STA 35C: Statistical Data Science III

Lecture 20: Basis Functions & Regression Splines

Dogyoon Song

Spring 2026, UC Davis

Announcements

Midterm 2 solutions and scores are posted online

- You can review your graded exam during tomorrow's discussion section
- Remember that only your higher score on the two midterms will count toward your final course grade

To submit regrade requests

- If you believe a score should be adjusted, please email the TA **by noon on Wednesday, May 20** with:
  - The specific problem(s) for which you are requesting a regrade
  - A clear explanation of why your answer merits more credit

Where we are now

So far, we have studied foundational topics in statistical learning:

- **Regression and Classification** basics
- **Resampling methods:** Cross-validation & bootstrap
- **Model selection:** Subset selection and regularization (ridge & lasso)
- **Multiple testing:** FP inflation, FWER and FDR controls

Now we move to two final themes for the rest of the course:

- **Beyond linear models:**
  - Basis functions & regression splines
- **Unsupervised learning:** structure in X without a response Y
  - Principal component analysis (PCA)
  - Clustering

Agenda

Today's topic: We continue the model-flexibility story. Instead of only selecting or shrinking existing predictors, we create richer transformed features using basis functions.

- **Basis functions**
  - Recall: polynomial regression
  - Step functions
  - How basis functions unify these ideas
- **Regression splines**
  - Piecewise polynomials
  - Smoothness constraints at knots
  - Truncated power basis representation
  - Preview: boundary behaviors and "natural" splines

Motivation for basis functions: Beyond linear models

Challenge: Linear models are powerful but can sometimes be too restrictive

- Assumes $Y \approx \beta_0 + \sum_{j=1}^{p} \beta_j X_j$, a linear function of the raw predictors
- Real data often exhibits more complex, nonlinear relationships

Goal: Capture nonlinear structure while keeping estimation as tractable as least squares

Examples:

- *Polynomial regression*: use $(X, X^2, X^3, \dots)$
- *Step functions*: approximate the regression function by piecewise-constant segments

Today's plan:

- Review polynomial regression & define step functions
- Unify these via *basis functions*
- Introduce *splines* for flexible piecewise-defined models

Example 1: Polynomial regression

Polynomial regression: Replace the standard linear model

$$Y = \beta_0 + \beta_1 X + \epsilon$$

with a polynomial:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_d X^d + \epsilon$$

Remarks:

- Although nonlinear in X, the model is linear in the coefficients β_j
    - Treat $X, X^2, \ldots, X^d$ as transformed predictors
    - The coefficients $\beta_1, \ldots, \beta_d$ can be estimated by standard least squares
- Typically, a moderate degree d (2–4) is used to avoid overfitting

Example 1: Polynomial regression (illustration)

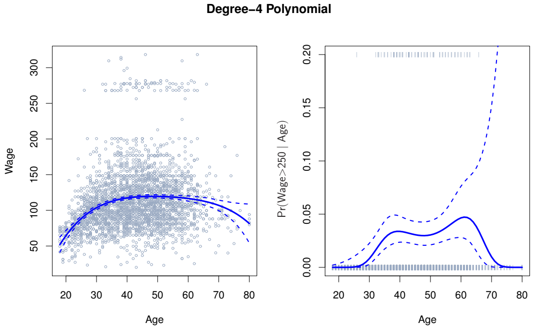

Figure: **Polynomial fit on `Wage` data.** (*Left*) A degree-4 polynomial of `wage` on `age`, with 95% confidence bands. (*Right*) Logistic regression for `wage`>$250k using a degree-4 polynomial [JWHT21, Figure 7.1].

