Calculation of the smoothing spline with weighted roughness measure

Carl de Boor

Introduction

The (cubic) smoothing spline, of Schoenberg [S64] and Reinsch [R67], [R71], has become the most commonly used spline, particularly after the introduction of generalized cross validation by Craven and Wahba [CW79] for an automatic choice of the smoothing parameter. It is the purpose of this note to derive the computational details, in terms of B-splines, for the construction of the *weighted* smoothing spline, in hopes of promoting its use.

The λ-weighted smoothing spline is the unique minimizer of

$$\rho \sum_j w_j (y_j - f(x_j))^2 + \int_a^b \lambda(t)(D^m f)(t)^2 \, \mathrm{d}t \tag{1}$$

over all f in

$$X := L_2^{(m)}[a \,..\, b],$$

given the data sites $x_1 < \cdots < x_N$ in $[a \,..\, b]$, the data $y = (y_j)$, the weight vector $w = (w_j)$ of positive weights (usually equal to 1), the smoothing parameter $\rho \in [0+ \,..\, \infty-]$, the natural number m, and the nonnegative integrable function λ.

As $\rho \to \infty$, this weighted smoothing spline converges to the (natural) λ-weighted *interpolating* spline, and the latter has been around for some time, at least since Cinquin's thesis [C81], and K. Salkauskas' independent paper [Sa84] (see also Foley [Fo86]), albeit only with a piecewise constant weight with breaks at the data sites and only for $m = 2$. In fact, more general weighted splines already appear in [KW71], but only for smooth λ since the development there is via L-splines. Weighted splines with the weight the reciprocal of a piecewise polynomial function q were introduced by Kulkarni and Laurent [KL91] (and given the name *Q-splines*). These share with the weighted splines for piecewise constant weight the happy property that they are piecewise polynomial, but are smoother if the weight function is at least continuous. They also occur, independently, in [BSa92]. The approximation of more general weights by piecewise constant weights is considered in [SaB92], [BSa93].

The above list of papers concerning the interpolating weighted spline is far from exhaustive. But I could find only one paper discussing a weighted *smoothing* spline (namely [KL91] which details the case $m = 2$ and $\lambda = 1/q$ with q a continuous broken line) and could find no programs for the construction of a weighted smoothing spline. On the other hand, the few people familiar with the weighted spline will not be surprised to learn that it takes very little effort to modify an existing program for the construction of the standard smoothing spline to also produce a weighted smoothing spline, as long as the weight λ is piecewise constant with breaks only at the data sites (or, more generally, the reciprocal of a piecewise polynomial with breaks only at the data sites). Even if breaks are also permitted to occur at places other than the data sites, the needed adjustments are quite simple. Carrying out such a modification seems eminently worthwhile since it makes available considerably more flexibility in the shaping of the smoothing spline.

Derivation

It seems simplest to me (and to some others, see, e.g., [A92] and the references there) to view the minimization of (1) as a special case of best approximation in an inner product space. Since this is not the standard approach, I take the time to give the details, making, of course, no claim that the results are new.

Use the linear maps

$$\begin{aligned} \alpha : X \to Y \ &:= \mathrm{I\!R}^N : f \mapsto f|_x := \big(f(x_j)\big)_{j=1}^N, \\ \beta : X \to Z \ &:= L_2[a \,..\, b] : f \mapsto D^m f, \end{aligned}$$

to embed X in the Hilbert space

$$H := Y \times Z$$

with natural inner product

$$\langle (f,g),(h,k)\rangle := \rho\langle f,h\rangle_Y + \langle g,k\rangle_Z$$

with

$$\langle f,g\rangle_Y := \sum_j w_j f_j g_j, \quad f,g \in \mathrm{I\!R}^N,$$

$$\langle f,g\rangle_Z = \int_a^b \lambda f g, \quad f,g \in L_2[a \,..\, b].$$

Assuming as I do that $0 < \rho < \infty$, the only issue here is whether

$$X \to H : f \mapsto (\alpha(f), \beta(f))$$

is an embedding and whether, with this embedding, X becomes a closed subspace of H. For the former, it is necessary and sufficient that

