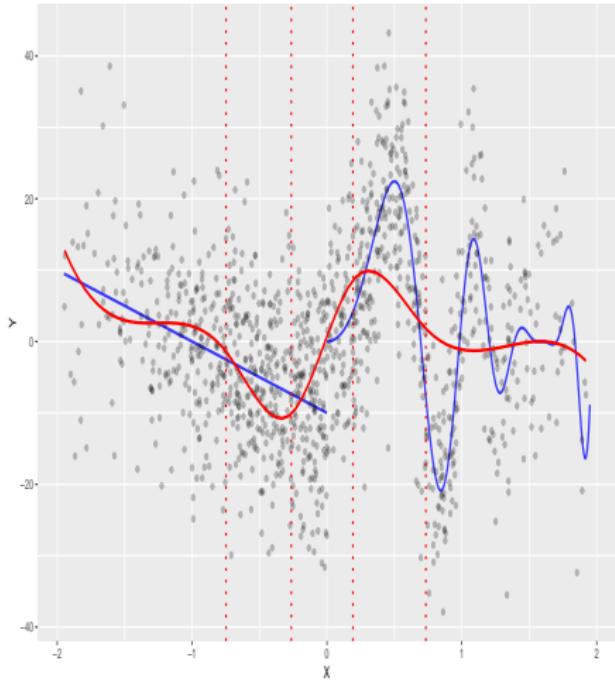
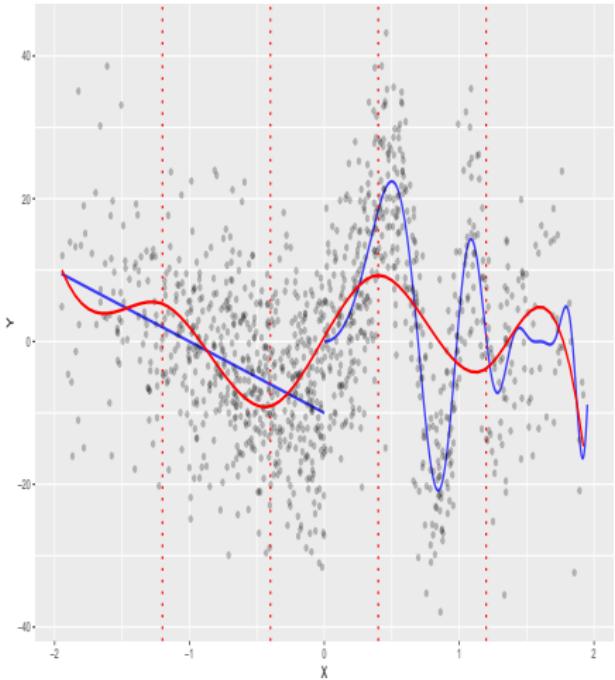
Natural Cubic Spline with 35 knots



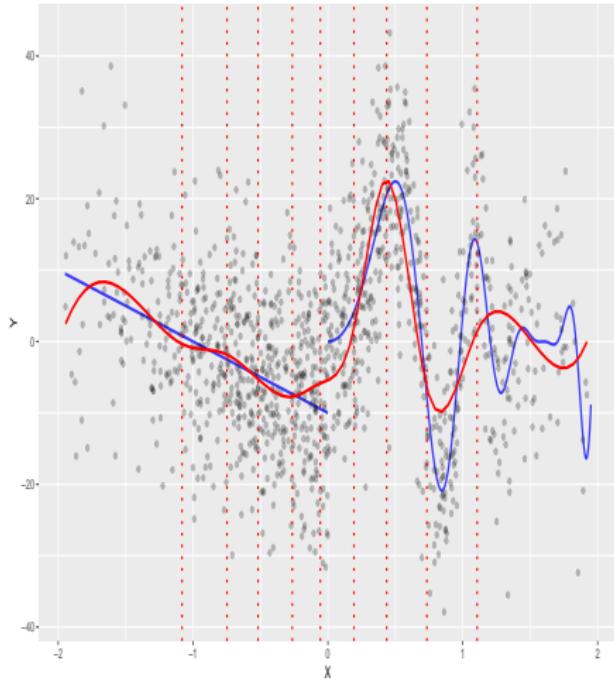
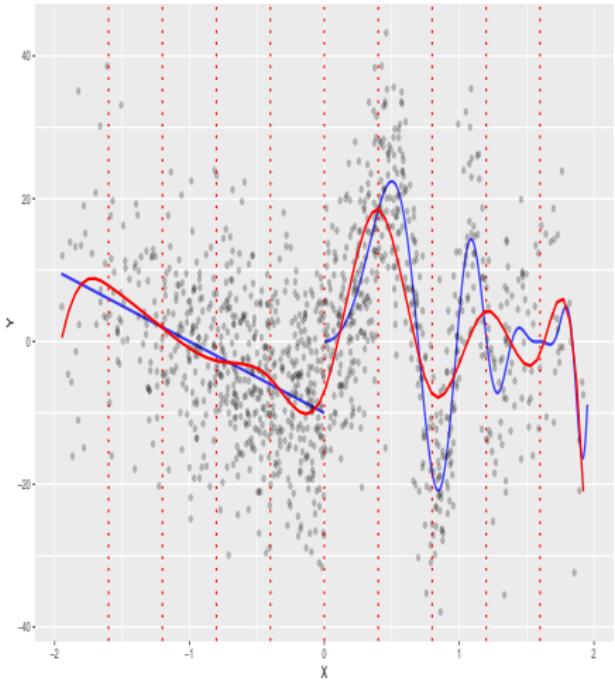
Natural cubic spline with 22 knots determined by 10-fold CV (left) and the cubic spline with 22 knots (right).



