Lecture 24: Generalized Linear Model and Nonlinear Regression

Ailin Zhang

2023-07-18

Moving Beyond Linearity

- The truth is never linear! Or almost never. But the linearity assumption is often good enough.
- Generalized linear model (GLM):
  - Structure of GLM
  - Statistical theory and regression diagnostic
  - Application for count data (Contingency Tables)
  - Design-based statistical inference
- Polynomial Regression And Step Functions
- Splines Regression

The Structure of Generalized Linear Models

$$Y_i = \mu_i + \epsilon_i$$

- A generalized linear model (or GLM) consists of three components:

1. Random Component: variability or randomness in the response variable (Y_i). The random component is typically specified using a distribution from the exponential family, which includes several common distributions such as the Gaussian, Bernoulli, Poisson, and multinomial distributions.

2. A linear predictor: a linear function of regressors

$$\eta_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_p X_{ip}$$

3. A smooth and invertible linearizing link function $g(.)$. It transforms the expection of the response variable $\mu_i = E(Y_i)$ to the linear predictor:

$$g(\mu_i) = \eta_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_p X_{ip}$$

Diagnostics for Generalized Linear Models

Most of the diagnostics for linear models extend relatively straightforwardly to GLMs

- Outlier, Leverage, and Influence Diagnostics
- Nonlinearity Diagnostics
- Multicollinearity

Outlier, Leverage, and Influence Diagnostics

- Hat matrix (from final iteration): $H = W^{1/2}X(X^TWX)^{-1}X^TW^{1/2}$, where W is the weight matrix from the final IWLS iteration.
- Residuals:
  - Response residuals: $Y_i - \hat{\mu}_i$ (least useful)
  - Pearson residuals: $\frac{\hat{\phi}^{1/2}(Y_i - \hat{\mu}_i)}{\sqrt{\hat{V}(Y_i|\eta_i)}}$, where $\hat{\phi}$ is the estimated dispersion parameter, and $\hat{V}(Y_i|\eta_i)$ is the conditional variance of response.
  - Standardized Pearson residuals: $\frac{\hat{\phi}^{1/2}(Y_i - \hat{\mu}_i)}{\sqrt{\hat{V}(Y_i|\eta_i)(1-h_i)}}$ correct for the conditional response variation and for the differential leverage of the observations.

Outlier, Leverage, and Influence Diagnostics

- Influence Measures

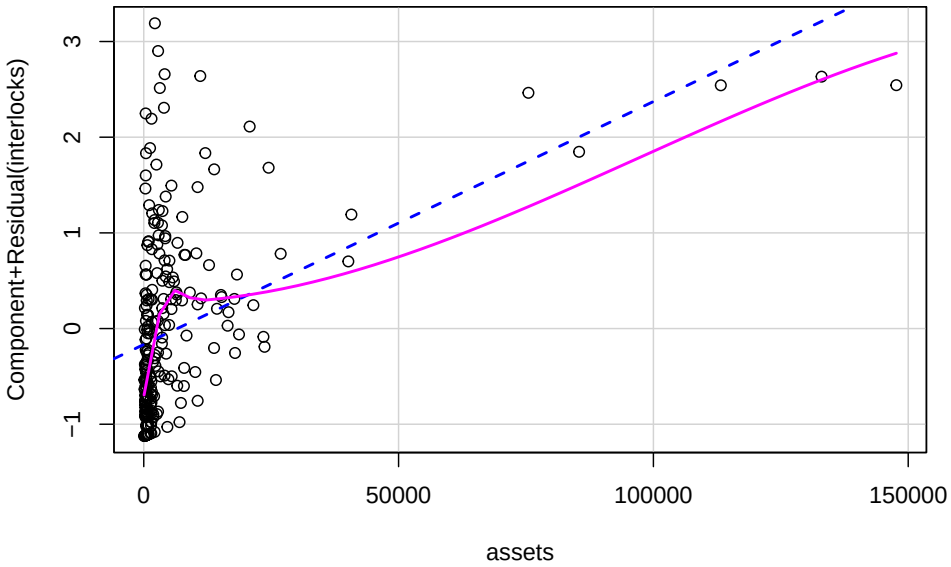
$$D_i = \frac{R_{Pi}^2}{p+1} \times \frac{h_i}{1-h_i}$$

Approximate values of influence measures for individual coefficients (Cook's distance)

Nonlinearity Diagnostics

Component-plus-residual plots also extend straightforwardly to GLMs. Nonparametric smoothing of the resulting scatterplots can be important to interpretation.

```
model1 <- glm(interlocks~.,data = Ornstein,family = "poisson")
crPlots(model1, terms= ~ assets)
```



The partial relationship between number of interlocks and assets is nonlinear, with a much steeper slope at the left than at the right.

Collinearity Diagnostics

```
vif(model1)
##                GVIF Df GVIF^(1/(2*Df))
## assets     7.932489  1        2.816467
## sector    11.110469  9        1.143132
## nation     1.535409  3        1.074082
```

Taking the 2df root of the GVIF is analogous to taking the square root of the VIF and makes generalized variance-inflation factors comparable across dimensions. None of these GVIFs are very large.

Design-based Inference

- The analysis takes into account the complex design features such as stratification, clustering, and unequal probability sampling.
- The goal is to obtain valid estimates of model parameters and appropriate standard errors that reflect the design features of the survey.
- To reflect the features properly, we need to specify the survey design features in the GLM analysis. This includes identifying stratification variables, clustering variables, and weight variables that reflect the sampling probabilities and adjustments for non-response, if applicable.
- Adjust the weights for estimation.
- R package `survey` will help