$$\ker\alpha \cap \ker\beta = \{0\},$$

and, since

$$\ker\alpha = \{f \in X : f|_x = 0\}, \quad \ker\beta = \Pi_{<m}$$

(the space of polynomials of degree $< m$), this will be so iff $N \geq m$, an assumption I make from now on. As to the latter, it is, in essence, the claim that D, hence D^m, is a closed linear map. Explicitly, I take for granted the standard representation theorem

$$f = T_{a,m} f + R\beta(f), \qquad \forall f \in X, \tag{2}$$

with $T_{a,m} f$ the Taylor polynomial of order m for f at a and with

$$R : Z \to X : g \mapsto \int_a^b (\cdot - s)_+^{m-1} g(s)\,\mathrm{d}s/(m-1)!.$$

This identifies X as the sum $\Pi_{<m} + R(Z)$ of a finite-dimensional linear subspace (which therefore is closed) and the subspace $R(Z)$ which is closed, hence X itself is closed.

Thus, the smoothing spline f_ρ is the unique best approximation from $X \subset H$ to the element $(y, 0) \in H$, hence is characterized by the fact that the error, $(y, 0) - (\alpha(f_\rho), \beta(f_\rho))$, is perpendicular to $X \subset H$, i.e.,

$$\rho\langle y - \alpha(f_\rho), \alpha(f)\rangle_Y + \langle -\beta(f_\rho), \beta(f)\rangle_Z = 0 \qquad \forall f \in X. \tag{3}$$

Since $\ker\beta = \Pi_{<m}$, (3) implies that

$$\langle y - \alpha(f_\rho), \alpha(f)\rangle_Y = 0 \qquad \forall f \in \Pi_{<m} \tag{4}$$

and, with (2) and this, (3) implies that

$$\rho\langle y - \alpha(f_\rho), \alpha(Rg)\rangle_Y = \langle \beta(f_\rho), g\rangle_Z \qquad \forall g \in Z. \tag{5}$$

Conversely, (4)–(5) imply (3).

Since the left side of (5) is a continuous linear functional as a function of g, it is expressible in the form $\langle z, g\rangle_Z$ for some $z \in Z$. Explicitly, with $h := y - \alpha(f_\rho)$,

$$\begin{aligned}
\langle y - \alpha(f_\rho), \alpha(Rg)\rangle_Y &= \sum_j w_j h_j \int_a^b (x_j - s)_+^{m-1} g(s)\,\mathrm{d}s/(m-1)! \\
&= \int_a^b \big(\sum_j w_j h_j\, (x_j - s)_+^{m-1}/(m-1)!\big) g(s)\,\mathrm{d}s \\
&= \langle \frac{1}{\lambda} \sum_j w_j h_j (x_j - \cdot)_+^{m-1}/(m-1)!, g\rangle_Z.
\end{aligned}$$

It follows that (5) is equivalent to

$$\rho \sum_j w_j (y_j - f_\rho(x_j))(x_j - \cdot)_+^{m-1}/(m-1)! \;=\; \lambda D^m f_\rho. \tag{6}$$

This shows that $\lambda D^m f_\rho$ is piecewise polynomial of order m with break sequence x, hence f_ρ itself is, by definition, a (λ-)weighted spline of order $2m$. Further, $\lambda D^m f$ vanishes to the right of x_N. However, by (4),

$$\sum_j w_j(y_j - f_\rho(x_j))(x_j - \cdot)^{m-1}/(m-1)! \;=\; 0,$$

hence, with $(x_j - t)_+^{m-1} = (x_j - t)^{m-1} - (-1)^{m-1}(t - x_j)_+^{m-1}$, also

$$\rho(-1)^m \sum_j w_j(y_j - f_\rho(x_j))(\cdot - x_j)_+^{m-1}/(m-1)! \;=\; \lambda D^m f_\rho, \tag{7}$$

showing that $\lambda D^m f_\rho$ also vanishes to the left of x_1. Consequently, we may write

$$\lambda D^m f_\rho =: \sum_k B_{k,m,x} c_k, \tag{8}$$

with $B_{k,m,x}$ the normalized B-spline with knots $x_k, \ldots, x_{k+m}$, i.e.,