- **Advantage:** More flexible than a strictly linear model
- **Limitation:** A polynomial is global; changing the fit in one region can affect the entire curve, especially wildly near boundaries

Example 2: Step functions

Idea: Partition a continuous variable X into intervals, treating each as a separate "level"

- For cutpoints $c_1 < c_2 < c_3 \ldots$, define indicator functions:

$$C_0(X) := I_{(-\infty, c_1]}(X), \qquad C_1(X) := I_{(c_1, c_2]}(X), \qquad C_2(X) := I_{(c_2, c_3]}(X) \; \ldots$$

  - The indicator function

$$I_S(x) = \begin{cases} 1 & \text{if } x \in S, \\ 0 & \text{if } x \notin S. \end{cases}$$

  - Equivalently, we often write $I(x \in S)$ for $I_S(x)$

- The fitted model is piecewise constant:

$$\mathbb{E}[Y \mid X = x] = \beta_0 + \beta_1 \cdot C_1(x) + \beta_2 \cdot C_2(x) + \cdots = \begin{cases} \beta_0 & x \le c_1, \\ \beta_0 + \beta_1 & c_1 < x \le c_2, \\ \beta_0 + \beta_2 & c_2 < x \le c_3, \\ \vdots & \vdots \end{cases}$$

Example 2: Step functions (illustration)

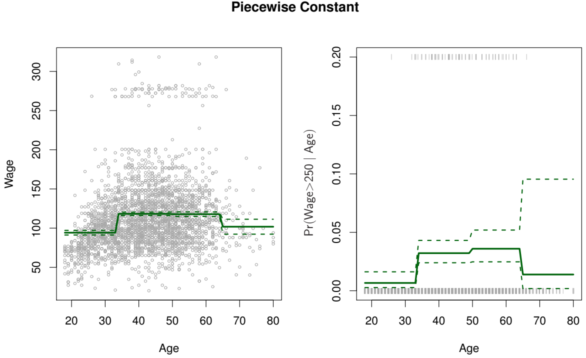

Figure: **Step-function fit on `Wage` data.** (*Left*) Piecewise-constant model for `wage` on `age`, with 95% confidence bands. (*Right*) Logistic regression for `wage`>$250k using a step function [JWHT21, Figure 7.2].

- **Advantage:** Easy to capture abrupt changes ("jumps")
- **Limitation:** Not smooth; can be too coarse with few cutpoints

Basis functions: Bridging polynomial, step, and more

Key idea: Transform the raw predictor X to construct new features $\{b_1(X), \ldots, b_K(X)\}$, then fit a linear model in those features:

$$Y \approx \beta_0 + \beta_1 b_1(X) + \cdots + \beta_K b_K(X)$$

Examples of basis functions:

- *Polynomials*: $b_1(X) = X, \; b_2(X) = X^2, \ldots$
  - Smooth but global
- *Step functions*: $b_1(X) = I(c_1 < X \leq c_2), \; b_2(X) = I(c_2 < X \leq c_3), \ldots$
  - Local but discontinuous
- ***Splines***: piecewise polynomials with continuity constraints
  - Best of both: local flexibility like step functions, but smoother like polynomials

Main benefits:

- Nonlinear in X: the basis functions can capture nonlinearities more flexibly
- **Important distinction:** while these models may be nonlinear as functions of X, they are *linear in the unknown coefficients* β_k, so least squares still applies

Pop-up quiz: Basis functions

Suppose we fit

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 I(X > 50) + \varepsilon.$$

Question: Which statement is most accurate?

A) The model must be fit separately for $X \leq 50$ and $X > 50$.

B) The fitted regression function is necessarily continuous at $X = 50$.

C) Least squares cannot be used because the model includes nonlinear functions of X.

D) Least squares can be used after creating the transformed predictors X^2 and $I(X > 50)$, because the model is linear in the coefficients.

Answer: D. The fitted function is nonlinear in X, but linear in the unknown coefficients; we can treat X^2 and $I(X > 50)$ as transformed features and fit by least squares.

Regression splines: Main concept

Want: More flexible than a single polynomial, but smoother than step functions

Idea:

- *Piecewise polynomials* of degree d, joined at *knots* (cutpoints)
- Within each interval between knots, fit a separate polynomial (e.g. cubic)
- Impose *smoothness constraints* at each knot, preventing abrupt jumps or kinks

Example (degree $d = 3$, one knot at c): $Y = f(X) + \epsilon$ where

$$f(x) = \begin{cases} \beta_{01} + \beta_{11}x + \cdots + \beta_{d1}x^d, & x \leq c, \\ \beta_{02} + \beta_{12}x + \cdots + \beta_{d2}x^d, & x > c. \end{cases}$$

For a standard degree-d regression spline, we require f and its first $d - 1$ derivatives to be continuous at each knot