Below are the cubic spline with 4 knots evenly distributed on x (left) and the one with 4 knots on data quintiles (right).

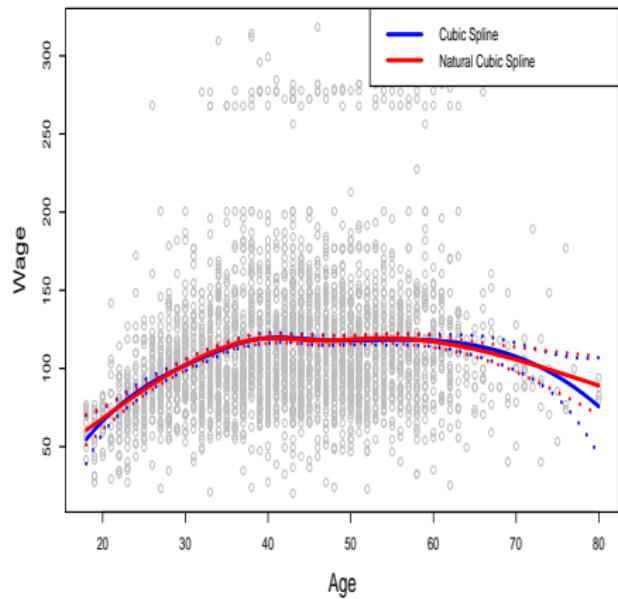
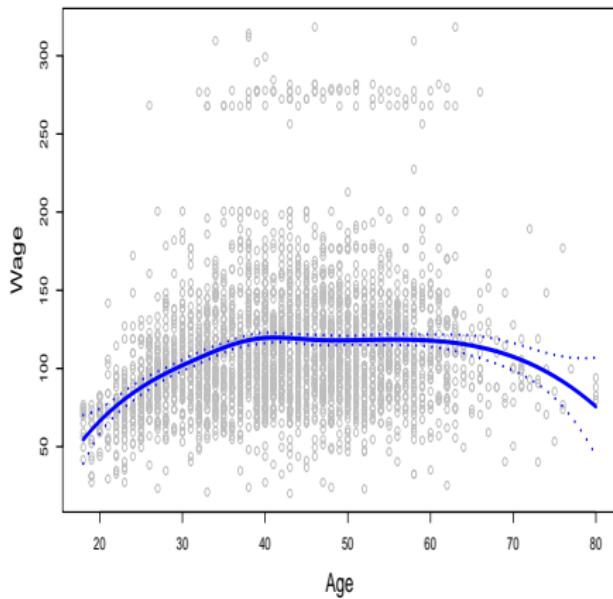


Below are the cubic spline with 9 knots evenly distributed on x (left), and the one with 9 knots on data deciles (right).



Example: Wage in the Mid-Atlantic Region

We fit a cubic spline and a natural cubic spline, with age the predictor and 4 knots determined by data quintiles. Note that the variance of the natural cubic spline (red) is smaller near the boundary of age.



Natural Splines vs. Polynomial Regressions

Polynomial regressions permit more flexibility by increasing the degree of polynomials and hence may create excess variation in some data regions (where only a polynomial with a lower degree is needed). On the other hand, natural splines permit flexibility by adjusting the number and locations of knots (but keeping the degree of spline fixed). As a result, natural splines result in more stable results in general. See pp. 300-301 of JWHT (2021).

Smoothing Splines

To fit a function g to a sample of n observations, it is natural to consider minimizing $\sum_{i=1}^n [y_i - g(x_i)]^2$. Without any constraint, this leads to a highly nonlinear g function, e.g., g may simply interpolate all y_i so that the error sum of squares is zero. This is not a desirable solution because it over fits the data. Instead, we may find a sufficiently **smooth** function that minimizes $\sum_{i=1}^n [y_i - g(x_i)]^2$. To this end, we minimize:

$$\sum_{i=1}^n [y_i - g(x_i)]^2 + \lambda \int g''(t)^2 dt,$$

where λ is a non-negative tuning parameter, and g'' is the second-order derivative of g . The resulting solution is known as a **smoothing spline**.

