Smoothing splines

Patrick Breheny

October 4

Introduction

- We are discussing ways to estimate the regression function f, where

$$\mathrm{E}(y|x) = f(x)$$

- One approach is of course to assume that f has a certain shape, such as linear or quadratic, that can be estimated parametrically
- A better – though still parametric – approach is to use splines, wherein the basis functions act locally, yet produce a smooth $\hat{f}$

Problems with knots

- Fixed-df splines are very useful tools, but they do have one shortcoming: the placement of knots
- Choices regarding the number of knots and where they are located are not particularly easy to make in a systematic and data-driven manner
- Furthermore, assuming that you place knots at quantiles or equally spaced intervals, models will not be nested inside each other, which complicates hypothesis testing

Controlling smoothness with penalization

- We can avoid the knot selection problem altogether by using penalization to formulate the problem in a nonparametric way
- Here, we directly solve for the function f that minimizes the following objective function, a penalized version of the least squares objective:

$$\sum_{i=1}^{n}\{y_i - f(x_i)\}^2 + \lambda \int \{f''(u)\}^2 du$$

- The first term captures the fit to the data, while the second penalizes curvature – note that for a line, $f''(u) = 0$ for all u

Connection with splines

- Here, λ is the smoothing parameter, and it controls the tradeoff between the two terms:
    - $\lambda = 0$ imposes no restrictions and f will therefore interpolate the data
    - $\lambda = \infty$ renders curvature impossible, thereby returning us to ordinary linear regression
- It may sound impossible to solve for such an f over all possible functions, but the solution turns out to be surprisingly simple: f must be a natural cubic spline

Terminology

- First, some terminology:
  - The parametric splines with fixed degrees of freedom that we have talked about so far are called *regression splines*
  - A spline that passes through the points $\{x_i, y_i\}$ is called an *interpolating spline*, and is said to interpolate the points $\{x_i, y_i\}$
  - A spline that describes and smooths noisy data by passing close to $\{x_i, y_i\}$ without the requirement of passing through them is called a *smoothing spline*

Natural cubic splines are the smoothest interpolators

Theorem: Out of all twice-differentiable functions passing through the points $\{x_i, y_i\}$, the one that minimizes

$$\lambda \int \{f''(u)\}^2 du$$

is a natural cubic spline with knots at every unique value of $\{x_i\}$

Natural cubic splines solve the nonparametric formulation

Theorem: Out of all twice-differentiable functions, the one that minimizes

$$\sum_{i=1}^{n}\{y_i - f(x_i)\}^2 + \lambda \int \{f''(u)\}^2 du$$

is a natural cubic spline with knots at every unique value of $\{x_i\}$

Design matrix

Let $\{N_j\}_{j=1}^n$ denote the collection of natural cubic spline basis functions and $\mathbf{N}$ denote the $n \times n$ design matrix consisting of the basis functions evaluated at the observed values:

- $N_{ij} = N_j(x_i)$
- $f(x) = \sum_{j=1}^n N_j(x)\beta_j$
- $f(\mathbf{x}) = \mathbf{N}\boldsymbol{\beta}$

Solution

- The penalized objective function is therefore

$$(\mathbf{y}-\mathbf{N}\boldsymbol{\beta})^T(\mathbf{y}-\mathbf{N}\boldsymbol{\beta})+\lambda\boldsymbol{\beta}^T\boldsymbol{\Omega}\boldsymbol{\beta},$$

where $\boldsymbol{\Omega}_{jk}=\int N_j''(t)N_k''(t)dt$

- The solution is therefore

$$\widehat{\boldsymbol{\beta}}=(\mathbf{N}'\mathbf{N}+\lambda\boldsymbol{\Omega})^{-1}\mathbf{N}'\mathbf{y}$$

Smoothing splines are linear smoothers

- Note that

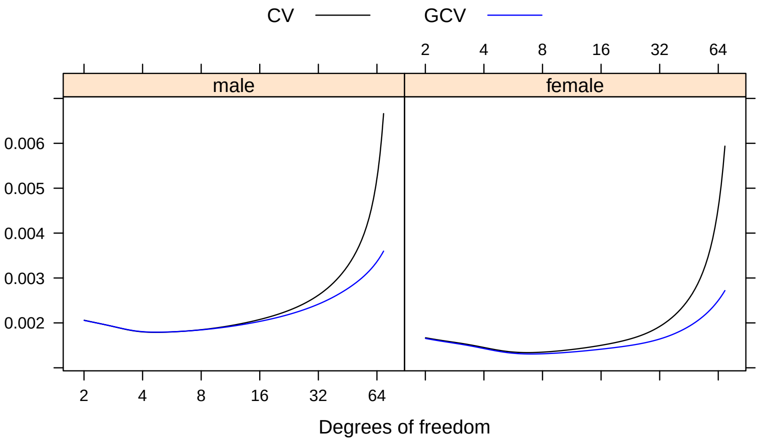
$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{N}(\mathbf{N}'\mathbf{N} + \lambda\boldsymbol{\Omega})^{-1}\mathbf{N}'\mathbf{y} \\ &= \mathbf{S}_\lambda \mathbf{y};\end{aligned}$$

  in other words, smoothing spline estimates are linear (nonparametric regression estimates with this property are said to be *linear smoothers*)
- As with ridge regression, this property provides us with a convenient way to calculate (or approximate) the leave-one-out cross-validation score as well as define the degrees of freedom of the estimate:

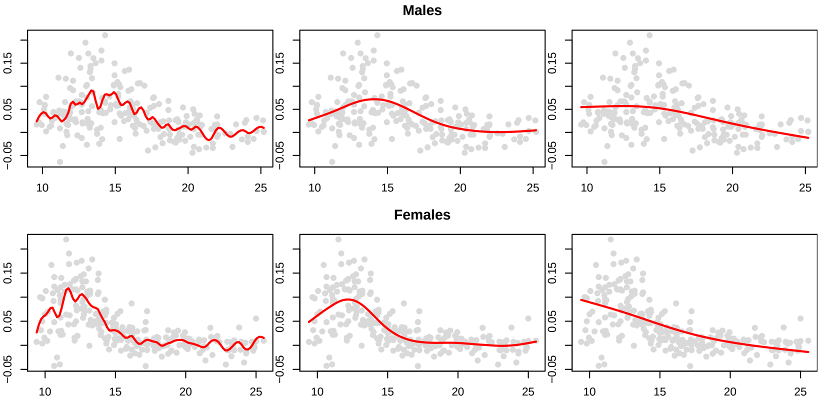
$$\mathrm{GCV} = \frac{1}{n}\sum_i \left(\frac{y_i - \hat{y}_i}{1 - \mathrm{tr}(\mathbf{S}_\lambda)/n}\right)^2$$

$$\mathrm{df}_\lambda = \mathrm{tr}(\mathbf{S}_\lambda)$$

CV, GCV for BMD example



Undersmoothing and oversmoothing of BMD data



Sampling distribution for smoothing splines

- The fact that smoothing splines are linear estimators greatly simplifies inference as well
- **Theorem:** Suppose that $y_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$; then

$$\hat{f}(\mathbf{x}) \sim N\left(\bar{f}(\mathbf{x}), \sigma^2 \mathbf{S}_\lambda \mathbf{S}_\lambda\right),$$

  where $\bar{f}(\mathbf{x}) = \mathbf{S}_\lambda f(\mathbf{x})$, the projection of $f(\mathbf{x})$ onto the space spanned by the natural cubic spline basis given the constraint on its integrated squared second derivative implied by λ
- In practice, we typically assume that $f(\mathbf{x}) - \bar{f}(\mathbf{x})$ is small, and use the above relationship to construct confidence intervals for $f(\mathbf{x})$ despite the fact that technically, they are intervals for $\bar{f}(\mathbf{x})$

$\mathbf{S}_\lambda$ versus $\mathbf{H}$

- Note that the smoothing matrix $\mathbf{S}_\lambda$ is quite similar to the projection matrix $\mathbf{H}$ from linear regression
- In particular, both $\mathbf{S}_\lambda$ and $\mathbf{H}$ are symmetric and positive semidefinite
- However, $\mathbf{H}$ is idempotent (*i.e.*, $\mathbf{HH} = \mathbf{H}$), whereas $\mathbf{S}_\lambda \mathbf{S}_\lambda$ is smaller than $\mathbf{S}_\lambda$ (in the sense that $\mathbf{S}_\lambda - \mathbf{S}_\lambda \mathbf{S}_\lambda$ is positive semidefinite), because $\mathbf{S}_\lambda$ introduces shrinkage, biasing estimates towards zero in order to reduce variance

Estimation of σ^2

- **Theorem:** For any linear smoother,

$$\mathrm{E}\sum_i (y_i - \hat{y}_i)^2 = \sigma^2 \mathrm{tr}\left((\mathbf{I} - \mathbf{S}_\lambda)^T(\mathbf{I} - \mathbf{S}_\lambda)\right) + \mathbf{b}^T\mathbf{b},$$

  where $\mathbf{b} = f(\mathbf{x}) - \bar{f}(\mathbf{x})$
- Thus, assuming that the bias term is small, the following is a nearly unbiased estimator for σ^2:

$$\hat{\sigma}^2 = \frac{\sum_i (y_i - \hat{y}_i)^2}{n - p^*},$$

  where $p^* = 2\mathrm{tr}(\mathbf{S}_\lambda) - \mathrm{tr}(\mathbf{S}_\lambda\mathbf{S}_\lambda)$
- The quantity p^* is known as the *equivalent number of parameters*, by analogy with linear regression, and differs slightly from the equivalent degrees of freedom

Pointwise confidence bands