Wage Data

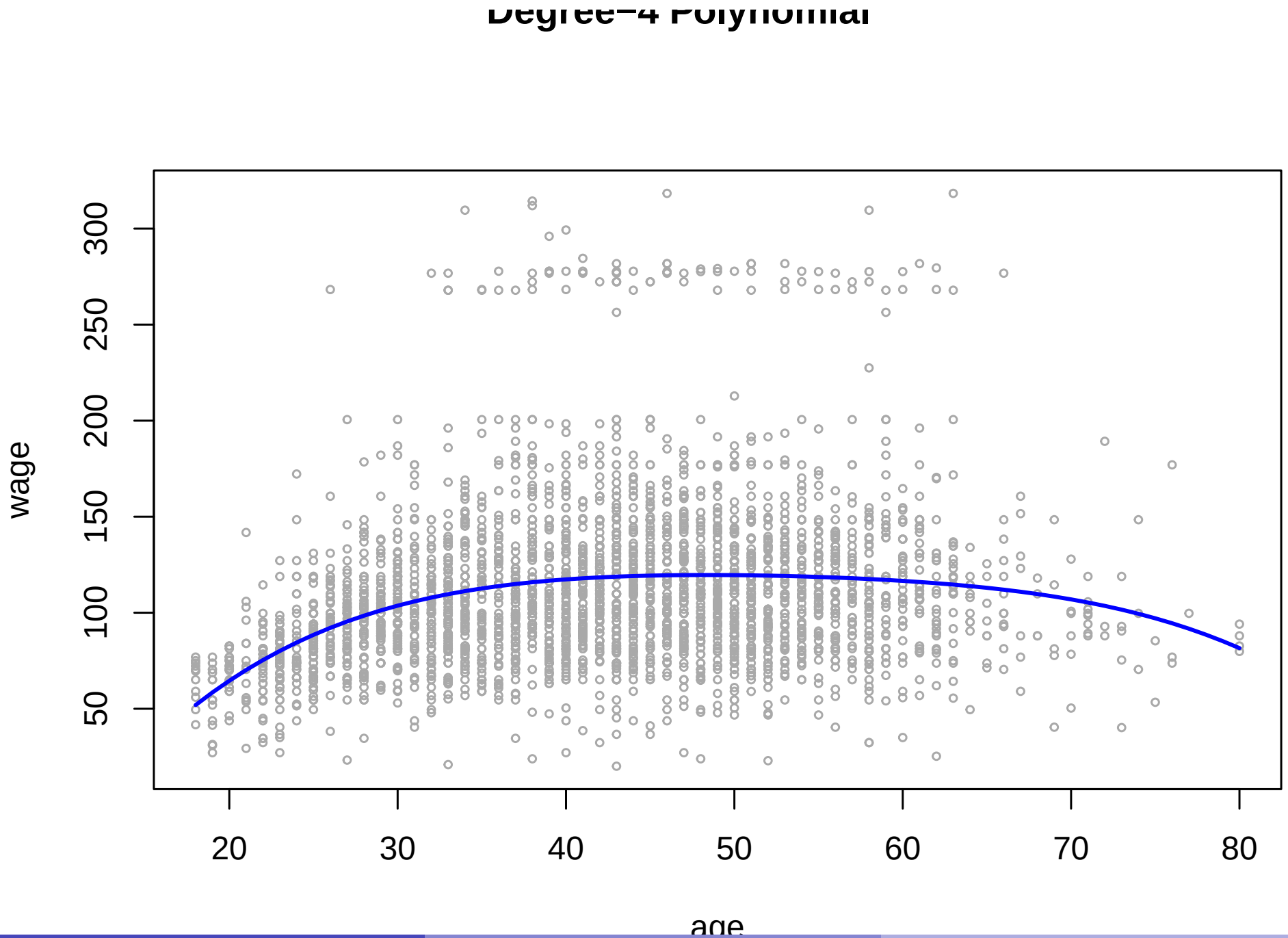
Wage and other data for a group of 3000 male workers in the Mid-Atlantic region.

- year: Year that wage information was recorded
- age: Age of worker
- maritl: A factor with levels 1. Never Married 2. Married 3. Widowed 4. Divorced and 5. Separated indicating marital status
- race: A factor with levels 1. White 2. Black 3. Asian and 4. Other indicating race
- education: A factor with levels 1. $<$ HS Grad 2. HS Grad 3. Some College 4. College Grad and 5. Advanced Degree indicating education level
- region: Region of the country (mid-atlantic only)
- jobclass: A factor with levels 1. Industrial and 2. Information indicating type of job
- health: A factor with levels 1. $<=$Good and 2. $>=$Very Good indicating health level of worker
- health_ins: A factor with levels 1. Yes and 2. No indicating whether worker has health insurance
- logwage: Log of workers wage
- wage: Workers raw wage

Polynomial Regression

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \cdots + \beta_d x_i^d + \epsilon_i$$

```
library(ISLR2)
attach(Wage)
model2 <- lm(wage ~ poly(age, 4), data = Wage)
summary(model2)
##
## Call:
## lm(formula = wage ~ poly(age, 4), data = Wage)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -98.707 -24.626  -4.993  15.217 203.693
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    111.7036     0.7287 153.283  < 2e-16 ***
## poly(age, 4)1  447.0679    39.9148  11.201  < 2e-16 ***
## poly(age, 4)2 -478.3158    39.9148 -11.983  < 2e-16 ***
## poly(age, 4)3  125.5217    39.9148   3.145  0.00168 **
## poly(age, 4)4  -77.9112    39.9148  -1.952  0.05104 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.91 on 2995 degrees of freedom
## Multiple R-squared:  0.08626,    Adjusted R-squared:  0.08504
## F-statistic: 70.69 on 4 and 2995 DF,  p-value: < 2.2e-16
```


Degree−4 Polynomial
wage
50
100
150
200
250
300
20
30
40
50
60
70
80
age

Step Functions

Using polynomial functions of the features as predictors in a linear model imposes a global structure on the non-linear function of X. We can instead use step functions in order to avoid imposing such a global structure.

$$y_i = \beta_0 + \beta_1 C_1(x_i) + \beta_2 C_2(x_i) + \cdots + \beta_K C_K(x_i)$$

where $C_0(x) = I(X < c_1), \ldots, C_j(x) = I(c_{j-1} \leq X < c_j), \ldots, C_K(X) = I(c_K \leq X)$

```
table(cut(age, 4))
##
## (17.9,33.5]   (33.5,49]   (49,64.5] (64.5,80.1]
##         750        1399         779          72
fit <- lm(wage ~ cut(age, 4), data=Wage)
coef(summary(fit))
##                         Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)            94.158392   1.476069 63.789970 0.000000e+00
## cut(age, 4)(33.5,49]   24.053491   1.829431 13.148074 1.982315e-38
## cut(age, 4)(49,64.5]   23.664559   2.067958 11.443444 1.040750e-29
## cut(age, 4)(64.5,80.1]  7.640592   4.987424  1.531972 1.256350e-01
```

