Regression Methods

Myrto Limnios

Autumn 2025 - Week 12

Regularisation - Splines

Basis functions

$y = X\beta + \varepsilon$

- We seek to estimate a function $\mu(x)$ based on data $(x_1, y_1), \ldots, (x_n, y_n)$.
- There are n parameters $\mu_1 = \mu(x_1), \ldots, \mu_n = \mu(x_n)$ (plus noise, ...), so we assume that $\mu(x)$ belongs to a suitable class of functions, defined for $x \in \mathcal{X}$.
- Simple linear model is

$$\mu_{n\times 1} = B_{n\times p}\beta_{p\times 1}, \quad \operatorname{rank}(B) = p \leq n,$$

  with the columns of B evaluations at $x_1, \ldots, x_n$ of **basis functions**.
- The basis functions may be
  - **global** (e.g., polynomials, trigonometric/Fourier functions),
  - **local** (e.g., splines),
  - **multiscale** (e.g., wavelets).
- We choose the basis for
  - suitability for the problem at hand (e.g., suitably smooth), and
  - computational reasons—want fast, preferably $\mathcal{O}(n)$, handling of $n \times n$ matrices.
- Focus on **spline functions**, on which there is a huge literature.

Aside: Polynomial regression

$$B = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{p-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_p & x_p^2 & & x_p^{p-1} \end{pmatrix}$$

- Classical approach is to fit a polynomial of degree $p-1$, i.e.,

$$\mu(x_j) = \beta_0 + \beta_1 x_j + \cdots + \beta_{p-1} x_j^{p-1},$$

and choose $\beta_0, \ldots, \beta_{p-1}$ to minimise the sum of squares

$$\sum_{j=1}^{n} \{y_j - \mu(x_j)\}^2 = \sum_{j=1}^{n} \left\{ y_j - (\beta_0 + \beta_1 x_j + \cdots + \beta_{p-1} x_j^{p-1}) \right\}^2,$$

giving $\widehat{\beta}_{p\times 1} = (B^{\mathrm{T}} B)^{-1} B^{\mathrm{T}} y$, where (j, i) element of $n \times p$ matrix B is x_j^{i-1}.

- Comments:
  - easily copes with missing values/unequally spaced observations;
  - use orthogonal polynomials to avoid numerical problems if n, k large;
  - sensitivity to observations at extremities of series often leads to poor fit;
  - usually doesn't work well because infinite differentiability everywhere is generally unnecessarily restrictive.

Piecewise linear basis

- Place **knots** of a univariate x at $x_1^* < \cdots < x_K^*$, and define **tent functions**

$$
\begin{aligned}
b_1(x) &= \begin{cases} (x_2^* - x)/(x_2^* - x_1^*), & x_1^* \le x \le x_2^*, \\ 0, & \text{otherwise,} \end{cases} \\
b_k(x) &= \begin{cases} (x - x_{k-1}^*)/(x_k^* - x_{k-1}^*), & x_{k-1}^* < x \le x_k^*, \\ (x_{k+1}^* - x)/(x_{k+1}^* - x_k^*), & x_k^* < x \le x_{k+1}^*, \end{cases} \quad k = 2, \ldots, K-1, \\
b_K(x) &= \begin{cases} (x - x_{K-1}^*)/(x_K^* - x_{K-1}^*), & x_{K-1}^* \le x \le x_K^*, \\ 0, & \text{otherwise :} \end{cases}
\end{aligned}
$$

  these are non-zero only in (x_{k-1}^*, x_{k+1}^*) (**compact support**) and take value 1 at x_k^*.