- $g''(t)$ measures the change of the slope $g'(t)$. If g is smooth, $g'(t)$ would not be changing much, so that the total change of the slope, $\int g''(t)^2 dt$, would be small; otherwise, $g'(t)$ would change a lot, resulting a large integral value. The penalty term $\lambda \int g''(t)^2 dt$ thus controls the **smoothness** (or **roughness**) of the solution.
- When $\lambda = 0$, there is no control of the smoothness of the solution. When λ increases, a rough function will be penalized more, so that the solution will tend to be smoother. In the extreme case that $\lambda \rightarrow \infty$, the solution will have to be perfectly smooth (a straight line), so that the second-order derivative is zero.

It can be shown that the solution to this problem is a piecewise cubic polynomial with (interior and boundary) knots at each of the distinct values of x_1, \dots, x_n and continuous first- and second-order derivatives. Also, it is linear outside the range of x_i . That is, the solution is a **natural cubic spline** with knots at each of x_1, \dots, x_n . See p. 302 of JWHT (2021).

A smoothing spline is, however, **not** the same as the natural spline with the fixed knots at x_1, \dots, x_n computed from the basis function approach, but it is a **shrunken version** of such spline, where the level of shrinkage is determined by the smoothing parameter λ ; see the slides below.

To see this, we note that a natural cubic spline can be represented using the basis functions N_j and hence can write the smoothing spline as:

$g(x) = \sum_{j=1}^n N_j(x)\theta_j$. Let \mathbf{N} be the $n \times n$ matrix with $N_{ij} = N_j(x_i)$. Then, $g(x_i)$ is the i^{th} row of $\mathbf{N}\boldsymbol{\theta}$, and the objective function for the smoothing spline can be expressed as:

$$(\mathbf{y} - \mathbf{N}\boldsymbol{\theta})'(\mathbf{y} - \mathbf{N}\boldsymbol{\theta}) + \lambda\boldsymbol{\theta}'\Omega\boldsymbol{\theta},$$

where Ω is the $n \times n$ matrix with $\Omega_{jk} = \int N''_j(t)N''_k(t) dt$. As in the Ridge regression, the minimizer is

$$\hat{\boldsymbol{\theta}} = (\mathbf{N}'\mathbf{N} + \lambda\Omega)^{-1}\mathbf{N}'\mathbf{y}.$$

The vector of the fitted smoothing spline is

$$\hat{\mathbf{g}}_\lambda = \mathbf{N}(\mathbf{N}'\mathbf{N} + \lambda\Omega)^{-1}\mathbf{N}'\mathbf{y} =: \mathbf{S}_\lambda\mathbf{y}.$$

In the context of linear regression with k parameters, recall that the vector of the OLS fitted values is $\hat{\mathbf{y}} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$, where the orthogonal projection matrix $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$ is $n \times n$ with rank k , which is also its trace (the sum of the diagonal elements):

$$\text{trace}(\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}') = \text{trace}(\mathbf{X}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}) = \text{trace}(\mathbf{I}_k) = k.$$

In analogy, we take the trace of $\mathbf{S}_\lambda = \mathbf{N}(\mathbf{N}'\mathbf{N} + \lambda\boldsymbol{\Omega})^{-1}\mathbf{N}'$ as the **effective degrees of freedom** of a smoothing spline, which depends on λ . The value of this trace in effect determines the coefficients for the basis functions that will not be shrunken towards zero, and hence the number of basis functions that really matter in computing the smoothing spline. The number of effective coefficients thus changes with λ .

Choosing the Smoothing Parameter

Although there is no need to determine the number of knots in fitting a smoothing spline, we need to choose the smoothing parameter λ . We may determine λ by cross validation, i.e., the λ that yields the smallest error sum of squares. In the light of leave-one-out cross validation (LOOCV) discussed in Lecture 6, the LOOCV error sum of squares for each λ can be computed using only the smoothing spline fits to all data. As such, we find the λ value that minimizes

$$\frac{1}{n} \sum_{i=1}^n \left(\frac{y_i - \hat{g}_\lambda(x_i)}{1 - s_{\lambda,i}} \right)^2.$$

where $s_{\lambda,i}$ is the i th diagonal element of S_λ .

Local Regressions

Local Regression is a **weighted least squares regression** that computes the fitted value at **each** point x_o using only those data in a **neighborhood** of x_o . To implement local regressions, we must determine the following.