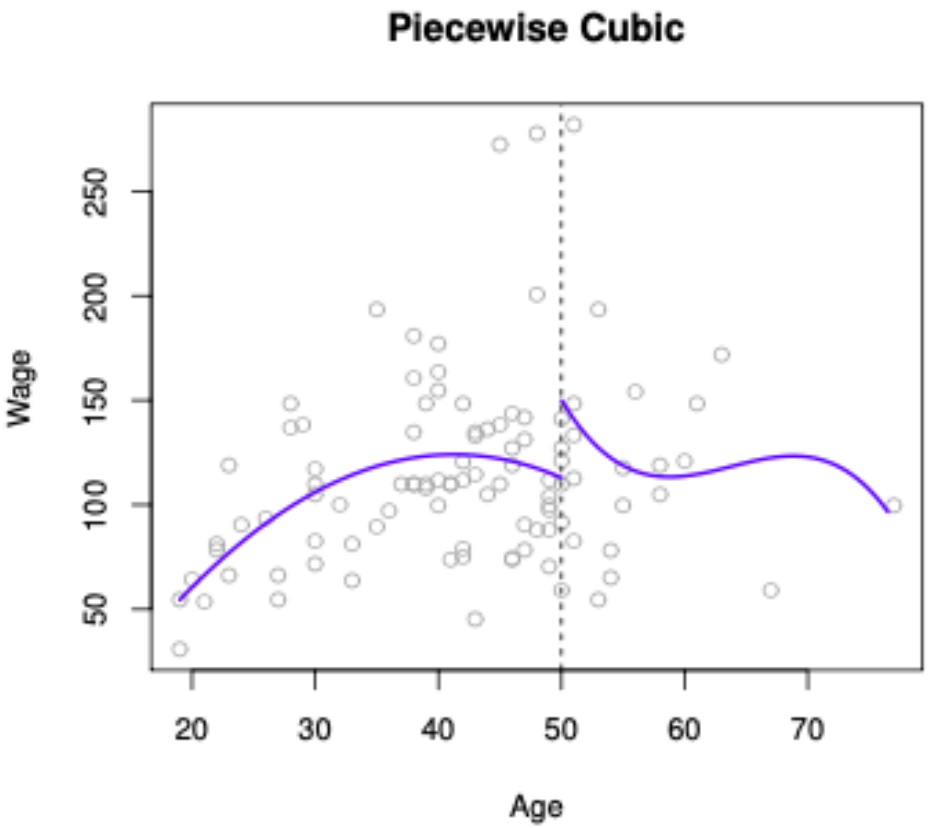
Piecewise Polynomials

A piecewise cubic polynomial with a single knot at a point c takes the form:

$$y_i = \begin{cases} \beta_{01} + \beta_{11} x_i + \beta_{21} x_i^2 + \beta_{31} x_i^3 + \epsilon_i & \text{if } x_i < c \\ \beta_{02} + \beta_{12} x_i + \beta_{22} x_i^2 + \beta_{32} x_i^3 + \epsilon_i & \text{if } x_i \geq c \end{cases}$$

- Each of these polynomial functions can be fit using least squares applied to simple functions of the original cubic polynomial regression.
- If we place K different knots throughout the range of X, then we will end up fitting K + 1 different cubic polynomials. (4K+4 parameters)

Piecewise Polynomials



The function is discontinuous and looks unrealistic.

Constraints and Splines

- Constraints: the prediction needs to be continuous, so for the derivatives.
- Each constraint that we impose on the piecewise polynomials effectively frees up one degree of freedom
- Knots: The points at which the lines join are called knots.
- Splines: a degree-d spline is that it is a piecewise degree-d polynomial, with continuity in derivatives up to degree d $-$ 1 at each knot.

Cubic Splines

Given a set of data points $(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$ where, a cubic spline $S(x)$ is a function that satisfies the following conditions:

1. On each interval $[x_i, x_{i+1}]$, $S(x)$ is a cubic polynomial.
2. $S(x)$ is continuous and has continuous first and second derivatives at all x_i.

The cubic spline function on the interval $[x_i, x_{i+1}]$ can be written as:

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$$

where a_i, b_i, c_i, d_i are the coefficients of the cubic polynomial on the interval $[x_i, x_{i+1}]$.

Cubic Splines

A cubic spline with K knots can be modeled as:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i + \beta_4 h(x_i, \xi_1) + \cdots + \beta_{K+3} h(x_i, \xi_K) + \epsilon_i$$

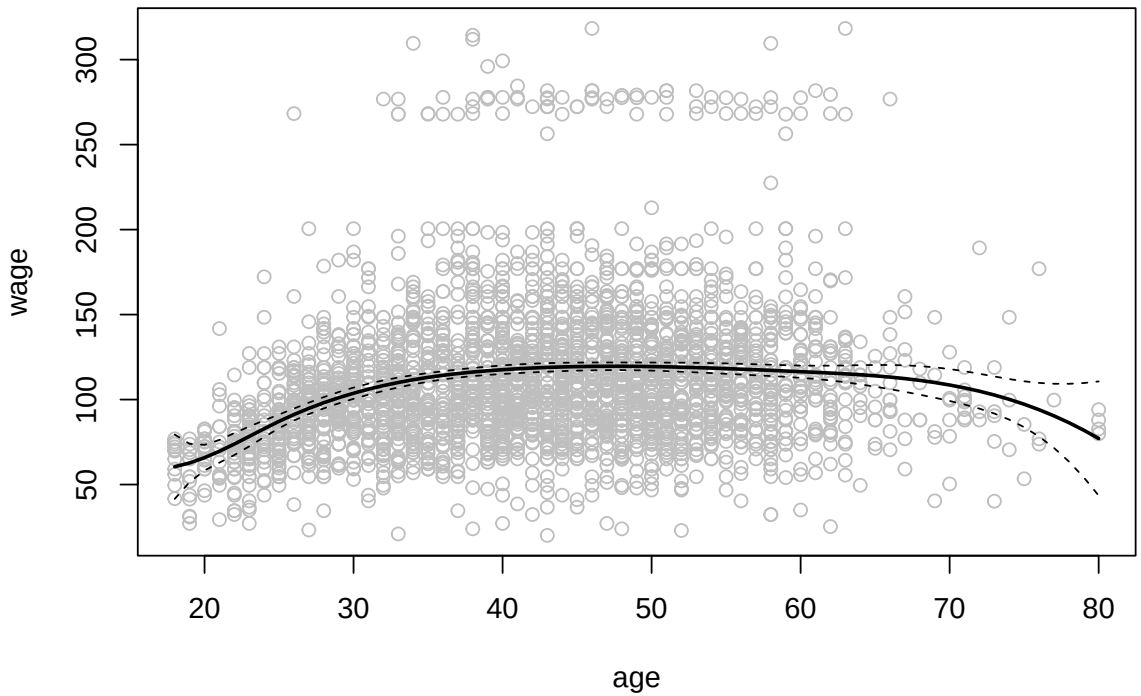
where $h(x, \xi) = (x - \xi)_+^3 = \begin{cases} (x-\xi)^3 & \text{if } x > \xi \\ 0 & \text{otherwise} \end{cases}$

