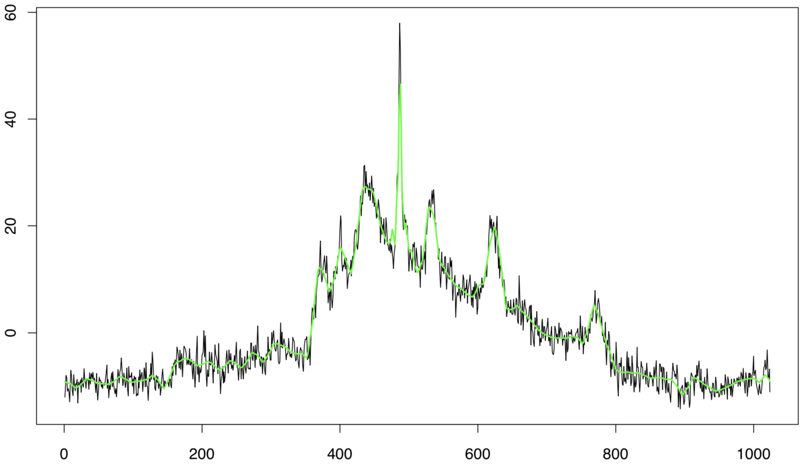
Nonlinear Methods

Damek Davis
School of ORIE, Cornell University
ORIE 4740 Lec 11–12 (March 3, March 8)


60
40
20
0
0
200
400
600
800
1000

Announcements

Recap: What we covered so far

- Concepts: model flexibility; bias-variance tradeoffs
- Linear regression: fitting and evaluation models
- Classification: Logistic regression; KNN
- Model selection and regularization: subset selection; Ridge; Lasso
- Cross-validation

Recap: Supervised Learning

Supervised learning:

- Regression
- Classification
- Regularization & variable selection: apply to both
- CV: estimate test errors to choose models (tunning parameters)

Recap: Linear vs. Nonlinear

Linear techniques:

- Linear regression
- Logistic regression
- *k*-means clustering (Later)
- Principal Component Analysis (Later)

Recap: Linear vs. Nonlinear

Linear techniques:

- Linear regression
- Logistic regression
- k-means clustering (Later)
- Principal Component Analysis (Later)

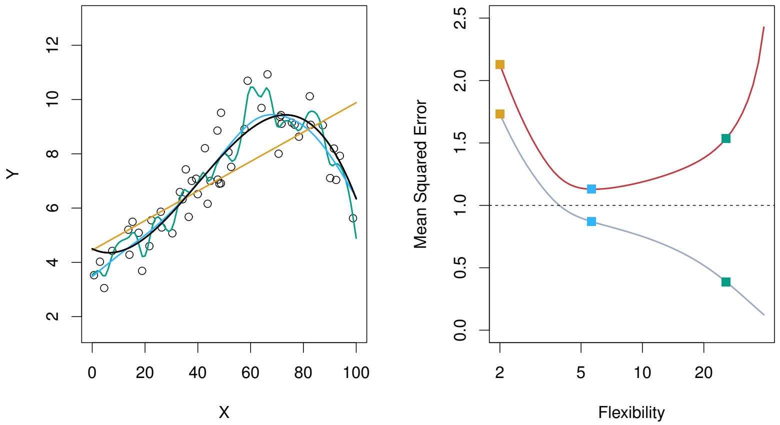
In terms of the bias-variance tradeoff, why might we fit a nonlinear model?

A. To reduce bias

B. To reduce variance

Limits of Linearity

Principle: Not all data is linear—nonlinear models are sometimes a necessity.



- This week: We're going to reduce bias, but possibly increase variance.

Recap: Linear vs. Nonlinear

Linear techniques:

- Linear regression
- Logistic regression
- *k*-means clustering
- Principal Component Analysis

Is KNN a linear or nonlinear technique?

A. Linear
B. Nonlinear

Recap: Linear vs. Nonlinear

Linear techniques:

- Linear regression
- Logistic regression
- k-means clustering
- Principal Component Analysis

Simple extensions of linear techniques:

- Adding high-order and interaction terms
- Converting to dummy variables

Recap: Linear vs. Nonlinear

Linear techniques:

- Linear regression
- Logistic regression
- *k*-means clustering
- Principal Component Analysis

Simple extensions of linear techniques:

- Adding high-order and interaction terms
- Converting to dummy variables

Nonlinear techniques:

- KNN

Recap: Linear vs. Nonlinear

Linear techniques:

- Linear regression
- Logistic regression
- *k*-means clustering
- Principal Component Analysis

Simple extensions of linear techniques:

- Adding high-order and interaction terms
- Converting to dummy variables

Nonlinear techniques:

- KNN

Next:

- More extensions to linear & logistic regression (Today)
- Decision Trees & Random Forest (Next)

Recap: Linear vs. Nonlinear

Linear techniques:

- Linear regression
- Logistic regression
- *k*-means clustering
- Principal Component Analysis

Simple extensions of linear techniques:

- Adding high-order and interaction terms
- Converting to dummy variables

Nonlinear techniques:

- KNN

Next:

- More extensions to linear & logistic regression (Today)
- Decision Trees & Random Forest (Next)

Outside the Scope:

- Neural networks (hard to fit computationally AND hard to analyze statistically)

Beyond Linear Regression and Logistic Regression

Goal: Learn a few classes of nonlinear models.

Beyond Linear Regression and Logistic Regression

Goal: Learn a few classes of nonlinear models.

- Nonlinear models with 1 predictor: $Y = f(X)$

- Nonlinear models with p predictors: $Y = f(X_1, X_2, \ldots, X_p)$

Beyond Linear Regression and Logistic Regression

Goal: Learn a few classes of nonlinear models.

- Nonlinear models with 1 predictor: $Y = f(X)$
  - The basis function approach
  - Regression Splines
  - Smoothing Splines
  - Local Regression (not covered)

- Nonlinear models with p predictors: $Y = f(X_1, X_2, \ldots, X_p)$
  - Generalized Additive Models (GAMs)

The Basis Function Approach (ISLR 7.1–7.3)

Linear regression: $Y \approx \beta_0 + \beta_1 X$

Logistic regression: $\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X$

The Basis Function Approach (ISLR 7.1–7.3)

Linear regression: $Y \approx \beta_0 + \beta_1 X$

Logistic regression: $\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X$

Adding high order terms:

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots$$

The Basis Function Approach (ISLR 7.1–7.3)

Linear regression: $Y \approx \beta_0 + \beta_1 X$

Logistic regression: $\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X$

Adding high order terms:

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots$$

Logarithmic terms:

$$\cdots\cdots \approx \beta_0 + \beta_1 \log(X)$$

The Basis Function Approach (ISLR 7.1–7.3)

Linear regression: $Y \approx \beta_0 + \beta_1 X$

Logistic regression: $\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X$

Adding high order terms:

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots$$

Logarithmic terms:

$$\cdots\cdots \approx \beta_0 + \beta_1 \log(X)$$

How do you perform regression with higher order or log terms?
You transform the...

A. outcome and the response variables and apply least squares.
B. outcome variables and apply least squares.
C. response variables and apply least squares.

The Basis Function Approach (ISLR 7.1–7.3)

Linear regression: $Y \approx \beta_0 + \beta_1 X$

Logistic regression: $\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X$

Adding high order terms:

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots$$

Logarithmic terms:

$$\cdots\cdots \approx \beta_0 + \beta_1 \log(X)$$

More generally:

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

- $b_1(\cdot), \ldots, b_K(\cdot)$: basis functions (pre-specified)

The Basis Function Approach (ISLR 7.1–7.3)

Linear regression: $Y \approx \beta_0 + \beta_1 X$

Logistic regression: $\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X$

Adding high order terms:

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots$$

Logarithmic terms:

$$\cdots\cdots \approx \beta_0 + \beta_1 \log(X)$$

More generally:

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

- $b_1(\cdot), \ldots, b_K(\cdot)$: basis functions (pre-specified)

Principle: To use basis functions, you just transform the predictor table.

Polynomial Basis Functions

Principle: To use basis functions, you just transform the predictor table.

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

Polynomial functions:

$$b_j(x) = x^j, \quad j = 1, \ldots, K$$

This leads to a polynomial model

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_K X^K$$

Example: Wage Dataset

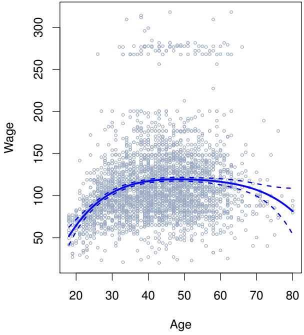
$y_i =$ wage of individual i $\qquad$ $x_i =$ age of individual i

Example: Wage Dataset

$y_i =$ wage of individual i $\qquad$ $x_i =$ age of individual i

Regression with polynomial basis functions up to degree 4:

$$\hat{y}_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \beta_4 x_i^4$$



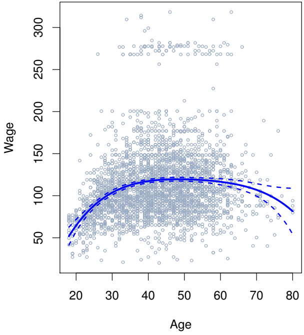
- Dotted lines: 95% confidence intervals of $\hat{y}_i$

Example: Wage Dataset

y_i = wage of individual i $\quad\quad$ x_i = age of individual i

Regression with polynomial basis functions up to degree 4:

$$\hat{y}_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \beta_4 x_i^4$$



Suppose we fit a linear model to a similar data set. Which of the following scenarios cannot be captured by such a model?

- **A.** Wage decreases with age
- **B.** Wage increases with age
- **C.** Wage stays constant as people age
- **D.** Wage is low at birth and death
- **E.** Wage is low at birth and death, but is substantially higher at age 40.

