Smoothing splines

Statistical Learning
CLAMSES - University of Milano-Bicocca

Aldo Solari

References

- Bowman, Evers. Lecture Notes on Nonparametric Smoothing. Section 3
- Eilers, Marx (1996). Flexible smoothing with B-splines and penalties. Statistical science, 11(2), 89–121.

Natural cubic spline

- A set of n points (x_i, y_i) can be exactly interpolated using a natural cubic spline with the $x_1 < \ldots < x_n$ as knots. The interpolating natural cubic spline is unique.
- Amongst all functions on $[a, b]$ which are twice continuously differentiable and which interpolate the set of points (x_i, y_i), a natural cubic spline with knots at the x_i yields the smallest roughness penalty

$$\int_a^b (f''(x))^2 \, dx$$

- $f''(x)$ is the second derivative of f with respect to x - it would be zero if f were linear, so this measures the curvature of f at x.

Smoothing spline

- Smoothing splines circumvent the problem of knot selection by performing regularized regression over the natural spline basis, placing knots at all inputs $x_1, \ldots, x_n$
- With inputs $x_1 < \ldots < x_n$ contained in an interval $[a, b]$, the minimiser of

$$\hat{f} = \arg\min_{f \in \mathcal{C}_2} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \int_a^b (f''(x))^2 dx$$

  amongst all twice continuously differentiable functions on $[a, b]$ is given by a a natural cubic spline with knots in the unique x_i

- The previous result tells us that we can choose natural cubic spline basis $B_1, \ldots, B_n$ with knots $\xi_1 = x_1, \ldots, \xi_n = x_n$ and solve

$$\hat{\beta}_\lambda = \arg\min_\beta \sum_{i=1}^n (y_i - \sum_{j=1}^n \beta_j B_j(x_i))^2 + \lambda \int_a^b \Big(\sum_{j=1}^n \beta_j B_j''(x) \Big)^2 dx$$

to obtain the smoothing spline estimate $\hat{f}(x) = \sum_{i=1}^n \hat{\beta}_j B_j(x)$

- Rewriting

$$\hat{\beta}_\lambda = \arg\min_\beta \|y - B\beta\|^2 + \lambda \beta^t \Omega \beta$$

where $B_{ij} = B_j(x_i)$ and $\Omega_{jk} = \int B_j''(x) B_k''(x)\, dx$, shows the smoothing spline problem to be a type of generalized ridge regression problem with solution

$$\hat{\beta}_\lambda = (B^t B + \lambda\Omega)^{-1} B^t y$$

- Fitted values in Reinsch form

$$\begin{aligned} \hat{y} &= B(B^t B + \lambda\Omega)^{-1} B^t y \\ &= (I_n + \lambda K)^{-1} y \end{aligned}$$

  where $K = (B^t)^{-1}\Omega B^{-1}$ does not depend on λ, and $S = (I_n + \lambda K)^{-1}$ is the $n \times n$ *smoothing matrix*

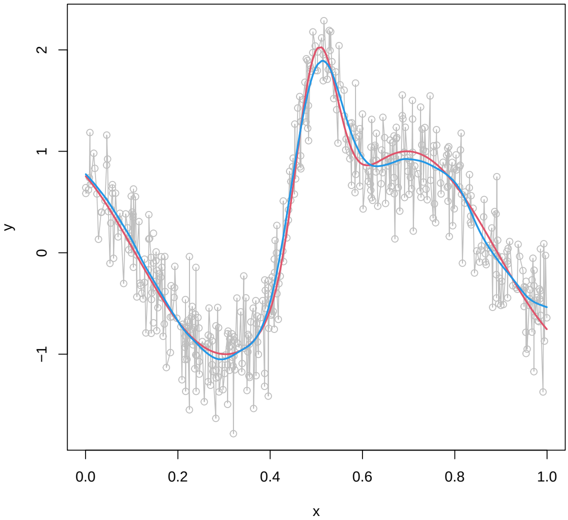
- Leave-one-out cross validation

$$\mathrm{LOO} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - S_{ii}}\right)^2$$

- Generalized cross validation

$$\mathrm{GCV} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - \mathrm{tr}(S)/n}\right)^2$$

  where $\mathrm{tr}(S)$ is the effective degrees of freedom



`smooth.spline` result with $\lambda = 0$ and 6.9e-15 by LOO

Reinsch original solution

- The original Reinsch (1967) algorithm solves the constrained optimization problem

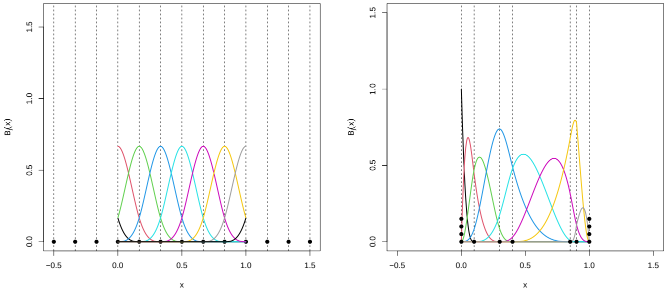
$$\hat{f} = \arg\min_{f \in \mathcal{C}_2} \int_a^b (f''(x))^2 dx \text{ such that } \sum_{i=1}^{n} (y_i - f(x_i))^2 \leq c$$