Fitting a cubic spline with K knots uses K+4 degrees of freedom.

- Natural Cubic Spline: At the endpoints x_0 and x_n, $S''(x_0) = S''(x_n) = 0$ (-2 df)
- Clamped Cubic Spline: $S'(x_0) = S'(x_n)$ (-1 df)

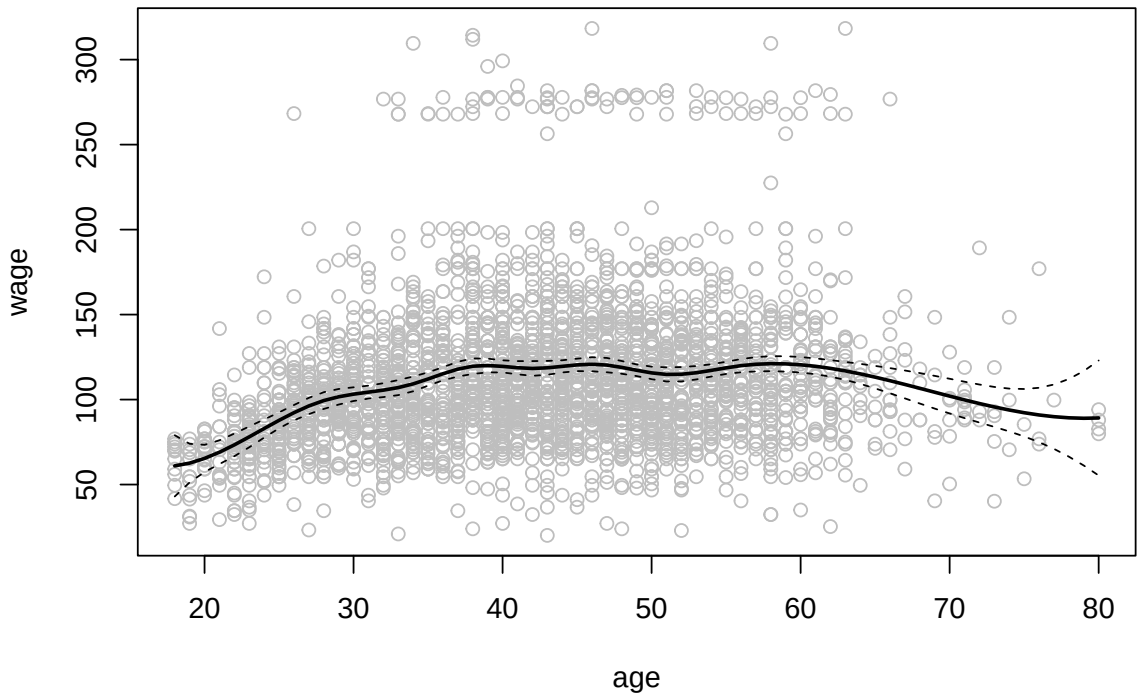
Cubic Splines

```
library(splines)
fit <- lm(wage ~ bs(age, knots = c(25, 40, 60)), data = Wage)
pred <- predict(fit, newdata = list(age = age.grid), se = T)
plot(age, wage, col = "gray")
lines(age.grid, pred$fit, lwd = 2)
lines(age.grid, pred$fit + 2 * pred$se, lty = "dashed")
lines(age.grid, pred$fit - 2 * pred$se, lty = "dashed")
```