$$\begin{aligned} B_{k,m,x}(t) &= (x_{k+m} - x_k)[x_k, \ldots, x_{k+m}](\cdot - t)_+^{m-1} \\ &=: \sum_j (x_j - t)_+^{m-1} c_{j,k}, \end{aligned}$$

hence

$$c_{j,k} = \begin{cases} (x_{k+m} - x_k)/\prod\{(x_j - x_i) : i \in \{k, \ldots, k+m\}\backslash\{j\}\}, & j = k, \ldots, k+m; \\ 0, & \text{otherwise.} \end{cases}$$

Following the standard terminology in the special case $\lambda = 1$, call any f_ρ satisfying (8) a *natural (λ-)weighted spline of order $2m$ with break sequence x.*

Now, on expressing $\lambda D^m f_\rho$ in (6) in this way and comparing coefficients of $(x_j - \cdot)_+^{m-1}$, we obtain

$$\rho W(y - \alpha(f_\rho)) = Cc, \tag{9}$$

with

$$W := \operatorname{diag}(w), \qquad C := (m-1)!(c_{j,k} : j = 1, \ldots, N; k = 1, \ldots, N-m),$$

and c the B-spline coefficient sequence for $\lambda D^m f_\rho$, as defined in (8). Next, note that, for any $f \in X$,

$$(C^t \alpha(f))_j \;=\; (m-1)!\,(x_{j+m} - x_j)[x_j, \ldots, x_{j+m}]f = \int_a^b B_{j,m,x} D^m f, \tag{10}$$

using the fact that the (appropriately normalized) B-spline is the Peano kernel for the divided difference. Therefore, any $\alpha(f_\rho)$ satisfying (9) automatically satisfies (4), since, for any such $\alpha(f_\rho)$ and any $f \in X$,

$$\langle y - \alpha(f_\rho), \alpha(f)\rangle_Y \;=\; \alpha(f)^t W(y - \alpha(f_\rho)) \;=\; \alpha(f)^t Cc/\rho \;=\; (C^t\alpha(f))^t c/\rho,$$

while $C^t\alpha(f) = 0$ for $f \in \Pi_{<m}$ by (10). Since (9) with (8) implies (6), hence (5), it follows that, with (8), (9) is equivalent to (3). We therefore now concentrate on (9).

For that, from (10) with (8),

$$C^t\alpha(f_\rho) = Ac, \tag{11}$$

with

$$A := (\int_a^b \frac{1}{\lambda} B_{j,m,x} B_{k,m,x} : j,k = 1,\dots,N-m).$$

Substitution of (9), in the form

$$\alpha(f_\rho) \;=\; y - \; W^{-1}C(c/\rho), \tag{12}$$

into (11) gives the equation

$$C^t y = (C^t W^{-1} C + \rho\, A)(c/\rho) \tag{13}$$

which may be solved stably for $u := c/\rho$, for given data y (since its coefficient matrix is symmetric positive definite). From this, we obtain the smoothed values $\alpha(f_\rho)$ directly from (12). To obtain f_ρ, integrate the resulting $D^m f_\rho = (1/\lambda)\sum_j B_{j,m,x} c_j$ m times, to obtain $f := D^{-m}(D^m f_\rho)$, which differs from f_ρ only by some $q \in \Pi_{<m}$. Determine this q as the unique $q \in \Pi_{<m}$ for which $\alpha(q + f) = \alpha(f_\rho)$, i.e., for which $\alpha(q) = \alpha(f_\rho) - \alpha(f)$, with the vector $\alpha(f_\rho)$ computed from (12).

