NONPARAMETRIC REGRESSION WITH PENALIZED SPLINES

Let

$$Y_i = m(X_i) + \varepsilon_i \quad , \quad i=1,\dots,n,$$

$\varepsilon_1,\dots,\varepsilon_n$ iid with $\mathbb{E}\,\varepsilon_1 = 0$, $\mathbb{E}\,\varepsilon_1^2 = \sigma^2$, independent of $X_1,\dots,X_n \in [0,1]$.

Before introducing "penalized splines", we introduce the "smoothing spline" estimator.

The smoothing spline estimator is defined as

$$\hat{m}_n^{sspl} = \underset{g \in W_2}{\operatorname{argmin}} \sum_{i=1}^n \left(Y_i - g(X_i)\right)^2 + \lambda \int_0^1 [g''(x)]^2\,dx,$$

where

$$W_2 = \left\{ g : [0,1] \to \mathbb{R} : \ g' \text{ is continuous and } \int_0^1 [g''(x)]^2\,dx < \infty \right\}.$$

The space of functions W_2 is called a Sobolev space (these can be defined more generally using higher-order derivatives).

Note that $\mathcal{H}(2,L)$ on $[0,1]$ is contained in W_2.

Idea is to tune the wiggliness of $\hat{m}_n^{sspl}$ by the choice of λ.

How do we find $\hat{m}_n^{sspl}$ in the space W_2 which minimizes our objective function?

It turns out that the solution $\hat{m}_n^{sspl}$ is a function that is

(i) a continuous function with 2 continuous derivatives on $[0,1]$

(ii) a polynomial of degree 3 on the intervals $[x_1,x_2),\dots,[x_{n-1},x_n)$

(iii) a polynomial of degree 1 on the intervals $[0,x_1)$ and $[x_n,1]$

See Wahba (1990), the Foreword. This is a fascinating result!

Functions on $[0,1]$ satisfying (i), (ii), and (iii) are called natural cubic splines.

We can construct a set of basis functions for this space of natural cubic splines and parameterize the problem.

Interesting: There is a knot at every single data point!

Our cubic B-spline basis functions from the previous lecture are not a basis for the natural cubic splines, because they build functions which are cubic instead of linear in the boundary intervals.

To learn how to construct a basis for the natural cubic splines, see Elements of Statistical Learning by Hastie, Tibshirani, and Friedman.

It turns out that B-spline bases afford computational advantages, since the matrix $B^T B$ is banded under B-splines, and this structure can be exploited.

see discussion on pg 189 of ESL 2nd Ed.

Due to our love of B-splines, we will now abandon the smoothing spline estimator (which requires a basis for natural splines) and consider an estimator which will be nearly identical in practice:

Let $\mathcal{M}_n$ be the space of cubic splines on $[0,1]$ based on some knots

$$\kappa_{-3} = \kappa_{-2} = \kappa_{-1} = \kappa_0 < \kappa_1 < \dots < \kappa_{K_n} = \kappa_{K_n+1} = \kappa_{K_n+2} = \kappa_{K_n+3}.$$

Then define the penalized spline estimator $\hat{m}_n^{pspl}$ of m as

$$\hat{m}_n^{pspl} = \underset{g \in \mathcal{M}_n}{\operatorname{argmin}} \sum_{i=1}^{n} \left(Y_i - g(X_i) \right)^2 + \lambda \int_0^1 [g''(x)]^2 \, dx.$$

Note that we have only changed the space in which we are searching for a minimizer from W_2 to $\mathcal{M}_n$.

The idea is to choose K_n to be quite large and then tune the wiggliness by selecting a value for λ.

Exercise: Let $b_1, \ldots, b_{d_n}$, $d_n = K_n + 3$ be the cubic B-spline functions comprising a basis for $\mathcal{M}_n$. Find a matrix representation of $\hat{m}_n^{pspl}(x_0)$.

