Splines, Knots, and Penalties

Paul H. C. Eilers
Department of Biostatistics
Erasmus Medical Centre
Rotterdam, The Netherlands
(p.eilers@erasmusmc.nl)

Brian D. Marx*
Department of Experimental Statistics
Louisiana State University
Baton Rouge, LA 70803 USA
(bmarx@lsu.edu)
*Corresponding Author

March 8, 2010

Abstract

Penalized splines have gained much popularity as a flexible tool for smoothing and semi-parametric models. Two approaches have been advocated: 1) use a B-spline basis, equally-spaced knots and difference penalties (Eilers and Marx, 1996) and 2) use truncated power functions, knots based on quantiles of the independent variable and a ridge penalty (Ruppert, Wand and Carroll, 2003). We compare the two approaches on many aspects: numerical stability, quality of the fit, interpolation/extrapolation, derivative estimation, visual presentation and extension to multi-dimensional smoothing. We discuss mixed model and Bayesian parallels to penalized regression. We conclude that B-splines with difference penalties are clearly to be preferred.

Keywords: P-splines, truncated power functions, difference penalty, interpolation, smoothing, mixed models.

1 Introduction

Almost 20 year ago we coined the name P-splines for a simple combination of two ideas for curve fitting: regression on a basis of B-splines and a difference penalty on the regression coefficients (Eilers and Marx, 1992). In a later article we fully developed this idea (Eilers and Marx, 1996). By using equally spaced knots and a large number of B-splines, the role of the basis is reduced to little more than a convenient smooth interpolation device. The penalty is the core ingredient of the model: smoothness is tuned by changing its weight. The basic idea is not new: O'Sullivan (1986) published a similar proposal. His penalty was more complicated, as it was discrete, but derived from the integrated squared derivative of the fitted curve. In contrast, P-splines use a purely discrete penalty, making it almost trivial to use differences of any order. No price has to be paid for this simplicity (Eilers and Marx, 1996). Interestingly, O'Sullivan's idea was revived recently by Wand and Ormerod (2008), who also supplied R code.

Ruppert and Carroll (2000) proposed a competing approach to smoothing, based on truncated power functions (TPF), unequally spaced knots (quantiles of x), and a ridge penalty. The material was greatly expanded in a book (Ruppert, Wand and Carrol, 2003). Both P-splines and TPF have become popular in statistics and in applied fields, as can be judged from citation counts. In an overview article, Ruppert, Wand and Carroll (2009) recently collected 314 references, from the period 2003-2007.

With the growing popularity, the nickname P-splines has gradually grown into a catch-all, blurring the distinction between the differences in basis functions and type of penalty. This can be confusing to people who enter the field. One goal of this article is to discuss and illustrate the qualitative differences between the two systems and to help users make a well-informed choice. As a start we propose more precise nicknames: *PB-splines* (for penalized B-splines) and *PT-splines* (for penalized truncated power functions).

Our second goal is to discuss some numerical aspects of both systems. The numerical condition of straight computations with PT-splines can be problematic, especially for quadratic or cubic splines. On the other hand, as is well known from the literature (de Boor, 2001; Dierckx, 1993), B-splines can be computed from TPF by a computing repeated differences. This is very useful to study equivalences and differences between the two ways of penalizing. The difference algorithm for B-splines does not have a good reputation with respect to numerical stability, but we show there is no need for concern.

A third goal is to make a convincing plea for equally spaced knots, also for PT-splines. The advantages are most clear when smoothly interpolating or extrapolating data. At the same time we attack the widely held idea that the number of splines has to be less than the number of observations. This simply is not true. Somewhat amazingly, the penalty automatically and gracefully handles the situation, even if there are many more splines than data points.

A fourth goal is to show that the difference penalty adaptively lends itself to extensions and generalizations, e.g. "designer penalties". Examples are: smoothing of circular of periodic data and reduction of "overshoot" by the use of multiple penalties.

PT-splines immediately lead to a mixed model approach, because they contain from the start an unpenalized part and a penalized part. The former (the global

polynomial part) can be interpreted as the fixed component and the latter as the random component. We show that PB-splines can be pressed into the same mold by a simple transformation of the B-spline basis and the addition of a basis of powers of x. However, it is more elegant and simple to introduce a model with only random components, in which an explicit fixed component does not occur at all, because it is taken care of automatically. This is also a convenient starting point for a Bayesian model, using the Gibbs sampler.

Along the way we discuss some interesting sidelines, like smoothing of two dimensional data with penalized tensor products of B-splines, and the use of the Whittaker smoother for data on regular grids, as presented in Eilers (2003). We also note that the field of application of P-splines has been extended in several directions: generalized additive models (GAM) (Marx and Eilers, 1998), multivariate calibration and signal regression (PSR) (Marx and Eilers, 1999, 2005) and models which contain a mix of building blocks chosen from GAM, PSR and varying-coefficient models (VCM) (Eilers and Marx, 2002). In all cases a B-spline basis is being used, with an uniform grid of knots, and (higher-order) differences in the penalty.

The article has been written in the style of an opinionated tutorial, in which theoretical ideas and practical illustrations go hand in hand. We have avoided unnecessary mathematical detail.

2 Smoothing with penalized spline regression

For completeness, we first briefly present the two approaches to penalized spline regression that we will compare. For a more extensive presentation the reader should consult Eilers and Marx (1996), which again we will refer to as EM or PB-splines, and Ruppert, Wand and Carroll (2003), which we will refer to as RWC or PT-splines. To keep the presentation simple, we do not consider the case of a spatially varying penalty until Section 13.