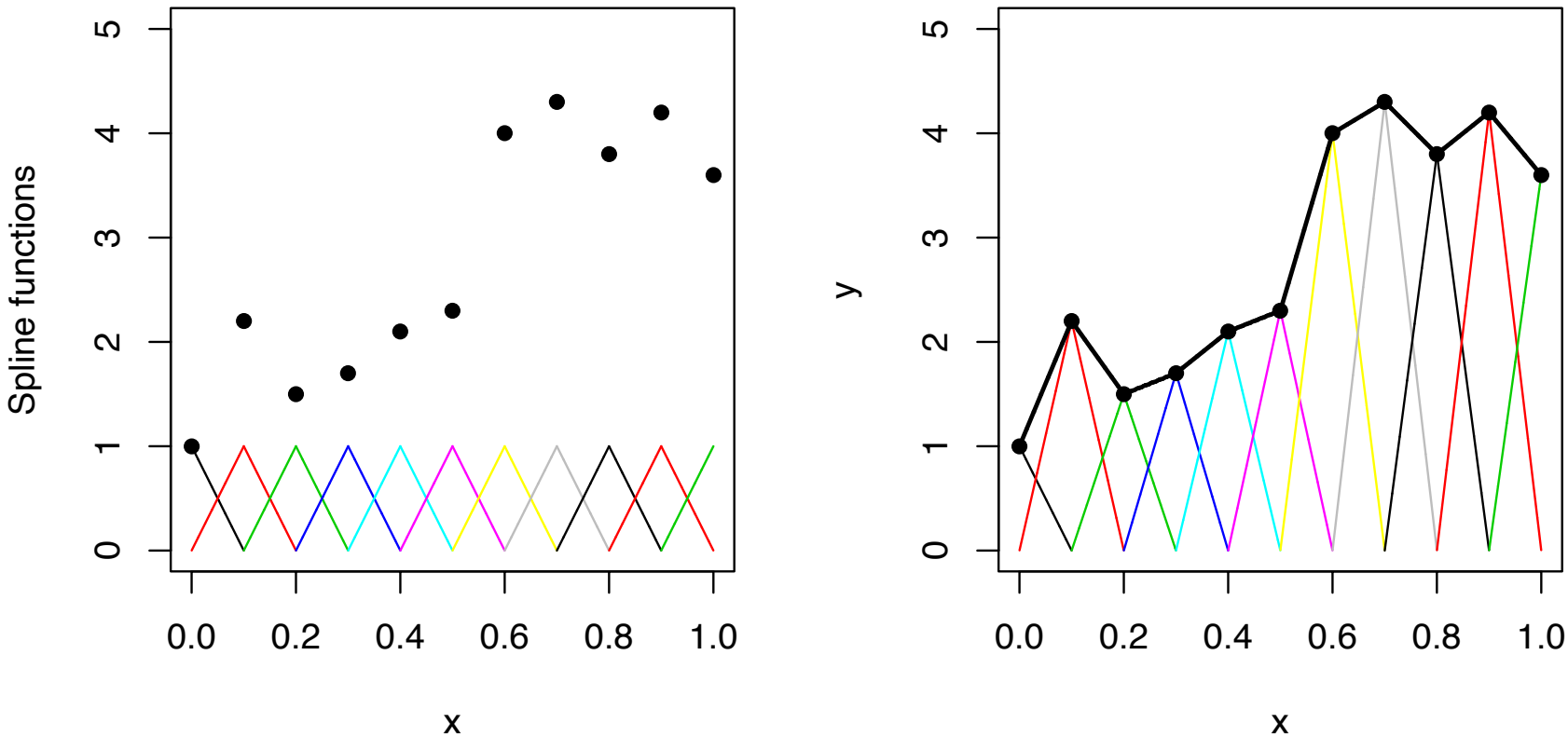
- An exact linear interpolant of data $y_1, \ldots, y_K$ at the knots is the function

$$\mu(x) = \sum_{k=1}^{K} b_k(x) y_k = B(x)^{\mathrm{T}} y,$$

  which by construction

  - passes through the points (x_k^*, y_k) and
  - is linear between the knots.

Piecewise linear basis



- Left: piecewise linear basis functions $b_k(x)$ and data (x_k^*, y_k).
- Right: functions $b_k(x)y_k$ and linear interpolant (bold).

Statistical use

- Aim for summary of the n observations, so interpolation not useful.
- Could use $K < n$ knots, but fit tends to depend heavily on their locations, so better to use high(ish) K and impose structure by penalising roughness of $\mu(x)$:

$$\widehat{\beta}_\lambda = \text{argmin}_\beta \left\{ \|y - B\beta\|^2 + \lambda \sum_{k=2}^{K-1} \left\{ \mu(x^*_{k-1}) - 2\mu(x^*_k) + \mu(x^*_{k+1}) \right\}^2 \right\}.$$

