STAT 151A Lecture 40

Henry Liev

3 December 2025

Remark 0.1 (Design Matrix for Regression Splines)

Many possible ways to parameterize the same column space → we'll use "truncated basis"
Focus on cubic splines (Polynomial has order 3)

$$\mathbf{X} = \begin{bmatrix} \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & & \uparrow \\ \vec{1} & \vec{x} & \vec{x}^2 & \vec{x}^3 & h(\vec{x}, c_1) & h(\vec{x}, c_2) & \cdots & h(\vec{x}, c_k) \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & & \downarrow \end{bmatrix}$$

Knots: $c_1 < c_2 < \cdots < c_k$

$h(x_i, c) = (x_i - c)_+^3 = \begin{cases} (x_i - c)^3 & x_i > c \\ 0 & x_i < c \end{cases}$

Let's consider $x_0 < c_1$ and $c_1 < x_1 < c_2$

Model for x_0

$y_0 \approx \hat\beta_0 + \hat\beta_1 x_0 + \hat\beta_2 x_0^2 + \hat\beta_3 x_0^3 + 0$

$y_0 \approx \hat\beta_0 + \hat\beta_1 x_1 + \hat\beta_2 x_1^2 + \hat\beta_3 x_1^3 + \hat\beta_4 (x_1 - c_1)^3$

$\frac{\partial \hat f(x_0)}{\partial x} \approx \cancel{\hat\beta_0} + \hat\beta_1 \cancel{x_0} + 2\hat\beta_2 x_0 + 3\hat\beta_3 x_0^{\cancelto{2}{3}}$

$\frac{\partial \hat f(x_1)}{\partial x} \approx \cancel{\hat\beta_0} + \hat\beta_1 \cancel{x_0} + 2\hat\beta_2 x_0 + 3\hat\beta_3 x_0^{\cancelto{2}{3}} + 3\hat\beta_4 (x_1 - c_1)^{\cancelto{2}{3}}$

Are also the same at c_1

This is true for all knots, and true for 2nd derivatives

Not true for 3rd derivatives

How many coefficients: $k + 4$

Remark 0.2 (Natural Splines)

Restrict regression spline further to force linearity for $x < c_1$ and $x > c_k$ (helps w/ extrapolation performance)

Changes basis in a complicated way (I won't show it) → ISLR/ESL

What to remember: How many coefficients? $k+2$ instead of $k+4$ because we have added extra restrictions

Remark 0.3 (Multiple $\vec{x}$'s)

How do we take regression spline idea and use it here?
Generalized additive model (GAMs)

$$y_i \approx \hat{\beta}_0 + \hat{f}_1(x_{1i}) + \hat{f}_2(x_{2i}) + \cdots + \hat{f}_p(x_{pi})$$

Where $\hat{f}_j(\cdot)$s are flexible nonlinear functions
Strategy: Use a different polynomial or spline basis for each $\vec{x}_j$

$$\mathbf{X} = \begin{bmatrix} \uparrow & \uparrow & \uparrow & & \uparrow & \uparrow & \uparrow & & \uparrow & \\ \vec{1} & b_1(\vec{x}_1) & b_2(\vec{x}_1) & \cdots & b_k(\vec{x}_1) & b_1(\vec{x}_2) & b_2(\vec{x}_2) & \cdots & b_k(\vec{x}_2) & \cdots \\ \downarrow & \downarrow & \downarrow & & \downarrow & \downarrow & \downarrow & & \downarrow & \end{bmatrix}$$

Remark 0.4 (Bigger Picture)

Zoom out: Our basic tool is linear combinations of x (or transformations thereof), maybe inside a fixed nonlinear function $\boxed{\mathbf{X}\beta}$
Instead of using linear combinations, use a list of decision rules