- The previous formulation with a Lagrange parameter on the integral smoothing term instead of the least squares term is equivalent
- See `casl_smspline` implementation in Section 2.6 of CASL

P-splines

B-spline basis

- The truncated power basis suffers from computational issues. The *B*-spline basis is a re-parametrization of the truncated power basis spanning an equivalent space
- The appearance of *B*-splines depends on their knot spacing, e.g.
  - uniform *B*-splines on equidistant knots;
  - non-uniform *B*-splines on unevenly spaced knots and repeated boundary;



Left plot: uniform cubic B-splines with equidistant knots

Right plot: non-uniform cubic B-splines with unevenly spaced knots and duplicated boundary knots

B-spline basis

– B-splines can be computed as differences of truncated power functions
– The general formula for equally-spaced knots is

$$B_j(x) = \frac{(-1)^{M+1}\Delta^{M+1} f_j(x, M)}{h^M M!}$$

satisfying

$$\sum_j B_j(x) = 1$$

where $f_j(x, M) = (x - \xi_j)_+^M$, h is is the distance between knots and Δ^O is the Oth order difference with
$\Delta f_j(x, M) = f_j(x, M) - f_{j-1}(x, M)$,
$\Delta^2 f_j(x, M) = \Delta(\Delta f_j(x, M)) = f_j(x, M) - 2f_{j-1}(x, M) + f_{j-2}(x, M)$

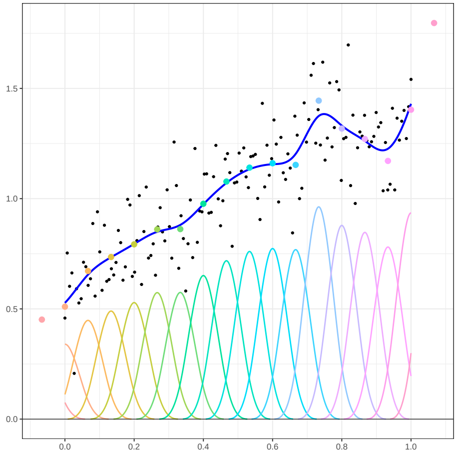
P-splines

- There is an intermediate solution between regression and smoothing splines, proposed more recently by Eilers and Marx (1996)
- P-splines use a basis of (quadratic or cubic) B-splines, B, computed on x and using equally-spaced knots. Minimize

  $$\|y - B\beta\|^2 + \lambda\|D\beta\|^2$$

  where $D = \Delta^O$ is the matrix of Oth order differences, with $\Delta\beta_j = \beta_j - \beta_{j-1}$, $\Delta^2\beta_j = \Delta(\Delta\beta_j) = \beta_j - 2\beta_{j-1} + \beta_{j-2}$ and so on for higher O. Mostly $O = 2$ or $O = 3$ is used.
- Minimization leads to the system of equations

  $$(B^tB + \lambda D^tD)\hat{\beta} = B^ty$$



The core idea of *P*-splines: a sum of B-spline basis functions, with gradually changing heights. The blue curve shows the *P*-spline fit, and the large dots the *B*-spline coefficients. R code in `f-ps-show.R`

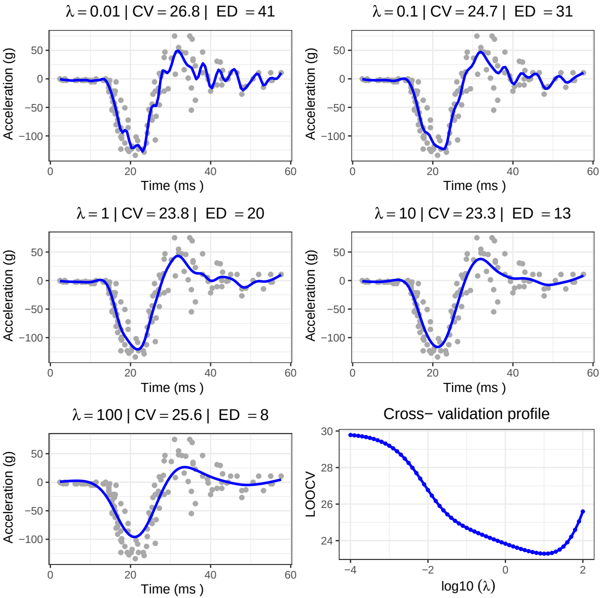
Cross-validation

- We have that $\hat{y} = B(B^tB + \lambda D^tD)^{-1}B^ty = Sy$
-
$$\mathrm{LOO} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i^{(-i)})^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - S_{ii}}\right)^2$$
-
$$\mathrm{GCV} = \frac{1}{n}\sum_{i=1}^{n}\frac{(y_i - \hat{y}_i)^2}{(1 - \mathrm{tr}(S)/n)^2}$$
- We can compute the trace of R without actually computing its diagonal, using

$$\mathrm{tr}(S) = \mathrm{tr}((B^tB + P)^{-1}B^tB) = \mathrm{tr}(I_n - (B^tB + P)^{-1}P)$$

where $P = \lambda D^tD$

mcycle



λ = 0.01 | CV = 26.8 | ED = 41
λ = 0.1 | CV = 24.7 | ED = 31
λ = 1 | CV = 23.8 | ED = 20
λ = 10 | CV = 23.3 | ED = 13
λ = 100 | CV = 25.6 | ED = 8
Cross− validation profile
Acceleration (g)
Time (ms)
LOOCV
log10 (λ)