Solution: Note that for any $g \in \mathcal{M}_n$ we may write

$$g(x) = \sum_{\ell=1}^{d_n} \alpha_\ell b_\ell(x) \quad \text{for some } \alpha_1, \ldots, \alpha_{d_n} \in \mathbb{R}.$$

Now, setting

$$B = \Big(b_\ell(X_i) \Big)_{1 \le i \le n,\, 1 \le \ell \le d_n},$$

we may write

$$\sum_{i=1}^{n} \Big(Y_i - g(X_i) \Big)^2 = \| \underset{\sim}{Y} - B \underset{\sim}{\alpha} \|_2^2 .$$

Moreover, we have $g''(x) = \sum_{\ell=1}^{d_n} \alpha_\ell b_\ell''(x)$, so that

$$\int_0^1 \left[g''(x) \right]^2 dx = \int_0^1 \sum_{\ell=1}^{d_n} \alpha_\ell b_\ell''(x) \sum_{j=1}^{d_n} \alpha_j b_j''(x)\, dx$$

$$= \sum_{\ell=1}^{d_n} \sum_{j=1}^{d_n} \alpha_\ell \alpha_j \int_0^1 b_\ell''(x) b_j''(x)\, dx$$

$$= \underset{\sim}{\alpha}^T \Omega \underset{\sim}{\alpha},$$

where

$$\underset{d_n \times d_n}{\Omega} = \left(\int_0^1 b_\ell''(x) b_j''(x) \right)_{1 \le \ell, j \le d_n} .$$

So the objective function is given by

$$\| \underset{\sim}{Y} - B \underset{\sim}{\alpha} \|_2^2 + \lambda \underset{\sim}{\alpha}^T \Omega \underset{\sim}{\alpha},$$

which is minimized in $\underset{\sim}{\alpha}$ at

$$\underset{\sim}{\hat{\alpha}} = \left(B^T B + \lambda \Omega \right)^{-1} B^T \underset{\sim}{Y}.$$

So we may write

$$\hat{m}_n^{\text{pspl}}(x_0) = b_{x_0}^T \underset{\sim}{\hat{\alpha}} \quad , \quad b_{x_0} = \left(b_1(x_0), \ldots, b_{d_n}(x_0) \right)^T.$$

Exercise: Derive a formula for computing the entries of Ω.

Solution: We have, for each $1 \le \ell, j \le d_n$,

$$\int_0^1 b_\ell''(x)\, b_j''(x)\, dx = \sum_{k=0}^{K-1} \int_{\kappa_k}^{\kappa_{k+1}} b_\ell''(x)\, b_j''(x)\, dx$$

$b_\ell''(x) = b_\ell(\kappa_k) + \frac{x-\kappa_k}{\kappa_{k+1}-\kappa_k}\left(b_\ell(\kappa_{k+1}) - b_\ell(\kappa_k)\right)$

$\tau = \frac{x-\kappa_k}{\kappa_{k+1}-\kappa_k} \iff x = \kappa_k + \tau(\kappa_{k+1}-\kappa_k)$

$dx = (\kappa_{k+1}-\kappa_k)\, d\tau$

$$= \sum_{k=0}^{K-1} (\kappa_{k+1}-\kappa_k) \int_0^1 \Big[\underbrace{b_\ell''(\kappa_k)}_{\theta_k} + \tau\big(\underbrace{b_\ell''(\kappa_{k+1})}_{\theta_{k+1}} - b_\ell''(\kappa_k)\big)\Big]\Big[\underbrace{b_j''(\kappa_k)}_{\gamma_k} + \tau\big(\underbrace{b_j''(\kappa_{k+1})}_{\gamma_{k+1}} - b_j''(\kappa_k)\big)\Big]\, d\tau$$

Exploit the fact that b_j'' is piecewise linear on each interval.

$$= \sum_{k=0}^{K-1} (\kappa_{k+1}-\kappa_k) \int_0^1 \left[\theta_k + \tau(\theta_{k+1}-\theta_k)\right]\left[\gamma_k + \tau(\gamma_{k+1}-\gamma_k)\right] d\tau$$

$$= \sum_{k=0}^{K-1} (\kappa_{k+1}-\kappa_k) \int_0^1 \Big\{ \theta_k \gamma_k + \tau\left[\gamma_k(\theta_{k+1}-\theta_k) + \theta_k(\gamma_{k+1}-\gamma_k)\right] + \tau^2 (\theta_{k+1}-\theta_k)(\gamma_{k+1}-\gamma_k) \Big\}\, d\tau$$