- The second term sums squared numerical second derivatives at the internal knots, and λ imposes the degree of penalisation:
  - $\lambda = 0$ (no penalty) gives the interpolant,
  - $\lambda \to \infty$ forces the second derivatives to be zero, so gives a straight-line fit.
- On setting $\beta_k = \mu(x^*_k)$ and writing

$$\begin{pmatrix} \beta_1 - 2\beta_2 + \beta_3 \\ \beta_2 - 2\beta_3 + \beta_4 \\ \beta_3 - 2\beta_4 + \beta_5 \\ \vdots \end{pmatrix} = \begin{pmatrix} 1 & -2 & 1 & 0 & 0 & 0 & \cdots \\ 0 & 1 & -2 & 1 & 0 & 0 & \cdots \\ 0 & 0 & 1 & -2 & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \end{pmatrix} = D_{(K-2)\times K}\beta_{K\times 1},$$

the penalty is
$\sum_{k=2}^{K-1}(\beta_{k-1} - 2\beta_k + \beta_{k+1})^2 = (D\beta)^{\mathrm{T}} D\beta = \beta^{\mathrm{T}} D^{\mathrm{T}} D\beta = \beta^{\mathrm{T}} S\beta$, say.

Penalized fit

- The penalty matrix S is of size $K \times K$ but of rank $K-2$, because

$$S1_K = Sx^*_{K\times 1} = 0_K :$$

the null space of S consists of all straight lines $\beta_0 1_K + \beta_1 x^*$, which are unpenalised.

- Hence (recalling ridge regression),

$$\widehat{\beta}_\lambda = \text{argmin}_\beta \left\{ \|y - B\beta\|^2 + \lambda \beta^{\mathrm{T}} S \beta \right\} = (B^{\mathrm{T}}B + \lambda S)^{-1} B^{\mathrm{T}} y$$

giving

$$\textbf{fitted values} \qquad \widehat{y} \quad = \quad B\widehat{\beta}_\lambda = B(B^{\mathrm{T}}B + \lambda S)^{-1} B^{\mathrm{T}} y = H_\lambda y,$$

$$\textbf{equivalent degrees of freedom} \quad \mathbf{df}_\lambda \quad = \quad \text{tr}(H_\lambda) = \sum_{k=1}^{K} \frac{1}{1+\eta_k \lambda},$$

where

- $\eta_1 \leq \cdots \leq \eta_K \in [0,1]$ are the eigenvalues of $(B^{\mathrm{T}}B)^{-1/2} S (B^{\mathrm{T}}B)^{-1/2}$,
- $\eta_1 = \eta_2 = 0$, corresponding to the null space of S, so
- $\mathbf{df}_\lambda$ is monotone decreasing in λ, with

$$(\lambda = 0) \qquad K \geq \mathbf{df}_\lambda \geq 2 \qquad (\lambda \to \infty).$$

Higher-order splines

- The *p*th **degree spline** basis with **knots** $x_1^* < \cdots < x_K^*$ is

$$1, x, \ldots, x^p, (x - x_1^*)_+^p, \ldots, (x - x_K^*)_+^p,$$

  where $u_+ = \max(u, 0)$ is the **positive part function**.

- The resulting basis matrix B is highly collinear and gives an implausible statistical model.

- *B*-**spline** bases span the same linear space, but have better numerical properties. They are defined by adding **boundary knots** x_0^* and x_{K+1}^* and setting up an **augmented knot sequence**

$$\tau_1 \leq \cdots \leq \tau_M \leq x_0^* \leq \tau_{M+1} = x_1^* \leq \cdots \leq \tau_{M+K} = x_K^* \leq x_{K+1}^* \leq \tau_{K+1+M} \leq \cdots \leq \tau_{K+2M};$$

  typically the τ_k outside $[x_0^*, x_{K+1}^*]$ are set to the boundary knot values. Then

$$B_{k,1}(x) = I(\tau_k \leq x < \tau_{k+1}), \quad k = 1, \ldots, K + 2M - 1,$$

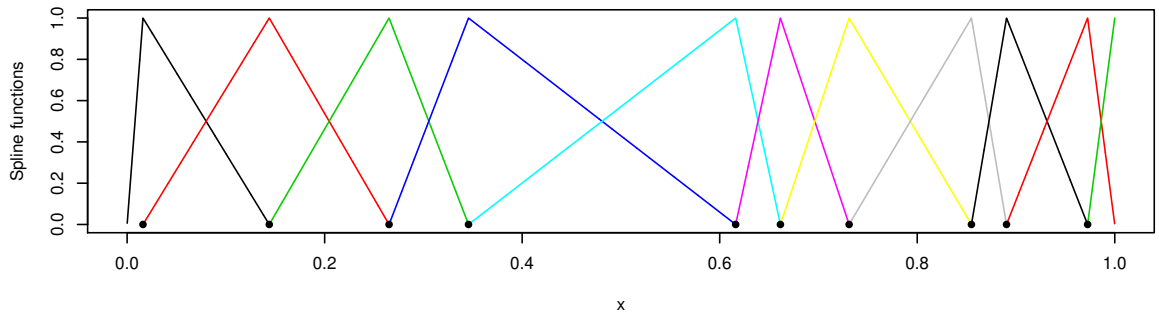
$$B_{k,m}(x) = \frac{x - \tau_k}{\tau_{k+m-1} - \tau_k} B_{k,m-1}(x) + \frac{\tau_{k+m} - x}{\tau_{k+m} - \tau_{k+1}} B_{k+1,m-1}(x), \quad k = 1, \ldots, K + 2M - m$$

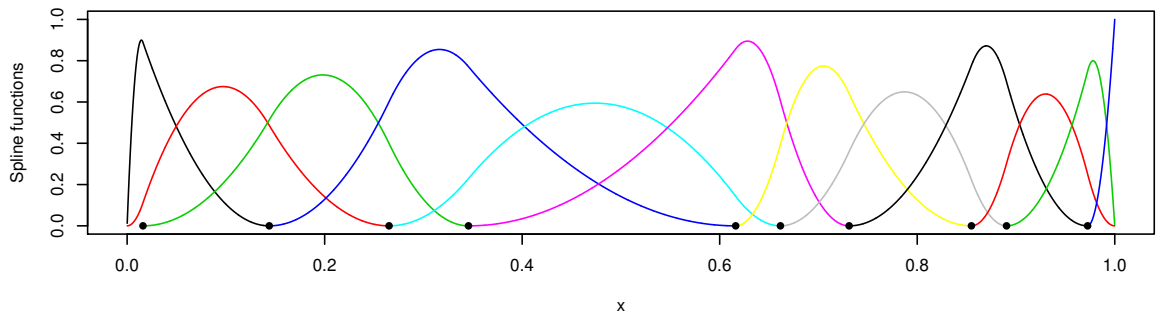
  where we set $B_{k,1} \equiv 0$ if $\tau_k = \tau_{k+1}$ (avoiding division by zero).

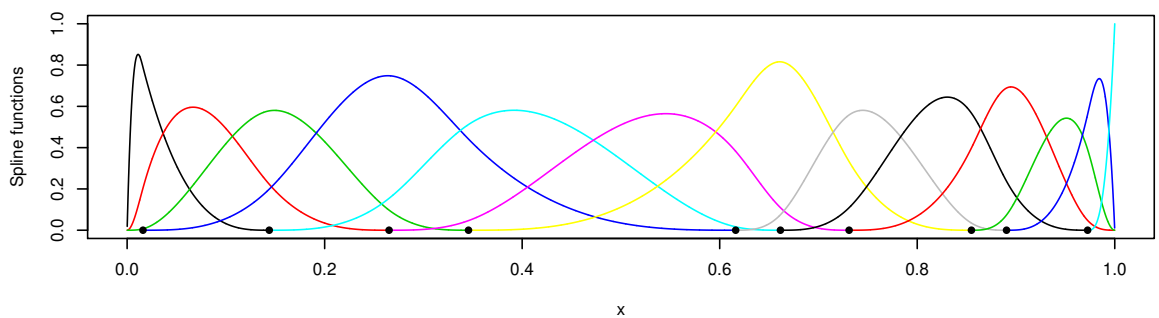
- Cubic splines ($p = 3$, $M = 4$) give visually smooth functions.

- $K = 10$ on the next slide, with $M = 2$ (linear), $M = 3$ (quadratic) and $M = 4$ (cubic), and the τ_k set to equal the boundary knots.

Linear, quadratic and cubic B-splines







Natural cubic spline

- Suppose the x_j are distinct (no loss of generality) and

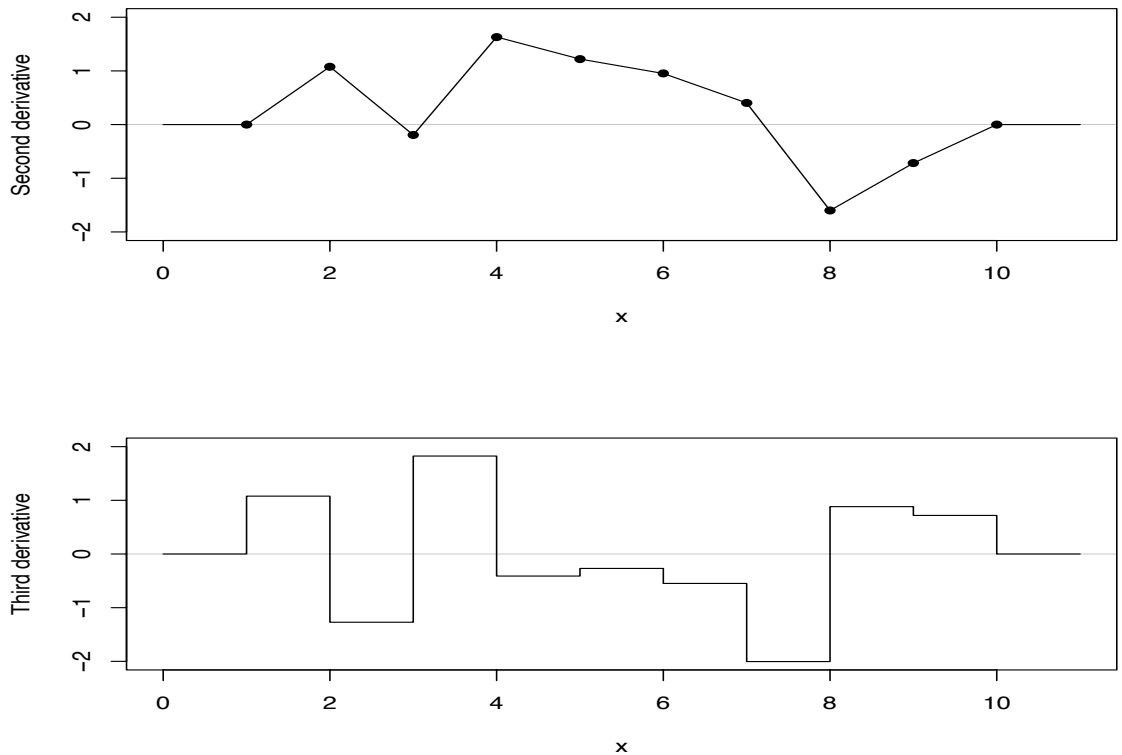
$$a < x_1 < \cdots < x_n < b, \quad \mathcal{X} = [a, b] \subset \mathbb{R}.$$

- A **natural cubic spline** adds the constraint that the function is linear outside $[x_1, x_n]$, and thus avoids high variance due to quadratic and higher terms outside this interval.
- A natural cubic spline
  - has $K = n$ knots, at $x_1 < \cdots < x_n$,
  - is a cubic polynomial on each interval between knots,
  - is continuous, with continuous first and second derivatives at each knot, and
  - is linear on $[a, x_1]$ and $[x_n, b]$, with zero second and third derivatives at x_1 and x_n,
  - has

$$2 + 4(n-1) + 2 \text{ parameters } - 3n \text{ linear constraints } = n$$

    degrees of freedom (df), which can be split into
    - 2 df for a linear fit, plus
    - $n - 2$ df for the second derivatives $\mu''(x_2), \ldots, \mu''(x_{n-1})$.

Natural cubic spline



- A natural cubic spline may be constructed by integrating a linear second derivative function $\mu''(x)$ which is determined by $\mu''(x_2), \ldots, \mu''(x_{K-1})$ and because $\mu''(x) \equiv 0$ for $x \notin (x_1, x_K)$.
- On integrating twice we gain two constants:
  $\mu(x) = \beta_0 + \beta_1 x + \int_0^x \int_0^{x'} \mu''(u)\, \mathrm{d}u\, \mathrm{d}x'$.
- Above $x_1 = 1, \ldots, x_{10} = 10$, so the spline is determined by $\mu''(2), \ldots, \mu''(9)$ and the line.

Optimality of natural cubic splines

- Let $\mathcal{S}_2(\mathcal{X})$ denote the set of functions μ differentiable on $\mathcal{X} = [a, b]$ with absolutely continuous first derivative μ': i.e., there exists an integrable function μ'' such that $\int_a^x \mu''(u)\mathrm{d}u = \mu'(x) - \mu'(a)$ for $x \in \mathcal{X}$.
- Clearly any μ with two continuous derivatives on $\mathcal{X}$ lies in $\mathcal{S}_2(\mathcal{X})$.

Theorem 20

Suppose $n \geq 2$, that $a < x_1 < \cdots < x_n < b$, and that μ is the natural cubic spline interpolating $y_1, \ldots, y_n$ at $x_1, \ldots, x_n$. If $\tilde{\mu} \in \mathcal{S}_2(\mathcal{X})$ also interpolates the y_j, then

$$\int_{\mathcal{X}} \tilde{\mu}''^2 \geq \int_{\mathcal{X}} \mu''^2,$$

with equality iff $\tilde{\mu} \equiv \mu$.