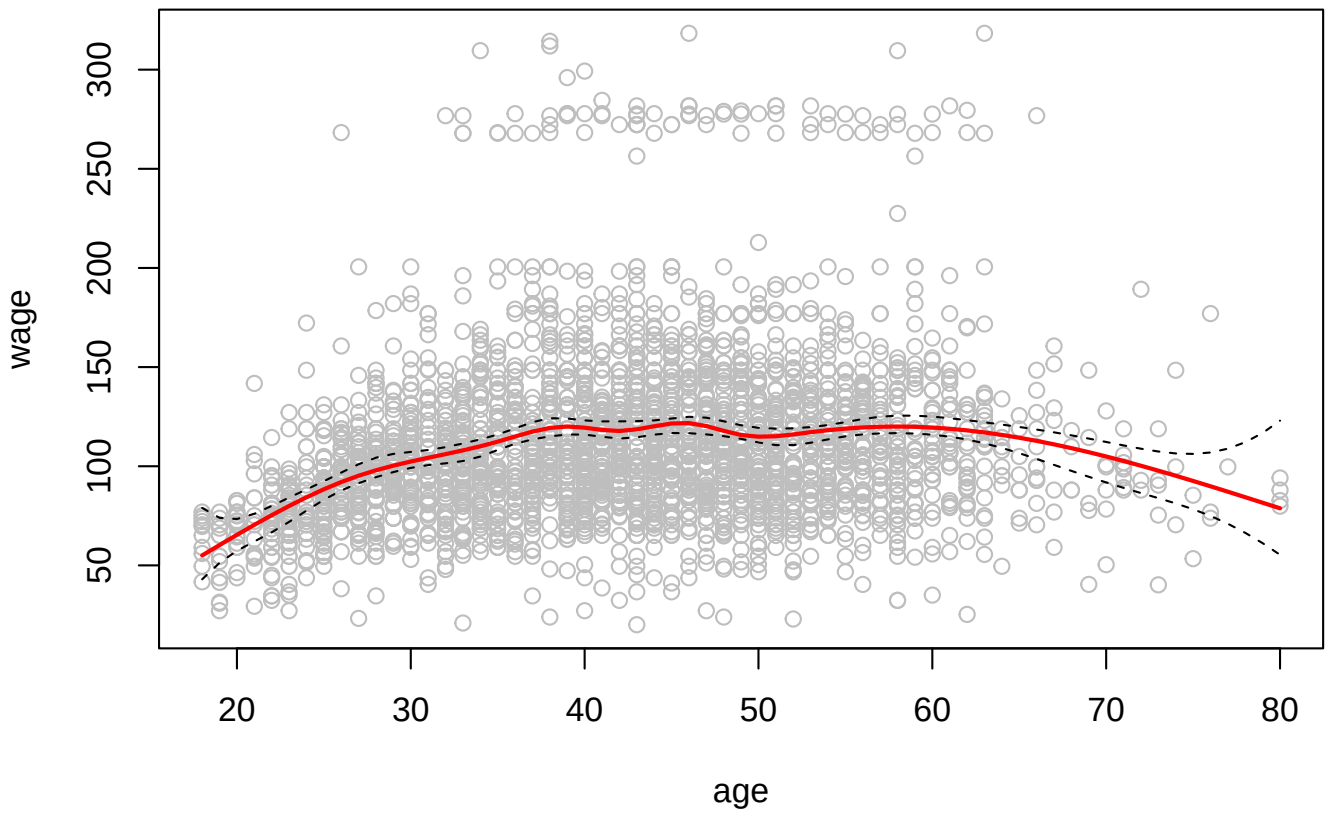
Cubic Splines

```
library(splines)
fit <- lm(wage ~ bs(age, df = 10), data = Wage)
pred <- predict(fit, newdata = list(age = age.grid), se = T)
plot(age, wage, col = "gray")
lines(age.grid, pred$fit, lwd = 2)
lines(age.grid, pred$fit + 2 * pred$se, lty = "dashed")
lines(age.grid, pred$fit - 2 * pred$se, lty = "dashed")
```

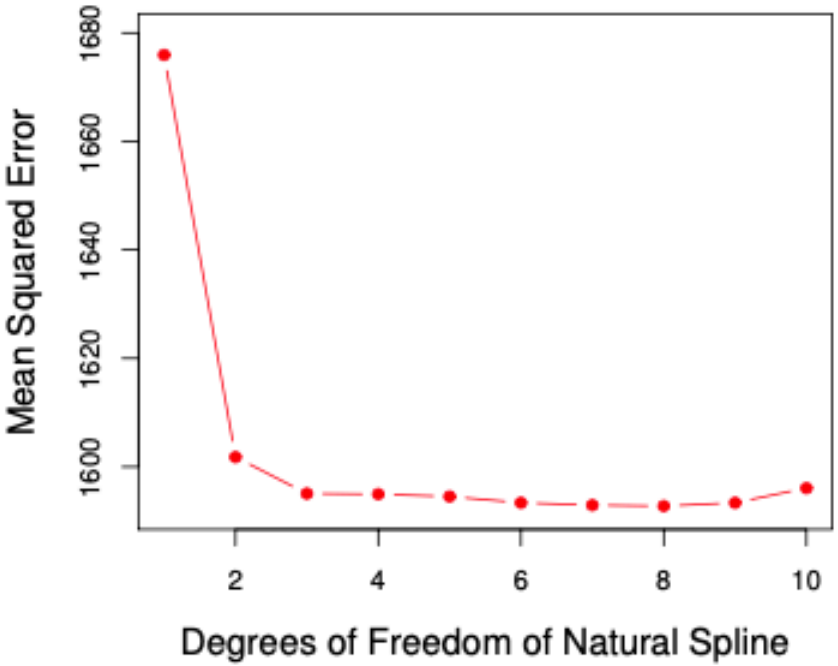


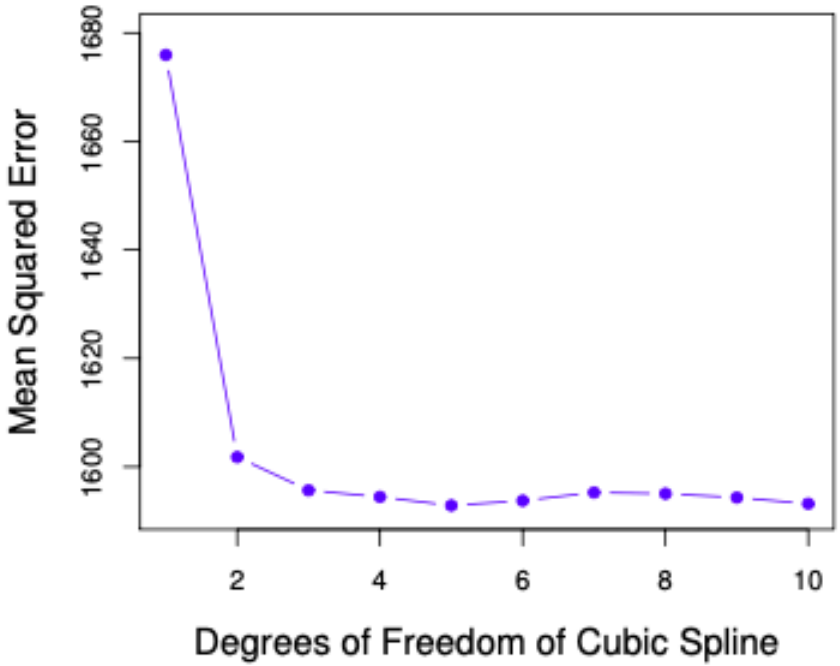
Natural Cubic Splines

```
fit2 <- lm(wage ~ ns(age, df = 10), data = Wage)
pred2 <- predict(fit2, newdata = list(age = age.grid),se = T)
plot(age, wage, col = "gray")
lines(age.grid, pred2$fit, col = "red", lwd = 2)
lines(age.grid, pred$fit + 2 * pred$se, lty = "dashed")
lines(age.grid, pred$fit - 2 * pred$se, lty = "dashed")
```