$$= \sum_{k=0}^{K-1} (\kappa_{k+1}-\kappa_k) \Big\{ \theta_k \gamma_k + \frac{1}{2}\left[\gamma_k(\theta_{k+1}-\theta_k) + \theta_k(\gamma_{k+1}-\gamma_k)\right] + \frac{1}{3}(\theta_{k+1}-\theta_k)(\gamma_{k+1}-\gamma_k) \Big\}$$

$$= \sum_{k=0}^{K-1} (\kappa_{k+1}-\kappa_k) \Big\{ \frac{1}{2}\left(\gamma_k \theta_{k+1} + \gamma_{k+1}\theta_k\right) + \frac{1}{3}(\theta_{k+1}-\theta_k)(\gamma_{k+1}-\gamma_k) \Big\}$$

$$= \sum_{k=0}^{K-1} (\kappa_{k+1} - \kappa_k) \Big[\tfrac{1}{2} \big(b_j''(\kappa_k) b_\ell''(\kappa_{k+1}) + b_j''(\kappa_{k+1}) b_\ell''(\kappa_k) \big) + \tfrac{1}{3} \big(b_\ell''(\kappa_{k+1}) - b_\ell''(\kappa_k) \big) \big(b_j''(\kappa_{k+1}) - b_j''(\kappa_k) \big) \Big].$$

Result: If $m \in W_2$ then for large enough n, we have

$$\mathbb{E} \int_0^1 [\hat{m}_n^{sspl}(x) - m(x)]^2 dx \leq C \cdot n^{-4/5}.$$

Note that the rate is like $n^{-\frac{2\beta}{2\beta+1}}$ with $\beta = 2$.

The proof is much more complicated than any we have done in the course.

The penalized spline estimator $\hat{m}_n^{pspl}$ is very similar to $\hat{m}_n^{sspl}$.

Analysis of the smoother matrix:

Let

$$\hat{\underset{\sim}{m}}_n^{pspl} = \left(\hat{m}_n^{pspl}(X_1), \dots, \hat{m}_n^{pspl}(X_n) \right)^T$$

be the $n \times 1$ vector with entries given by $\hat{m}_n^{pspl}$ evaluated at $X_1, \dots, X_n$.

So $\hat{\underset{\sim}{m}}_n^{pspl}$ is the vector of "fitted values".

Note that we may write

$$\hat{\underset{\sim}{m}}_n^{pspl} = B \hat{\underset{\sim}{\alpha}} = B(B^TB + \lambda\Omega)^{-1} B^T Y = S Y,$$

where $S = B(B^TB + \lambda\Omega)^{-1}B^T$.

The matrix S is called the smoother matrix.

Row i of S gives the weights $w_1, \dots, w_n$ such that $\hat{m}_n^{pspl}(X_i) = \sum_{j=1}^n w_j Y_j$.

The values in each row of S look like weights that could come from a kernel.

Silverman (1984) investigated this and found that, asymptotically, smoothing splines are the same as kernel smoothing (N-W estimator) under a specific choice of kernel and with a local choice of the bandwidth:

$$K(u) = \frac{1}{2} e^{-|u|/\sqrt{2}} \cdot \sin\left(|u|/\sqrt{2} + \frac{\pi}{4}\right), \qquad h(x) = \left(\frac{\lambda/n}{f(x)}\right)^{1/4}.$$

Note: Every linear estimator has a smoother matrix:

An estimator $\hat{m}_n$ of m is called a linear estimator if

$$\hat{m}_n(x) = \sum_{i=1}^{n} W_{ni}(x)\, Y_i$$

for some weights $W_{n1}(x), \ldots, W_{nn}(x)$ for each x.

For any linear estimator, we can write

$$\begin{bmatrix} \hat{m}_n(X_1) \\ \vdots \\ \hat{m}_n(X_n) \end{bmatrix} = \underbrace{\begin{bmatrix} W_{n1}(X_1) & \cdots & W_{nn}(X_1) \\ & & \\ W_{n1}(X_n) & \cdots & W_{nn}(X_n) \end{bmatrix}}_{=S, \text{ the smoother matrix}} \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}$$

Exercise: Plot some rows of the smoothing matrix for penalized splines.
Plot the same rows of that for the N-W estimator under the Silverman kernel with bandwidth $h = \lambda^{1/4}$. Generate $X_1, \ldots, X_n \overset{ind}{\sim} U(0,1)$.