It is only at this point, of m-fold integration, that the choice of the weight λ in the roughness measure begins to matter (other than an assumption that λ be measurable and essentially positive, to ensure that $\langle\ ,\ \rangle_Z$ is an inner product). For the special case $\lambda = 1$, one would use the standard formula, see, e.g., [pgs : p.150], to carry out the integration, obtaining, in that case, f_ρ as a natural spline of order $2m$ with simple interior knots $(x_i : i = 2,\dots,N-1)$. While the integration can be carried out in closed form for a somewhat larger class, I will, at this point, restrict attention to those λ for which f_ρ is still piecewise polynomial and, specifically, on the simplest of these, namely the piecewise constants with breaks only at the x_i, i.e.,

$$\lambda \in \Pi_{0,x}. \tag{14}$$

It would not be very hard to consider more general choices, namely a λ that is the reciprocal of a piecewise polynomial function, as in [KL91] and [BSa92].

The behavior of the error as a function of ρ

According to (13), as $\rho \to 0$, $u := c/\rho$ converges to $(C^t W^{-1} C)^{-1} C^t y$, hence $c = \rho u \to 0$, therefore f_{0+} is the unique polynomial $q \in \Pi_{<m}$ that minimizes $\|y - \alpha(q)\|$. At the other extreme, as $\rho \to \infty$, (13) approaches the equation $C^t y = Ac$, hence, with (11), $f_{\infty -}$ is the unique natural (λ)-weighted spline of order $2m$ with knots x that interpolates to the given data.

As a function of ρ, the error

$$E_\rho := \|y - \alpha(f_\rho)\|^2$$

decreases with increasing ρ, as can be seen as follows: For each $f \in X$,

$$\mathbb{R}_+ \to \mathbb{R} : \rho \mapsto \rho \|y - \alpha(f)\|^2 + \|\beta(f)\|^2$$

is a straight line, hence the function

$$F : \mathbb{R}_+ \to \mathbb{R} : \rho \mapsto \min_{f \in X} \left(\rho \|y - \alpha(f)\|^2 + \|\beta(f)\|^2 \right),$$

as the pointwise minimum of a collection of straight lines with nonnegative y-intercepts and nonnegative slopes, is continuous, nondecreasing, concave downward and is bounded (above) by its asymptote at infinity, the constant line of height $\|\beta(f_{\infty -})\|^2$, while the asymptote at the other extreme (i.e., the tangent at the origin) is the line through the origin with slope $\|y - \alpha(f_{0+})\|^2$. Since the straight line

$$\rho \mapsto \rho \|y - \alpha(f_\rho)\|^2 + \|\beta(f_\rho)\|^2$$

is the tangent to F at ρ, we have

$$DF(\rho) = E_\rho,$$

showing that E_ρ decreases with increasing ρ and that, correspondingly, $\|\beta(f_\rho)\|^2$ (the y-intercept of the tangent) increases with increasing ρ.

For this reason, Reinsch and others have proposed to choose the smoothing parameter ρ as small as possible subject to the constraint that E_ρ not exceed a given tolerance, *tol*. Further, Reinsch has pointed out that the function

$$G : \rho \mapsto 1/\|y - \alpha(f_\rho)\|$$

is concave upward and becomes ever more linear with growing ρ, hence Newton's method applied to the equation

$$1/E_\rho^{1/2} - 1/(tol)^{1/2} = 0 \tag{15}$$

for ρ and started at $\rho = 0$ is bound to converge, and to converge quite fast, particularly if the solution is 'large'. Further, since

$$E_\rho = u^t C^t W^{-1} C u$$

by (12), one gets $DE_\rho = 2u^t C^t W^{-1} C Du$, while, from (13), Du uniquely solves the equation

$$-Au = (C^t W^{-1} C + \rho A) Du.$$

In particular, for $\rho = 0$, this says that $DE_\rho = -2u^t Au$, with $u = (C^t W^{-1} C)^{-1} C^t y$ needed in any case for the calculation of $\alpha(f_\rho)$ via (12). This provides the slope needed for the starting step, at $\rho = 0$, of Newton's method applied to (15). For subsequent steps, I would avoid calculation of DE_ρ (which requires solution of a linear system) by using the secant method instead.