Choosing the Number of the Knots: Cross Validation

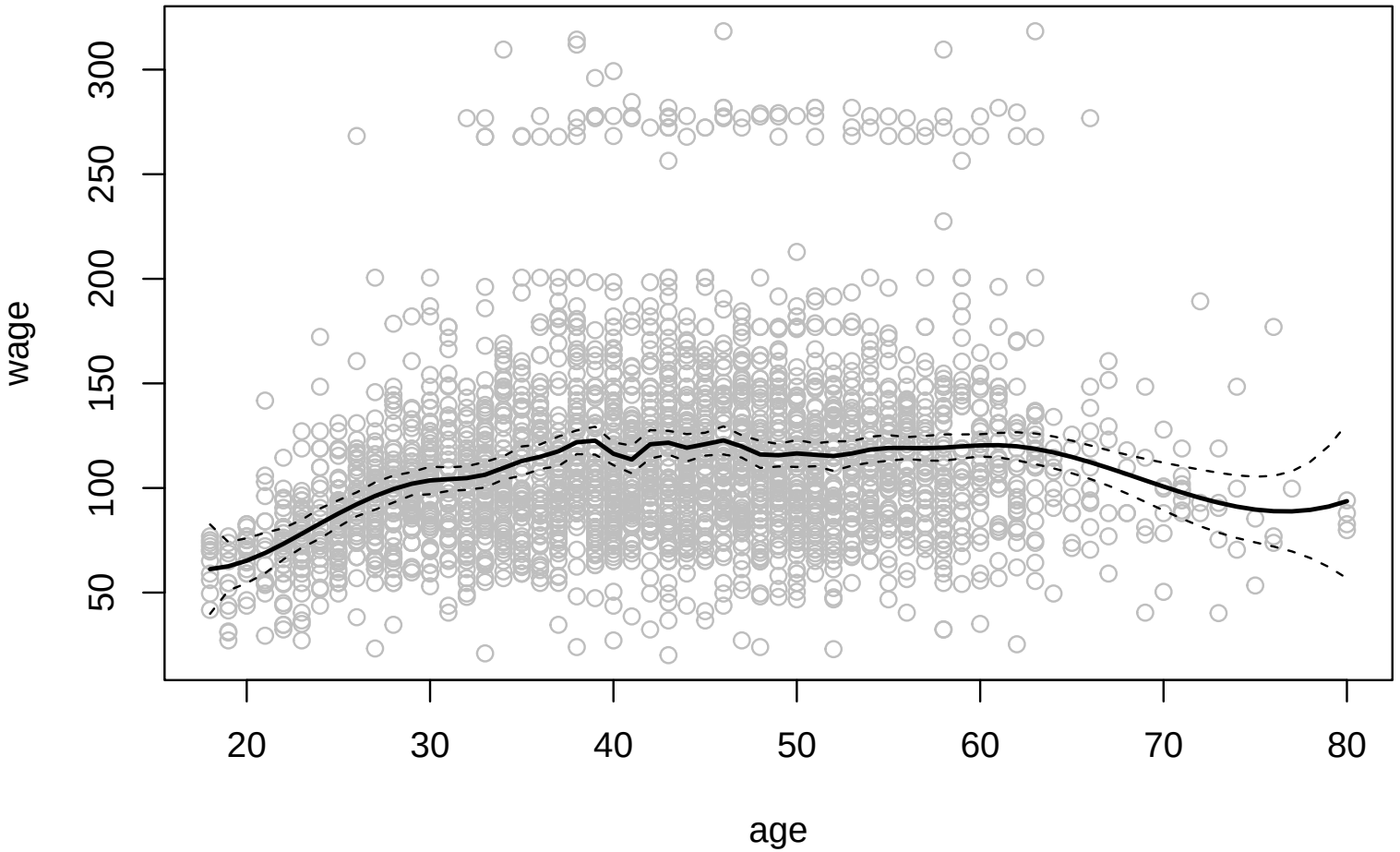




Smoothing Splines

Such a function would overfit the data—it would be far too flexible.

What we really want is a function g that makes RSS small, but that is also smooth.