We can also learn something from the eigendecomposition of S.

Let

$$S = U \Lambda U^T,$$

where $\underset{n\times n}{U}$ has orthonormal columns $\left(U^T U = I\right)$ and

$$\Lambda = \mathrm{diag}\left(\lambda_1, \ldots, \lambda_n\right),$$

with $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$. Then we may write

$$\hat{\underset{\sim}{m}}_n^{pspl} = S \underset{\sim}{Y} = U \Lambda U^T \underset{\sim}{Y} = U \Lambda \underbrace{(U^T U)^{-1} U^T \underset{\sim}{Y}}_{\hat{\underset{\sim}{\gamma}}} = U \Lambda \hat{\underset{\sim}{\gamma}}$$

where we can interpret $\hat{\underset{\sim}{\gamma}}$ as coefficients of a least-squares regression of $\underset{\sim}{Y}$ onto the columns of U, which are the eigenvectors of S.

So the fitted values in $\hat{\underset{\sim}{m}}_n^{pspl}$ result from projecting $\underset{\sim}{Y}$ onto the columns of U, but then shrinking, via Λ, the coefficients towards zero, such that eigenvectors with smaller eigenvalues are suppressed.

Exercise: Plot the eigenvectors of S from penalized splines.
Check what they look like at different λ values.

The eigenstructure of the smoother matrices $B(B^TB)^{-1}B^T$ and the penalized splines counterpart $B(B^TB + \lambda\Omega)^{-1}B^T$ are very distinct. While the eigenvalues of the latter decrease smoothly from 1 to 0, those of the former are all equal to 1 or 0.

Result: If B is an $n\times d$ matrix with full rank, the matrix $B(B^TB)^{-1}B^T$ has exactly d nonzero eigenvalues, and these are all equal to 1.

Proof: Idempotent matrices have eigenvalues of only 0 and 1, and $B(B^TB)^{-1}B^T$ is idempotent. Moreover, the trace of a matrix is equal to the sum of its eigenvalues, and

$$tr\left(B(B^TB)^{-1}B^T\right) = tr\left(B^TB(B^TB)^{-1}\right) = tr\left(I_d\right) = d.$$

Therefore $B(B^TB)^{-1}B^T$ has exactly d nonzero eigenvalues, and these are 1.

Selection of λ:

Note that $\hat{m}_n^{pspl}$ is a linear estimator — that is, it can be written as

$$\hat{m}_n^{pspl}(x) = \sum_{i=1}^n W_{ni}(x) Y_i \quad \text{for some weights } W_{n1}(x), \dots, W_{nn}(x) \text{ for each } x.$$

So we can choose λ to minimize (See Lec 04 slides)

$$CV_n(h) = \frac{1}{n} \sum_{i=1}^n \left[\frac{Y_i - \hat{m}_{n,-i}^{pspl}(X_i)}{1 - W_{ni}(X_i)} \right]^2 .$$

Helper results:

Result: Idempotent matrices have eigenvalues of 0 and 1.

Prove: Let A be idempotent and let $A\underset{\sim}{x} = \lambda \underset{\sim}{x}$ for some λ and some $\underset{\sim}{x}$.

Then $AA\underset{\sim}{x} = \lambda A \underset{\sim}{x} \implies A\underset{\sim}{x} = \lambda^2 \underset{\sim}{x} \implies \lambda \underset{\sim}{x} = \lambda^2 \underset{\sim}{x} \implies \lambda \in \{0, 1\}$.

Result: The trace of a matrix is equal to the sum of its eigen values.

Let A have eigendecomposition $A = U \Lambda U^T$ with $UU^T = U^T U = I$.

Then $tr(A) = tr(U \Lambda U^T) = tr(U^T U \Lambda) = tr(\Lambda)$.

Note that the above results involve <u>square</u> matrices.