$\rightarrow$ See the next slide for an illustration

Illustration of cubic spline: Ensuring smoothness at knots

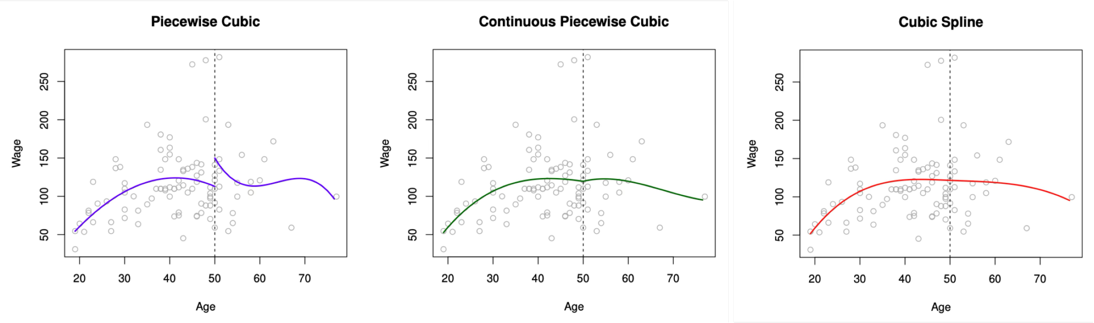

Figure: Various piecewise polynomials fit to a subset of the `Wage` data, with a knot at `age`=50. **Left:** Unconstrained polynomials cause a discontinuity. **Center:** Imposing continuity at the knot eliminates jumps but can form a "kink." **Right:** Additionally constraining continuity of first and second derivatives yields a smooth cubic spline. Excerpted from [JWHT21, Figure 7.3].

Illustration: Comparing cubic vs. linear splines

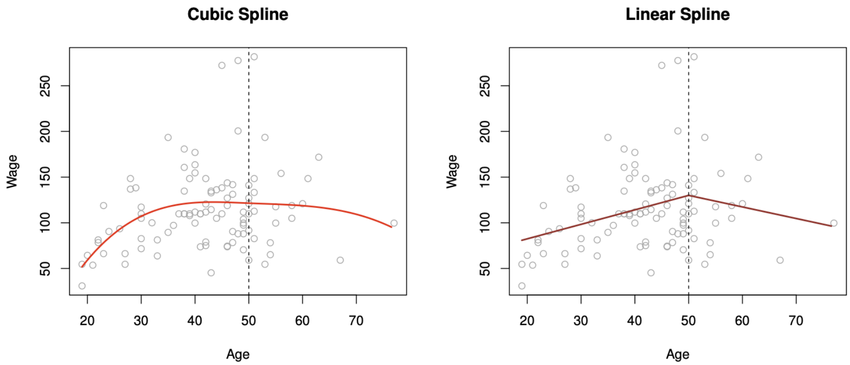

Figure: Again with a knot at age=50 on a subset of Wage. **Left:** A cubic spline, enforcing continuity of function plus its 1st and 2nd derivatives. **Right:** A linear spline, only requiring continuity of function at the knot age=50. Excerpted from [JWHT21, Figure 7.3].

Degrees of freedom for splines

Degrees of freedom measure the effective number of fitted coefficients, hence the flexibility of the spline model

Piecewise polynomial of degree d with K knots:

- $(d+1)$ polynomial coefficients in each of the $(K+1)$ intervals
  $\Longrightarrow$ $(K+1) \times (d+1)$ parameters in total *before* smoothness constraints
- Each knot imposes d constraints: continuity of f and its first $d-1$ derivatives
  - $K \times d$ constraints in total
- After smoothness constraints, the degrees of freedom are

$$(K+1)(d+1) - Kd = d + 1 + K.$$

  - e.g., a cubic spline $(d=3)$ with K knots has $(3+1)+K = K+4$ parameters

Trade-off: We want

- Enough degree and knots to capture possible nonlinearities
- But not so many that we overfit or lose interpretability

Spline basis representation: Truncated power basis

Key question: How to systematically fit a piecewise polynomial, enforcing the smoothness constraints at the knots?