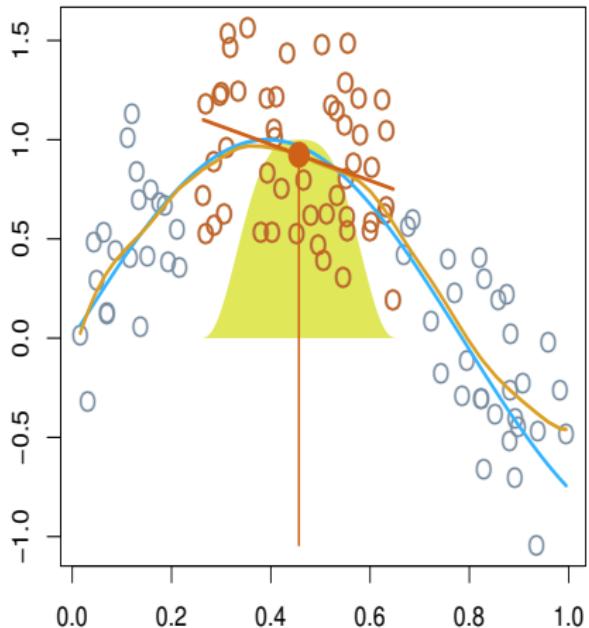
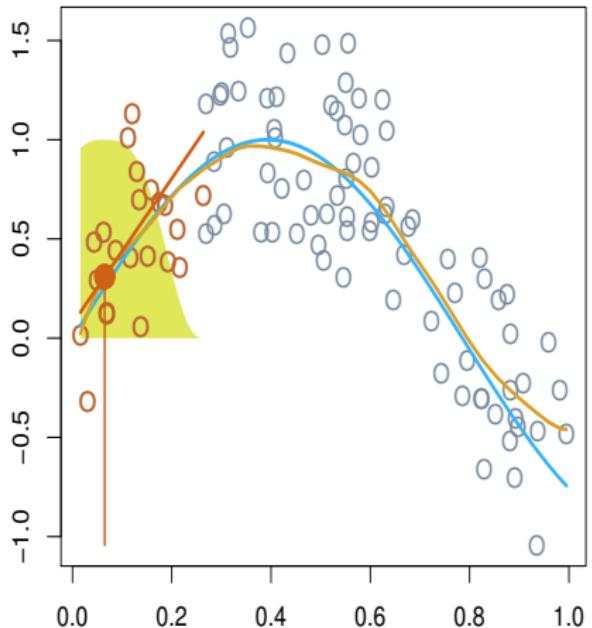
- ① Choose a regression function f : It may be linear, quadratic, or cubic.
- ② Select a fraction (**span**) of the data, say $s\%$, that are closest to x_o .
- ③ Assign **weights** $K(x_i, x_o)$ to the data x_i in the selected span of x_o , where a point closer to x_o is assigned a larger weight, and those outside the span receive zero weight.

For each x_o , we then minimize:

$$\sum_{i=1}^n K(x_i, x_o) [y_i - f(x_i)]^2;$$

the fitted value of x_o is $\hat{f}(x_o)$.

Local Regression



Source: Figure 7.9 of JWHT (2021)

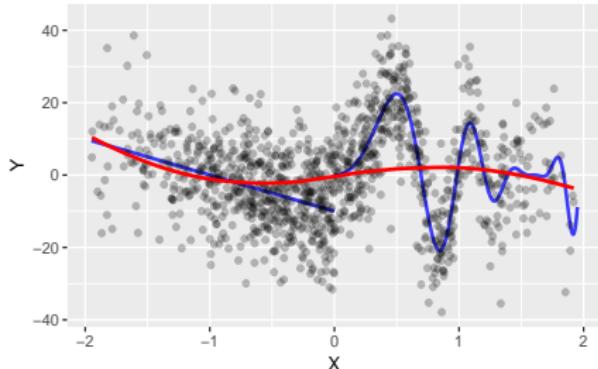
Note that **loess** (locally estimated scatterplot smoothing) of R uses the “tricubic” weights: for x_i in the span of x_o ,

$$K(x_i, x_o) = \left\{1 - [d(x_i, x_o)/\text{max-dist}]\right\}^3,$$

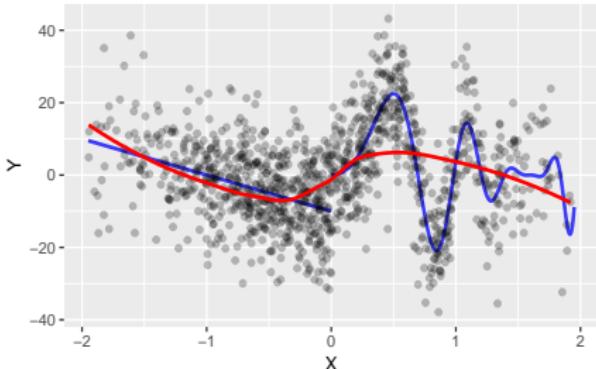
where $d(x_i, x_o)$ is a measure of the distance between x_i in the span and x_o , and max-dist is the largest distance between the point in the span and x_o . Clearly, a point is away from x_o has larger distance and receives a smaller weight; otherwise, it receives a larger weight. As only part of the sample is used for each local fit, a large sample is needed for better fit of data.

Below are the estimated local quadratic regressions with tricubic weights. It is readily observed that a smaller s yields a more **local** and **rough** fit and that a large s leads to a **less local** but **smoother** fit.