NONPARAMETRIC REGRESSION WITH TREND FILTERING

For trend filtering at first assume $X_i = i/n$, $i=1,\dots,n$, so that $X_1,\dots,X_n$ are equally spaced. We can relax this assumption later.

For $Y_i = m(X_i) + \varepsilon_i$, $i=1,\dots,n$, we wish to estimate

$$\underset{\sim}{\mu} = \left(m(X_1),\dots, m(X_n)\right)^T,$$

which is the vector containing the values of m at the design points. The trend filtering estimate of $\underset{\sim}{\mu}$ of order k is given by

$$\hat{\underset{\sim}{\mu}} = \underset{\underset{\sim}{u} \in \mathbb{R}^n}{\operatorname{argmin}} \left\| \underset{\sim}{Y} - \underset{\sim}{u} \right\|_2^2 + \lambda \left\| D^{(k+1)} \underset{\sim}{u} \right\|_1 ,$$

where k is a positive integer and where $D^{(k+1)}$ is constructed with

$$D^{(1)} = \underbrace{\begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix}}_{(n-1)\times n}$$

and the recursion

$$D^{(k+1)} = \underset{(n-k-1)\times(n-k)}{D^{(1)}} \quad \underset{(n-k)\times n}{D^{(k)}} .$$

Note the change in dimension.

For example, we have

$$D^{(2)} = \underset{(n-2)\times(n-1)}{D^{(1)}} \; \underset{(n-1)\times n}{D^{(1)}}$$

$$= \underbrace{\begin{bmatrix} -1 & 1 & 0 & & \\ 0 & -1 & 1 & & \\ & & & & \\ & & & -1 & 1 \end{bmatrix}}_{(n-2)\times(n-1)} \begin{bmatrix} -1 & 1 & & & \\ 0 & -1 & 1 & & \\ & & & & \\ & & & -1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -2 & 1 & & & \\ 0 & 1 & -2 & 1 & & \\ & & & \ddots & & \\ & & & 1 & -2 & 1 \end{bmatrix}$$

and

$$D^{(3)} = \underbrace{\begin{bmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & & & \\ & & & -1 & 1 \end{bmatrix}}_{(n-2-1)\times(n-2)} \underbrace{\begin{bmatrix} 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & & & & \\ & & & 1 & -2 & 1 \end{bmatrix}}_{(n-2)\times n}$$

$$= \begin{bmatrix} -1 & 3 & -3 & 1 & & & \\ & -1 & 3 & -3 & 1 & & \\ & & & & \ddots & & \\ & & & -1 & 3 & -3 & 1 \end{bmatrix}.$$

Note that for $x \in \mathbb{R}^n$, $\|x\|_1 = \sum_{i=1}^{n} |x_i|$, so for $k=1$, the trend filtering estimator is

$$\hat{\underset{\sim}{\mu}} = \underset{\underset{\sim}{u} \in \mathbb{R}^n}{\operatorname{argmin}} \|y - \underset{\sim}{u}\|_2^2 + \lambda \sum_{j=1}^{n-1} |u_{j+1} - u_j| .$$

So the penalty $\lambda \|D^{(1)} \underset{\sim}{u}\|_1$ penalizes the number of change points in $\hat{\underset{\sim}{\mu}}$.
(discontinuities)

Exercise: Interpret the penalties $\lambda \| D^{(2)} \underset{\sim}{\mu} \|_1$, $\lambda \| D^{(3)} \underset{\sim}{\mu} \|_2$, $\lambda \| D^{(4)} \underset{\sim}{\mu} \|_2$.

Computation of Trend filtering estimator:

The following minimization is known as the Generalized Lasso problem:

$$\underset{\sim}{\hat{\beta}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \; \| \underset{n\times 1}{\underset{\sim}{Y}} - \underset{n\times p}{\mathbb{X}} \underset{p\times 1}{\underset{\sim}{\beta}} \|_2^2 + \lambda \| \underset{m\times p}{D} \underset{p\times 1}{\underset{\sim}{\beta}} \|_1$$

To compute the trend filtering estimator $\underset{\sim}{\hat{\mu}}$, solve above problem with $\mathbb{X} = I_n$ and a special choice of D.

Use genlasso package.

Exercise: Fit this estimator on some data. Use $k = 1, 2, 3, 4$.