Smoothing Splines

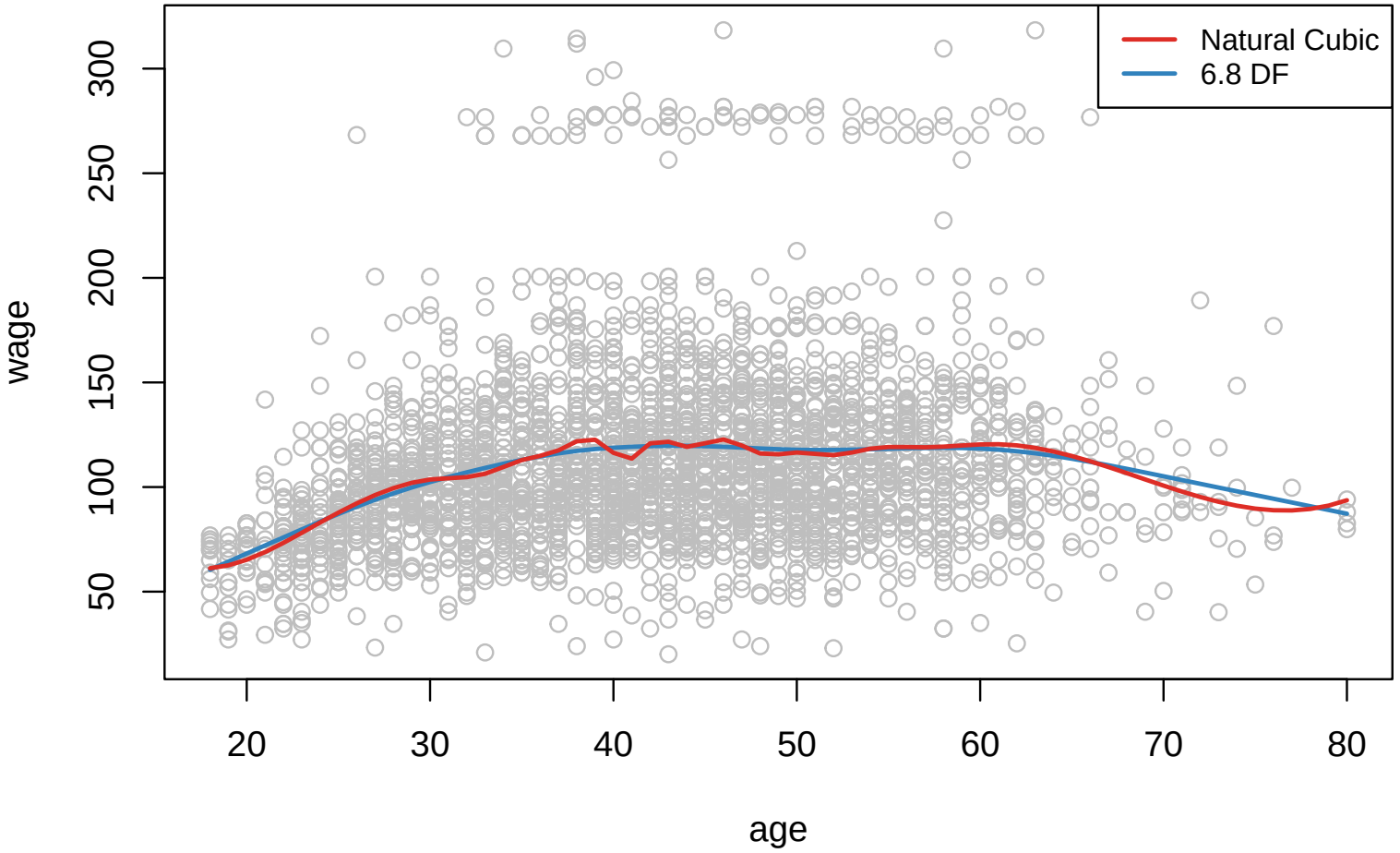
$$\sum_{i=1}^{n}(y_i - g(x_i))^2 + \lambda \int g''(t)^2 dt$$

where λ is a nonnegative tuning parameter. The function g that minimizes the objective function is known as a smoothing spline.

Effective degress of freedom: a measure of the flexibility of a statistical model. It is defined as the trace of the hat matrix.

```
fit2=smooth.spline(age,wage,cv=TRUE)
fit2$df # effective degres of freedom
## [1] 6.794596
```

Smoothing Splines



Local Regression

Local regression is a flexible non-linear functions, which involves computing the fit at a target point x_0 using only the nearby training observations.

Algorithm:

1. Gather the fraction $s = k/n$ of training points whose x_i are closest to x0.
2. Assign a weight $K_{i0} = K(x_i, x_0)$ to each point in this neighborhood, so that the point furthest from x_0 has weight zero, and the closest has the highest weight.
3. Fit a weighted least squares regression of the y_i on the x_i using the aforementioned weights, that minimize:

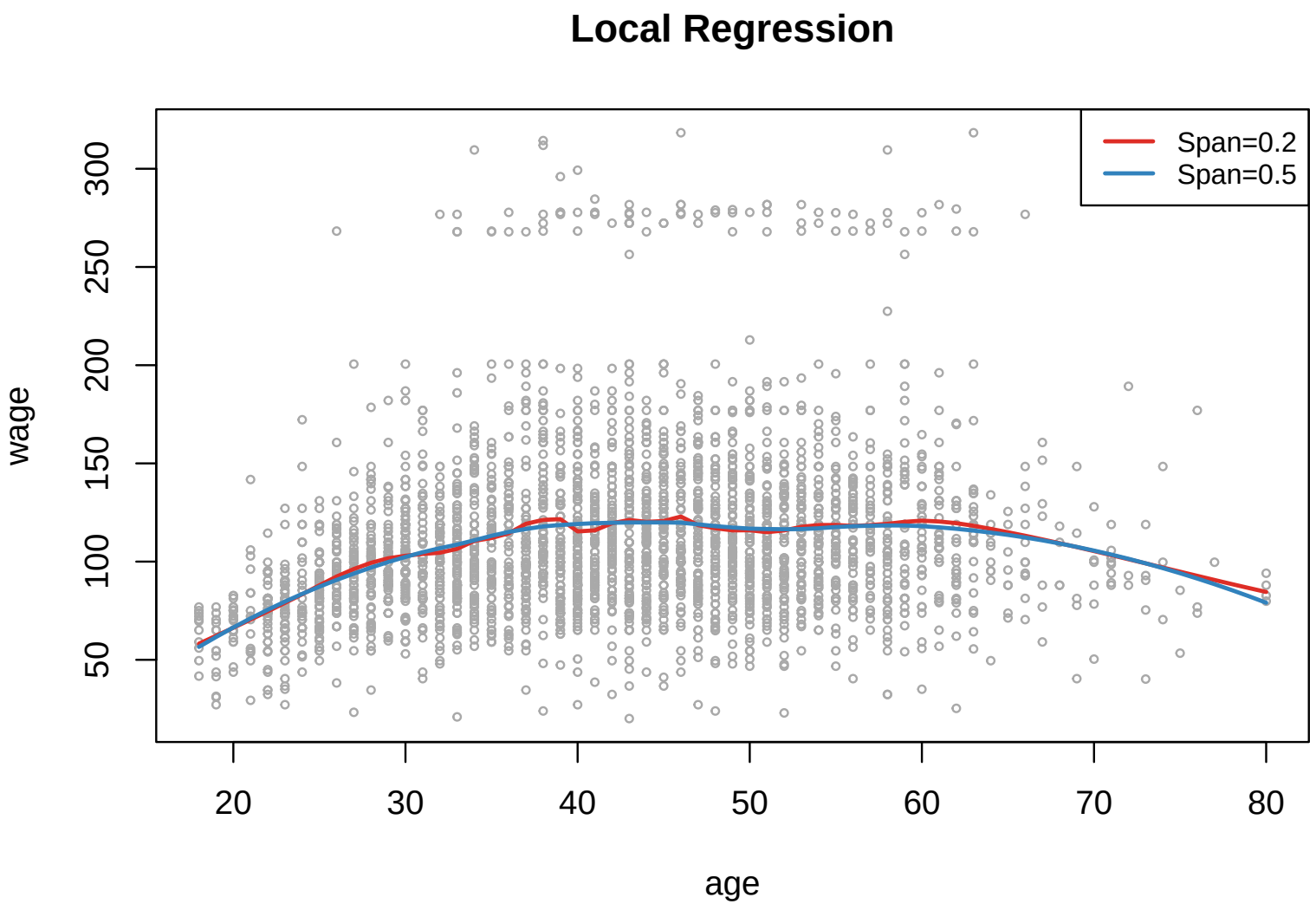
$$\sum_{i=1}^{n} K_{i0}(y_i - \beta_0 - \beta_1 x_i)^2$$

4. The fitted value at x_0 is given by $\hat{f}(x_0) = \hat{\beta}_0 + \hat{\beta}_1 x_0$

- Great for generalization:
  - High dimensions: new weight matrix
  - Global in some variables, but local in another

Local Regression

```
fit=loess(wage~age,span=.2,data=Wage)
```



Generalized Additive Models (GAMs)

Generalized additive models (GAMs) provide a general framework for extending a standard linear model by allowing non-linear functions of each of the variables,while maintaining additivity.

$$y_i = \beta_0 + \sum_{j=1}^{p} f_j(x_{ij}) + \epsilon_i$$

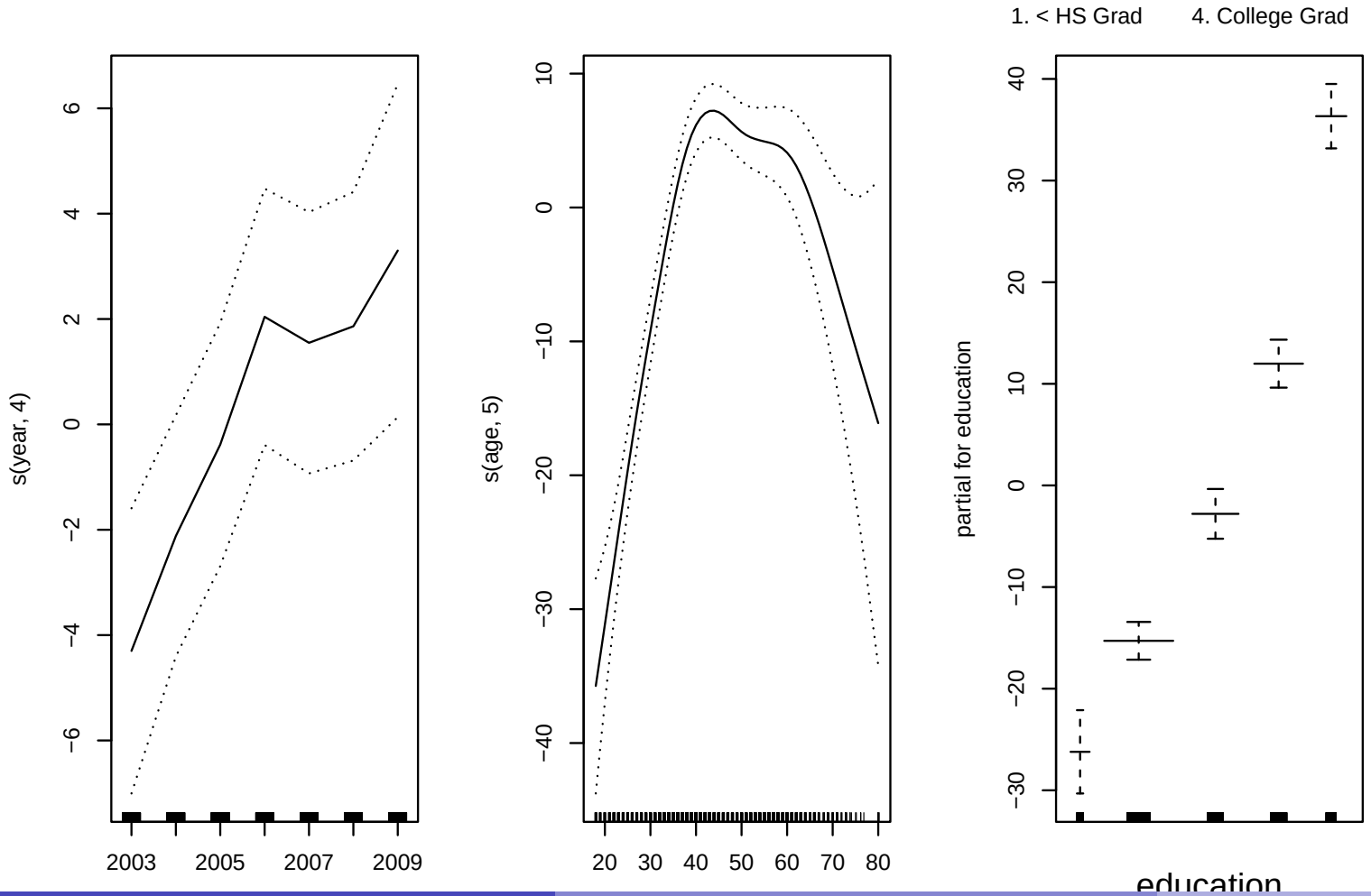
It is called an additive model because we calculate a separate f_j for each X_j, and then add together all of their contributions.

```
library(gam)
model<-gam(wage~s(year,4)+s(age,5)+education,data=Wage)
```

The `s(year,4)` and `s(age,5)` terms are specifying that year and age should be included in the model as smooth splines, with the df $= 4$ and 5

GAM with smooth splines

```
par(mfrow=c(1,3))
plot(model, se=TRUE)
```



GAM with loess

```
model1=gam(wage~s(year,df=4)+lo(age,span =0.7)+education,data=
```