- Thus μ minimises the **roughness penalty** $\lambda \int_{\mathcal{X}} \mu''^2$ in a larger class of functions than that to which it belongs, making it a natural choice as an interpolant, because minimising

$$\sum_{j=1}^{n} \{y_j - \tilde{\mu}(x_j)\}^2 + \lambda \int_{\mathcal{X}} \tilde{\mu}''(x)^2 \, \mathrm{d}x$$

for $\tilde{\mu} \in \mathcal{S}_2(\mathcal{X})$ will automatically result in a natural cubic spline μ: if $\tilde{\mu}(x_j) = \mu(x_j)$, then the penalty is reduced by using μ.

Proof Theorem 20

Let $\nu = \tilde{\mu} - \mu \in \mathcal{S}_2(\mathcal{X})$, and note that $\nu(x_j) = 0$ for each j, since $\mu(x_j) = \tilde{\mu}(x_j) = y_j$. The natural boundary conditions imply that $\mu''(a) = \mu''(b) = 0$, so integration by parts yields

$$0 = \big[\mu''(x)\nu'(x)\big]_a^b = \int_{\mathcal{X}} (\mu''\nu')' = \int_{\mathcal{X}} \mu''\nu'' + \int_{\mathcal{X}} \mu'''\nu',$$

and hence the facts that μ''' is piecewise constant and that $\nu(x_j) = 0$ yields

$$\int_{\mathcal{X}} \mu''\nu'' = -\int_{\mathcal{X}} \mu'''\nu' = -\sum_{j=1}^{n-1} \mu'''(x_j^+) \int_{x_j}^{x_{j+1}} \nu' = -\sum_{j=1}^{n-1} \mu'''(x_j^+)\{\nu(x_{j+1}) - \nu(x_j)\} = 0.$$

Hence

$$\int_{\mathcal{X}} \tilde{\mu}''^2 = \int_{\mathcal{X}} (\mu''+\nu'')^2 = \int_{\mathcal{X}} \mu''^2 + 2\int_{\mathcal{X}} \mu''\nu'' + \int_{\mathcal{X}} \nu''^2 = \int_{\mathcal{X}} \mu''^2 + \int_{\mathcal{X}} \nu''^2 \geq \int_{\mathcal{X}} \mu''^2,$$

wth equality iff $\nu''(x) \equiv 0$. This occurs iff $\nu(x)$ is linear, but since $\nu(x_j) = 0$ at at least two points, $\nu(x) = 0$ for all $x \in \mathcal{X}$.

More splines

- Sometimes cyclic effects (e.g., seasonality, diurnal variation) must be modelled smoothly, so (e.g.) December joins smoothly onto January. Then the penalty and spline basis must be modified accordingly, to give a **cyclic (cubic) spline**.
- **P-splines** are a version of B-splines (usually with equally-spaced knots) in which a difference penalty is applied to the parameters to control the wiggliness of μ, e.g.,

$$\sum_{k=1}^{K-1} w_k(\beta_{k+1}-\beta_k)^2 = \beta^{\mathrm{T}} D^{\mathrm{T}} W D \beta, \quad \text{with} \quad D = \begin{pmatrix} -1 & 1 & 0 & 0\cdots & \\ 0 & -1 & 1 & 0 & \cdots \\ \cdot & \cdot & \cdot & \cdot & \cdots \end{pmatrix},$$

  and $W = \mathrm{diag}(w_1, \ldots, w_{K-1})$. These are easy to set up and flexible, but messy if the knots are not equi-spaced, and the penalty is less readily interpreted.
- For an **adaptive spline** we can let $w_k \equiv w_k(x)$ vary with x, for example setting $w(x) = B(x)\lambda_{L\times 1}$ and thus having $D^{\mathrm{T}} W D = \sum_l \lambda_l D^{\mathrm{T}} \mathrm{diag}\{B_l(x)\} D$, where $B_l(x)$ is the lth column of $B(x)$, then estimating the vector λ.
- Other possibilities include (Wood, 2017, Chapter 5)
  - **shape-constrained splines** to impose, e.g., monotonicity on the fit;
  - **thin-plate**, **Duchon** and **tensor product** splines used in spatial problems; and
  - **soap film splines** used when smoothing over complex domains.

Motorcycle data: adaptive fit

Standard (left) and adaptive (right) spline fits, the latter with $K = 40$ and $L = 5$: