Statistics & Probability Letters 6 (1988) 331–334
North-Holland

April 1988

BLOCK DIAGONAL SMOOTHING SPLINES

Brian S. YANDELL

Departments of Horticulture and of Statistics, University of Wisconsin, Madison, WI, USA

Received June 1987
Revised July 1987

Abstract: Algorithms for generalized cross validation are modified to handle stratified nonparametric problems and generalized additive models. This is particularly useful when the smoothness penalties can be combined additively with only one tuning constant to determine. Specific changes are suggested to the package GCVPACK (Bates et al., 1987, Comm. Statist. B) for implementation.

Keywords: generalized cross validation, ridge regression, thin plate smoothing spline.

1. Introduction

We show that algorithms for generalized cross validation used for thin plate smoothing splines and related problems can be easily modified to handle problems with matrices of block-diagonal form. Taking advantage of block-diagonal forms where possible can lead to considerable savings of computing space and time. We present enhancements to GCVPACK (Bates et al., 1987) which allow one to use most of this package of subroutines unchanged.

One example of a problem with a block-diagonal design arises in the study of smooth functional relationships between predictors and response in which different "strata" may require differently shaped smooth functions. As another example, one may have data on disconnected regions and not wish to impose continuity or smoothness between regions, but may still wish to impose a global penalty for smoothness across all regions. Both of these can be depicted with the thin-plate spline model

$$y_{ij} = f_i(\boldsymbol{x}_{ij}) + \varepsilon_{ij}, \quad i = 1, \ldots, r, \quad j = 1, \ldots, n_i, \tag{1}$$

with the $(\boldsymbol{x}_{ij}, y_{ij})$ observed data, the f_i unknown functions assumed to be reasonably smooth, and the ε_{ij} independent zero-mean random variables with finite variance σ^2. Smoothness is imposed on f_i by introducing a global penalty $J_i(f_i)$. The penalized least squares solution minimizes the objective function

$$S_\lambda(f) = \frac{1}{N} \sum_{i=1}^{r} \sum_{j=1}^{n_i} \left(y_{ij} - f_i(\boldsymbol{x}_{ij})\right)^2 + \lambda \sum_{i=1}^{r} \alpha_i J_i(f_i), \tag{2}$$

with $N = \sum n_i$ and λ and α_i some constants. Härdle and Marron (1986) considered tests of functional shape using model (1), while Yandell and Hogg (1988) considered penalized likelihood estimation in a generalized linear model analog of (2). The case in which the f_i are parallel is a special case of the partial spline, or semi-parametric model.

Generalized additive models (Stone, 1985) can sometimes fit within a block-diagonal framework (Chen, 1986). Consider the model

$$y_j = \sum_{i=1}^{r} f_i(\boldsymbol{x}_j) + \varepsilon_j, \quad j = 1, \ldots, n, \tag{3}$$

in which one may wish to impose different penalties J_i on different smooth functions f_i. The objective function for this problem is

$$S_\lambda(f)=\frac{1}{n}\sum_{j=1}^{n}\left(y_j-\sum_{i=1}^{r}f_i(x_j)\right)^2+\lambda\sum_{i=1}^{r}\alpha_i J_i(f_i), \tag{4}$$

with λ and α_i some constants. Still another example concerns solving a large system with a general design matrix and a general smoothing penalty. Here, computer storage space and processing time are critical, and some mild assumptions leading to a block-diagonal penalty or design matrix can save on both accounts.

We enhance algorithms presented in Bates et al. (1987) (referred to below as GCVPACK) for the choice of λ in (2) and (4) by generalized cross validation with α_i fixed. Section 2 examines thin plate smoothing splines with no replicates while section 3 concerns the general design with a semi-norm penalty. Details of the use of GCVPACK subroutines for block diagonal matrices occur in section 4.

2. Thin plate smoothing splines

The minimizer of (2) can be represented as a member of a reproducing kernel Hilbert space, with the reproducing kernel implicitly defined by the penalities J_i. Thus the model (1) can be written in matrix form

$$y=T_i\beta_i+K_i\delta_i+\varepsilon_i,$$

in which $y_i^{\mathrm{T}}=(y_{i1},\ldots,y_{in_i})$, T_i is an $n_i\times t$ matrix whose columns span the null space, and K_i is an $n_i\times n_i$ non-negative semi-definite matrix corresponding to the penalty. In many applications, and in the thin plate smoothing spline routine of GCVPACK, *dtpss*, the penalty is the integrated squared m-th derivative of f_i. In the case where there are no replicated design points, which we consider here, the penalty can be written in matrix form as $J_i(f_i)=\delta_i^{\mathrm{T}}K_i\delta_i$.

Define T and K as block-diagonal matrices with diagonal blocks T_i and α_iK_i, respectively, and off-block elements being 0. Define $y^{\mathrm{T}}=(y_1^{\mathrm{T}},\ldots,y_r^{\mathrm{T}})$, and similarly β and δ are catenations of β_i and $\alpha_i^{-1/2}\delta_i$, respectively. We can write the objective function (4) in matrix form as

$$S_\lambda(\beta,\delta)=\frac{1}{N}\|y-T\beta-K\delta\|^2+\lambda\delta^{\mathrm{T}}K\delta.$$

Natural choices for α_i are $\alpha_i=1$ or $\alpha_i=n_i/N$. The constant λ will be chosen by generalized cross validation, as outlined below.

The linear algebra for the solution of this quadratic problem can proceed for each i as in GCVPACK. In other words, we take a QR decomposition of T_i as (Dongarra et al., 1979, Chapter 9)

$$T_i=F_iG_i=[F_{i1}:F_{i2}]\begin{bmatrix}G_{i1}\\ 0\end{bmatrix}=F_{i1}G_{i1}.$$

This is followed by a Cholesky decomposition (Dongarra et al., 1979, Chapter 8) of

$$F_{i2}^{\mathrm{T}}K_iF_{i2}=L_i^{\mathrm{T}}L_i,$$

with L_i square upper triangular of size n_i-t. A singular value decomposition (Dongarra et al., 1979, Chapter 10) of

$$L_i^{\mathrm{T}}=U_iD_iV_i^{\mathrm{T}}$$

leads to a convenient diagonal form, with D_i being diagonal and U_i and V_i being orthogonal, all of size n_i-t. If one defines $F_{(1)}$, $F_{(2)}$, U and D as block diagonal matrices with diagonal blocks F_{i1}, F_{i2}, U_i and $\alpha_i^{-1/2}D_i$, and $F=[F_{(1)}:F_{(2)}]$, then the function estimator is $\hat{y}=A(\lambda)y$, with the "hat" matrix of the form (2.7) of GCVPACK, namely

$$A(\lambda)=F\begin{bmatrix}I&0\\0&U\end{bmatrix}\begin{bmatrix}I&0\\0&D^2(D^2+N\lambda I)^{-1}\end{bmatrix}\times\begin{bmatrix}I&0\\0&U^{\mathrm{T}}\end{bmatrix}F^{\mathrm{T}}. \tag{5}$$

The generalized cross validation (GCV) function, which can be minimized to approximately minimize the predictive mean square error (Craven and Wahba, 1979), can be written as

$$V(\lambda)=\frac{N\sum_{i=1}^{r}\sum_{j=1}^{n_i-t}z_{ij}^2\left(1+d_{ij}^2/(N\alpha_i\lambda)\right)^{-2}}{\left(\sum_{i=1}^{r}\sum_{j=1}^{n_i-t}\left(1+d_{ij}^2/(N\alpha_i\lambda)\right)^{-2}\right)^2},$$

with $z_i=(z_{i1},\ldots,z_{in_i})^{\mathrm{T}}=U_i^{\mathrm{T}}F_{i2}^{\mathrm{T}}y_i$. Note that while one could optimize this over λ and α_i, this would be a time-consuming process. Fixing α_i allows a quick minimization, say by golden section on λ as done in GCVPACK.

Once λ is chosen, the estimates of β_i and δ_i can proceed separately for each function f_i as detailed in GCVPACK. In other words, for each i,

$$\delta_{i\lambda} = F_{i2} U_i \left(D_i^2 + N\lambda\alpha_i I\right)^{-1} U_i^{\mathrm{T}} F_{i2}^{\mathrm{T}} y_i,$$

and $\beta_{i\lambda}$ can be found by solving

$$G_{i1}\beta_{i\lambda} = F_{i1}^{\mathrm{T}}(y_i - K_i\delta_{i\lambda}).$$

3. General design matrix

More generally, we have a model

$$y = X\theta + \varepsilon, \tag{6}$$

with θ a p-dimensional vector, y an n-dimensional vector and X an $n \times p$ design matrix, subject to a penalty $J(\theta) = \theta^{\mathrm{T}}\Sigma\theta$, with Σ a $p \times p$ positive semi-definite symmetric matrix. This model has the objective function

$$S_\lambda(\theta) = \frac{1}{n} \| y - X\theta \|^2 + \lambda\theta^{\mathrm{T}}\Sigma\theta.$$

We assume here that Σ has a block-diagonal structure with diagonal blocks Σ_i of dimension $p_i \times p_i$, $i = 1, \ldots, r$. For convenience, let $\theta^{\mathrm{T}} = (\theta_1^{\mathrm{T}}, \ldots, \theta_r^{\mathrm{T}})$. The generalized additive model (3) can be placed in this form, with $\alpha_i^{-1}\Sigma_i$ the penalty matrix for f_i.

It is also possible to improve the algorithm if X has block diagonal form, in which the blocking of θ is a superset of that done for Σ. That is, we have $\theta^{\mathrm{T}} = (\phi_1^{\mathrm{T}}, \ldots, \phi_q^{\mathrm{T}})$, with $q \leqslant r$ and, for $1 \leqslant j \leqslant q$, $\phi_j^{\mathrm{T}} = (\theta_{j1}^{\mathrm{T}}, \ldots, \theta_{jk}^{\mathrm{T}})$ for some subset $j_1, \ldots, j_k$ of $1, \ldots, r$. We shall call such an X properly blocked.

The decomposition of the r block of a block-diagonal Σ can proceed separately. That is, we perform a pivoted Cholesky decomposition (Dongarra et al., 1979, Champer 8) followed by a QR decomposition, to arrive at the reparameterization (cf. GCVPACK)

$$\begin{pmatrix} \gamma_i \\ \beta_i \end{pmatrix} = \begin{bmatrix} R_{i1}^{-\mathrm{T}} & 0 \\ 0 & I \end{bmatrix} Q_i^{\mathrm{T}} E_i^{\mathrm{T}} \theta_i.$$

Let E, $R_{(1)}$, $Q_{(1)}$ and $Q_{(2)}$ be block diagonal matrices composed respectively of E_i, R_{i1}, Q_{i1} and Q_{i2}, with $Q_i = [Q_{i1} : Q_{i2}]$ and $Q = [Q_{(1)} : Q_{(2)}]$. We proceed to the matrix

$$Z = \left[Z_{(1)} : Z_{(2)}\right] = XEQ \begin{bmatrix} R_{(1)}^{-\mathrm{T}} & 0 \\ 0 & I \end{bmatrix}.$$

At this point, no further savings accrue unless X is properly blocked. If X is block diagonal but not properly blocked, $Z_{(1)}$ and $Z_{(2)}$ need not be block diagonal and must be treated in a general way. If X is properly blocked, then $Z_{(1)}$ and $Z_{(2)}$ are also block diagonal. Let the blocks be denoted Z_{j1} and Z_{j2}, $j = 1, \cdots, q$. For each j perform a QR decomposition of $Z_{j2} = F_j G_j$, followed by a singular value decomposition of $F_{j2}^{\mathrm{T}} Z_{j1}$, as in GCVPACK. This leads to a block diagonal form for $A(\lambda)$ similar to (5). One can then proceed to choose λ by generalized cross validation and to find parameter estimates in an analogous fashion to Section 2.

4. GCVPACK routines

In order to use GCVPACK for block diagonal problems, a few routines must be changed. For the thin plate smoothing splines, the driver dtpss must be modified to make r repeated calls to the subroutines dsetup, dqrdc, dftkf and dsgdcl, which set up and manipulate the K_i and T_i matrices. In addition, dtpss must keep track of the blocks K_i and T_i, and other ancillary information required for each call. The generalized cross validation routine dgcvl must be modified to call drsap repeatedly r times and to create long vectors of the d's and the z's. The vector of singular values should have d_{ij} replaced by $d_{ij}\alpha_i^{-1/2}$, allowing one to call dvlop without further modification. Once λ is determined, dgcvl must call dcfcrl repeatedly r times to obtain the estimates of β_i and δ_i.

Modifications to the general driver dsnsm can allow blocks for Σ and for X. Only minor modifications are needed to dsnsm, to keep track of storage space. The routine ddcom has to be modified to call dsgdc r times and dcrtz and dzdc q times. The routine dgcv must be changed in much the same way as for the thin plate smoothing spline case.

We note that with minor adjustments the other cases discussed in GCVPACK, replicated x values and partial splines can be easily handled along the same lines. In addition, the same general approach could be taken with the one-dimensional natural spline algorithms for generalized cross validation (Hutchinson and de Hoog, 1985).

Acknowledgements

This research has been supported in part by United States Department of Agriculture CSRS grant 511-100, and National Sciences Foundation grant DMS-8404970. Computing was performed on the UW-Madison Statistics VAX 11/750 Research Computer.

References

Bates, D.M., M.J. Lindstrom, G. Wahba and B.S. Yandell (1987), GCVPACK-Routines for Generalized Cross Validation, *Comm. Statist. B – Simul. Comput.* **16**, 263–297 (Algorithms Section).

Chen, Z. (1986), A stepwise approach for the purely periodic interaction spline model, Technical Report #792, Dept. of Statistics, Univ. of Wisconsin.

Craven, P. and G. 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.

Dongarra, J.J., J.R. Bunch, C.B. Moler and G.W. Stewart (1979), *Linpack Users' Guide* (SIAM, Philadelphia).

Härdle, W. and J.S. Marron (1986), Semiparametric comparison of regression curves, Technical Report #A-93, Projektbereich A, Universität Bonn.

Hutchinson, M.F. and F.R. de Hoog (1985), Smoothing noisy data with spline functions, *Numer. Math.* **47**, 99–106.

Stone, C.J. (1985), Additive regression and other nonparametric models, *Ann. Statist.* **13**, 689–705.

Yandell, B.S. and D.B. Hogg (1988), Modelling insect natality using splines, *Biometrics* **44** (to appear).