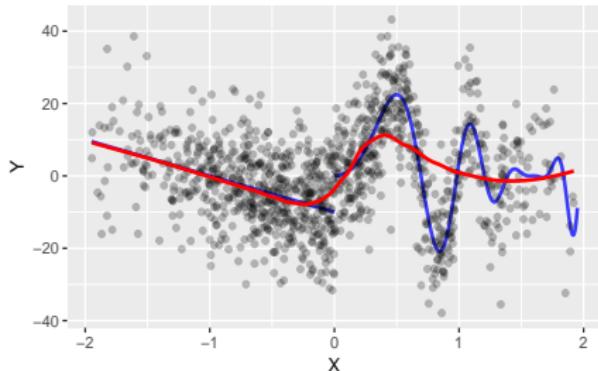
Local Regression with span = 100%



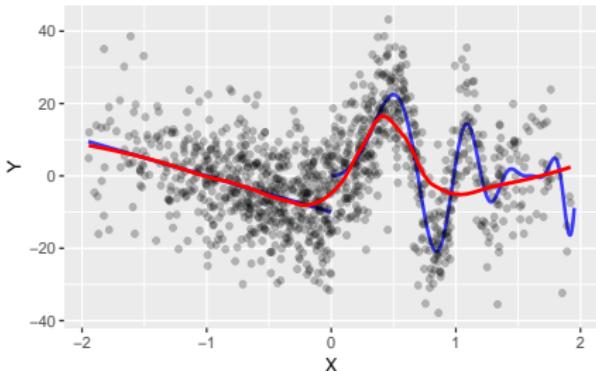
Local Regression with span = 80%



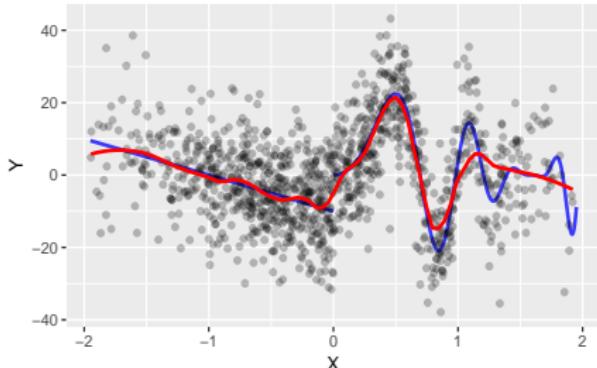
Local Regression with span = 60%



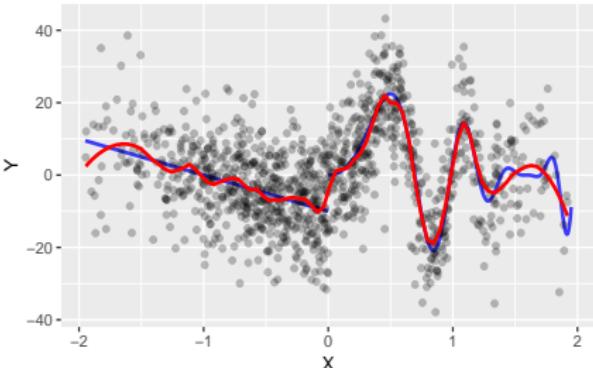
Local Regression with span = 40%



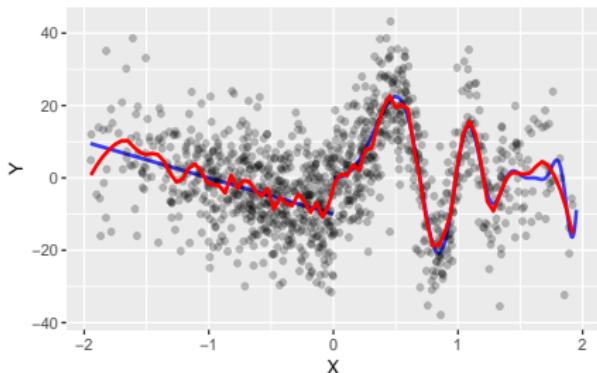
Local Regression with span = 20%



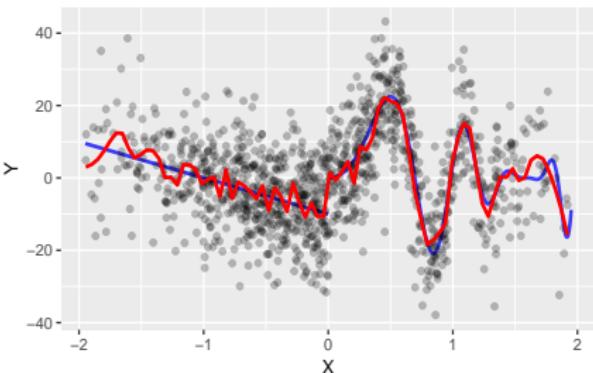
Local Regression with span = 10%



Local Regression with span = 5%

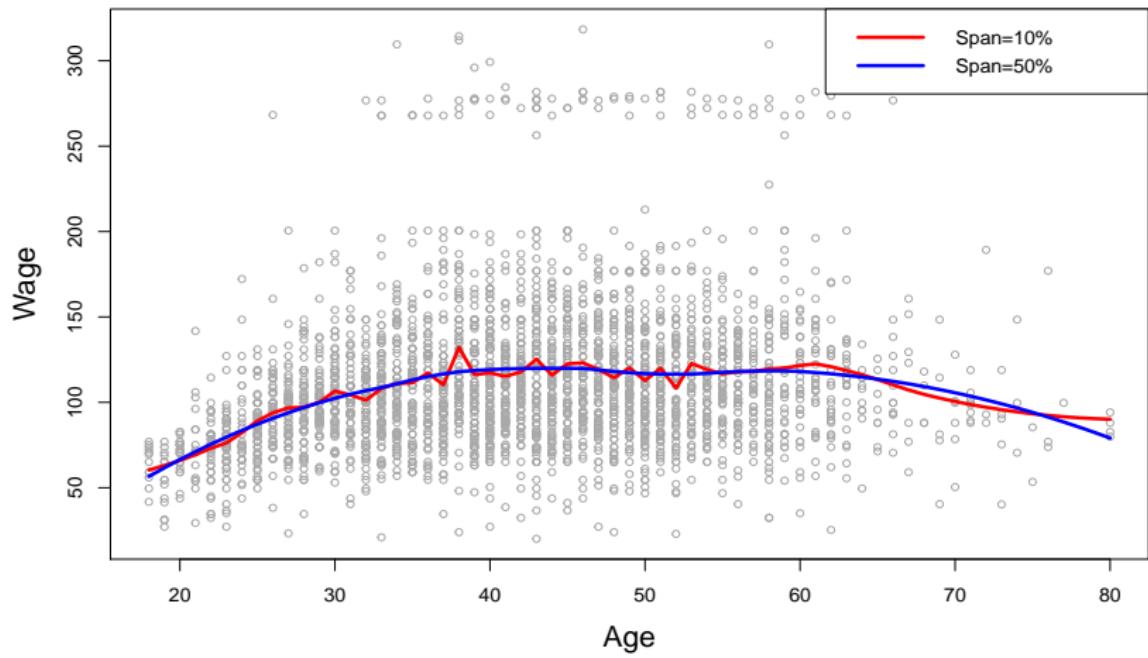


Local Regression with span = 3%



Example: Wage in the Mid-Atlantic region

We estimate local quadratic regressions of wage on age, with the span $s = 10\%$ and 50% . Note the curve with $s = 10\%$ is very rough.



Generalized Additive Models

We now extend the previous analysis to models with multiple predictors x_1, x_2, \dots, x_p . The **Generalized Additive** (GA) model can be expressed as:

$$y_i = \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + \cdots + f_p(x_{ip}) + u_i,$$

where f_j may be a polynomial or cubic spline.

- Instead of jointly modelling all predictors x_1, x_2, \dots, x_p , this model admits nonlinear functions for **each** predictor and adds them up. The additivity form excludes more complex interactions between predictors, but it permits study of the marginal effect of each predictor while holding other predictors fixed.