Let the data be m pairs (x_i, y_i). PB-splines use a basis of (quadratic or cubic) B-splines, B, computed on x and using equally-spaced knots. They write the model as $E(y) = \mu = B\alpha$ and minimize the following objective function:

$$Q_B = ||y - B\alpha||^2 + \lambda||D_d\alpha||^2, \tag{1}$$

where D_d is a matrix such that $D_d\alpha = \Delta^d\alpha$ constructs the vector of dth differences of α, and λ is a non-negative tuning parameter. Remember that $\Delta\alpha_j = \alpha_j - \alpha_{j-1}$, $\Delta^2\alpha_j = \Delta(\Delta\alpha_j) = \alpha_j - \alpha_{j-1} - (\alpha_{j-1} - \alpha_{j-2}) = \alpha_j - 2\alpha_{j-1} + \alpha_{j-2}$, and so on for higher d. Mostly $d = 2$ or $d = 3$ is used. Minimization of Q_B leads to the system of equations

$$(B'B + \lambda D_d'D_d)\hat{\alpha} = B'y. \tag{2}$$

Notice that for $\lambda = 0$ this reduces to the normal equations for linear regression of y on B. The number of basis functions in B is chosen "too large", which means that for $\lambda = 0$ the fitted curve is over-fitting the data, giving a result with too many fluctuations. Depending on the application, the size of the basis can be anywhere from 10 to over 1000. By increasing λ the smoothness can be tuned. In the limit of a very large λ a linear ($d = 2$) or quadratic ($d = 3$) fit is obtained.

Alternatively, PT-splines use a basis, F, of truncated power functions (TPF). For a given degree p, column j of F is given by

$$f_{ij} = (x_i - t_j)^p I(x_i > t_j), \tag{3}$$

$I(u)$ is the indicator function; it is 1 when $u \geq 0$ and 0 when $u < 0$. The vector t contains the knots. They are chosen as quantiles of the x variable. The model for $E(y) = \mu$ is given by

$$\mu_i = \sum_{k=0}^{p} \beta_k x_i^k + \sum_{j=1}^{n-1} b_j f_{ij}, \tag{4}$$

or

$$\mu = X\beta + Fb, \tag{5}$$

where X is a m by $p+1$ matrix with x_i^0 to x_i^p in row i. The objective is to minimize

$$Q_F = ||y - X\beta - Fb||^2 + \kappa||b||^2, \tag{6}$$

in which we recognize a ridge penalty on b. Minimization of Q_F leads to the system of equations

$$\begin{bmatrix} X'X & X'F \\ F'X & F'F + \kappa I \end{bmatrix} \begin{bmatrix} \beta \\ b \end{bmatrix} = \begin{bmatrix} X'y \\ F'y \end{bmatrix}. \tag{7}$$

RWC mostly use $p = 1$ in their PT-spline applications. The number of knots is chosen such that the fitted curve is over-fitting the data for small κ. Increasing κ increases the smoothness and in the limit a polynomial fit of degree p is obtained. Quantiles of x are chosen for the positions of the knots, hence they will generally not be equally-spaced.

Note that the penalty is essential. It might be small, but λ or κ should always have a positive value. In some applications, especially when interpolation occurs, B or F itself may be singular, but not $B'B + \lambda D'D$ or $F'F + \kappa I$.

3 Splines and knots

In this section we look at basis functions and their mutual relationships in more detail. We begin our presentation with TPF and equally-spaced knots since they are somewhat easier to explain, and B-splines can be derived from them.

Again, let the data be pairs $(x_i,\ y_i)$, $i = 1 \ldots m$. To simplify the presentation without loss of generality, we assume that all x lie between 0 and 1. It is clear that any set of x's can be linearly transformed to conform to this condition. Let $t_j = (j-1)/n$, $j = 2, \ldots, n$ be a set of $n-1$ equally-spaced knots.

The simplest system of truncated power functions (TPF) uses $p = 0$; it consists of step functions with jumps of size 1 at the knots. The right branch of a TPF of degree p looks like the right branch of $(x - t_j)^p$; the left branch is zero. Figure 1 shows linear and quadratic TPF bases, with equally-spaced knots.

B-splines can be computed as differences of TPF (de Boor, 2001; Dierckx, 1993). Take, as an example, TPF of degree zero. The difference

$$B_j(x; 0) = f_{j-1}(x; 0) - f_j(x; 0) = -\Delta f_j(x; 0) \tag{8}$$

is a rectangular function, which is 1 between t_{j-1} and t_j and zero everywhere else. Performing this computation for $n+1$ TPF, we get a basis with n B-splines of degree zero. In a similar way we can combine $n+2$ triples of degree one TPF to get $n+1$ triangular (degree one) B-splines:

$$B_j(x; 1) = f_{j-2}(x; 1) - 2f_{j-1}(x; 1) + f_j(x; 1) = \Delta^2 f_j(x; 1). \tag{9}$$

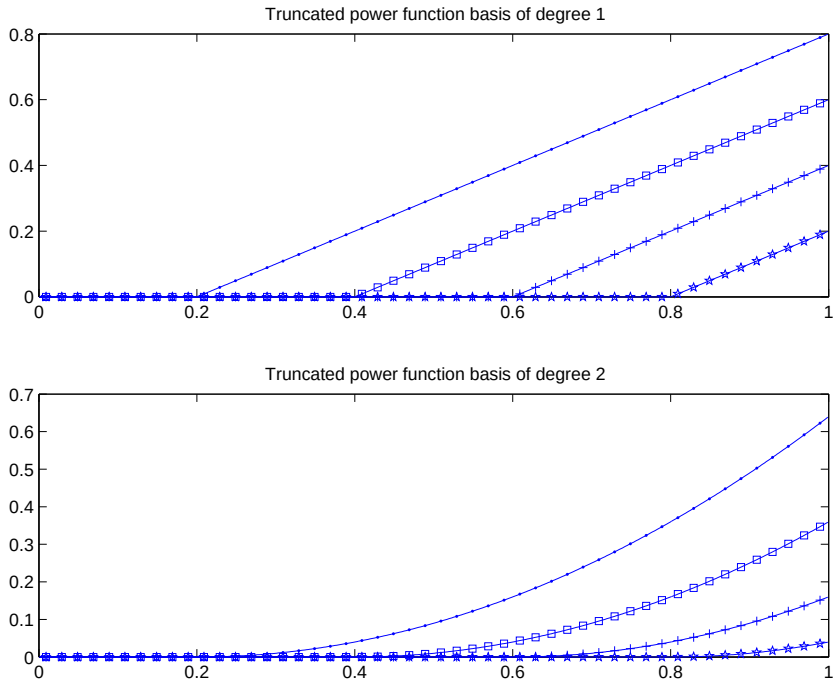

Figure 1: Truncated power function bases with equally-spaced knots. Upper panel: linear; lower panel: cubic.

Actually, a correction factor is needed, to realize the convenient condition that $\sum_j B_j(x;p) = 1$ for any degree p. The general formula is

$$B_j(x;p) = (-1)^{p+1}\Delta^{p+1} f_j(x;p)/(h^p p!), \tag{10}$$

where h is the distance between knots. These results only hold for equally-spaced knots. Somewhat more complicated results can be obtained for arbitrarily spaced knots, using divided differences (de Boor, 2001). However, it should become clear in the applications that we discuss, there is not any need to choose unequally-spaced knots, whether we use TPF or B-splines.

Notice that we need an extra $2p+2$ knots for the TPF, referred to as the expanded basis $\breve{F}$, to generate a complete B-spline basis. Thus in general $\Delta^{p+1} f$ is performed on $n+1+2p$ different $(p+1)$-tuples of degree p truncated polynomials resulting in $n+k$ B-splines. In languages likes R and Matlab, it is nearly trivial to compute a TPF basis and take differences to get a B-spline basis, as the following code fragment shows.

```
tpower <- function(x, t, p){
   # Truncated p-th power function
   (x - t) ^ p * (x > t)
}

bbase <- function(x, xl, xr, ndx, deg){
   # Construct a B-spline basis of degree 'deg'
   dx <- (xr - xl) / ndx
   knots <- seq(xl - deg * dx, xr + deg * dx, by = dx)
```

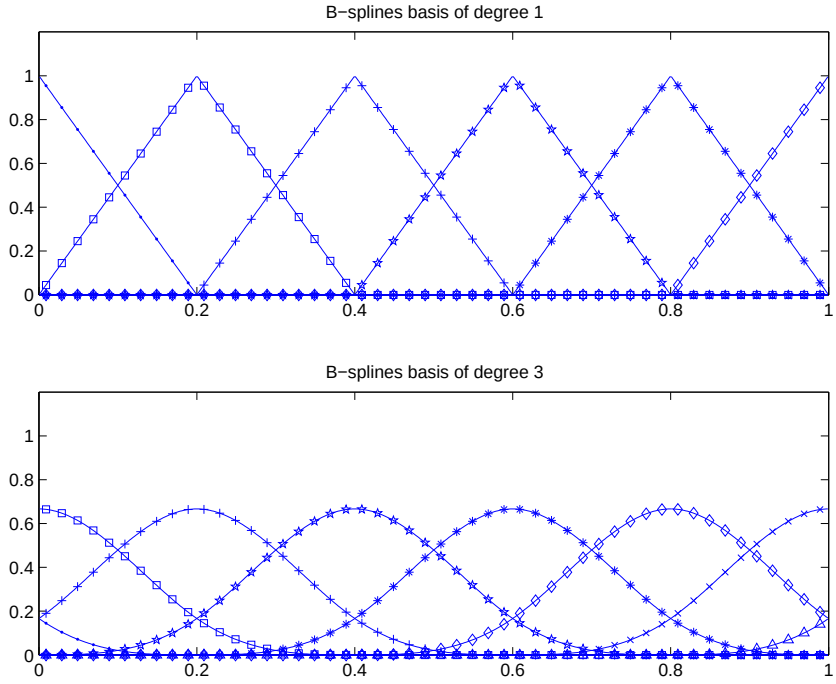

Figure 2: B-spline bases with equally-spaced knots. Upper panel: linear; lower panel: quadratic

```
    P <- outer(x, knots, tpower, deg)
    n <- dim(P)[2]
    D <- diff(diag(n), diff = deg + 1) / (gamma(deg + 1) * dx ^ deg)
    B <- (-1) ^ (deg + 1) * P %*% t(D)
    B
}
```

Figure 2 shows linear and cubic B-spline bases. All basis function have the same shape, but they are shifted horizontally by a multiple of the knot distance. This is also true at the boundaries, in contrast with other schemes, like natural B-splines, where near the boundaries the basis functions have different shapes.

Surprisingly, the differencing algorithm is not used often. Rather a recursive formula, deriving B-spline of degree p from those of degree $p-1$, starting at $p=0$, is more popular (de Boor, 2001).

We use the term "degree" to indicate B-splines that consist of $(p+1)$ segments of degree p. In the B-spline literature, it is common to use "order", which is $p+1$. We make this choice to avoid confusion with the order of difference penalties.

RWC strongly emphasize the use of unequally-spaced knots, based on quantiles of x, with PT-splines. This is good advice in regression without penalties. For one, it is crucial to avoid placing knots in "empty" regions of the domain of x, to avoid singularities. However, insisting on this choice of knots for penalized splines underestimates the power of the penalty. In Section 6, we will see that large stretches without data, but with many knots, will be interpolated automatically and smoothly. In addition equally-spaced knots are easy to specify and report, and they simplify the computations.

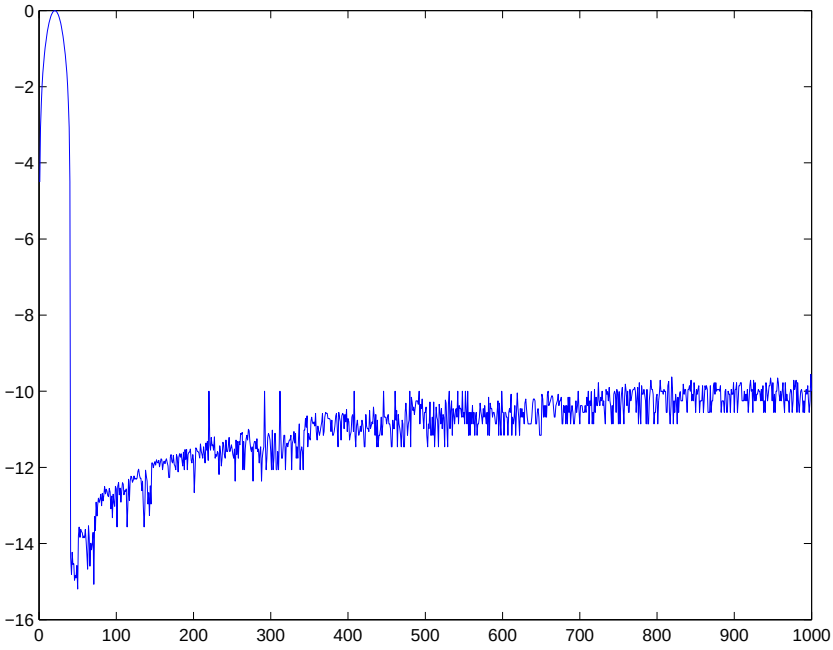

Figure 3: Logarithm of the absolute value of one cubic B-spline, computed from the fourth difference of cubic truncated polynomials. The B-splines has been scaled to a maximum of 1.

One might be worried by this simple approach to the computation of B-spline bases. De Boor (2001) warns against computing B-splines as differences of truncated power functions. It is interesting to study this in some detail.

Consider one cubic B-spline. Only four segments are non-zero. In the segments to the left of it no error will be made, because the TPF are all zero there and so will be their differences. On the right there is an opportunity to make errors. The worst case will occur for the leftmost B-splines at the right end of the domain. Figure 3 shows the absolute value of one cubic B-spline, scaled to a maximum of 1. The distance between the knots is 0.01. We see that the largest error is of the order of $10^{-10.}$.

To understand de Boor's warning better, we must take into account that he did his research in the 1970's, when single precision (4 bytes) was the default. In present-day computers it is the double precision IEEE 754 standard. IEEE 754 single precision has a relative precision of 2^{23}, or a little less than 7 digits, while it is 2^{54}, or over 16 digits, in double precision. Also de Boor considered B-spline of very high degree (up to 20, in his Exercise 9.2).

However, whether the error is small or not, we can completely eliminate it in a very simple way: we know that a B-spline has to be zero past its fourth knot, so we can simply give it a zero value there– resulting in no error. The trend of the error suggests that in the (nominally) non-zero part of the B-spline the error is very small. This was confirmed by a direct comparison with de Boor's recursive algorithm.

4 Penalties and coefficients

PB-splines use discrete penalties to tune the amount of smoothness. EM put a difference penalty on the coefficients of the B-spline basis functions. The degree of the B-splines and the order of the penalty can be chosen independently. EM advise to investigate several orders of the penalty and plot an information criterion (e.g. AIC) or a cross-validation measure against the effective dimension to get a good impression of the limiting behavior (which might indicate a polynomial model). RWC 's PT-splines always have a ridge penalty on the TPF, whatever their degree. We will show that this is equivalent to a B-spline basis and the order of the difference penalty equal to one higher than the degree of these TPF.

Without loss of generality, let the domain of x run from 0 to 1, and let the spacing of the knots be $1/n$. Then the $n-p$ truncated power functions of degree p in the basis F start at knot positions p/n to $(n-1)/n$. A B-spline basis B of degree p contains $n+p$ B-splines. To compute B as differences of order $p+1$ of a TPF basis $\breve{F}$, there have to be $n+2p+1$ basis functions in $\breve{F}$, corresponding to the knots $-p/n$ to $(n+p)/n$. In $B = \breve{F}\breve{D}'_{p+1}$, the order $p+1$ differencing matrix $\breve{D}_{p+1}$ has $n+2p+1$ columns and $n+p$ rows. We can write $F = \breve{F}S$ if S is the identity matrix of size $n-p$, bordered by $p+1$ rows of zeros on the top and at the bottom. Thus post-multiplication by S selects the $n-p$ middle columns of $\breve{F}$. We have that $S'S = I_{n-p}$ and also that $\breve{D}_{p+1}S = D'_{p+1}$, the transpose of the $n-p$ by $n+p$ differencing matrix of order $p+1$.

Write a PB-spline and the PT-spline fits as

$$B\alpha = B(\gamma + a) = X\beta + Fb = \breve{F}\breve{D}'_{p+1}(\gamma + a) = X\beta + \breve{F}Sb, \tag{11}$$

with $B\gamma = X\beta$ and $Ba = Fb$. We thus have that $\breve{D}'_{p+1}a = Sb$, and multiplying both sides by S' gives $S'\breve{D}'_{p+1}a = D_{p+1}a = S'Sb = b$. We also have that $D_{p+1}\gamma = 0$, because if $B\gamma = X\beta$, γ has to be a sequence of degree p, and hence $D_{p+1}\gamma = 0$. This finally proves that $D_{p+1}\alpha = b$ and that a ridge penalty on b is equivalent to a difference penalty of order $p+1$ on α.

If we go in the reverse direction, we can interpret TPF as (repeated) sums of B-splines. The columns of X in $\mu = X\beta + Fb$ are needed to recover the powers of x that disappear when taking differences.

5 Visualization

The details of a PB-spline model can be visualized in an attractive way. This is not important for everyday smoothing, but may be useful and instructive when introducing new users to the method. Figure 4 illustrates simulated data that have been fit with a rich B-spline basis and a second order penalty. The individual B-splines are shown, scaled by their coefficients. Also shown, as encircled dots, are the coefficients of the individual B-splines at the positions of their maxima (knot positions for odd degree, half-way between knots for even degree). These points are close to the fitted curve and present the skeleton of the fit. The B-splines put the flesh on this skeleton, its softness determined by their degree.

This type of presentation also helps to make clear what the difference penalty is doing: it forces the skeleton, i.e. the coefficients, to follow a smooth pattern. Consequently the full curve that follows from them will also be smooth.

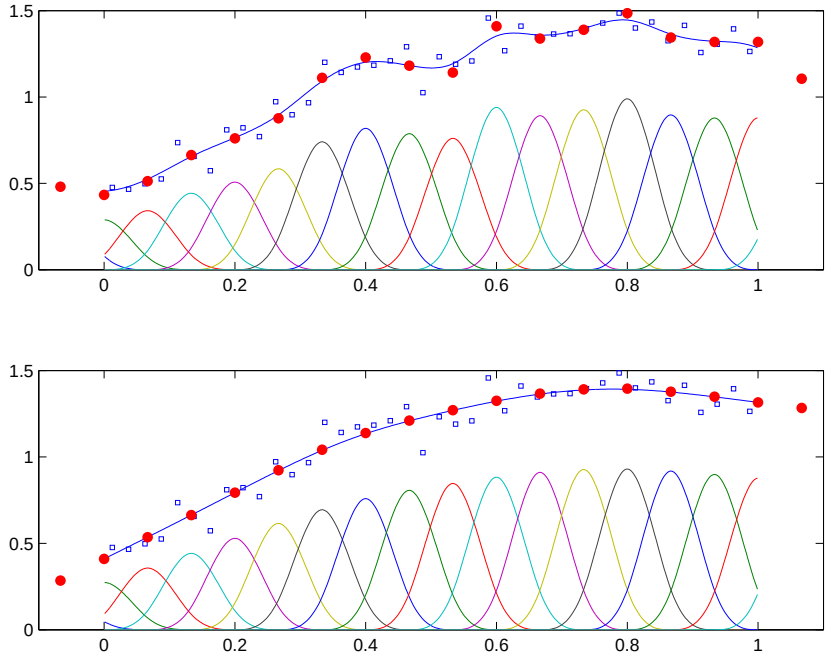

Figure 4: Components of a fit with 18 cubic B-splines and a second order penalty to simulated data (squares). The encircled dots show the coefficients of the B-splines. Top: $\lambda = 0.01$, bottom: $\lambda = 10$.

Notice that also the coefficients of the "end splines" are presented. The dots remind us of the "control points" that are used in computer-aided design and graphical software to shape Bezier curves (Gasson, 1983).

PT-splines with TPF do not lend themselves to such an insightful presentation. Figure 5 shows an example. The coefficients cannot be connected to the data (they are second differences of the coefficients of the corresponding B-spline basis) and a plot of the scaled basis functions shows many crossing straight lines.

6 Interpolation and extrapolation

Penalized splines allow straightforward smooth interpolation and extrapolation. In this area the power of penalties becomes most clearly visible. As we will see the choice of knots is influential.

In Figure 6 we illustrate interpolation and extrapolation with PB-splines, using cubic B-splines on a fine grid of knots and a second order penalty. The individual B-splines and the coefficients are also shown. Notice that in large parts of the domain of x there are no data but ample knots. Yet the automatically generated interpolating curves are smooth and look natural. This is the work of the penalty. When interpolating, the B-spline coefficients form a polynomial sequence of degree $2d - 1$, and for extrapolation the degree is $d - 1$. Thus, when $d = 2$, we get cubic interpolation and linear extrapolation.