Truncated power basis for a degree-d spline:

$$\underbrace{1,\ X,\ X^2,\ldots,X^d}_{\text{base polynomials}} \ \cup\ \{\quad \underbrace{(X-c_k)_+^d}_{\text{truncated power basis}} \quad : k = 1\ldots K\}$$

where $(x-c)_+^d = \max\{x-c, 0\}^d$

- Then, we can write

$$f(x) = \beta_0 + \beta_1 x + \cdots + \beta_d x^d + \sum_{k=1}^{K} \beta_{d+k}\,(x-c_k)_+^d$$

- The term $(x-c_k)_+^d$ is inactive before the knot and changes the polynomial shape after it
- This representation automatically gives a degree-d spline with the desired smoothness

A toy numerical example: Piecewise linear spline

Example

Let X range in $[0, 8]$ with knots at $x = 2, 5$. Use piecewise linear polynomials (degree $d = 1$). By the DoF formula $(d + 1) + K = (1 + 1) + 2 = 4$ total parameters.

Basis representation:

$$b_0(x) = 1, \quad b_1(x) = x, \quad b_2(x) = (x - 2)_+, \quad b_3(x) = (x - 5)_+, \qquad (u)_+ := \max(u, 0).$$

Then the resulting linear spline model, fit by least squares, is

$$\widehat{y}(x) = \beta_0 \, b_0(x) + \beta_1 \, b_1(x) + \beta_2 \, b_2(x) + \beta_3 \, b_3(x).$$

Interpretation:

- β_0 is the base intercept.
- β_1 is the slope for $0 \leq x \leq 2$.
- β_2 changes the slope after $x = 2$, so the slope for $2 < x \leq 5$ is $\beta_1 + \beta_2$.
- β_3 further changes the slope after $x = 5$, so the slope for $5 < x \leq 8$ is $\beta_1 + \beta_2 + \beta_3$.

Pop-up quiz: Regression spline

Suppose we fit a **cubic spline** with $K = 3$ knots.

Question: How many degrees of freedom does this spline have after imposing the usual smoothness constraints?

A) 4, because a cubic polynomial has four coefficients.

B) 3, because there are three knots.

C) $K + 4 = 7$, because a cubic spline has $4 + K$ degrees of freedom.

D) $4(K + 1) = 16$, because there are four polynomial pieces and each has four coefficients.

Answer: C. Before constraints, there are $4(K + 1) = 16$ coefficients. The $K = 3$ knots impose $3K = 9$ smoothness constraints, leaving $16 - 9 = 7 = K + 4$ degrees of freedom.

Preview: Why natural splines?

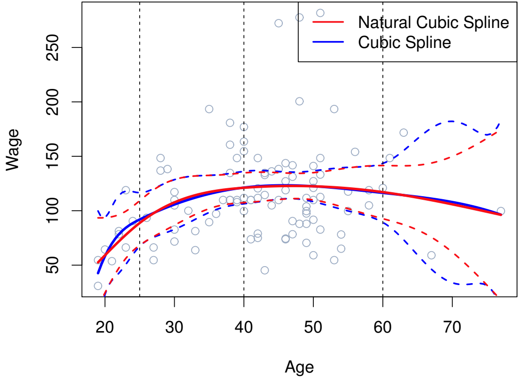

Figure: A cubic spline and a natural cubic spline, with three knots, fit to a subset of the Wage data. The dashed lines denote the knot locations [JWHT21, Figure 7.4].

Preview for next lecture:

- Cubic splines can behave erratically near the boundaries.
- Natural cubic splines add boundary constraints, making the tails linear.
- We will also discuss knot placement and smoothing splines.

Wrap-up & next steps

- **Basis functions** create transformed features $b_1(X), \ldots, b_K(X)$, allowing nonlinear fits while still using least squares
- **Regression splines** use piecewise polynomials joined smoothly at knots
- **Truncated power bases** give a compact way to represent splines and fit them using ordinary least squares
- **Next lecture:** natural splines, knot placement, and smoothing splines

Suggested reading: [JWHT21, Ch. 7.1–7.4]

References

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani.
An Introduction to Statistical Learning: with Applications in R, volume 112 of *Springer Texts in Statistics*.
Springer, New York, NY, 2nd edition, 2021.