- Dotted lines: 95% confidence intervals of $\hat{y}_i$

Example: Wage Dataset

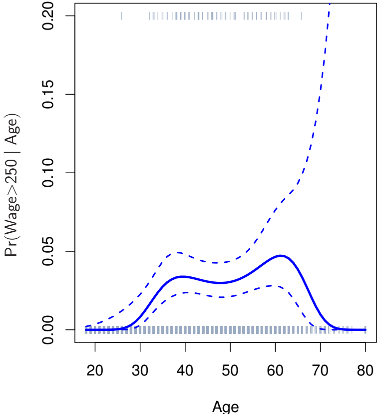
$y_i =$ wage of individual i $\qquad x_i =$ age of individual i

Example: Wage Dataset

y_i = wage of individual i $\qquad$ x_i = age of individual i

Classification with polynomial basis functions up to degree 4:

$$\log\left[\frac{\hat{\Pr}(y_i > 250)}{1 - \hat{\Pr}(y_i > 250)}\right] = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \beta_4 x_i^4$$

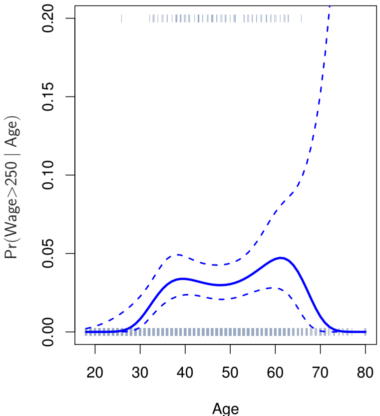


Example: Wage Dataset

$y_i =$ wage of individual i $\qquad$ $x_i =$ age of individual i

Classification with polynomial basis functions up to degree 4:

$$\log\left[\frac{\hat{\Pr}(y_i > 250)}{1 - \hat{\Pr}(y_i > 250)}\right] = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \beta_4 x_i^4$$



Suppose we fit a linear logistic model to a similar data set and then interpret the model. Is it possible for our model to suggest that a 1 year old has wage < 250, that an 80 year old has wage < 250, but a 40 year old has wage > 250?

A. Yes

B. No

Step Basis Functions

Principle: To use basis functions, you just transform the predictor table.

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

Step functions: Given knots $c_1, c_2, \ldots, c_K$

$$\begin{aligned}
b_1(x) &= C_1(x) \triangleq I(c_1 \leq x < c_2) \\
b_2(x) &= C_2(x) \triangleq I(c_2 \leq x < c_3) \\
&\vdots \\
b_{K-1}(x) &= C_{K-1}(x) \triangleq I(c_{K-1} \leq x < c_K) \\
b_K(x) &= C_K(x) \triangleq I(c_K \leq x)
\end{aligned}$$

This leads to a piecewise-constant model

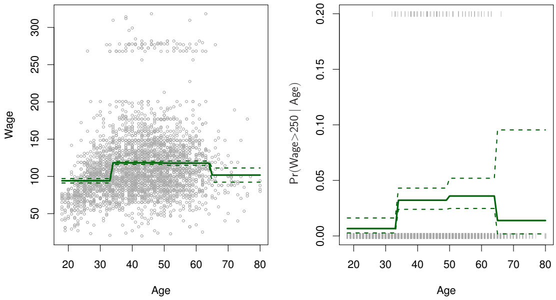
- knots need to be pre-specified

Example: Wage Dataset

y_i = wage of individual i $\qquad$ x_i = age of individual i

Use step basis functions with 2 or 3 knots:

$$y_i \approx \beta_0 + \beta_1 C_1(x_i) + \beta_2 C_2(x_i) \qquad \log\left[\frac{\hat{\Pr}(y_i > 250)}{1 - \hat{\Pr}(y_i > 250)}\right] \approx \beta_0 + \cdots + \beta_3 C_3(x_i)$$

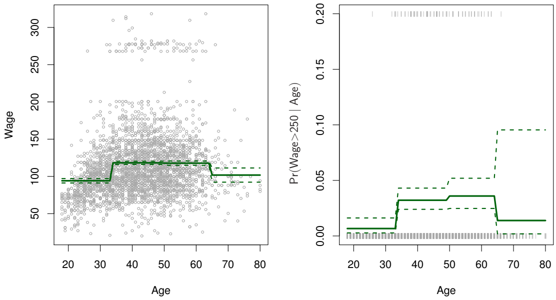


Example: Wage Dataset

y_i = wage of individual i $\qquad$ x_i = age of individual i

Use step basis functions with 2 or 3 knots:

$$y_i \approx \beta_0 + \beta_1 C_1(x_i) + \beta_2 C_2(x_i) \qquad \log\left[\frac{\hat{\Pr}(y_i > 250)}{1 - \hat{\Pr}(y_i > 250)}\right] \approx \beta_0 + \cdots + \beta_3 C_3(x_i)$$



Consider the left figure, which has 2 knots. Suppose we add a new person of age 50 to our dataset and then refit the model. Which of the following is true?

A. Our estimate of β_1 do not change.

B. Our estimate of β_0 and β_2 dop not change.

Basis Function Approach: Fitting and Inference

Principle: To use basis functions, you just transform the predictor table.

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

Basis Function Approach: Fitting and Inference

Principle: To use basis functions, you just transform the predictor table.

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

Fit using least squares

Basis Function Approach: Fitting and Inference

Principle: To use basis functions, you just transform the predictor table.

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

Fit using least squares

Can use all the tools from linear regression:

- Standard errors & confidence intervals for $\hat{\beta}_j$
- p-values for each $\hat{\beta}_j$
- p-values for the entire model

Basis Function Approach: Fitting and Inference

Principle: To use basis functions, you just transform the predictor table.

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

Fit using least squares

Can use all the tools from linear regression:

- Standard errors & confidence intervals for $\hat{\beta}_j$
- p-values for each $\hat{\beta}_j$
- p-values for the entire model

Other choices of basis functions:

- $b_1(x) = \sqrt{x}$
- $b_1(x) = \log(x)$
- Regression Splines (next)
- Based on wavelets or Fourier series (not covered)

Regression Splines (ISLR 7.4)

Using step functions, we fit a piecewise constant model

$$Y \approx \beta_0 + \beta_1 \underbrace{C_1(X)}_{\text{Step Function}} = \begin{cases} \beta_0 & \text{if } X < c \\ \beta_0 + \beta_1 & \text{if } X \geq c \end{cases}$$

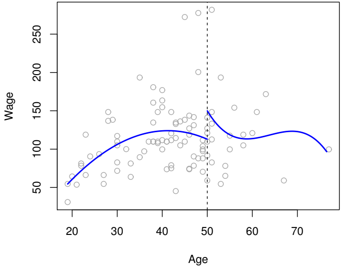
Regression Splines (ISLR 7.4)

Using step functions, we fit a piecewise constant model

$$Y \approx \beta_0 + \beta_1 \underbrace{C_1(X)}_{\text{Step Function}} = \begin{cases} \beta_0 & \text{if } X < c \\ \beta_0 + \beta_1 & \text{if } X \geq c \end{cases}$$

More generally, we can fit a piecewise polynomial model

$$Y \approx \begin{cases} \beta_{01} + \beta_{11}X + \beta_{21}X^2 + \beta_{31}X^3 & \text{if } X < c \\ \beta_{02} + \beta_{12}X + \beta_{22}X^2 + \beta_{32}X^3 & \text{if } X \geq c \end{cases}$$



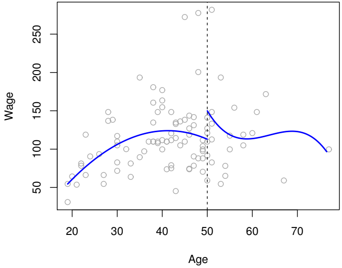
Regression Splines (ISLR 7.4)

Using step functions, we fit a piecewise constant model

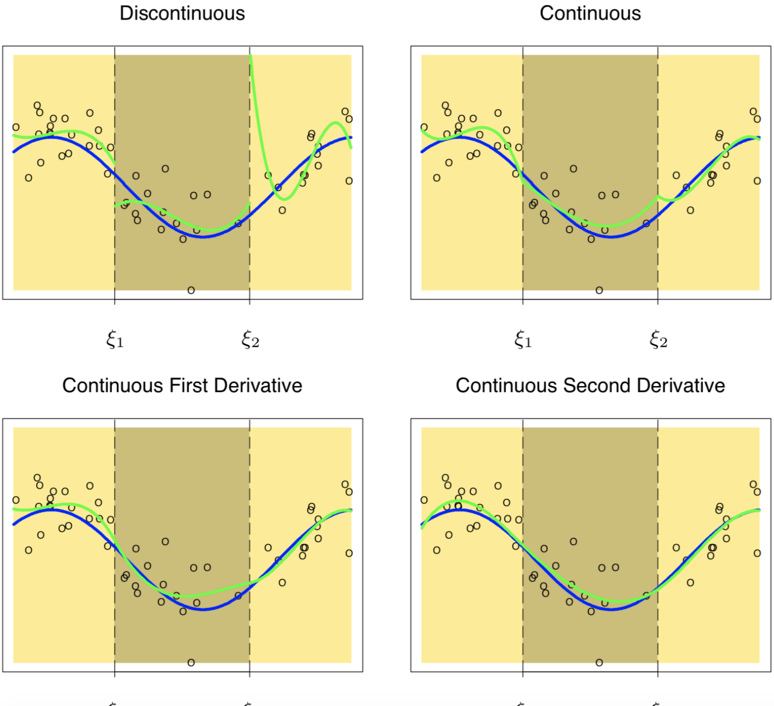
$$Y \approx \beta_0 + \beta_1 \underbrace{C_1(X)}_{\text{Step Function}} = \begin{cases} \beta_0 & \text{if } X < c \\ \beta_0 + \beta_1 & \text{if } X \geq c \end{cases}$$

More generally, we can fit a piecewise polynomial model

$$Y \approx \begin{cases} \beta_{01} + \beta_{11}X + \beta_{21}X^2 + \beta_{31}X^3 & \text{if } X < c \\ \beta_{02} + \beta_{12}X + \beta_{22}X^2 + \beta_{32}X^3 & \text{if } X \geq c \end{cases}$$

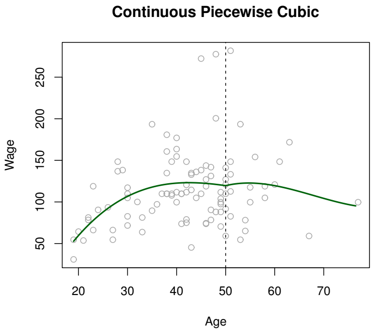


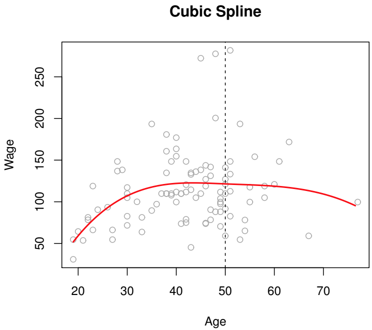
- 8 degrees of freedom (too flexible)


Discontinuous
Continuous
ξ_1
ξ_2
ξ_1
ξ_2
Continuous First Derivative
Continuous Second Derivative

Regression Splines

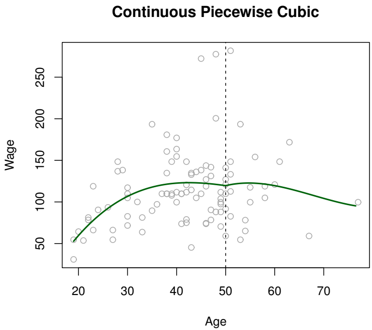
Regression splines:
Piecewise polynomial models that are continuous and smooth at the knots
(smoothness = continuity of derivatives)

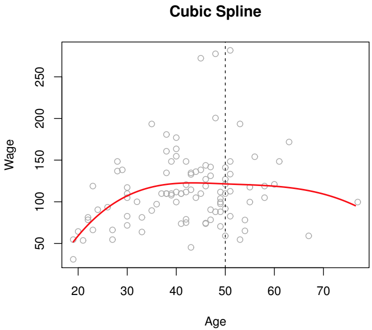




Regression Splines

Regression splines:
Piecewise polynomial models that are continuous and smooth at the knots (smoothness = continuity of derivatives)





Most popular: Cubic splines

- Continuous piecewise cubic models with continuous first two derivatives
- K knots: $K + 4$ degrees of freedom (instead of $4K + 4$)
- Reduce flexibility/variance; increase bias

Cubic Splines

- A cubic splines with one knot at $x = 1$ can be written as
$$Y \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 \max\{(X-1)^3, 0\}.$$

Cubic Splines

- A cubic splines with one knot at $x = 1$ can be written as

$$Y \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 \max\{(X-1)^3, 0\}.$$

- More generally, a cubic splines with K knots at $\xi_1, \xi_2, \ldots, \xi_K$ (i.e., $K + 4$ DF) can be written as

$$Y \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_{K+3} b_{K+3}(X)$$

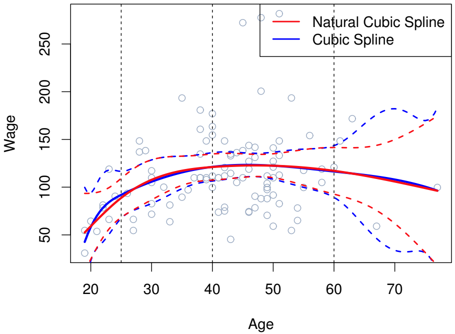
with basis functions

$$\begin{aligned}
b_1(X) &= X \\
b_2(X) &= X^2 \\
b_3(X) &= X^3 \\
b_4(X) &= \max\{(X-\xi_1)^3, 0\} \\
b_5(X) &= \max\{(X-\xi_2)^3, 0\} \\
&\vdots \\
b_{K+3}(X) &= \max\{(X-\xi_K)^3, 0\}
\end{aligned}$$

(cf. ISLR 7.4.3)

Cubic Splines

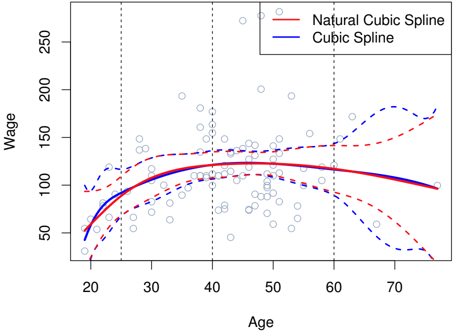
- Hence, cubic splices can be fitted using least squares



- Problem: Cubic splines may appear wild at boundary (conf. int. big).

Cubic Splines

- Hence, cubic splices can be fitted using least squares



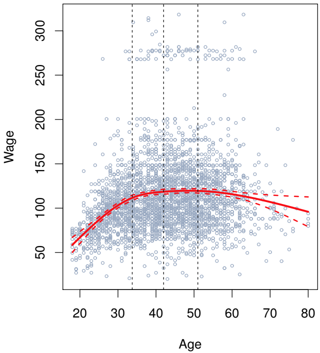
- Problem: Cubic splines may appear wild at boundary (conf. int. big).

- Natural cubic splines: linear at the boundary
(further reduce df/flexibility/variance)

Cubic Splines: Choosing the Knots

Locations of knots:

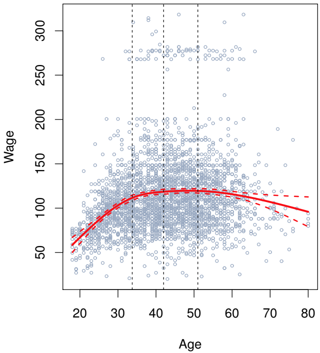
- Placed at uniform quantiles of data



Cubic Splines: Choosing the Knots

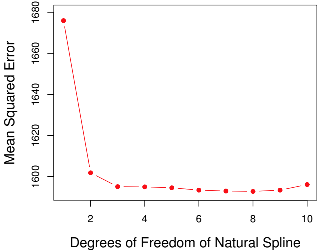
Locations of knots:

- Placed at uniform quantiles of data



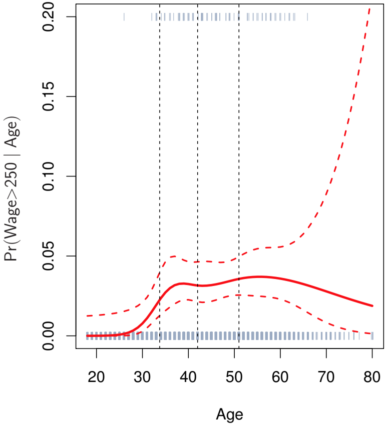
Number of knots:

- Equivalent to choosing degrees of freedom
- Choose the best-looking curve, or...
- By cross-validation



Cubic Splines

Apply to classification (logistic regression) as well



Polynomial Regression vs. Cubic Splines

Polynomial regression (the basis function approach with polynomial basis)

$$Y \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_K X^K$$

- Flexibility/DF determined by degree of polynomials K

Cubic splines

- Flexibility/DF determined by number of knots K

Polynomial Regression vs. Cubic Splines

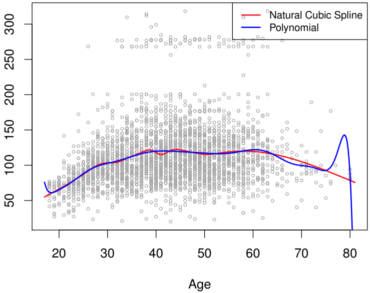
Polynomial regression (the basis function approach with polynomial basis)

$$Y \approx \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_K X^K$$

- Flexibility/DF determined by degree of polynomials K

Cubic splines

- Flexibility/DF determined by number of knots K



Same degrees of freedom (=15)

Cubic splines often more stable (esp. at the boundaries)

Principle: When faced with a new data set, try splines before polynomials.

Suppose our training samples (x_i, y_i) satisfy

$$y_i = 1 + x_i + x_i^2 + x_i^3 \qquad \text{i = 1, \ldots, 10}$$

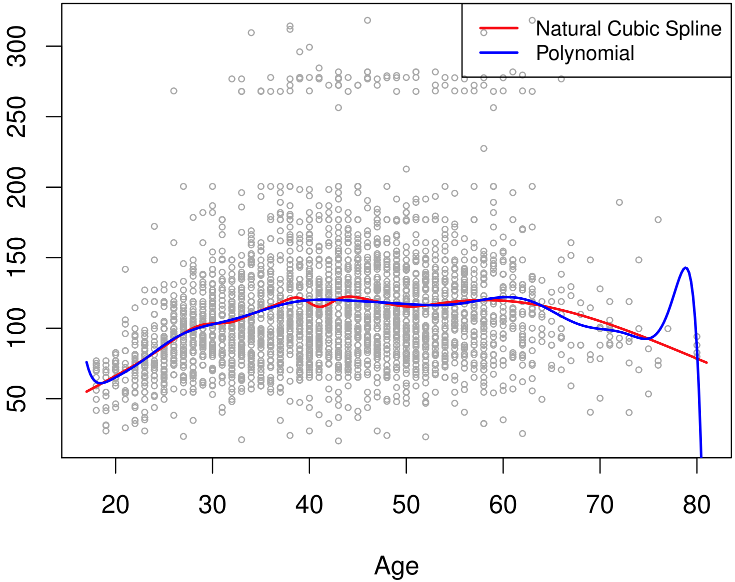
for distinct points $x_1 < x_2 < \ldots < x_{10}$.

Which model will have better training error?

A. Regression with degree 3 polynomial basis.

B. A continuous piecewise cubic function with one knot at x_2.

C. Both models will have identical training error.


Natural Cubic Spline
Polynomial
300
250
200
150
100
50
20
30
40
50
60
70
80
Age

Which of the following is not true

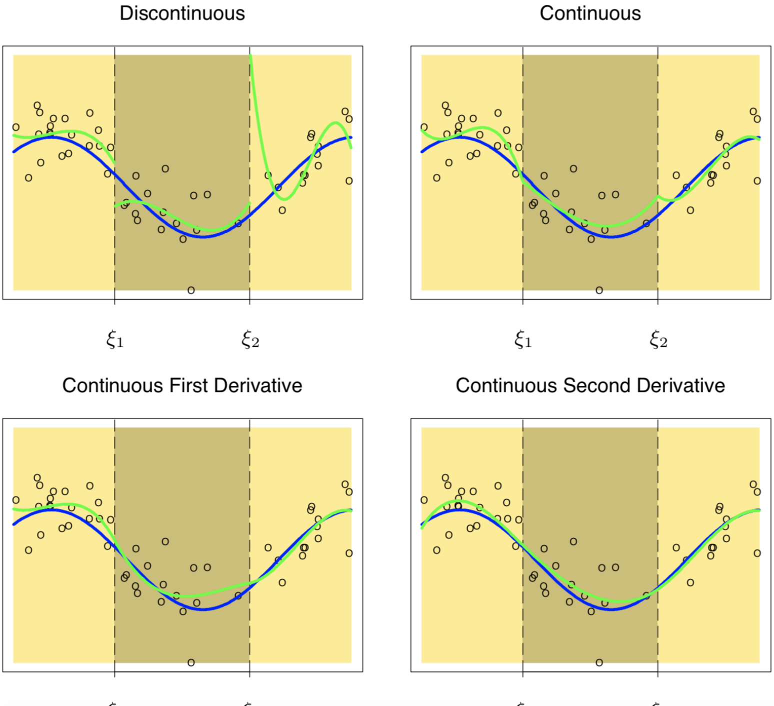
A. One can use fit a cubic spline using the basis function approach.

B. Standard polynomial basis function models do not have continuous second derivatives.

C. Cubic splines enforce continuity of first and second derivatives at the boundaries of regions.

D. A cubic spline can be more flexible than a degree 100 standard polynomial basis function model.

E. Compared to cubic splines, natural cubic splines are less wiggly at the boundaries of data sets.


Discontinuous
Continuous
ξ1
ξ2
ξ1
ξ2
Continuous First Derivative
Continuous Second Derivative

Which model is more flexible?

A. A cubic Spline with $K = 13$ knots

B. A degree 15 polynomial

Recap

Principle: To use basis functions, you just transform the predictor table.

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_K b_K(X)$$

Polynomial functions:

$$b_j(x) = x^j, \quad j = 1, \ldots, K$$

Cubic splines: with K knots at $\xi_1, \xi_2, \ldots, \xi_K$ (i.e., $K + 4$ DF) can be written as

$$Y \approx \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + \cdots + \beta_{K+3} b_{K+3}(X)$$

with basis functions

$$\begin{aligned}
b_1(X) &= X \\
b_2(X) &= X^2 \\
b_3(X) &= X^3 \\
b_4(X) &= \max\{(X - \xi_1)^3, 0\} \\
&\vdots \\
b_{K+3}(X) &= \max\{(X - \xi_K)^3, 0\}
\end{aligned}$$

(cf. ISLR 7.4.3)

Smoothing Splines (ISLR 7.5)

Recall:

(Cubic) Regression splines:

- Problem: Specify knots (or DF)
- Cubic polynomials between knots
- Smoothness at knots
- Fitting: convert to a basis function model and solved by LS

Smoothing Splines (ISLR 7.5)

Recall:

(Cubic) Regression splines:

- Problem: Specify knots (or DF)
- Cubic polynomials between knots
- Smoothness at knots
- Fitting: convert to a basis function model and solved by LS

Smoothing splines: Another way of fitting a smooth curve $g(\cdot)$

- Nonparametric regression!
- Specify tuning parameter λ
- Find the curve as the solution to the optimization problem

$$\min_g \quad \underbrace{\sum_{i=1}^{n}(y_i - g(x_i))^2}_{\text{Loss (RSS)}} + \underbrace{\lambda \int g''(t)^2 dt}_{\text{Regularization}}$$

Smoothing Splines

$$\min_g \quad \underbrace{\sum_{i=1}^{n}(y_i - g(x_i))^2}_{\text{Loss (RSS)}} + \underbrace{\lambda \int g''(t)^2 dt}_{\text{Regularization}}$$

- Loss term: encourage $g(\cdot)$ to fit data well
- Regularization: encourage smoothness
- $g''(t)$: second derivative
- Small $g''(t)$: less wiggly near t
- Larger $\lambda \;\Rightarrow\;$ Smaller $g''(t) \;\Rightarrow\; g(\cdot)$ more smooth

Smoothing Splines

$$\min_g \quad \underbrace{\sum_{i=1}^{n}(y_i - g(x_i))^2}_{\text{Loss (RSS)}} + \underbrace{\lambda \int g''(t)^2 dt}_{\text{Regularization}}$$

- Loss term: encourage $g(\cdot)$ to fit data well
- Regularization: encourage smoothness
- $g''(t)$: second derivative
- Small $g''(t)$: less wiggly near t
- Larger $\lambda \;\Rightarrow\;$ Smaller $g''(t) \;\Rightarrow\; g(\cdot)$ more smooth

Suppose h is a function with $h''(t) = 0$ at every t. Then

A. h is linear

B. h is quadratic

Smoothing Splines

$$\min_g \quad \underbrace{\sum_{i=1}^{n}(y_i - g(x_i))^2}_{\text{Loss (RSS)}} + \underbrace{\lambda \int g''(t)^2 dt}_{\text{Regularization}}$$

- Loss term: encourage $g(\cdot)$ to fit data well
- Regularization: encourage smoothness
- $g''(t)$: second derivative
- Small $g''(t)$: less wiggly near t
- Larger $\lambda \;\Rightarrow\;$ Smaller $g''(t) \;\Rightarrow\; g(\cdot)$ more smooth

If $\lambda = 0$, then any solution g will

A. Perfectly fit the training data (if possible).

B. Perfectly fit the test data (if possible).

Smoothing Splines

$$\min_{g} \quad \underbrace{\sum_{i=1}^{n}(y_i - g(x_i))^2}_{\text{Loss (RSS)}} + \underbrace{\lambda \int g''(t)^2 dt}_{\text{Regularization}}$$

- Loss term: encourage $g(\cdot)$ to fit data well
- Regularization: encourage smoothness
- $g''(t)$: second derivative
- Small $g''(t)$: less wiggly near t
- Larger $\lambda \;\Rightarrow\;$ Smaller $g''(t) \;\Rightarrow\; g(\cdot)$ more smooth

The optimal solution

- Can show: the optimal $g(\cdot)$ is a natural cubic spline
- knots are located at $x_1, x_2, \cdots, x_n$.
- Benefit: n knots, but less than $n + 4$ DF (b/c of λ)

Smoothing Splines: Choosing λ

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

(Recall: In regression splines, flexibility determined by $\#$ knots K, or equivalently DF $= K + 4$)

[1] (Optional) For more mathematical details, see Sec 5.4.1 in *Elements of Statistical Learning* by T. Hastie, R. Tibshirani, J. Friedman.

Smoothing Splines: Choosing λ

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

(Recall: In regression splines, flexibility determined by $\#$ knots K, or equivalently DF $= K + 4$)

For smoothing splines:

- Flexibility determined by λ

[1] (Optional) For more mathematical details, see Sec 5.4.1 in *Elements of Statistical Learning* by T. Hastie, R. Tibshirani, J. Friedman.

Smoothing Splines: Choosing λ

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

(Recall: In regression splines, flexibility determined by $\#$ knots K, or equivalently DF $= K + 4$)

For smoothing splines:

- Flexibility determined by λ

  How should flexibility depend on λ?

  A. As λ increases, flexibility increases.
  B. As λ increases, flexibility decreases.

[1] (Optional) For more mathematical details, see Sec 5.4.1 in *Elements of Statistical Learning* by T. Hastie, R. Tibshirani, J. Friedman.

Smoothing Splines: Choosing λ

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

(Recall: In regression splines, flexibility determined by $\#$ knots K, or equivalently DF $= K + 4$)

For smoothing splines:

- Flexibility determined by λ
- Corresponding to an effective degree of freedom, $df_\lambda \in [2, n]$

[1] (Optional) For more mathematical details, see Sec 5.4.1 in *Elements of Statistical Learning* by T. Hastie, R. Tibshirani, J. Friedman.

Smoothing Splines: Choosing λ

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

(Recall: In regression splines, flexibility determined by $\#$ knots K, or equivalently DF $= K + 4$)

For smoothing splines:

- Flexibility determined by λ
- Corresponding to an effective degree of freedom, $df_\lambda \in [2, n]$
- Closed form expression for df_λ (cf. ISLR 279)[1]

[1] (Optional) For more mathematical details, see Sec 5.4.1 in *Elements of Statistical Learning* by T. Hastie, R. Tibshirani, J. Friedman.

Smoothing Splines: Choosing λ

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

(Recall: In regression splines, flexibility determined by $\#$ knots K, or equivalently DF $= K + 4$)

For smoothing splines:

- Flexibility determined by λ
- Corresponding to an effective degree of freedom, $df_\lambda \in [2, n]$
- Closed form expression for df_λ (cf. ISLR 279)[1]
- Choose λ (equivalently, df_λ) by CV

[1] (Optional) For more mathematical details, see Sec 5.4.1 in *Elements of Statistical Learning* by T. Hastie, R. Tibshirani, J. Friedman.

Smoothing Splines: Choosing λ

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

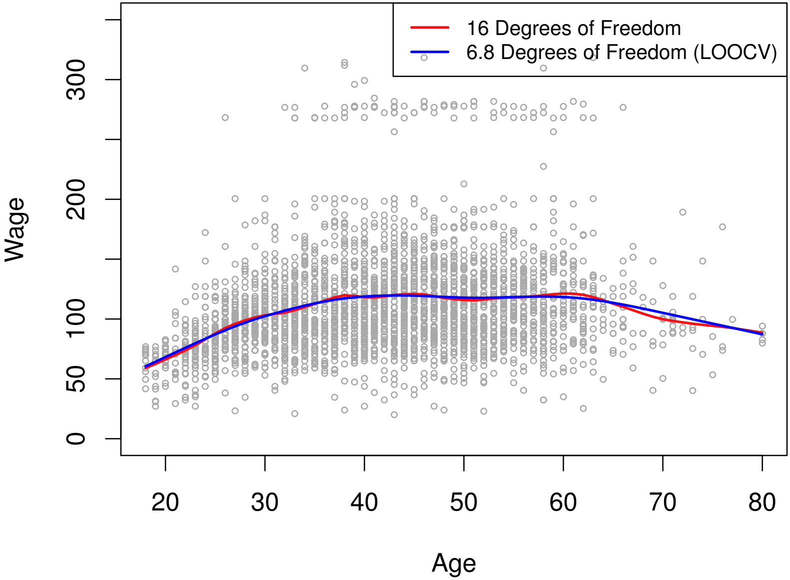
(Recall: In regression splines, flexibility determined by $\#$ knots K, or equivalently DF $= K + 4$)

For smoothing splines:

- Flexibility determined by λ
- Corresponding to an effective degree of freedom, $df_\lambda \in [2, n]$
- Closed form expression for df_λ (cf. ISLR 279)[1]
- Choose λ (equivalently, df_λ) by CV
- LOOCV can be done very efficiently

[1] (Optional) For more mathematical details, see Sec 5.4.1 in *Elements of Statistical Learning* by T. Hastie, R. Tibshirani, J. Friedman.

Smoothing Splines: Choosing λ



Smoothing Splines: Choosing λ, another example

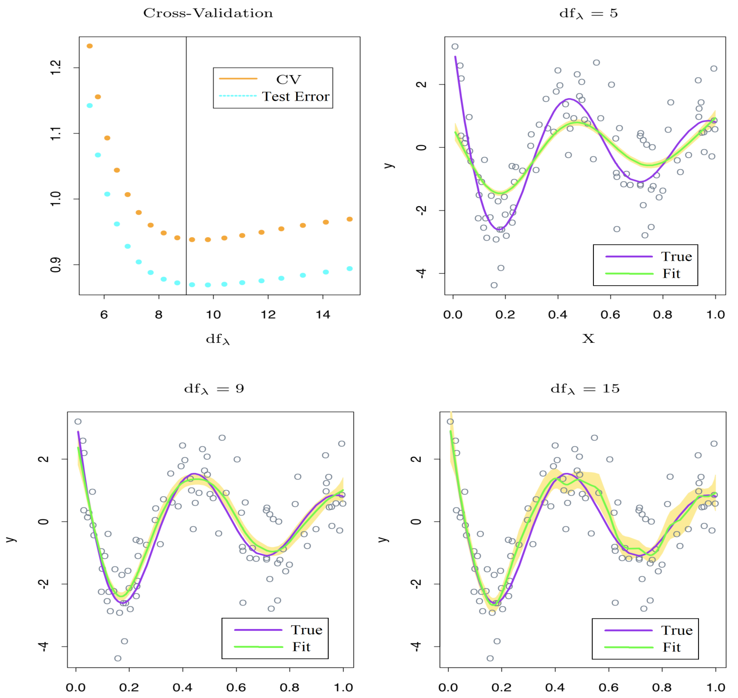
- Suppose data is generated from the true model

$$Y = f(X) + \epsilon, \qquad \text{where}$$
$$f(X) = \frac{\sin(12X + 2.4)}{X + 0.2}.$$

Not a polynomial.

- Our training set consists of $n = 100$ data points.

- We fit a smoothing spline different values of λ (i.e., different effect degrees of freedom df_λ).



Cross-Validation
CV
Test Error
df_λ
$\mathrm{df}_\lambda = 5$
True
Fit
X
y
$\mathrm{df}_\lambda = 9$
True
Fit
y
$\mathrm{df}_\lambda = 15$
True
Fit
y

Suppose you have a cubic spline with 5 knots. Which of the following splines have a similar model complexity?

A. Smoothing spline with 100 knots and 5 degrees of freedom.

B. Smoothing spline with 5 knots and 100 degrees of freedom.

How does a smoothing spline (with $\lambda > 0$) decay outside the boundaries of a dataset?

A. Linearly

B. Cubically

C. Either are possible, but it depends on λ.

Logistic Regression using Smoothing Splines

(Optional; cf. Sec 5.6 of ESL[2])

- Also known as *Nonparametric Logistic Regression*

[2] ESL = *Elements of Statistical Learning*.

Logistic Regression using Smoothing Splines

(Optional; cf. Sec 5.6 of ESL[2])

- Also known as *Nonparametric Logistic Regression*

- The model is

$$\boxed{\text{log odds} \approx g(X)} \qquad (\text{compare to log odds} \approx \beta_0 + \beta_1 X)$$

- where we fit g by solving the regularized maximum likelihood problem:

$$\max_g \quad \underbrace{\sum_{i=1}^{n} \left[y_i g(x_i) - \log\left(1 + e^{g(x_i)}\right) \right]}_{\text{Log likelihood}} - \lambda \underbrace{\int g''(t)^2 dt}_{\text{Regularization}}.$$

[2] ESL = *Elements of Statistical Learning*.

Logistic Regression using Smoothing Splines

(Optional; cf. Sec 5.6 of ESL[2])

- Also known as *Nonparametric Logistic Regression*

- The model is

$$\boxed{\text{log odds} \approx g(X)} \qquad (\text{compare to log odds} \approx \beta_0 + \beta_1 X)$$

- where we fit g by solving the regularized maximum likelihood problem:

$$\max_g \quad \underbrace{\sum_{i=1}^{n} \left[y_i g(x_i) - \log\left(1 + e^{g(x_i)}\right) \right]}_{\text{Log likelihood}} - \lambda \underbrace{\int g''(t)^2 dt}_{\text{Regularization}} .$$

- The optimal $g(\cdot)$ is a natural cubic spline with knots at $x_1, x_2, \cdots, x_n$!

Mystery of nonparametric models.

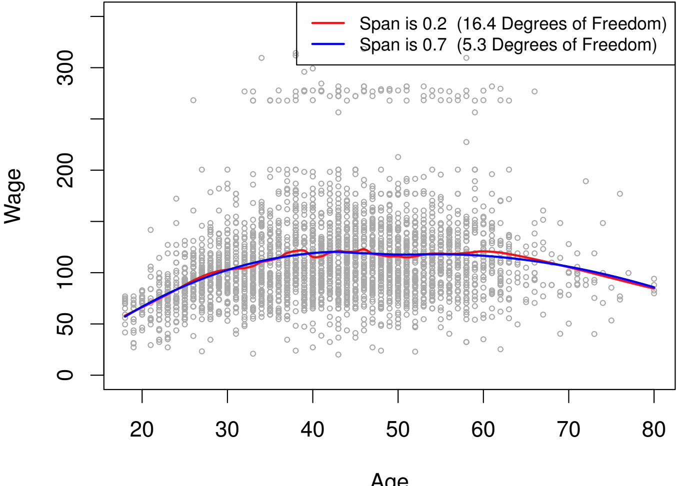
[2]ESL = *Elements of Statistical Learning*.

Local Regression

(Not covered; ISLR 7.6)

A third way of fitting smooth curves

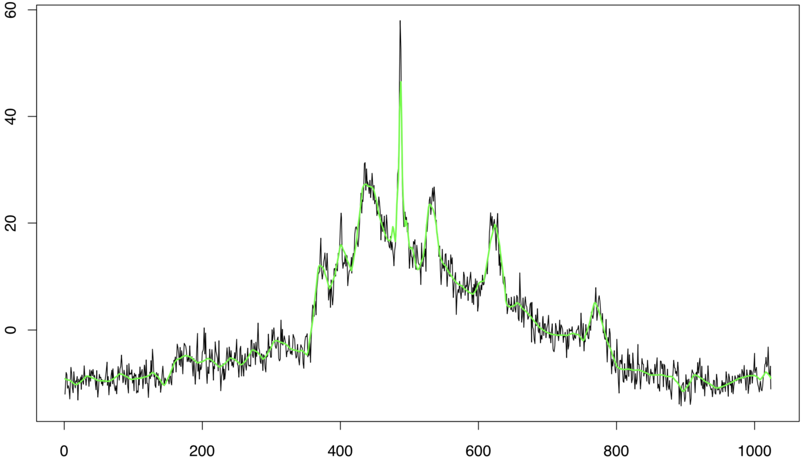
- Flexibility determined by a tuning parameter s (span)
- Corresponding to some effective DF
- Limitation: Like KNN need all training data at testing time.



Fourier and Wavelet Basis

(Not covered; ESL 5.9)

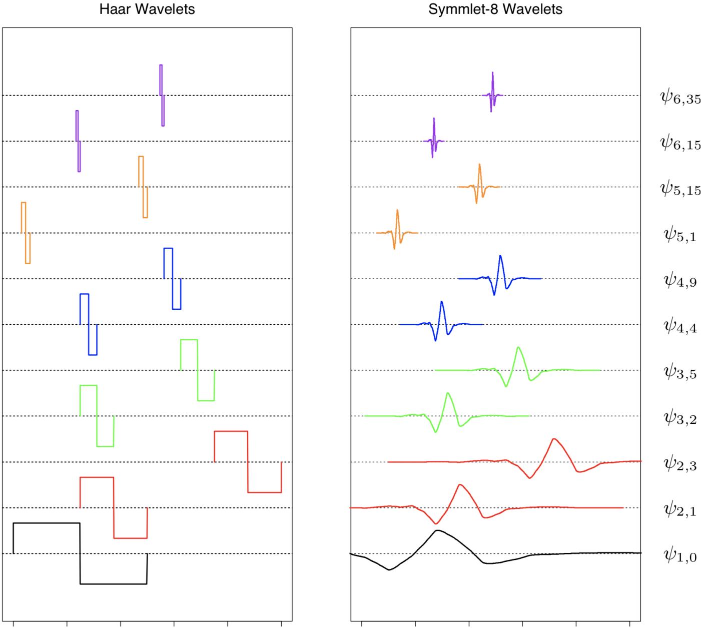
- Fitting/de-noising an NMR (nuclear magnetic resonance) signal using the Symmlet-8 wavelet basis



- Application: MRI

Principle: Use domain knowledge to choose basis functions!

- Wavelet basis are non-periodic and localized



Fourier and Wavelet Basis

(Not covered; ESL 5.9)

- Wavelet basis are non-periodic and localized

- They have the form

$$b_{j,k}(x) = 2^{j/2} \cdot b(2^j x - k), \qquad j = 0, 1, 2 \ldots, \quad k = 0, 1, 2, \ldots$$

where $b(\cdot)$ (called *mother wavelet*) is some function that equals 0 outside the interval $[0, 1]$.

Fourier and Wavelet Basis

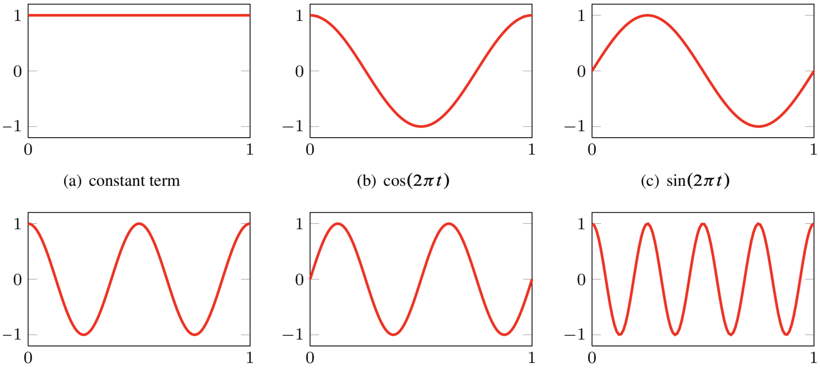
(Not covered; ESL 5.9)

- Popular in signal and image processing

Fourier and Wavelet Basis

(Not covered; ESL 5.9)

- Popular in signal and image processing
- Fourier basis are periodic (sines and cosines)



(a) constant term (b) $\cos(2\pi t)$ (c) $\sin(2\pi t)$

(d) $\cos(4\pi t)$ (e) $\sin(4\pi t)$ (f) $\cos(6\pi t)$

Mini Summary

1 predictor: $Y = f(X)$

- Basis function approach: $f(X) = \sum_j \beta_j b_j(X)$
- Regression Splines: $f(X) =$ piecewise polynomials joined smoothly
- Smoothing Splines: $f(X) =$ solution to $f''(\cdot)$-regularized least squares

Mini Summary

1 predictor: $Y = f(X)$

- Basis function approach: $f(X) = \sum_j \beta_j b_j(X)$
- Regression Splines: $f(X) =$ piecewise polynomials joined smoothly
- Smoothing Splines: $f(X) =$ solution to $f''(\cdot)$-regularized least squares

Principle: Ridge regression is to least squares as smoothing splines are to natural cubic splines.

Mini Summary

1 predictor: $Y = f(X)$

- Basis function approach: $f(X) = \sum_j \beta_j b_j(X)$
- Regression Splines: $f(X) =$ piecewise polynomials joined smoothly
- Smoothing Splines: $f(X) =$ solution to $f''(\cdot)$-regularized least squares

Principle: Ridge regression is to least squares as smoothing splines are to natural cubic splines.

p predictors: $Y = f(X_1, X_2, \ldots, X_p)$

- Generalized Additive Models (GAMs)

Generalized Additive Models (ISLR 7.7)

Recall: Multiple linear regression

$$Y \approx \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Generalized Additive Models (ISLR 7.7)

Recall: Multiple linear regression

$$Y \approx \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Generalized Additive Model: Maintains only additivity

$$Y \approx \beta_0 + f_1(X_1) + \cdots + f_p(X_p)$$

Generalized Additive Models (ISLR 7.7)

Recall: Multiple linear regression

$$Y \approx \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Generalized Additive Model: Maintains only additivity

$$Y \approx \beta_0 + f_1(X_1) + \cdots + f_p(X_p)$$

- $f_j(\cdot)$: Any of the univariate nonlinear functions we just learned
- E.g. polynomials, linear combination of basis functions, cubic/smoothing splines
- Build multivariate nonlinear models by adding up univariate ones

Example: Wage Dataset

Fit a GAM of the form

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

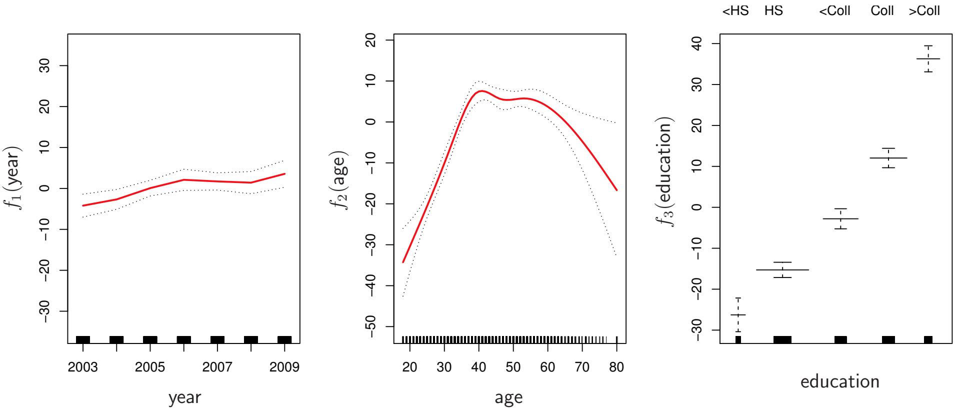
Example: Wage Dataset

Fit a GAM of the form

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

where

- $f_1(\cdot)$, $f_2(\cdot)$: natural cubic splines
- education: categorical w/ 5 levels `<HS, HS, <Coll, Coll, >Coll`
- $f_3(\cdot) =$ a different value for each level of education
  - i.e., encode education w/ 4 four dummy variables and fit a usual linear model



▶ Interpretable: See contribution of each variable to the overall model.

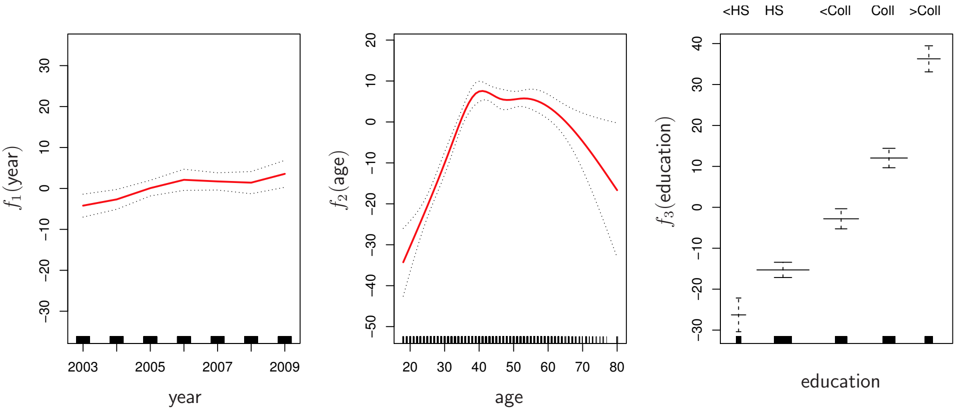
What can't we conclude from these plots?

A. There was a downturn in wages in 2008

B. Most people's salaries peak around age 40

C. One must do well in college to get a higher wage

D. People who do not finish college, generally have lower wages.



- Interpretable: See contribution of each variable to the overall model.

What can't we conclude from these plots?

A. There was a downturn in wages in 2008

B. Most people's salaries peak around age 40

C. One must do well in college to get a higher wage

D. People who do not finish college, generally have lower wages.

- Easy to fit: A big regression onto spline basis and dummy variables.

GAMs for Classification

Recall: Logistic regression

$$\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

GAMs for Classification

Recall: Logistic regression

$$\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Logistic regression GAM:

$$\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + f_1(X_1) + \cdots + f_p(X_p)$$

GAMs for Classification

Recall: Logistic regression

$$\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Logistic regression GAM:

$$\log\left(\frac{\Pr(Y=1|X)}{1-\Pr(Y=1|X)}\right) \approx \beta_0 + f_1(X_1) + \cdots + f_p(X_p)$$

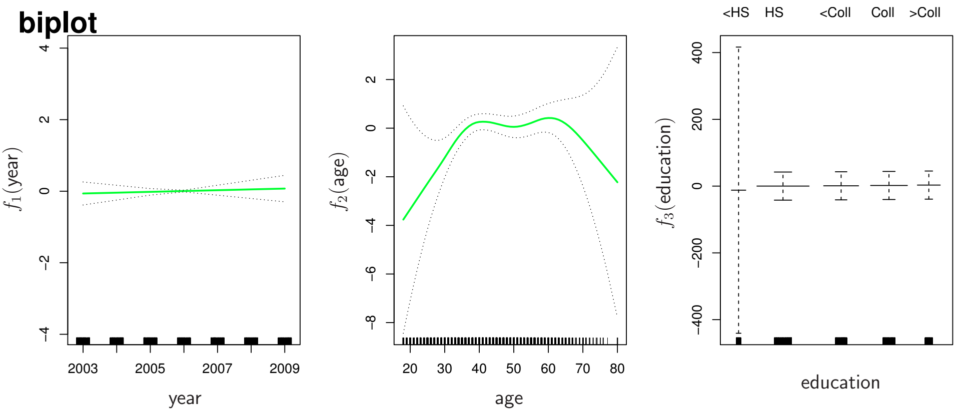
Example: Wage Dataset
Fit a GAM of the form

$$\log\left(\frac{\Pr(\text{wage} > 250)}{\Pr(\text{wage} \leq 250)}\right) \approx \beta_0 + \beta_1 \times \text{year} + f_2(\text{age}) + f_3(\text{education})$$

where

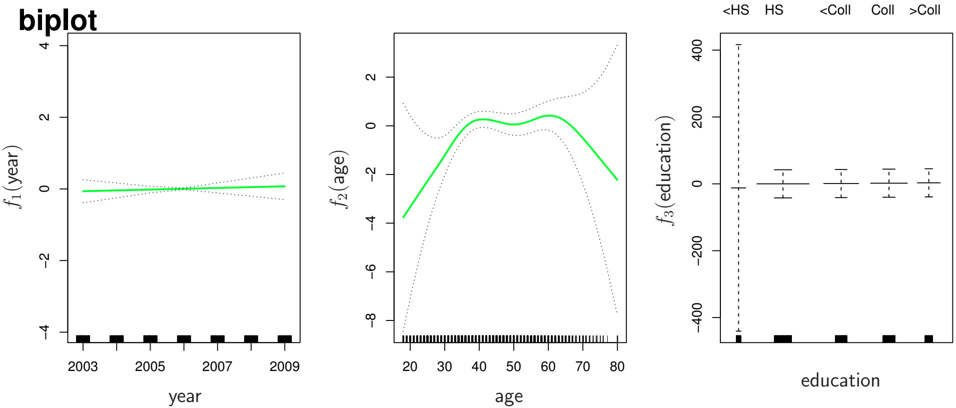
- $f_2(\cdot)$: smoothing splines with $df = 5$
- $f_3(\cdot)$ constant for each level of education

biplot



$f_1(\text{year})$
4
2
0
−2
−4
2003
2005
2007
2009
year
$f_2(\text{age})$
2
0
−2
−4
−6
−8
20
30
40
50
60
70
80
age
<HS
HS
<Coll
Coll
>Coll
$f_3(\text{education})$
400
200
0
−200
−400
education

biplot



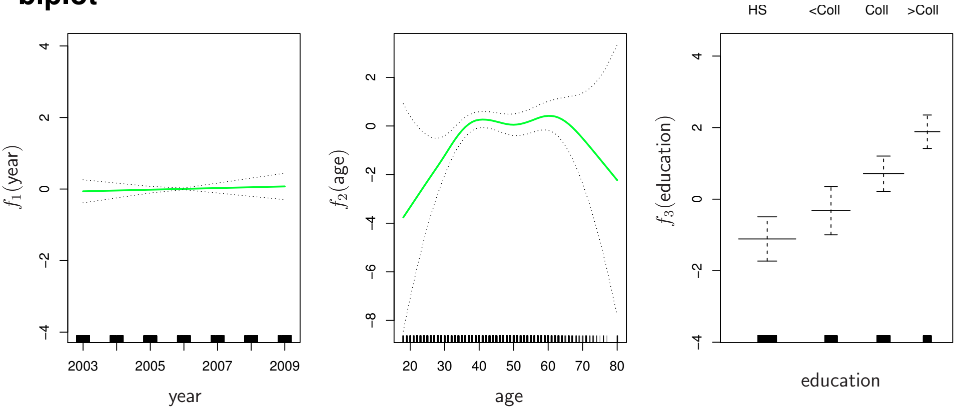
What's more likely?

A. Almost no one in the data set makes more than 250K a year.

B. Almost no one with a college education makes more than 250K a year.

C. Almost no one with less than a high school education makes more than 250K a year.

biplot



- Fitting: Backfitting (Optional):
  - First fit linear model to year and dummy education variables.
  - Then fit residual using smoothing spline.

GAMs: Pros and Cons

$$Y \quad \text{or} \quad \text{log-odds}(Y) \approx \beta_0 + f_1(X_1) + \cdots + f_p(X_p)$$

- Combine simple univariate nonlinear models $f_j(\cdot)$ to build p-variate models
- Flexible choices for each $f_j(\cdot)$
- (Natural) Cubic Spline is a popular choice
- Control flexibility by specifying degrees-of-freedom

- Interaction/synergy effects b/w predictors not captured

Nonlinear Modeling in R (ISLR 7.8)

- Polynomial Regression

```
> fit.1 = lm(wage~age, data=Wage)
> fit.4 = lm(wage~poly(age,4), data=Wage)
```

Nonlinear Modeling in R (ISLR 7.8)

- Polynomial Regression

```
> fit.1 = lm(wage~age, data=Wage)
> fit.4 = lm(wage~poly(age,4), data=Wage)
```

- Polynomial Logistic Regression

```
> fit = glm(I(wage>250)~poly(age,3), data=Wage, family=binomial)
```

Nonlinear Modeling in R (ISLR 7.8)

- Regression with step functions of a numeric predictor

```
> 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
```

Nonlinear Modeling in R (ISLR 7.8)

- Regression with step functions of a numeric predictor

```
> 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
```

- Regression with step functions of a categorical predictor

```
> fit = lm(wage~education, data=Wage)
> coef(summary(fit))
_                           Estimate Std. Error   t value      Pr(>|t|)
(Intercept)                 84.10441   2.231165 37.695283 7.807516e-255
education2.HS Grad          11.67894   2.520332  4.633887  3.741870e-06
education3.Some College     23.65115   2.651529  8.919815  7.953154e-19
education4.College Grad     40.32349   2.631679 15.322344  4.467568e-51
education5.Advanced Degree  66.81336   2.847782 23.461545 6.916139e-112
```

Nonlinear Modeling in R (ISLR 7.8)

- Cubic Splines with pre-specified knots

```
> library(splines)
> fit1 = lm(wage~bs(age, knots=c(25,40,60)), data=Wage)   # DF= 7
```

- Cubic Splines with $df = 6$ (plus 1 intercept)

Knots at 3 uniform quantiles (25%, 50%, 75%)

```
> fit2 = lm(wage~bs(age, df=6), data=Wage)
```

Nonlinear Modeling in R (ISLR 7.8)

- Cubic Splines with pre-specified knots

```
> library(splines)
> fit1 = lm(wage~bs(age, knots=c(25,40,60)), data=Wage)   # DF= 7
```

- Cubic Splines with *df* $= 6$ (plus 1 intercept)

Knots at 3 uniform quantiles (25%, 50%, 75%)

```
> fit2 = lm(wage~bs(age, df=6), data=Wage)
```

- Natural Cubic Splines *df* $= 4$ (plus 1 intercept)

Knots at uniform quantiles

```
> fit3 = lm(wage~ns(age, df=4), data=Wage)
```

Nonlinear Modeling in R (ISLR 7.8)

- Cubic Splines with pre-specified knots

```
> library(splines)
> fit1 = lm(wage~bs(age, knots=c(25,40,60)), data=Wage)   # DF= 7
```

- Cubic Splines with $df = 6$ (plus 1 intercept)

Knots at 3 uniform quantiles (25%, 50%, 75%)

```
> fit2 = lm(wage~bs(age, df=6), data=Wage)
```

- Natural Cubic Splines $df = 4$ (plus 1 intercept)

Knots at uniform quantiles

```
> fit3 = lm(wage~ns(age, df=4), data=Wage)
```

- Smoothing Splines $df = 16$

```
> fit4 = smooth.spline(age, wage, df=16)
```

Nonlinear Modeling in R (ISLR 7.8)

- Cubic Splines with pre-specified knots

```
> library(splines)
> fit1 = lm(wage~bs(age, knots=c(25,40,60)), data=Wage)   # DF= 7
```

- Cubic Splines with *df* $= 6$ (plus 1 intercept)
Knots at 3 uniform quantiles (25%, 50%, 75%)

```
> fit2 = lm(wage~bs(age, df=6), data=Wage)
```

- Natural Cubic Splines *df* $= 4$ (plus 1 intercept)
Knots at uniform quantiles

```
> fit3 = lm(wage~ns(age, df=4), data=Wage)
```

- Smoothing Splines *df* $= 16$

```
> fit4 = smooth.spline(age, wage, df=16)
```

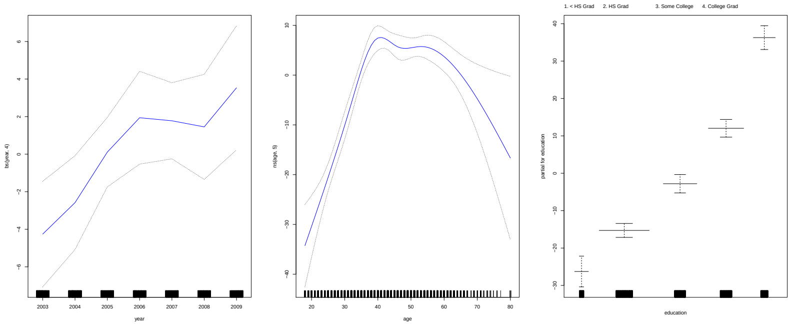
- Smoothing Splines with *df* chosen by CV

```
> fit5 = smooth.spline(Wage$age, Wage$wage, cv=TRUE)
> fit5$df
[1] 6.794596
```

Nonlinear Modeling in R (ISLR 7.8)

- GAM with (natural) cubic splines

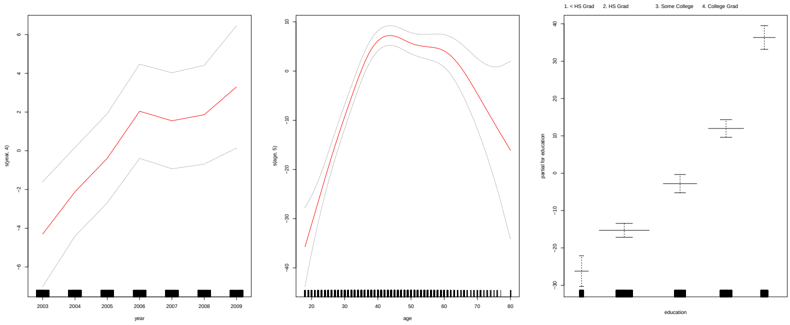
```
> gam1 = lm(wage~bs(year,4)+ns(age,5)+education, data=Wage )

> library(gam)
> plot.gam(gam1, se=TRUE, col="blue")
```



Nonlinear Modeling in R (ISLR 7.8)

- GAM with smoothing splines

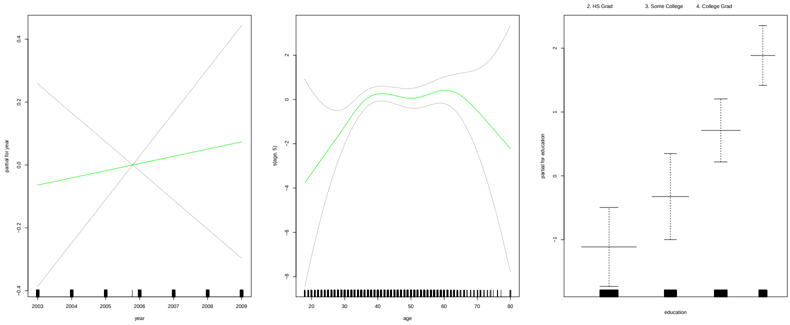
```
> library(gam)
> gam2 = gam(wage~s(year,4)+s(age,5)+education, data=Wage)

> plot(gam2, se=TRUE, col="red")
```



Nonlinear Modeling in R (ISLR 7.8)

- Logistic regression GAM with smoothing splines
Excluding observations with less than a high school education

```
> library(gam)
> gam.lr = gam(I(wage>250)~year+s(age,5)+education, family=
+ binomial, data=Wage, subset=(education!="1. < HS Grad"))
> plot(gam.lr, se=TRUE, col="green")
```



Model Selection using ANOVA (ISLR 7.8.3; Lab 5)

Capturing Interaction Effect

(Optional)

- We can fit 2-variable nonlinear functions that capture their interaction / synergy effect.

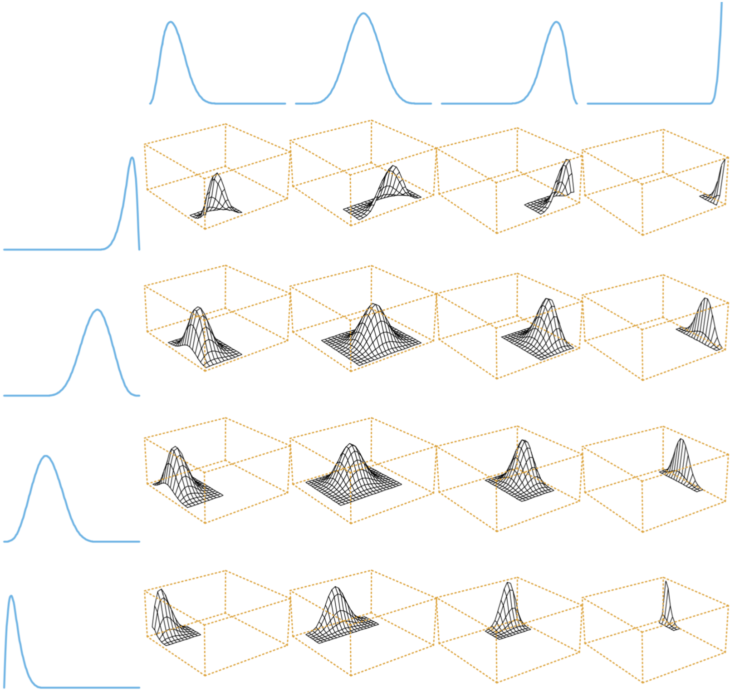
Capturing Interaction Effect

(Optional)

- We can fit 2-variable nonlinear functions that capture their interaction / synergy effect.

- For example, a two-dimensional spline that is the product of two one-dimensional splines

$$b(X_1, X_2) = g(X_1)h(X_2).$$



Capturing Interaction Effect

(Optional; ESL 5.8)

Generalization to *k*-variable (*k*th-order) splines

$$b(X_1, X_2, X_3, \ldots, X_k)$$

Capturing Interaction Effect

(Optional; ESL 5.8)

Generalization to k-variable (kth-order) splines

$$b(X_1, X_2, X_3, \ldots, X_k)$$

Can build multi-variable models similarly to GAMs:

$$f(X_1, X_2, \ldots, X_p) = \beta_0 + b_1(X_1, X_2) + b_2(X_2, X_3 X_4) + b_3(X_9) + \cdots$$

Capturing Interaction Effect

(Optional; ESL 5.8)

Generalization to k-variable (kth-order) splines

$$b(X_1, X_2, X_3, \ldots, X_k)$$

Can build multi-variable models similarly to GAMs:

$$f(X_1, X_2, \ldots, X_p) = \beta_0 + b_1(X_1, X_2) + b_2(X_2, X_3 X_4) + b_3(X_9) + \cdots$$

Danger of overfitting!

Capturing Interaction Effect

(Optional; ESL 5.8)

Generalization to k-variable (kth-order) splines

$$b(X_1, X_2, X_3, \dots, X_k)$$

Can build multi-variable models similarly to GAMs:

$$f(X_1, X_2, \dots, X_p) = \beta_0 + b_1(X_1, X_2) + b_2(X_2, X_3X_4) + b_3(X_9) + \cdots$$

Danger of overfitting!

Need to choose carefully:

- Maximum order of interaction k;
- Which terms to include;
- What functions b_i's to use.

Capturing Interaction Effect

(Optional; ESL 5.8)

Generalization to k-variable (kth-order) splines

$$b(X_1, X_2, X_3, \ldots, X_k)$$

Can build multi-variable models similarly to GAMs:

$$f(X_1, X_2, \ldots, X_p) = \beta_0 + b_1(X_1, X_2) + b_2(X_2, X_3 X_4) + b_3(X_9) + \cdots$$

Danger of overfitting!

Need to choose carefully:

- Maximum order of interaction k;
- Which terms to include;
- What functions b_i's to use.

Automatic procedure: MARS (Multivariate Adaptive Regression Splines), implemented in R package `earth`

Capturing Interaction Effect

(Optional; ESL 5.8)

Generalization to k-variable (kth-order) splines

$$b(X_1, X_2, X_3, \dots, X_k)$$

Can build multi-variable models similarly to GAMs:

$$f(X_1, X_2, \dots, X_p) = \beta_0 + b_1(X_1, X_2) + b_2(X_2, X_3 X_4) + b_3(X_9) + \cdots$$

Danger of overfitting!

Need to choose carefully:

- Maximum order of interaction k;
- Which terms to include;
- What functions b_i's to use.

Automatic procedure: MARS (Multivariate Adaptive Regression Splines), implemented in R package `earth`

Or, use decision trees and random forests (next week).

Nonlinear Modeling Summary

1 predictor: $Y = f(X)$

- Basis function approach: $f(X) = \sum_j \beta_j b_j(X)$
- Regression Splines: $f(X) =$ piecewise polynomials joint smoothly
- Smoothing Splines: $f(X) =$ solution to $f''(\cdot)$-regularized least squares
- Local Regression

Nonlinear Modeling Summary

1 predictor: $Y = f(X)$

- Basis function approach: $f(X) = \sum_j \beta_j b_j(X)$
- Regression Splines: $f(X) =$ piecewise polynomials joint smoothly
- Smoothing Splines: $f(X) =$ solution to $f''(\cdot)$-regularized least squares
- Local Regression

p predictors: $Y = f(X_1, X_2, \ldots, X_p)$

- Generalized Additive Models (GAMs)

$$Y \quad \text{or} \quad \text{log-odds}(Y) \;\approx\; \beta_0 + f_1(X_1) + \cdots + f_p(X_p)$$

where $f_j(\cdot)$ is a polynomial, step function, cubic/smoothing spline, local regression, ...

Some of the figures in this presentation are taken from “An Introduction to Statistical Learning, with applications in R” (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani
Slides based on Yudong Chen’s slides.