Another limiting case of interest concerns the confluence of some of the x_j. If the data y come from a smooth function and the relevant weights behave appropriately, then confluence of $r \le m$ neighboring points leads to the smoothing problem in which α also involves all the derivatives of order $< r$ at the multiple point and, correspondingly, f_ρ has only $2m-1-r$ continuous derivatives across that multiple point. Of course, the relevant formulæ for such an α can be derived directly in the above way, using divided differences with repeated nodes and, correspondingly, B-splines with repeated knots, in the standard way. In particular, there is some practical use for the *complete cubic smoothing spline* for which $\alpha(f) = (Df(x_1), f|_x, Df(x_N))$.

Numerical construction of the B-spline Gramian

There is one final hurdle to writing a program for the computation of f_ρ for general m, namely the construction of the matrix A of (weighted) inner products of B-splines. This is the second point at which the choice of λ becomes important. With our choice of

$$\lambda =: \sum_{j=1}^{N-1} \lambda_j \chi_{(x_j .. x_{j+1})} \in \Pi_{0,x}$$

instead of just $\lambda = 1$, the calculation of the entries of A is not at all complicated since, for small m, the integrals

$$A_{j,k} = \int_a^b B_{j,m,x} B_{k,m,x} / \lambda, \qquad j,k = 1, \ldots, N-m,$$

are most easily evaluated break interval by break interval anyway. To be sure, for $\lambda = 1$, there are stable recurrence relations for the integrals available in the literature, e.g., in [BLS76]. In that case, one might also make use of the formula

$$\begin{aligned}\int B_{i,k}B_{j,k} =&(-1)^k \frac{(2k-1)!}{(k!)^2}(t_{i+k}-t_i)(t_{j+k}-t_j)\\ &\times [t_i,\ldots,t_{i+k}]_x[t_j,\ldots,t_{j+h}]_y(x-y)_+^{2k-1},\end{aligned}$$

which, in slightly different form, appears already for that purpose in [JS68].

For $m = 1, 2$ (and, perhaps, even $m = 3$) it is easy to work out the matrix entries directly.

case $m = 1$**:** In this case, $B_{j,m,x} = \chi_{[x_j\,..\,x_{j+1})}$. Correspondingly, $Df_\rho = c_j/\lambda_j$ on $(x_j\,..\,x_{j+1})$. Further, A is the diagonal matrix with diagonal entries $\Delta x_j/\lambda_j$, $j = 1,\ldots,N-1$.

case $m = 2$**:** In this case, $B_{j,m,x}$ is the piecewise linear function that is zero at all its breaks x, except at x_{j+1}, where it is 1. Correspondingly,

$$D^2 f_\rho(t) = \begin{cases} c_j/\lambda_j & \text{for } t = x_j^+;\\ c_{j-1}/\lambda_{j-1} & \text{for } t = x_j^-.\end{cases}$$

Hence, with $\alpha(f_\rho)$ computed from (12), construction of the local cubic pieces is immediate once c (or c/ρ) is obtained from (13).

Further, with $t = x_j + s\Delta x_j$,

$$\int_{x_j}^{x_{j+1}} B_{j,2}(t)^2\,\mathrm{d}t = \Delta x_j \int_0^1 s^2\,\mathrm{d}s = \Delta x_j/3,$$

while

$$\begin{aligned}\int_{x_j}^{x_{j+1}} B_{j-1,2}(t)B_{j,2}(t)\,\mathrm{d}t \;&= \int_{x_j}^{x_{j+1}} B_{j,2}(t)\,\mathrm{d}t - \int_{x_j}^{x_{j+1}} B_{j,2}(t)^2\,\mathrm{d}t\\ &= \Delta x_j/2 - \Delta x_j/3 = \Delta x_j/6.\end{aligned}$$

Consequently, A is the tridiagonal matrix with general row

$$(\frac{\Delta x_j}{\lambda_j}, 2(\frac{\Delta x_j}{\lambda_j} + \frac{\Delta x_{j+1}}{\lambda_{j+1}}), \frac{\Delta x_{j+1}}{\lambda_{j+1}})/6, \qquad j = 1,\ldots,N-2.$$

Note that, for $\lambda = 1$, the entries in such a row add up to $(x_{j+2} - x_j)/2 = \int B_{j,2,x}$, exactly as they should.