- Alternatively, we may also consider a function for a small number of predictors, e.g., $f(x_1, x_2)$, and adds such functions up.

For example, a GA model with 2 predictors x_1 and x_2 such that f_1 and f_2 are a cubic spline with respective k knots: ξ_1, \dots, ξ_k and ζ_1, \dots, ζ_k :

$$\begin{aligned}y_i &= \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + u_i \\&= \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i1}^2 + \beta_3 x_{i1}^3 \\&\quad + \beta_4 h^3(x_{i1}, \xi_1) + \cdots + \beta_{3+k} h^3(x_{i1}, \xi_k) \\&\quad + \gamma_1 x_{i2} + \gamma_2 x_{i2}^2 + \gamma_3 x_{i2}^3 \\&\quad + \gamma_4 h^3(x_{i2}, \zeta_1) + \cdots + \gamma_{3+k} h^3(x_{ik}, \zeta_k) + u_i.\end{aligned}$$

More generally, a GA model may employ different functions for x_1 and x_2 , say, a cubic spline for x_1 and a local regression for x_2 .

Note: The GA model is **linear in parameters** in general and can be easily estimated using OLS. Yet, OLS is not applicable when a “smoothing spline” is involved; see JWHT (2021, p. 309).

Example: Wage in the Mid-Atlantic Region

In addition to the variables **wage** and **age** studied earlier, we now also consider the following variables:

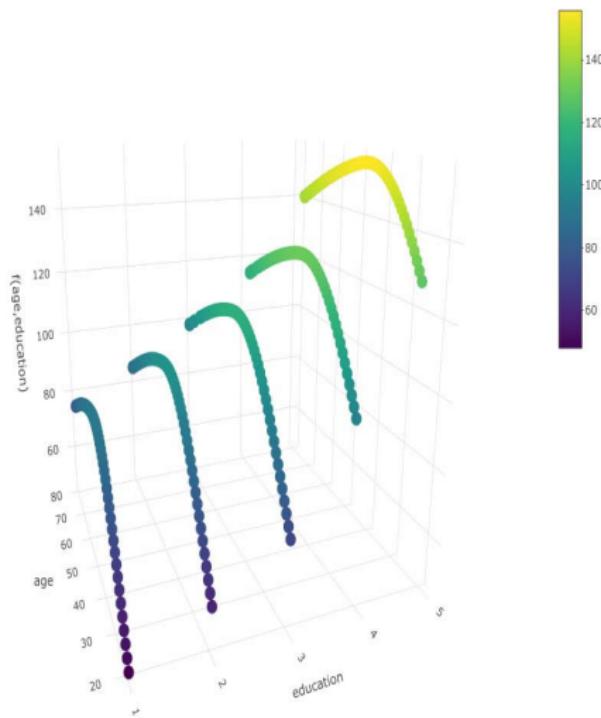
- **year**: Year that wages were recorded (from 2003 to 2009).
- **education**: 5 education levels of the worker: (1) < HS Grad, (2) HS Grad, (3) Some College, (4) College Grad, (5) Advanced Degree.

We first fit the following GA model:

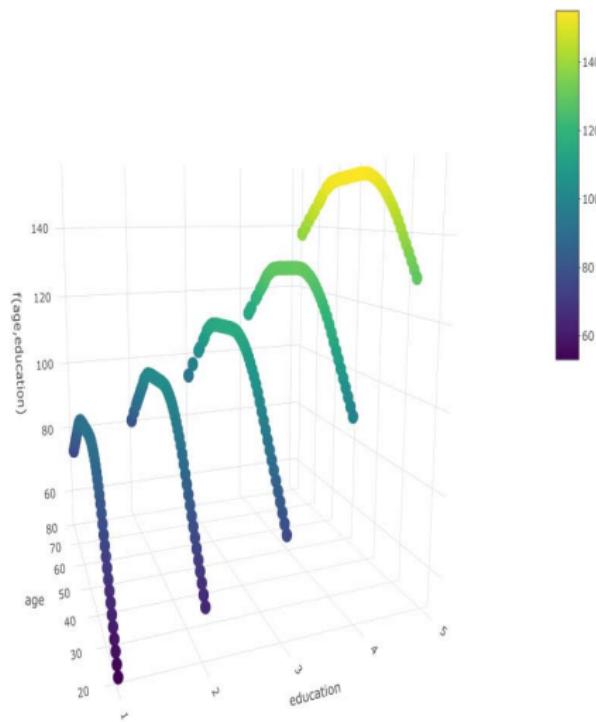
$$\text{wage} = \beta_0 + f_1(\text{age}) + f_2(\text{education}) + u.$$

We consider two cases: (1) f_1 is a natural cubic spline with 3 knots, and f_2 includes dummies for education levels; (2) f_1 is a local regression with span 50%, and f_2 includes education dummies.

Case 1: Natural cubic spline on age (3 knots) and education dummies



Case 2: Local regression on age (span 50%) and education dummies



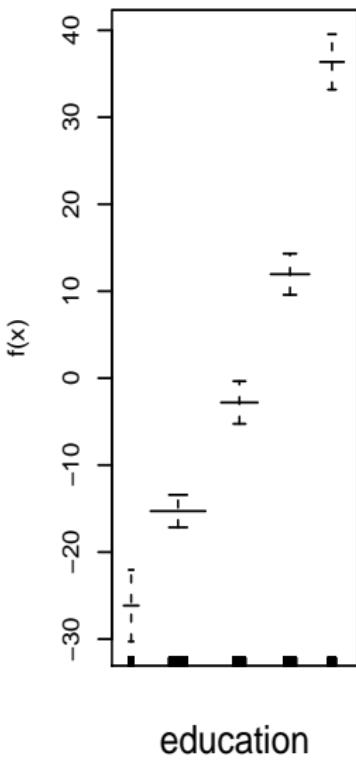
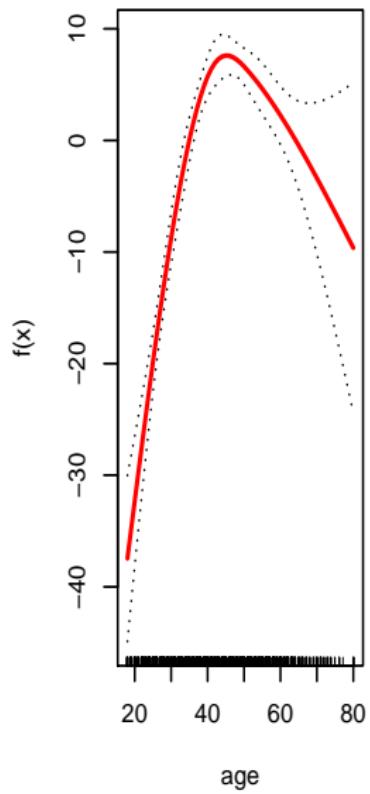
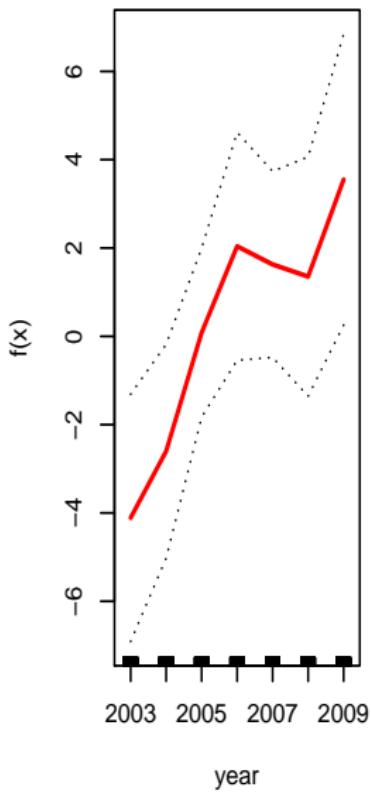
We then fit the following GA model with 3 predictors:

$$\text{wage} = \beta_0 + f_1(\text{age}) + f_2(\text{education}) + f_3(\text{year}) + u.$$

We consider two cases: (1) f_1 and f_3 are natural cubic splines with 3 knots, and f_2 education dummies; (2) f_1 is a local regression with span 50%, and f_2 and f_3 are the same as in (1). The results in the next two slides are summarized below.

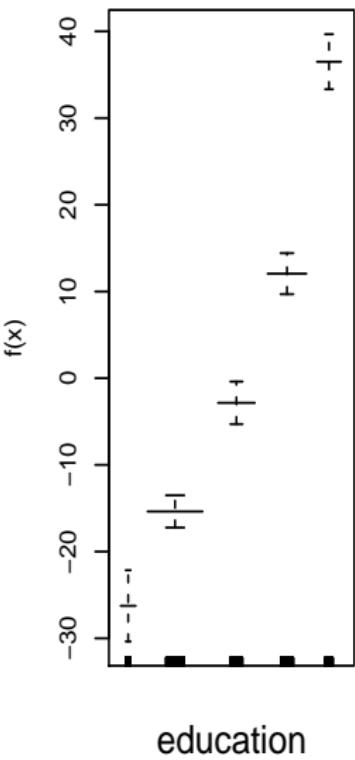
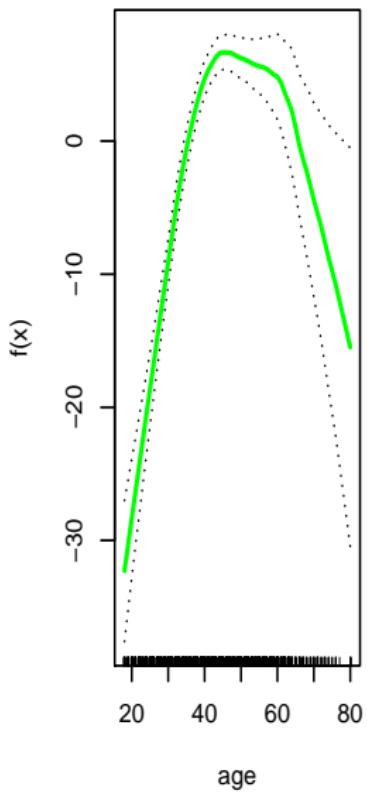
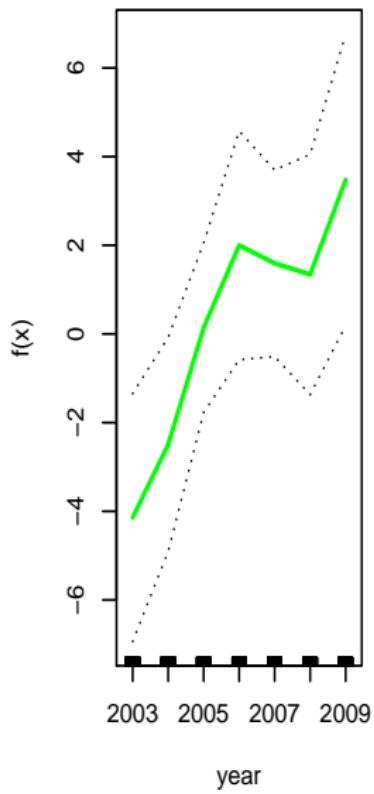
- Left panel: Holding **age** and **education** fixed, **wage** tends to increase with **year**, with a drop during the 2008 financial crisis.
- Middle panel: Holding **year** and **education** fixed, **wage** is higher for the middle-age workers.
- Right panel: Holding **year** and **age** fixed, **wage** increases with **education**.

Case1: Natural cubic spline: age, year; dummies: education



Case 2: Local regression: age; natural cubic spline: year; dummies: edu

1 2 3 4 5



References and Acknowledgement

References

- ① James, G., D. Witten, T. Hastie, and R. Tibshirani (2021). *An Introduction to Statistical Learning, with Applications in R*, 2nd edition, New York: Springer. (JWHT (2021))
- ② Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning*, Second Edition, New York: Springer.

Some of the figures in this presentation are taken from JWHT (2021) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani