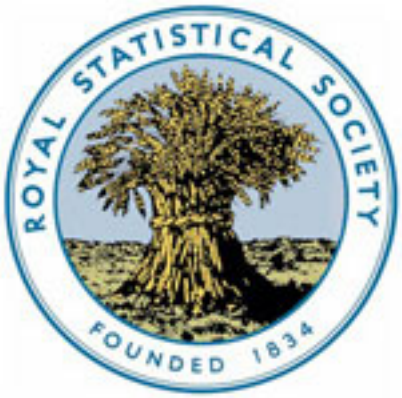

J. R. Statist. Soc. B (1996)
58, *No.* 2, *pp.* 379–396

Pseudosplines

By TREVOR HASTIE†

Stanford University, USA

[Received February 1994. Revised March 1995]

SUMMARY

We describe a method for constructing a family of low rank, penalized scatterplot smoothers. These *pseudosplines* have shrinking behaviour that is similar to that of smoothing splines. They require two ingredients: a basis and a penalty sequence. The smoother is then computed by a generalized ridge regression. The family can be used to approximate existing high rank smoothers in terms of their dominant eigenvectors. Our motivating example uses linear combinations of orthogonal polynomials to approximate smoothing splines, where the linear combination and the penalty sequence depend on the particular instance of the smoother being approximated. As a leading application, we demonstrate the use of these pseudosplines in additive model computations. Additive models are typically fitted by an iterative smoothing algorithm, and any features other than the fit itself are difficult to compute. These include standard error curves, degrees of freedom, generalized cross-validation and influence diagnostics. By using a low rank pseudospline approximation for each of the smoothers involved, the entire additive fit can be approximated by a corresponding low rank approximation. This can be computed exactly and efficiently, and opens the door to a variety of computations that were not feasible before.

Keywords: CUBIC SMOOTHING SPLINES; EIGENDECOMPOSITION; PENALIZED LEAST SQUARES; RIDGE REGRESSION

1. INTRODUCTION

Let $\mathbf{x}$ and $\mathbf{y}$ denote a set of n observations. A scatterplot smoother of $\mathbf{y}$ against $\mathbf{x}$ is a function of the data: $s(x_0) = \mathcal{S}(x_0|\mathbf{x}, \mathbf{y})$, which at each x_0 summarizes the dependence of $\mathbf{y}$ on $\mathbf{x}$, usually in a flexible but smooth way. A smoother is *linear* if

$$\mathcal{S}(x_0|\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} s(i, x_0, \mathbf{x})y_i$$

for some weights $s(i, x_0, \mathbf{x})$ which do not depend on $\mathbf{y}$. Popular linear smoothers are smoothing splines, kernel smoothers and local regression. If we concentrate on the computation of the fit only at the points in $\mathbf{x}$, we can write a linear smoother as a linear map S: $\mathbb{R}^n \mapsto \mathbb{R}^n$ defined by $\hat{\mathbf{y}} = S\mathbf{y}$. S is commonly referred to as a smoother matrix (Buja *et al.*, 1989; Hastie and Tibshirani, 1990).

S is the smoothing analogue of the *hat* or projection matrix in regression. Although S typically has full rank (n), we shall see that most of its action is concentrated in a much lower dimensional subspace. Consequently we can approximate S by a lower

†*Address for correspondence*: Department of Statistics, Sequoia Hall, Stanford University, Stanford, CA 94305, USA.
E-mail: trevor@playfair.stanford.edu

 0035–9246/96/58379

dimensional operator. Although the techniques that we discuss are quite general, they are motivated by and focus on smoothing splines.

In this paper we describe a method for constructing a family of low rank, penalized scatterplot smoothers. These *pseudosplines* have shrinking behaviour that is similar to that of smoothing splines. They require two ingredients: a basis and a penalty sequence; the smoother is then computed by a generalized ridge regression. The family can be used to approximate existing high rank smoothers in terms of their dominant eigenvectors. Our leading example uses linear combinations of orthogonal polynomials to approximate smoothing splines, where the linear combination and the penalty sequence depend on the particular instance of the smoother being approximated, but to a negligible extent on the value of the smoothing parameter.

There are several reasons why such a representation is appealing.

(a) The family is simple and low dimensional, like polynomial regression. However, instead of selecting the degree in integral steps, the *ridge* parameter allows us access to a continuum of models. Besides being a compelling application of ridge regression, this simple model offers much insight into penalized smoothers (see Fig. 2 later).
(b) The family provides a good low rank approximation to an existing smoother S. Although the matrix S is not explicitly required to compute the fit, it is needed for *secondary* characteristics of the smoother, such as standard errors, degrees of freedom and diagnostics.
(c) Smoothers are often used in a compound way, such as in generalized additive models (Hastie and Tibshirani, 1990), projection pursuit regression (Friedman and Stuetzle, 1981; Roosen and Hastie, 1994) and recently in non-linear autoregression models (Chen and Tsay, 1993) and other time series applications (Green and Silverman, 1994). Simple approximations allow the fit to be computed directly without iteration. This is especially important in cases where iterative algorithms are inefficient or may fail; autoregressive and time series models with highly correlated predictors fall into this class. Again they also make available secondary characteristics which are even less accessible for these more complicated models.

2. SMOOTHING SPLINES

In this section we give a brief review of smoothing splines, which motivate our pseudosplines.

A cubic smoothing spline minimizes the penalized least squares criterion

$$\sum_{1}^{n} \{y_i - g(x_i)\}^2 + \lambda \int_{-\infty}^{+\infty} g''(z)^2 \, dz \tag{1}$$

over a suitable Sobolev space W_2 of functions (Silverman, 1985; Wahba, 1990). The solution $\hat{g}(x)$ is a natural cubic spline with knots at each distinct x_i, and for the moment we assume that all the x_i in the sample are unique (in Section 7 we show how to deal with ties). The smoothing parameter λ trades off smoothness of the curve with its closeness to the y-values. As $\lambda \to 0$, the solution approaches an interpolating spline, whereas, as $\lambda \to \infty$, the solution approaches the least squares line.

One can show that the cubic smoothing spline is a linear smoother and hence write down the smoother matrix for producing the fit at the sample points. Although the standard representation is in terms of the computationally attractive B-spline basis functions, for our purposes that given in Green and Yandell (1985) is more useful:

$$S = (I + \lambda K)^{-1}.$$

This representation has the n fitted values as *parameters*. The criterion (1) reduces to $\|\mathbf{y} - \mathbf{f}\|^2 + \lambda \mathbf{f}^{\mathrm{T}} K \mathbf{f}$, and the quadratic form in the penalty matrix K can be seen roughly to accumulate squared second differences.

Further insight is gained from the eigendecomposition of S or equivalently of K itself. Since S is symmetric, it has a decomposition

$$S = U D_\phi U^{\mathrm{T}} = \sum_{i=1}^{n} \phi_i \mathbf{u}_i \mathbf{u}_i^{\mathrm{T}} \tag{2}$$

where the columns $\mathbf{u}_i$ of U are orthonormal and D_ϕ is diagonal with elements $\phi_i \in (0, 1]$ and decreasing in i. This Demmler and Reinsch (1975) basis has intuitive appeal. The eigenvalue ϕ_i shows us how much damping is done to the function $\mathbf{u}_i$ when the smoother is applied, since $S\mathbf{u}_i = \phi_i \mathbf{u}_i$ (this also shows that the $\mathbf{u}_i$ themselves are natural splines). The columns of U are like the sequence of orthonormal polynomials defined on $\mathbf{x}$, in that the number of zero crossings appears to increase with the order. Demmler and Reinsch indeed showed that for $k \geqslant 3$ the number of sign changes in the kth eigenvector of a cubic smoothing spline is $k - 1$. This decomposition suggests analogies with the traditional smoothing methods for time series (see Rice and Rosenblatt (1983)). $U^{\mathrm{T}}\mathbf{y}$ expresses $\mathbf{y}$ in terms of the basis defined by the columns of U (similar to a Fourier transform). The ϕ_i play the role of a taper.

Fig. 1(a) shows the results of applying a cubic smoothing spline to some air pollution data (128 observations). Two fits are given: a *smoother* fit corresponding to a larger penalty λ and a *rougher* fit for a smaller penalty. A convenient way to calibrate the amount of smoothing is via the *effective degrees of freedom*, defined by $\mathrm{df}_\lambda = \mathrm{tr}(S_\lambda)$ (Hastie and Tibshirani, 1990). We have used 5df and 10df respectively. Fig. 1(b) gives the eigenvalues for these smoothers. We notice several things. The first two eigenvalues are 1, since the first two eigenvectors span the space of linear functions (null space of K) which the smoother passes unchanged. From then on the eigenvalues decline smoothly to 0, and by number 8 those for 5df are within 10% of 0. Fig. 1(c) gives the third–sixth eigenvectors and shows how much shrinking is done. Of course the rate at which the eigenvalues decrease will differ depending on the value df_λ; those for 10df decline far more slowly. In fact, a more natural decomposition uses the eigenvalues θ_j of K, with $\phi_j = 1/(1 + \lambda\theta_j)$ and θ_j independent of λ. It is also important to note that the eigenvectors of S do not depend on the particular value of the smoothing parameter. Speckman (1982) discussed this decomposition in more detail and showed its use in describing bias and variance for smoothing splines.

The contributions of higher order functions to the eigendecomposition decrease rapidly with the order. We could think of approximating the smoother by using a low rank approximation based on its eigendecomposition. Thus a rank k approximation would have the form

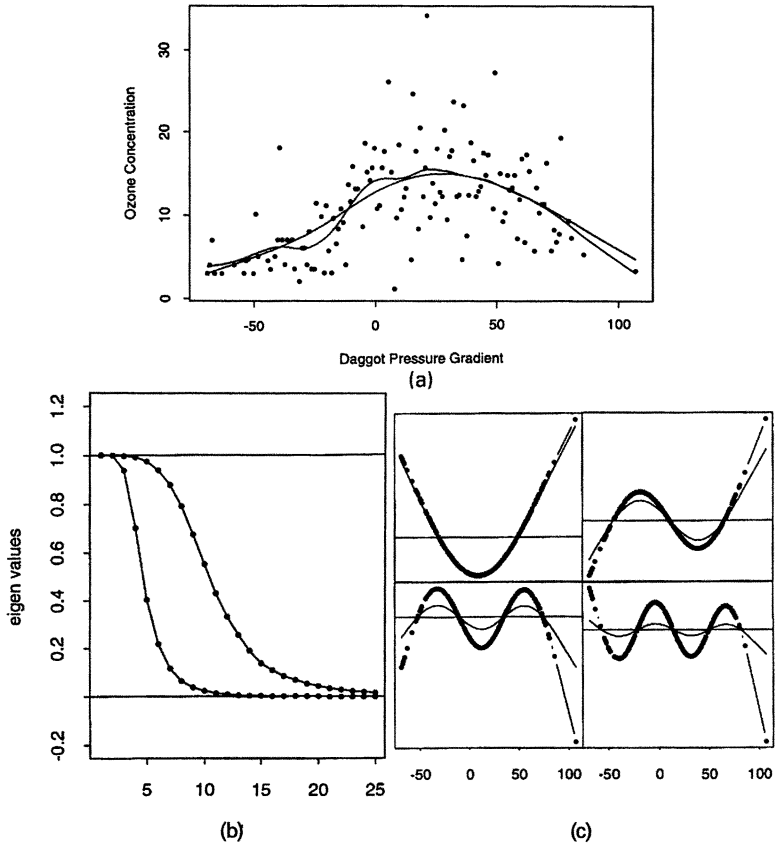

Fig. 1. (a) Smoothing spline fit of ozone concentration *versus* Daggot pressure gradient (the two fits correspond to different values of the smoothing parameter, chosen to achieve 5 and 10 effective degrees of freedom, defined by $df_\lambda = \mathrm{tr}(S_\lambda)$); (b) first 25 eigenvalues for the two smoothing spline matrices (the first two are exactly 1, and all are greater than or equal to 0); (c) third–sixth eigenvectors of the spline smoother matrices (in each case, $\mathbf{u}_j$ is plotted against x and as such is viewed as a function of x; the dots on the functions indicate the occurrence of data points; the damped functions represent the smoothed versions of these functions (using the 5df smoother))

$$S_k = UD_\phi^k U^{\mathrm{T}} \tag{3}$$

where D_ϕ^k is a truncated version of D_ϕ with diagonal elements from $k+1$ onwards set to 0. In fact, S_k is the best rank k approximation to S (Frobenius norm).

3. PSEUDOSPLINES

The smoothing spline suggests a way to parameterize a general class of smoothers. All we need are a sequence of orthonormal basis functions and a penalty sequence. For the analogy to be complete, the basis functions should be ordered in complexity. Let $\mathbf{p}(x)$ be a k-vector of such functions and θ_j, $j = 1, \ldots, k$, the penalties. Then the corresponding *pseudospline* with parameter λ minimizes

$$Q_\lambda(\beta, \mathbf{y}) = \|\mathbf{y} - P\beta\|^2 + \lambda\beta^T D_\theta \beta \tag{4}$$

where P is the matrix of evaluations of $\mathbf{p}$ at the data and $D_\theta = \text{diag}(\theta_1, \ldots, \theta_k)$. The solution has smoother matrix $\hat{S}_\lambda(P, \theta) = P(P^TP + \lambda D_\theta)^{-1}P^T$. If in addition the bases are orthonormal with respect to the observed $\mathbf{x}$ (sample measure), then $P^TP = I$ and our smoother simplifies to $\hat{S}_\lambda(P, \theta) = P(I + \lambda D_\theta)^{-1}P^T$. This has the form of the truncated smoothing spline (3) with $(I + \lambda D_\theta)^{-1}$ corresponding to the non-zero block of D_ϕ^k and P to the corresponding columns of U.

Fig. 2 illustrates the action of a pseudospline in terms of its eigenvalues.

How do we choose the bases and penalty sequences? Some obvious choices are orthogonal polynomials, cosinusoids (Rice and Rosenblatt, 1983) or Legendre or Chebyshev polynomials, where orthogonality is defined in terms of a continuous measure. Rice (1982) studied the rates of convergence of penalized polynomials using the last two systems.

All these candidates are naturally hierarchical—they have a complexity ordering (for this reason the popular B-spline bases are not natural candidates). Often it is natural for some of the basis functions to remain unpenalized (their θ_js are 0). For example, we may want the first two basis functions to span the space of linear functions (or possibly a higher order polynomial subspace), and $\theta_1 = \theta_2 = 0$; this is a direct analogy with the null penalty space of cubic smoothing splines. Some applications may call for specially tailored basis functions and null spaces. Whatever the choice, we end up with a parameterized family that gives us access to a spectrum of models ranging from the fit on the null space at one extreme to the unpenalized fit on the full basis set at the other.

Recently Donoho and Johnstone (1994) have applied non-linear shrinkage schemes to bases of orthonormal wavelets; these are more adaptive schemes than the framework discussed here, but similar in spirit.

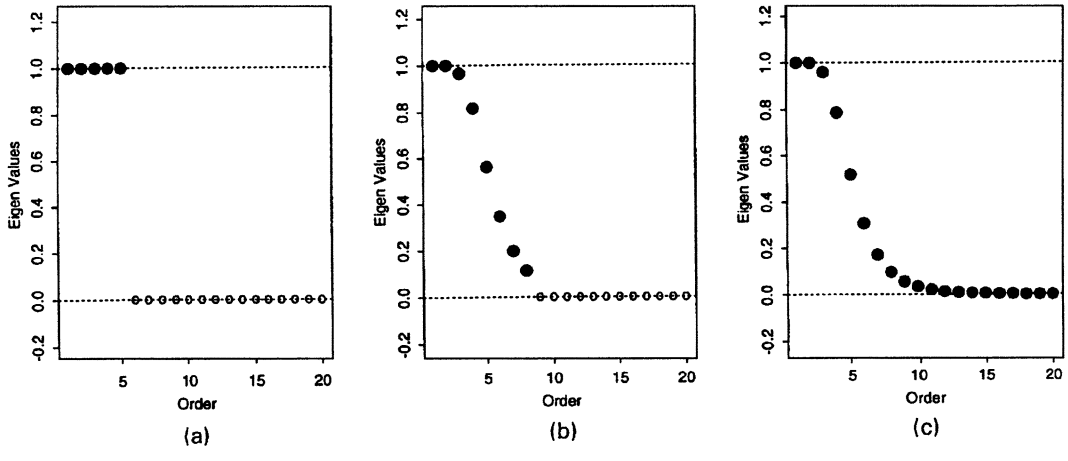

Fig. 2. Illustration of three different types of smoother based on a series of orthogonal basis functions (shown are the first 20 eigenvalues, and all smoothers have 5 effective degrees of freedom): (a) traditional series smoother, a projection using the first five basis functions; (b) pseudospline, here using eight basis functions and shrinking their effect down to 5 effective degrees of freedom; (c) smoothing spline—none of the eigenvalues are 0

For the remainder of this paper we shall focus on using these pseudosplines to approximate existing smoothers, in particular smoothing splines.

4. PSEUDO-EIGENDECOMPOSITION

To fix ideas suppose that we wish our pseudospline to approximate the action of the smoothing spline used in Fig. 1. Our approach is to estimate its truncated eigendecomposition (3). It does not matter which version (rough or smooth) we use, since the eigenvectors are the same, and the eigenvalues ψ_j give us θ_j up to a constant; hence we simply refer to the smoother as S.

We could simply compute S itself and truncate its eigendecomposition. This is expensive ($O(n^3)$), requires S explicitly and defeats the purpose of the approximation. Instead we supply a surrogate or *pseudobasis* P, which we use to define a *pseudo-eigendecomposition* of S:

$$\hat{S}(P) = PD_\psi P^{\mathrm{T}} \tag{5}$$

where D_ψ is a $k \times k$ diagonal matrix of *pseudo-eigenvalues* with elements $\psi_j = \mathbf{p}_j^{\mathrm{T}} S\mathbf{p}_j = \mathbf{p}_j^{\mathrm{T}}\hat{\mathbf{p}}_j$, and $\hat{\mathbf{p}}_j$ is simply the result of smoothing $\mathbf{p}_j$ ($O(n)$ computations). Proposition 1 shows that this choice of D_ψ is optimal in a least squares sense. Natural choices for P are the orthonormal polynomials in $\mathbf{x}$. If P is U_k itself, then the ψ_j are the corresponding eigenvalues of S. Fig. 3(a) shows the third–sixth-order orthonormal polynomials superimposed on the corresponding eigenvectors of S for our example.

It turns out that we can do better than equation (5) with very little extra work. Consider the $k \times k$ eigendecomposition $P^{\mathrm{T}}SP = VD_\psi^* V^{\mathrm{T}}$, and define $P^* = PV$. Then $\hat{S}(P^*)$ is a better approximation to S than $\hat{S}(P)$ is.

Proposition 1. Let P be any $n \times k$ orthonormal basis, S a symmetric (smoother) matrix and P^* be defined as above. Then

(a) $\|S - \hat{S}(P)\|_F = \min_{D \text{ diagonal}} \|S - PDP^{\mathrm{T}}\|_F$,

(b) $\|S - \hat{S}(P^*)\|_F = \min_M \|S - PMP^{\mathrm{T}}\|_F$,

(c) $\|S - \hat{S}(P^*)\|_F \leqslant \|S - \hat{S}(P)\|_F$.

Proof. Both $\|S - PMP^{\mathrm{T}}\|_F$ and $\|P^{\mathrm{T}}SP - M\|_F$ are minimized by the same matrix M. The results follow immediately by matching elements of M (or D) with the corresponding elements of $P^{\mathrm{T}}SP$. The inequality in (c) is immediate since the approximations minimize the norm subject to conditions of nested generality. □

The pseudo-eigenvalues are indistinguishable from the corresponding genuine components in Fig. 1. On the log-scale, we start to see differences in the very small eigenvalues (10^{-5}). The pseudo-eigenvectors are also very close, especially for low order—see Fig. 3(b).

Remark 1. If $S = (I + \lambda K)^{-1}$, then it is not difficult to show that $\hat{S}(P^*)$ solves

$$\min_\beta \|\mathbf{y} - P\beta\|^2 + \lambda(P\beta)^{\mathrm{T}} K(P\beta). \tag{6}$$

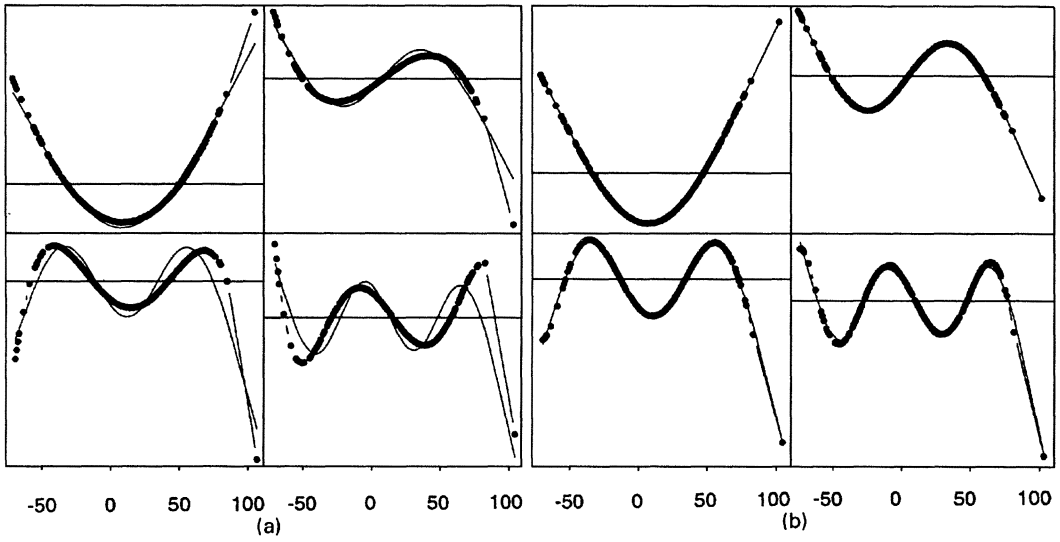

Fig. 3. (a) Third–sixth eigenvectors of a smoothing spline (———), along with the orthonormal polynomials of the same order (•) (they are similar in shape and in zero-crossing behaviour; the polynomials appear to be wilder in the tails, behaviour that is far worse for higher order polynomials); (b) as in (a) but showing the improved basis $P^* = PV$ (these are very close to the genuine eigenvectors)

(This was pointed out by the Associate Editor.) This amounts to solving the original penalized least squares problem over the subspace spanned by P. Note that the V above that diagonalizes P^TSP also diagonalizes P^TKP, which is the penalty matrix for the βs. Their respective eigenvalues are linked via the relationship $\psi_j^* = 1/(1 + \lambda\theta_j^*)$. From a computing aspect, P^TSP is far more attractive than P^TKP, since it simply involves the action of S on the columns of P (see the next section), whereas K is often implicit or buried in the code.

Remark 2. The eigenvalues of P^TSP and hence P^TKP, along with the P^*, give us a penalty sequence θ_k^* and basis to be used in equation (4), and the resulting pseudospline is an approximation to S. The smoothing parameter used in S is not critical, especially in the case above when S is a spline-type smoother, since the eigenvectors of P^TSP do not depend on it. Since the pseudospline has a built-in smoothing parameter, a single approximation to a particular version of S gives us an entire family of pseudosplines.

Remark 3. Although motivated by smoothing splines, and similar in structure to smoothing splines, the pseudospline can be used to approximate any linear smoother. This allows us to understand the action of S in terms of regularization in a particular basis. If S is not symmetric, the calculation of V and hence P^* requires some modification. Our current strategy is to symmetrize P^TSP by averaging it with its transpose.

Remark 4. In our examples our seed basis P are polynomials. Since each column of P^* is a linear combination of the columns of P, they are also polynomials, and P^*

spans the same space as P since S has full rank. The smoother $\hat{S}(P^*) = P(P^TSP)P^T$ operates by projecting first onto $C(P)$, smoothing using S, and then reprojecting onto $C(P)$.

Remark 5. Notice that there will be equality in proposition 1(c) under at least two conditions:

(a) if $P = U$, a subset of the eigenvectors of S, or
(b) if $P = P^*$ (so iterating the improvement will not help).

Fig. 4 illustrates the differences for the smoothing spline used in Fig. 1—$\hat{S}(P^*)$ approaches S as k increases, whereas $\hat{S}(P)$ does not.

5. COMPUTATIONAL DETAILS AND REFINEMENTS

An important feature of our construction is that the matrix S itself is not explicitly required; we simply need to be able to compute the action of S on the k n-vectors $\mathbf{p}_j$, an operation that can typically be performed in $O(n)$ operations. Our recipe above can of course be used for approximating any smoother. We have had experience with smoothing splines and locally weighted running lines (Cleveland, 1979), and for both of these it works well.

The rank required for the approximation will depend on $\mathrm{df} = \mathrm{tr}(S)$ for the smoother S—larger df will require higher rank. Our approach for developing a pseudospline approximation to S is therefore adaptive. Standard sequential algorithms exist for computing orthogonal polynomials. We always include the first two polynomials (constant and linear), since they are known eigenvectors. The computations proceed sequentially with each new polynomial $\mathbf{p}_j$, and hence basis of

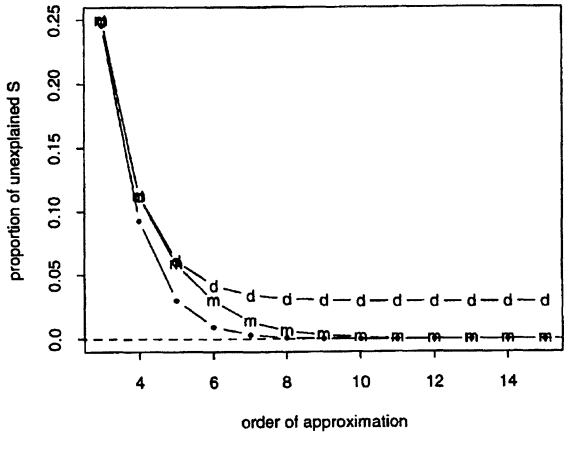

Fig. 4. Each curve represents the accuracy of a smoother approximation using a particular basis: d, $\|S - \hat{S}(P_k)\|^2/\|S\|^2$ as a function of k, the number of orthogonal columns in P_k, the orthogonal polynomial basis; m, the corresponding curve using the basis P_k^*; •, using the basis U_k, the ordered eigensubspace of S, corresponding to the best rank k approximation $\hat{S}(U_k)$ (S itself is the smoother used in Fig. 1 corresponding to $\mathrm{df}_\lambda = \mathrm{tr}(S_\lambda) = 5$

vectors P_j. Although we could compute P_j^* sequentially each time, we only need to diagonalize $P_j^{\mathrm{T}} S P_j$ once the approximation has been found to be satisfactory.

We need to decide how many terms J are sufficient. Ideally, a criterion of the form

$$F^2 = \frac{\|S - \hat{S}(P_j^*)\|_F}{\|S\|_F} \tag{7}$$

would be informative, but this is unavailable without S. For the present example we used $J = 8$ polynomials, and F^2 was 3.3%. In practice we continue to add terms until

$$\frac{\|\hat{S}(P_J^*) - \hat{S}(P_{J-1}^*)\|_F}{\|\hat{S}(P_J^*)\|_F} = \frac{2\|P_{J-1}^{\mathrm{T}} S \mathbf{p}_j\|^2 + \mathbf{p}_j^{\mathrm{T}} S \mathbf{p}_j}{\|P_J^{\mathrm{T}} S P_J\|_F} \tag{8}$$

is below some small threshold (0.001). Once the approximation is satisfactory, we diagonalize $P_J^{\mathrm{T}} S P_J$ to form P_J^* as described earlier.

To compute the fit $\hat{\mathbf{f}}$ at $\mathbf{x}$ when smoothing using $\hat{S}(P^*)$, we use

$$\hat{\mathbf{f}} = \hat{S}(P^*)\mathbf{y} = \sum_{j=1}^{J} \mathbf{p}_j^* \psi_j \langle \mathbf{p}_j^*, \mathbf{y} \rangle, \tag{9}$$

and, to compute the fit $\hat{f}(x_0)$ at a value x_0 not among the original x_i, we use

$$\hat{f}(x_0) = \sum_{j=1}^{J} p_j^*(x_0) \psi_j \langle \mathbf{p}_j^*, \mathbf{y} \rangle \tag{10}$$

where

$$p_j^*(x_0) = \sum_{k=1}^{J} p_k(x_0) V_{kj}$$

(remember that $P^* = PV$). We can include a smoothing parameter by replacing ψ_j by $1/(1 + \lambda\theta_j)$, where $\theta_j = 1/\psi_j - 1$.

Although $\hat{S}(P^*)$ performs better than $\hat{S}(P)$, they both are based on polynomials and might be dangerous especially when a higher rank approximation is needed. It turns out that we can improve *any* basis P as an approximation to an eigensubspace of S by smoothing each of the columns using S, followed by an orthogonalization. In the matrix algebra literature this corresponds to an iteration of the Q–R algorithm for finding an eigensubspace of a symmetric matrix. The Q–R algorithm is a generalization of the *power* method for iteratively finding a single eigenvector. Let $QR = SP$, where Q is orthogonal and R is the upper triangular matrix that orthogonalizes SP. Then $Q(Q^{\mathrm{T}} S Q)Q^{\mathrm{T}}$ is a better approximation to S than is $P(P^{\mathrm{T}} S P)P^{\mathrm{T}}$, or, in terms of $\hat{S}$, $\hat{S}(Q^*)$ is better than $\hat{S}(P^*)$.

We can be more precise about these improvements.

Proposition 2. Let P be any $n \times k$ orthonormal basis and let $QR = SP$ define an *improved* orthonormal basis Q for approximating the symmetric smoother S with eigenvalues in [0, 1]. Then

$$\|S - \hat{S}(Q^*)\|_F \leqslant \|S - \hat{S}(P^*)\|_F \tag{11}$$

where $\hat{S}(P^*) = P(P^TSP)P^T$ and $\hat{S}(Q^*)$ is defined similarly.

There is strict equality if the columns of P coincide with a subset of the columns of U, the eigenvectors of S. Notice that the proposition is not stated for $\hat{S}(P)$ and $\hat{S}(Q)$; there are counter-examples. A proof of the proposition is given in Appendix A and depends on a lemma proved by Jeff Lagarias. Lagarias (1991) explored inequalities of this nature in a more general context.

Each iteration of this Q–R algorithm requires $k-2$ additional applications of the smoother, and an order k eigendecomposition. The need for this added accuracy depends on the particular application. So far we have found that $\hat{S}(P^*)$ is sufficiently accurate for our applications, which typically involve small df.

6. APPLICATION: ADDITIVE MODELS

Our motivation for developing pseudosplines was to facilitate some of the difficult computations required for analysing additive models — this section describes some of these. These are by no means the only applications. Any scenario where smoothing splines and other smoothers are used, especially in a compound, non-standard fashion, can benefit from the parsimonious representation.

The penalized least squares criterion (1) is easily generalized for fitting an additive model

$$y_i = \alpha + \sum_{j=1}^{p} f_j(x_{ij}) + \epsilon_i$$

(Buja *et al.*, 1989):

$$\min_{f_j(\,)\in W_2} \sum_{i=1}^{n} \left\{ y_i - \alpha - \sum_j f_j(x_{ij}) \right\}^2 + \sum_{j=1}^{p} \lambda_j \int f_j''(t)^2 \, \mathrm{d}t. \tag{12}$$

It can be shown that the solutions satisfy

$$\begin{pmatrix} I & S_1 & S_1 & \dots & S_1 \\ S_2 & I & S_2 & \dots & S_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ S_p & S_p & S_p & \dots & I \end{pmatrix} \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \vdots \\ \mathbf{f}_p \end{pmatrix} = \begin{pmatrix} S_1\mathbf{y} \\ S_2\mathbf{y} \\ \vdots \\ S_p\mathbf{y} \end{pmatrix} \tag{13}$$

where each of the S_j is the appropriate smoothing spline matrix (each S_j actually represents $E_j^TS_jE_j$ where E_j orders $\mathbf{x}_j$) and $\mathbf{f}_j$ the vector of evaluations of the jth function. This $np \times np$ system is prohibitively expensive to solve directly — $O\{(np)^3\}$ — although by taking into account the special structure of the smoothing spline matrices can be reduced to an $n \times n$ system and hence is $O(n^3)$. Buja *et al.* (1989) described a backfitting or blockwise Gauss–Seidel algorithm for solving the system iteratively. It is particularly well suited for the job, since the operations $S_j\mathbf{z}$ can be

computed in $O(n)$ operations ($O(n \log n)$ if the data are to be sorted), and thus the whole solution can be obtained in $O(npq)$ operations, where q is the number of complete cycles. Sometimes the iterations converge slowly, especially if the smoothing windows are small and/or the variables are near collinear or *concurvous*.

Suppose that for each smoother S_j we have a rank k_j approximation $\hat{S}_j = P_j D_{\psi_j} P_j^{\mathrm{T}}$, and as before $D_{\psi_j} = (I + D_{\theta_j})^{-1}$. What happens if we plug them into equation (13)? Since $\mathbf{f}_j$ lies in $\mathcal{C}(P_j)$ we can write $\mathbf{f}_j = P_j \beta_j$. It is not difficult to show that equation (13) reduces to

$$(P^{\mathrm{T}} P + D_\theta)\beta = P^{\mathrm{T}} \mathbf{y}, \tag{14}$$

where $P = (P_1 \colon P_2 \colon \ldots \colon P_p)$, and D_θ and β are similarly composite versions of the separate penalties and coefficients.

This system has dimension $K = \Sigma_j k_j$, typically between $6p$ and $10p$, and much smaller than the original np. In fact each P_j smoother includes a constant column which we do not replicate in P, so the real dimension is $K - p + 1$.

The estimate has the form of a generalized ridge regression as in Section 2 for the single smoother, with criterion

$$Q(\beta) = \|\mathbf{y} - P\beta\|^2 + \beta^{\mathrm{T}} D_\theta \beta. \tag{15}$$

From our knowledge of the form of the contributions to D_θ, we see that the higher order components belonging to each variable are penalized simultaneously. We could add an additional parameter λ_j for each term as in Section 2.3, or else a global shrinking parameter λ.

Fig. 5 shows the pseudo-additive model fit for three variables from the air pollution data set of Breiman and Friedman (1985). The functions each have approximately 4 degrees of freedom, and each are approximated by seven pseudo-eigenvectors. The dotted functions were obtained by using the backfitting algorithm with the same smoothing splines used in the approximations.

6.1. *Hat Matrices*

The fitted functions in Fig. 5 have been enhanced by plotting standard error curves. Later we discuss generalized cross-validation (GCV), sensitivity analysis and diagnostics. The main ingredient for computing all of these is the *hat matrix* G for the fit $\hat{\mathbf{f}}_+ = \Sigma_j \hat{\mathbf{f}}_j = G\mathbf{y}$ where $G = P(P^{\mathrm{T}} P + D_\theta)^{-1} P^{\mathrm{T}}$, as well as the G_j that produce the individual fitted functions $\hat{\mathbf{f}}_j = G_j \mathbf{y}$. Here $G_j = P_j (P^{\mathrm{T}} P + D_\theta)_j^{-1} P^{\mathrm{T}}$ where $(P^{\mathrm{T}} P + D_\theta)_j^{-1}$ denotes the appropriate submatrix consisting of k_j of the K rows of $(P^{\mathrm{T}} P + D_\theta)^{-1}$.

Here we see the real strength of the additive model approximations. Hastie and Tibshirani (1987) used the backfitting algorithm itself to compute G and the individual G_j. They simply ran the backfitting algorithm n times, each time using for $\mathbf{y}$ a column of the $n \times n$ identity matrix, and hence built up G and the G_j a column at a time. Since this is $O(n^2)$ (with a large constant) it is typically used only once at the end of a series of fits in an analysis. The approximations, in contrast, can be routinely computed along with each fit. Not only are they available cheaply, but when used their factored form can be exploited to reduce the particular computations.

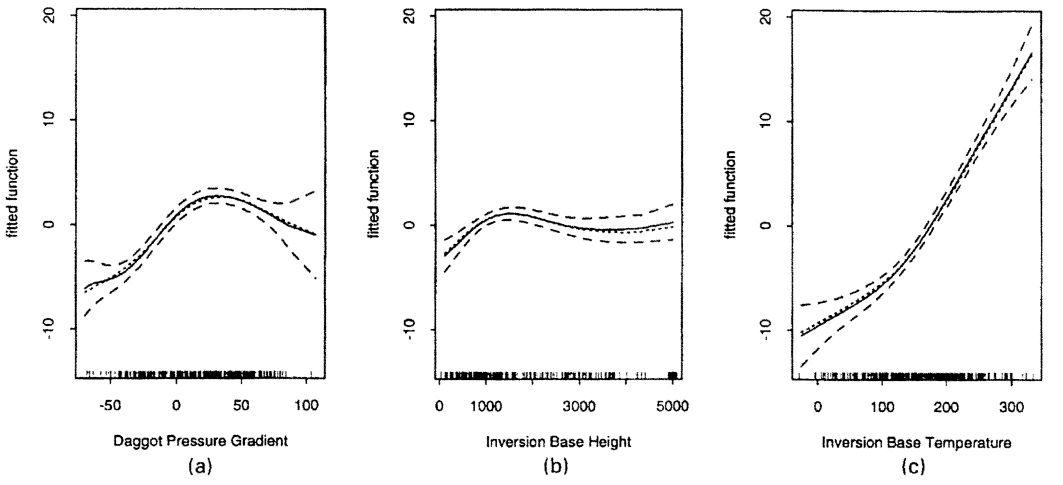

Fig. 5. Pseudoadditive model fitted to some air pollution data: the fitted functions are plotted on the same scale and superimposed on the plots (·····) are the fitted functions obtained using the backfitting algorithm with the original smoothing splines; included also are bands of twice the standard error curves, which also give an indication of influence; the rug plot at the base of each figure indicates the occurrence of data, jittered to represent ties

6.2. *Standard Errors*

If we assume that the y_i are independent and identically distributed with variance σ^2, from equation (14)

$$\operatorname{cov}(\hat{\beta}) = (P^{\mathrm{T}}P + D_\theta)^{-1}P^{\mathrm{T}}P(P^{\mathrm{T}}P + D_\theta)^{-1}\sigma^2. \tag{16}$$

Thus $\operatorname{cov}(\hat{\mathbf{f}}_j) = G_j G_j^{\mathrm{T}}\sigma^2 = P_j \operatorname{cov}(\hat{\beta})_j P_j^{\mathrm{T}}$ where $\operatorname{cov}(\hat{\beta})_j$ denotes the appropriate $k_j \times k_j$ submatrix of equation (16). The standard error curve for $\mathbf{f}_j$ requires the diagonal of $G_j G_j^{\mathrm{T}}$ and an estimate for σ^2.

As in Buja *et al.* (1989) and Cleveland and Devlin (1988), we use $\mathrm{RSS}/(n - d_1)$ to estimate σ^2 where $n - d_1$ is an appropriate estimate of residual degrees of freedom. Since $E(\mathrm{RSS}) = \operatorname{tr}\{(I - G)^{\mathrm{T}}(I - G)\sigma^2\}$ + bias term we use $d_1 = 2\operatorname{tr}(G) - \operatorname{tr}(G^{\mathrm{T}}G)$.

Alternatively we can follow the Bayesian route for smoothing splines (Hastie and Tibshirani (1990), section 5.4). Again these hat matrices are essential—for example, the posterior covariance of $\mathbf{f}_+$ under the natural prior is $G\sigma^2$.

6.3. *Generalized Cross-validation*

When fitting additive models, we need to specify the amount of smoothing for each of the terms in the model. One approach is to generalize the automatic methods for univariate smoothers, such as cross-validation or GCV. Gu *et al.* (1989) described a GCV approach for smoothing splines which uses the Newton method to optimize $\mathrm{GCV}(\lambda)$ with respect to its vector argument λ, consisting of a parameter λ_j for each of the p variables in the model. In our case the GCV criterion is

$$\mathrm{GCV}(\lambda) = \frac{n\|I - G(\lambda)\mathbf{y}\|^2}{\mathrm{tr}\{I - G(\lambda)\}^2} \tag{17}$$

where $G(\lambda) = P(P^{\mathrm{T}}P + D_{\lambda\theta})^{-1}P^{\mathrm{T}}$ and $D_{\lambda\theta}$ is block diagonal having $\lambda_j D_{\theta_j}$ as the ith block. Gu *et al.* (1989) have a similar form for GCV and discuss several decompositions for optimizing it efficiently. We shall not go into further details here but point out that all their algorithms are $O(K^3)$, where K is the rank of P. In their case, $K = n$ and represents a significant computational cost; here we can use their same algorithms and trade off this computational cost with that incurred by using a rank K approximation to the system.

6.4. *Concurvity*

We can use approximation (14) to gain insight into the nature and stability of the solutions, exactly here and approximately for the system (13). Buja *et al.* (1989) introduced the concept of concurvity, which we illustrate here for the case $p = 2$. Equation (14) reduces to

$$\left\{\begin{pmatrix} I & P_1^{\mathrm{T}}P_2 \\ P_2^{\mathrm{T}}P_1 & I \end{pmatrix} + \begin{pmatrix} D_{\theta_1} & 0 \\ 0 & D_{\theta_2} \end{pmatrix}\right\}\begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} P_1^{\mathrm{T}}\mathbf{y} \\ P_2^{\mathrm{T}}\mathbf{y} \end{pmatrix}. \tag{18}$$

Suppose that a column of P_1, say $\mathbf{u}$, has correlation 1 with a column of P_2, say $\mathbf{v}$. This means that two columns of the left matrix $P^{\mathrm{T}}P$ are identical. However, if there are corresponding non-zero entries in the penalty part $D_\theta = \mathrm{diag}(D_{\theta_1}, D_{\theta_2})$, this will not result in a degeneracy. The only cases where degeneracies can occur are where the two corresponding penalty columns are zero; this will happen if the linear functions are collinear. This is a simple demonstration (which carries over to the general p case as well) of the result in Buja *et al.* (1989) for general smoothers: *the only exact concurvity that can exist is collinearity*. Exact and approximate concurvities are defined as appropriately constrained low order eigenvectors of $(P^{\mathrm{T}}P + D_\theta)$; details are beyond the scope of this paper. Donnell *et al.* (1994) defined and discussed concurvity in general, and the related concept of additive principal components. They made use of the approximations developed here in some of their examples.

7. WEIGHTS

There are several situations that call for weighted smoothers or additive model fits.

(a) *Tied predictors*: when there are tied values for a predictor, the correct approach for smoothing splines is
   (i) to collapse the responses to their averages at the unique values of the predictors and
   (ii) to perform a weighted fit with weights proportional to the numbers of duplicates.

(b) *Heteroscedasticity*: if the responses are measured with different precision, or known to have different variances, a weighted fit is more efficient.

(c) *Generalized additive models*: the Newton–Raphson algorithm for fitting generalized additive models (Hastie and Tibshirani, 1990) calls for a weighted additive model fit at each iteration.

We are now faced with a possible dilemma when approximating a weighted smoother S_W, since its eigenvectors will not be the same as those of S. This would seem to imply that each time we changed the weights (for example in the third item above) we would have to compute a new approximation.

Our approach is to use the same basis vectors and penalties derived for the unweighted case, and simply to compute a weighted ridge regression when the observation weights change. We can justify this when approximating smoothing splines. The eigenvectors of $S = (I + \lambda K)^{-1}$ are also those of K, and the eigenvalues of K are θ_k. If we view our approximation in the unweighted case as a method for approximating K, then we can use it to construct the weighted version of S as well. This gives $\hat{S}_W = (W + PD_\theta P^T)^{-1} W$, and since we confine $\mathbf{f} \in \mathcal{C}(P)$ this is equivalent to $\hat{\mathbf{f}} = P(P^T W P + D_\theta)^{-1} P^T W \mathbf{y}$ — a weighted ridge regression, which is what we wanted. See Appendix A for computing this estimate.

A point of possible confusion is when we use the pseudobases of several variables (each with a different number of ties) in an additive model fit. We simply replicate the bases vectors for the tied observations and deal with full size n-vectors for each covariate. Once again this is equivalent in the univariate case to performing the weighted generalized ridge regression.

8. DISCUSSION

Bates and Wahba (1983) discussed approximations for computing the GCV statistic for smoothing splines and similar problems. Their approximation involves a pivoted Q–R-decomposition of the B-spline or other cubic spline basis used to compute the smoothing spline, and thus intimate knowledge of the smoother used. Our approximation is similar but can be used for any smoother. Although we have used orthogonal polynomials to 'seed' our approximations, other more suitable candidates can be used. Recall that our preferred pseudospline $\hat{S}(P^*)$ relies only on the fact that $\mathcal{C}(P) \approx \mathcal{C}(U_k)$, whereas the eigendecomposition of $P^T S P$ sorts out the order. Thus a system that is better behaved than polynomials, such as trigonometric series or fixed knot splines, may provide a good approximation with a gain in stability; we have not explored this area.

Regression splines are an alternative low rank method for smoothing and additive modelling (Stone and Koo, 1985; Friedman and Silverman, 1989). We need to select a regression basis for each variable, a popular choice being piecewise cubic polynomials. These in turn require the choice of knots, whose number determines the dimension of the basis, and whose position determines their nature. Given a basis for each variable, the regression is computed by projection onto the union of the bases. For reasonably low rank models, it becomes crucial where these one or two interior knots are placed on a variable.

Smoothing splines and similar 'shrinking' smoothers represent an alternative philosophy. They use a high dimensional basis for each variable, but then rather than compute the regression by projection they use penalized least squares; this dampens the influence of elements of the basis in a structured way. This in turn reduces the effective dimension of the fit but allows access to the richer class of functions.

Pseudosplines come somewhere in between; they use a 'medium' rank basis but also perform shrinking. In doing so they expose the structure of the class of shrinking smoothers in a parsimonious way (see Fig. 2). If the basis is chosen to estimate the

important components of a smoothing spline basis, not much is lost. They provide an analytical tool for understanding the behaviour of a number of such smoothers operating jointly as in an additive model fit.

If too small a rank k is chosen, the family of pseudosplines will be limited to fits of total rank k which may not be sufficient. We have not pursued any systematic way of determining an 'optimal' value for k, since typically we intend to use the pseudospline as a building block. It seems reasonable to use a generous value for k and then to explore smoother submodels via the parameterized form (4), especially if this permits complicated compound fits such as in the additive model. In our applications we have found $k = 10$ to be reasonable.

We have only touched on some applications in this paper. Other important application areas are as follows:

(a) Hastie and Tibshirani (1990) derived diagnostic measures for additive models, which generalize the univariate versions developed for smoothing splines (Eubank, 1985), as well as those developed for ridge regression (e.g. Eubank and Gunst (1986) and Walker and Birch (1988))—typically the smoothing matrices G_j and G of Section 6.1, or at least their diagonals, are required; the approximations can be used instead;
(b) constructing influence measures based on the joint behaviour of the covariates as well as residuals, analogous to the influence diagnostics of linear regression;
(c) understanding the effect on the influence diagnostics when the amount of smoothing is changed, as well as the dimension of the pseudobases;
(d) understanding the effects of near concurvity, and the causes;
(e) fitting additive models in complex scenarios where iterative algorithms are not easily available (Cox model) or where they have convergence problems (autoregressive models);
(f) providing explicit solutions for more general penalized multivariate functional models, such as functional canonical correlation analysis and ACE (Breiman and Friedman, 1985).

ACKNOWLEDGEMENTS

This paper has benefited from many discussions with Andreas Buja, John Chambers, Jeff Lagarias, Colin Mallows, Vijay Nair, Daryl Pregibon and Rob Tibshirani, as well as the comments of the referees on earlier drafts. Section 9.3.6 of Hastie and Tibshirani (1990) was based on an earlier and longer version of this paper; the present version has been shortened to avoid overlap.

This research was done while the author was a member of the Statistics and Data Analysis Group, AT&T Bell Laboratories, Murray Hill, New Jersey.

APPENDIX A

A.1. *Iterating Pseudospline Approximation*

Proposition 2. Let P be any $n \times k$ orthonormal basis and let $QR = SP$ define an *improved* orthonormal basis Q for approximating the symmetric smoother S with eigenvalues in [0, 1]. Then

$$\|S - \hat{S}(Q^*)\|_F \leqslant \|S - \hat{S}(P^*)\|_F$$

where $\hat{S}(P^*) = P(P^TSP)P^T$ and $\hat{S}(Q^*)$ is defined similarly.

Proof. Since $Q^TQ = I$, we have that $R^TR = P^TS^2P$. Let $S = UDU^T$ be the eigen-decomposition of S, and let $V = U^TP$. Note that V is also $n \times k$ orthonormal. Expanding the squared norm on the left-hand side we obtain

$$\|S - \hat{S}(Q^*)\|^2 = \text{tr}\{(S - QQ^TSQQ^T)^T(S - QQ^TSQQ^T)\} \tag{19}$$

$$= \text{tr}(S^TS) - \text{tr}\{(Q^TSQ)^2\} \tag{20}$$

and similarly for the norm on the right-hand side.

We therefore need to show that $\text{tr}(Q^TSQ)^2 \geqslant \text{tr}(P^TSP)^2$. Now

$$\text{tr}(Q^TSQ)^2 = \text{tr}(R^{-T}P^TS^3P^TR^{-1}R^{-T}P^TS^3P^TR^{-1})$$

$$= \text{tr}(V^TD^3V)(V^TD^2V)^{-1}(V^TD^3V)(V^TD^2V)^{-1} \tag{21}$$

$$= \text{tr}\{(V^TD^2V)^{-1/2}(V^TD^3V)(V^TD^2V)^{-1/2}\}^2. \tag{22}$$

It is sufficient to show that

$$(V^TD^2V)^{-1/2}(V^TD^3V)(V^TD^2V)^{-1/2} \geqslant V^TDV,$$

since then the trace inequality (for any positive power) follows. This is shown in lemma 1 below. □

Lemma 1 (Lagarias, 1991). Define V and D as above. Then

$$(V^TD^3V) \geqslant (V^TD^2V)^{1/2}(V^TDV)(V^TD^2V)^{1/2}.$$

Since $(V^TD^2V)^{1/2}$ is positive definite, this is equivalent to what is required in the proof of proposition 2.

Proof. By Ando (1979), corollary 4.2, part ii, $V^TAV \geqslant (V^TA^\alpha V)^{1/\alpha}$ for $\alpha \in (\frac{1}{2}, 1)$ and A positive definite. Taking $A = D^3$ and $\alpha = \frac{2}{3}$, we obtain $V^TD^3V \geqslant (V^TD^2V)^{3/2}$. Applying the Ando result again, we obtain $V^TD^2V \geqslant (V^TDV)^2$. Now if $B \geqslant C \geqslant 0$, B and C symmetric, then $B^\beta \geqslant C^\beta$ for $\beta \in (0, 1)$ (e.g. Chan and Kwong (1985)). This gives $(V^TD^2V)^{1/2} \geqslant (V^TDV)$, and thus

$$V^TD^3V \geqslant (V^TD^2V)^{3/2}$$

$$= (V^TD^2V)^{1/2}(V^TD^2V)^{1/2}(V^TD^2V)^{1/2}$$

$$\geqslant (V^TD^2V)^{1/2}(V^TDV)(V^TD^2V)^{1/2}.$$

□

A.2. *Computational Details*

Even though the computational burden has been dramatically reduced, it can still be costly to manipulate regressions with $10p$ predictors unless we are careful. We outline an approach for the unweighted case since it is slightly simpler and the ideas are the same as in the weighted case.

A well-known trick (for example see Golub and van Loan (1983)) reduces the generalized ridge regression problem (14) to an ordinary least squares problem. Define the augmented regression matrix and response variable

$$P^a = \begin{pmatrix} P \\ D_\theta^{1/2} \end{pmatrix} \quad \text{and} \quad \mathbf{y}^* = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix}. \tag{23}$$

Then least squares regression of $\mathbf{y}^*$ onto P^a gives the correct coefficients $\hat{\beta}$. Let

$$P^a = QR = \begin{pmatrix} Q_1 \\ Q_2 \end{pmatrix} R \tag{24}$$

denote the Q–R-decomposition of P^a, where Q is $(n+k) \times k$ orthogonal and R is $k \times k$ upper triangular. Then $P = Q_1 R$, $D_\theta^{1/2} = Q_2 R$ with $Q^T Q = Q_1^T Q_1 + Q_2^T Q_2 = I$. The following are easily derived:

(a) $\hat{\beta} = R^{-1} Q_1^T \mathbf{y}$;

(b) under independent and identically distributed errors

$$\operatorname{cov}(\hat{\beta}) = \sigma^2 R^{-1}(I - R^{-T} D_\theta R^{-1}) R^{-T} = \sigma^2 \Sigma;$$

(c) $G = Q_1 Q_1^T$ and $\operatorname{cov}(\hat{\mathbf{f}}_+) = GG^T\sigma^2 = Q_1(I - Q_2^T Q_2)Q_1\sigma^2 = Q_1(I - R^{-T} D_\theta R^{-1})Q_1^T\sigma^2$.

The G_j require a little more work, since we destroy the Q–R structure when we look at subsets. Nevertheless, $G_j = P_j(R^{-1})_j Q_1^T$, $\operatorname{cov}(\hat{\mathbf{f}}_j) = P_j(\Sigma)_j P_j^T$, and we can exploit the upper triangular structure of R^{-1} and Σ (and their jth partitions) in the computations.

REFERENCES

Ando, T. (1979) Concavity of certain maps on positive definite matrices and applications to hadamard products. *Lin. Alg. Applic.*, **26**, 203–241.

Bates, D. and Wahba, G. (1983) A truncated singular value decomposition and other methods for generalized cross-validation. *Technical Report 715*. Department of Statistics, University of Wisconsin, Madison.

Breiman, L. and Friedman, J. (1985) Estimating optimal transformation for multiple regression and correlation (with discussion). *J. Am. Statist. Ass.*, **80**, 580–619.

Buja, A., Hastie, T. and Tibshirani, R. (1989) Linear smoothers and additive models (with discussion). *Ann. Statist.*, **17**, 453–555.

Chan, N. and Kwong, M. (1985) Hermitian matrix inequalities and a conjecture. *Am. Math. Mnthly*, **92**, 533–541.

Chen, R. and Tsay, R. (1993) Nonlinear additive arx models. *J. Am. Statist. Ass.*, **88**, 955–967.

Cleveland, W. S. (1979) Robust locally weighted regression and smoothing scatterplots. *J. Am. Statist. Ass.*, **74**, 829–836.

Cleveland, W. S. and Devlin, S. J. (1988) Locally weighted regression: an approach to regression analysis by local fitting. *J. Am. Statist. Ass.*, **83**, 596–610.

Demmler, A. and Reinsch, C. (1975) Oscillation matrices and spline smoothing. *Numer. Math.*, **24**, 375–382.

Donnell, J., Buja, A. and Stuetzle, W. (1994) Analysis of additive dependencies using smallest additive principal components. *Ann. Statist.*, **22**, 1635–1673.

Donoho, D. and Johnstone, I. (1994) Adapting to unknown smoothness via wavelet shrinking. *Technical Report*. Stanford University, Stanford.

Eubank, R. L. (1985) Diagnostics for smoothing splines. *J. R. Statist. Soc.* B, **47**, 332–341.

Eubank, R. L. and Gunst, R. F. (1986) Diagnostics for penalized least-squares estimators. *Statist. Probab. Lett.*, **4**, 265–272.

Friedman, J. and Silverman, B. (1989) Flexible parsimonious smoothing and additive modelling (with discussion). *Technometrics*, **31**, 3–39.

Friedman, J. and Stuetzle, W. (1981) Projection pursuit regression. *J. Am. Statist. Ass.*, **76**, 817–823.

Golub, G. and Van Loan, C. (1983) *Matrix Computations*. Baltimore: Johns Hopkins University Press.

Green, P. and Silverman, B. (1994) *Nonparametric Regression and Generalized Linear Models*. London: Chapman and Hall.

Green, P. and Yandell, B. (1985) Semi-parametric generalized linear models. *Lect. Notes Statist.*, **32**, 44–55.

Gu, C., Bates, D., Chen, Z. and Wahba, G. (1989) The computation of GCV functions through householder tridiagonalization with application to the fitting of interaction spline models. *SIAM J. Matrix Anal.*, **10**, 457–480.

Hastie, T. and Tibshirani, R. (1987) Non-parametric logistic and proportional odds regression. *Appl. Statist.*, **36**, 260–276.

——— (1990) *Generalized Additive Models*. New York: Chapman and Hall.

Lagarias, J. (1991) Monotonicity properties of the toda flow, the qr-flow, and subspace iteration. *SIAM J. Matrix Anal.*, **12**, 449–462.

Rice, J. (1982) Penalized orthogonal polynomial regression. Unpublished.

Rice, J. and Rosenblatt, M. (1983) Smoothing splines: regression, derivatives and deconvolution. *Ann. Math. Statist.*, **11**, 141–156.

Roosen, C. and Hastie, T. (1994) Automatic smoothing spline projection pursuit. *J. Comput. Graph. Statist.*, **3**, 235–248.

Silverman B. (1985) Some aspects of the spline smoothing approach to non-parametric regression curve fitting (with discussion). *J. R. Statist. Soc.* B, **47**, 1–52.

Speckman, P. (1982) Efficient nonparametric regression with cross-validated smoothing splines. Unpublished.

Stone, C. and Koo, C. (1985) Additive splines in statistics. *Proc. Statist. Comput. Sect. Am. Statist. Ass.*, 45–48.

Wahba, G. (1990) *Spline Models for Observational Data*. Philadelphia: Society for Industrial and Applied Mathematics.

Walker, E. and Birch, J. B. (1988) Influence measures in ridge regression. *Technometrics*, **30**, 221–227; correction, 469–470.