case $m > 2$**:** Already for $m = 3$, the explicit expressions for the integrals $\int_{x_r}^{x_{r+1}} B_i B_j/\lambda$ become complex enough that it seems easier simply to employ Gauss quadrature, mindful of the fact that the integrand is a polynomial of order $2m$, hence the m-point Gauss-Legendre quadrature rule

$$\int_{x_r}^{x_{r+1}} f \approx \Delta x_r \sum_{j=1}^{m} w_j f(x_r + \Delta x_r \tau_j)$$

will give the exact integral (up to rounding error). Formally,

$$A = B^t V B,$$

with

$$V := \operatorname{diag}((\Delta x_r/\lambda_r) w : r = 1, \ldots, N-1),$$

and B the B-spline collocation matrix

$$B := (B_{j,m,x}(t_i) : i = 1, \ldots, m(N-1); j = 1, \ldots, N-m),$$

where

$$t := (x_r + \Delta x_r \tau : r = 1, \ldots, N-1).$$

However, one would want to take full advantage of the fact that all matrices here are almost block diagonal (in the sense introduced in [BW80]), and how that is done will depend crucially on the programming language used.

References

[A92] Attéia, M. (1992): Hilbertian Kernels and Spline Functions. Elsevier Science Publishers, Amsterdam

[pgs] Boor, C. de (1978): A Practical Guide to Splines. Springer Verlag, New York

[BLS76] Boor, C. de, T. Lyche, and L. L. Schumaker (1976): On calculating with B-splines II. Integration. In: L. Collatz, G. Meinardus and H. Werner, eds, Numerische Methoden der Approximationstheorie Vol. 3, ISNM 30, 123–146. Birkhäuser Verlag, Basel

[BW80] Boor, C. de and Richard Weiss (1980): SOLVEBLOK : A package for solving almost block diagonal linear systems. ACM Trans. Math. Software **6**, 80–87

[BSa92] Bos, L. and K. Salkauskas (1992): Weighted splines based on piecewise polynomial weight functions. In: H. Hagen, ed, Curve and Surface Design, 87–98. SIAM Publications, SIAM, Philadelphia PA

[BSa93] Bos, L. and K. Salkauskas (1993): Limits of weighted splines based on piecewise constant weight functions. Rocky Mountain J. Math. **23(2)**, 483–493

[C81] Cinquin, Ph. (1981): Splines unidimensionelles sous tension et bidimensionelles paramétrées: deux applications médicales. dissertation, Thèse, Université de Saint-Etienne, 28 octobre

[CW79] Craven, Peter and Grace Wahba (1979): Smoothing noisy data with spline functions estimating the correct degree of smoothing by the method of generalized cross validation. Numer. Math. **31**, 377–403

[Fa82] Farin, Gerald E. (1982): Visually C^2 cubic splines. Computer-Aided Design **14(3)**, 137–139

[Fo86] Foley, T. A. (1986): Local control of interval tension using weighted splines. Comput. Aided Geom. Design **3**, 281–294

[JS68] Jerome, J. and L. L. Schumaker (1968): A note on obtaining natural spline functions by the abstract approach of Atteia and Laurent. SIAM J. Numer. Anal. **5**, 657–663

[KW71] Kimeldorf, G. and G. Wahba (1971): Some results on Tchebycheffian splines functions. J. Math. Anal. Appl. **33**, 82–95

[KL91] Kulkarni, R. and P. J. Laurent (1991): Q-splines. Numer. Algorithms **1**, 45–73

[R67] Reinsch, C. H. (1967): Smoothing by spline functions. Numer. Math. **10**, 177–183

[R71] Reinsch, Christian H. (1971): Smoothing by spline functions. II. Numer. Math. **16**, 451–454

[Sa84] Salkauskas, K. (1984): C^1 splines for interpolation of rapidly changing data. Rocky Mountain J. Math. **14**, 239–250

[SaB92] Salkauskas, K. and L. Bos (1992): Weighted splines as optimal interpolants. Rocky Mountain J. Math. **22(2)**, 705–717

[S64] Schoenberg, I. J. (1964): Spline functions and the problem of graduation. Proc. Amer. Math. Soc. **52**, 947–950