Let W be a diagonal weight matrix. Consider interpolation: because a part of the diagonal of W in $B'WB$ contains zeros, a number of rows (equal to the number of zeros in the diagonal of W) of $B'WB$, as well as the corresponding elements of $B'Wy$,

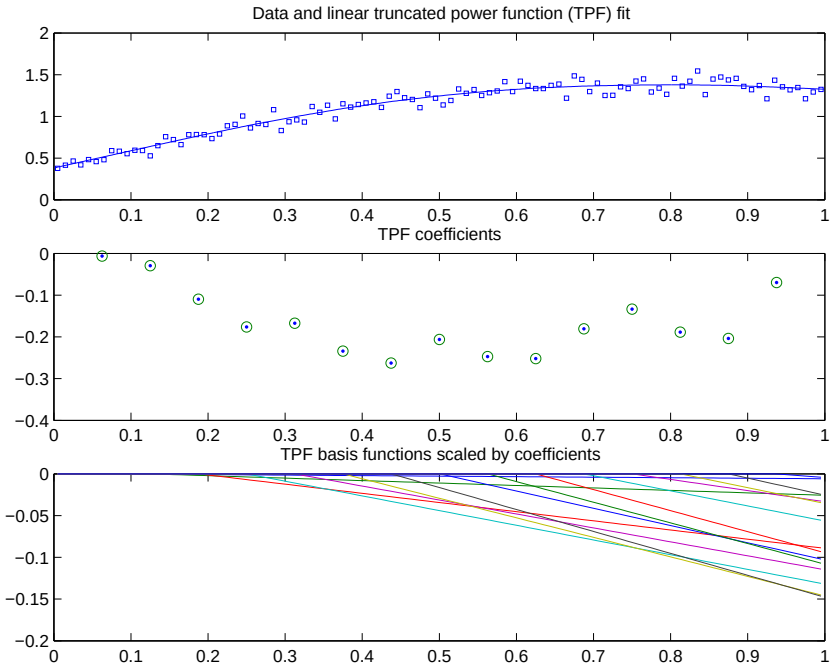

Figure 5: Components of a fit with 15 ($\kappa = 0.1$) linear truncated power functions to simulated data.

will contain only zeros. If $P_d = D_d'D_d$, it follows from $(B'WB + \lambda D_d'D_d)\hat{\alpha} = B'Wy$ that $\sum_k p_{jk}\hat{\alpha}_k = 0$ holds for these rows. Consider the upper left parts of P_d for orders 1 and 2:

$$P_1 = \begin{bmatrix} 1 & -1 & 0 & 0 & \dots \\ -1 & 2 & -1 & 0 & \dots \\ 0 & -1 & 2 & -1 & \dots \\ 0 & 0 & -1 & 2 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}; \quad P_2 = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 & 0 & \dots \\ -2 & 5 & -4 & 1 & 0 & 0 & \dots \\ 1 & -4 & 6 & -4 & 1 & 0 & \dots \\ 0 & 1 & -4 & 6 & -4 & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots & & \end{bmatrix}. \tag{12}$$

From row 2 (up to row $n-1$), the pattern of elements in the rows of P_1 is the same as that of the rows of D_2, but shifted to the left by one position. From row 3 (up to row $n-2$), the pattern of elements in the rows of P_2 is the same as that of the rows of D_4, but shifted to the left by two positions. In general we find in the rows $d+1$ to $n-d$ of P_d the pattern of D_{2d}, shifted by d positions to the left. It follows that $\Delta^{2d}\hat{\alpha}_{k+d} = 0$ for the interpolated part of $\hat{\alpha}$ that corresponds to zero rows of $B'WB$. This will only hold if that part of $\hat{\alpha}$ is a polynomial (in the index) of degree $2d-1$. The coefficients of this polynomial are determined by the "boundary conditions" imposed by the surrounding equations with non-zero rows of $B'WB$. It will not be easy to show algebraically that the connection is smooth, but any non-smooth connection would needlessly increase the penalty.

At first sight it may seem that the same reasoning holds for extrapolation, but this is not true. In $D_1'D_1$ the rows 2 to $n-1$ correspond to the coefficients of second differences and thus would allow a linear sequence for the corresponding part of $\hat{\alpha}$. But the first row says that the first two elements of $\hat{\alpha}$ have to be equal, annihilating the linear part. Consequently extrapolation will be by a constant value

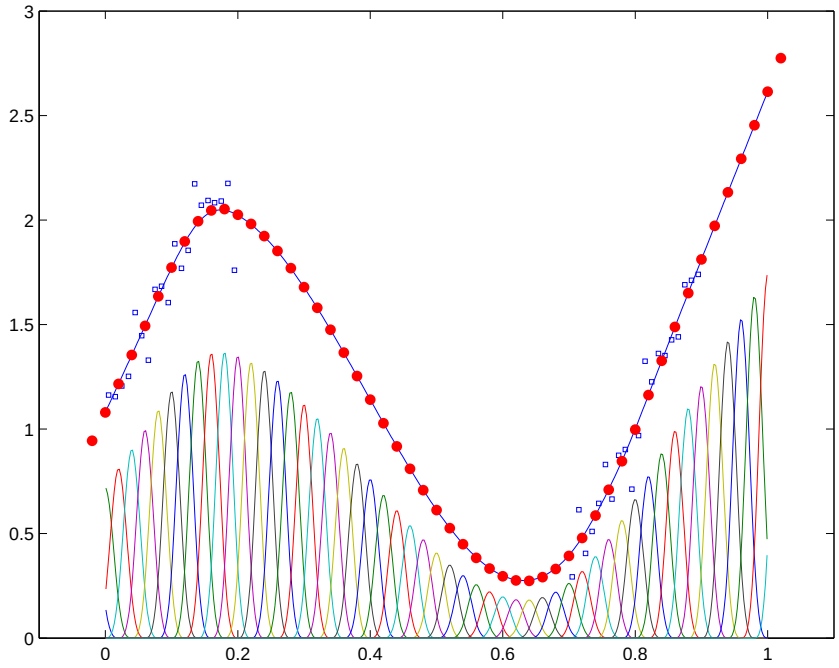

Figure 6: Smoothing and interpolation of simulated data with a large basis of cubic B-splines and a second order penalty ($\lambda = 10$). The scaled B-splines are shown on the bottom of the graph. Their sum gives the full line, which is the fitted curve. The encircled dots represent the value of the B-spline coefficients.

when $d = 1$. Similarly, rows 3 and further of $D_2'D_2$ would allow a cubic sequence for the extrapolated part of $\hat{\alpha}$, but the first two equations annihilate the quadratic and cubic components. In this case extrapolation is by a linear sequence. This reasoning applies to any value of d: the first d equations annihilate components of degree larger than d. By symmetry, the same holds for extrapolation to the right.

With PB-splines, the detailed behavior of the curve between the knots depends on the degree of the B-splines. To get a cubic interpolating curve (when $d = 2$), the B-splines need to be cubic too. However, if the number of knots is large it may well be that linear interpolation *between the knots* is acceptable and then a linear B-spline basis will suffice.

Perhaps surprisingly, PT-spline interpolation with linear TPF, a ridge penalty, and equally-spaced knots gives the same pleasant result, as Figure 7 shows. Of course it should, in view of the mathematical equivalence with a linear B-spline basis and a second order penalty. Notice however that between the knots the curve consists of linear pieces. The PT-spline scheme of RWC does not allow separate choices of degree of the basis and order of the penalty. A natural option may be to move up to cubic TPF basis functions, but then we implicitly choose a fourth order penalty in the equivalent P-spline model. Interpolation (extrapolation) then will be by a seventh (third) degree polynomial (in the coefficients), which might introduce more flexibility than needed.

Taking knots as quantiles of x is generally not a good idea in interpolation, as Figure 8 shows. To illustrate this point in the extreme case, a knot is placed at each measured x, (an extreme case of taking quantiles). The gap in the middle is bridged by a linear segment, which is less attractive than the one with many equally-spaced

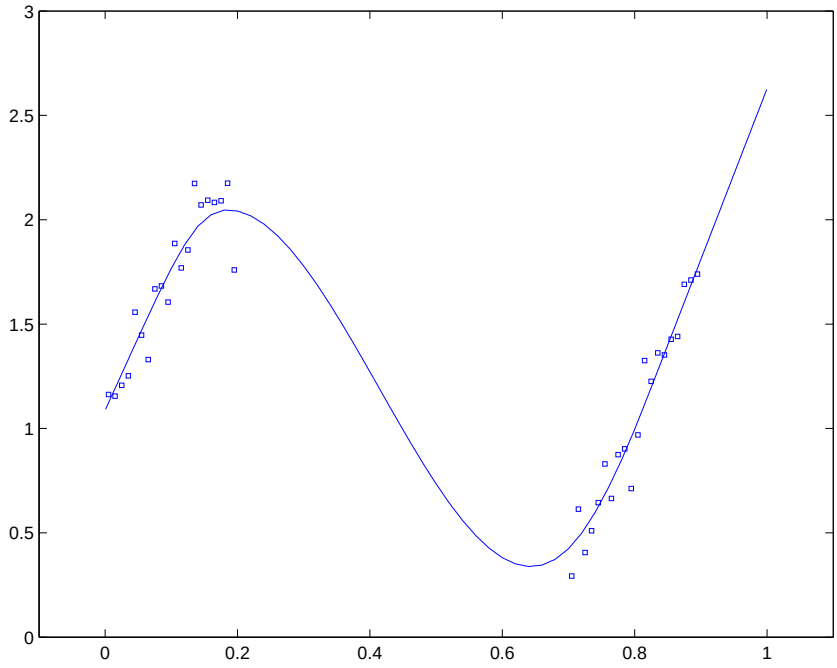

Figure 7: Smoothing and interpolation of simulated data with a basis of linear truncated power functions, with 100 equally-spaced knots. Ridge penalty with $\kappa = 0.1$.

knots (Figure 7). In this case, the fit to the available data is worse. RWC strongly emphasize the use of non-equispaced knots. In view of the results presented here quantile knots prevent the penalty from doing the best it can.

In theory, equally spaced knots carry the danger of ill numerical condition with them. Without a penalty, regression on the B-spline or TPF basis will generally lead to singular equations. The penalty removes the singularity, but still the condition might be poor when the data show a large gap. In practice this danger can be neglected, as a numerical experiment shows. We simulate 100 equally observations on the domain from 0 to 1 and leave out a gap in the middle; see Figure 9. The ratio of the largest to the smallest singular value of of the augmented matrix $[B'\sqrt{\lambda}D]'$ is computed as the condition number. For very small and very large values of λ this number can get larger than 10^6, but it is very improbable that a very small value of λ will be used in practice. The cross-validation criterion in the lower left graph in Figure 9 clearly point to a value in the neighborhood of $\lambda = 10$. The upper right graph shows data and fit for this value of λ. If λ is chosen too small, a strong overshoot can occur, as the lower right graph shows.

7 Computational aspects

In principle TPF can be used directly as a basis for regression. This is not to be recommended, as their numerical condition can be poor, especially when the number of knots is large and $p \geq 1$. The logarithm (base 10) of the condition number (the ratio of the maximum to minimum singular values) approximately indicates the number of significant numbers that can be lost when solving regression

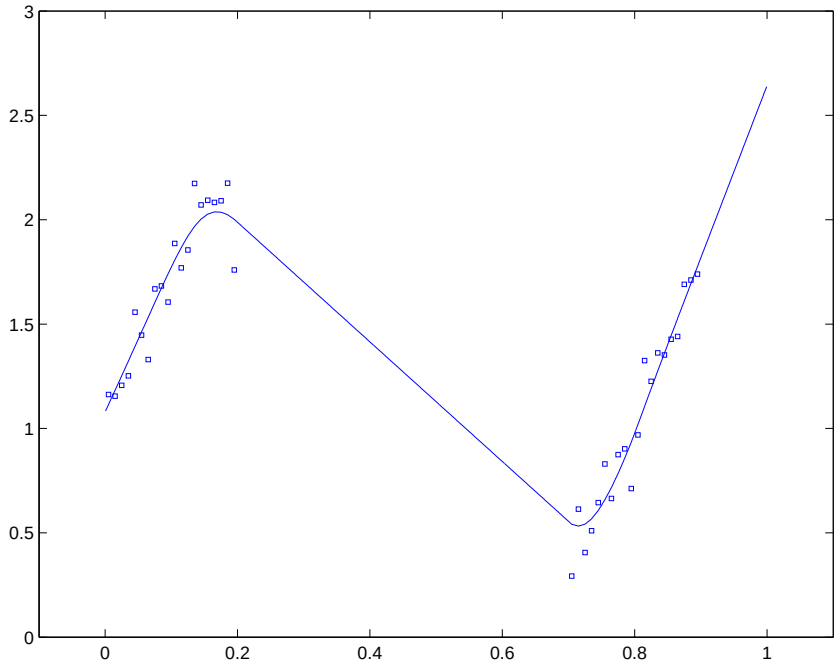

Figure 8: Smoothing and interpolation of simulated data with a basis of linear truncated power functions, with knots at unique values of x. Ridge penalty with $\kappa = 0.1$.

problems with this basis. This holds if the computations are not organized carefully, using a QR decomposition or Householder rotations. If, however, inner products are involved, double the number of digits can get lost (Golub and Van Loan, 1989). It is considered good practice to avoid condition problems in statistical computations.

Figure 10 shows condition numbers for TPF bases of degrees one, two and three, with several different numbers (n) of equispaced knots and different sample sizes (m) of equispaced x. The condition number strongly increases with the number of knots, but is relatively unaffected by changes in sample size. If smoothing is the only goal, TPF of degree 1 might be workable, but higher degrees, which are needed when one wishes to smoothly estimate (second) derivatives are essentially unusable. This is noted by RWC : they trust on the sophistication of existing functions or procedures in commercial statistical software and use the singular value decomposition to stabilize their own algorithms.

In very large problems extra care is needed when constructing a B-spline basis. Consider an engineering application from our own consulting experience, in which long series (15000 observations) of echo-sounding measurements must be smoothed and reduced to a uniform grid, where we used bases with 2000 B-splines. The sparse matrix abilities of Matlab were very useful in this setting. Of course, it is not a good idea to first fill a matrix of this size (30 million elements, or 240 Mb) with a TPF basis and then compute differences to reduce it to a sparse matrix. A nice property of B-splines with equispaced knots is that we get the same set of values, but shifted by k columns, if we shift x by (integer) k times the knot distance. Hence we can reduce all x to the interval between one chosen pair of knots, compute the basis in a matrix with $p+1$ columns, and transfer the results to the right columns of a sparse matrix.

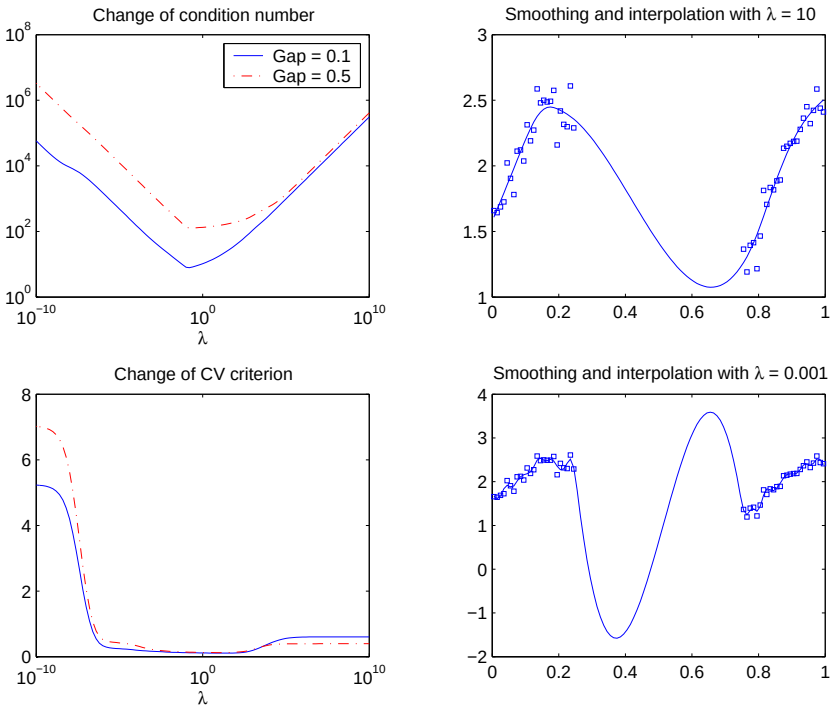

Figure 9: An illustration of optimal smoothing and interpolation with many B-splines and a large gap. The upper left panel shows the numerical condition, and the lower left panel the leave-one-out cross-validation profile. The panels to the right shows results of smoothing for the approximately optimal λ and for small λ, the latter showing overshoot. The number of cubic B-splines is 53 and the order of the penalty is 2.

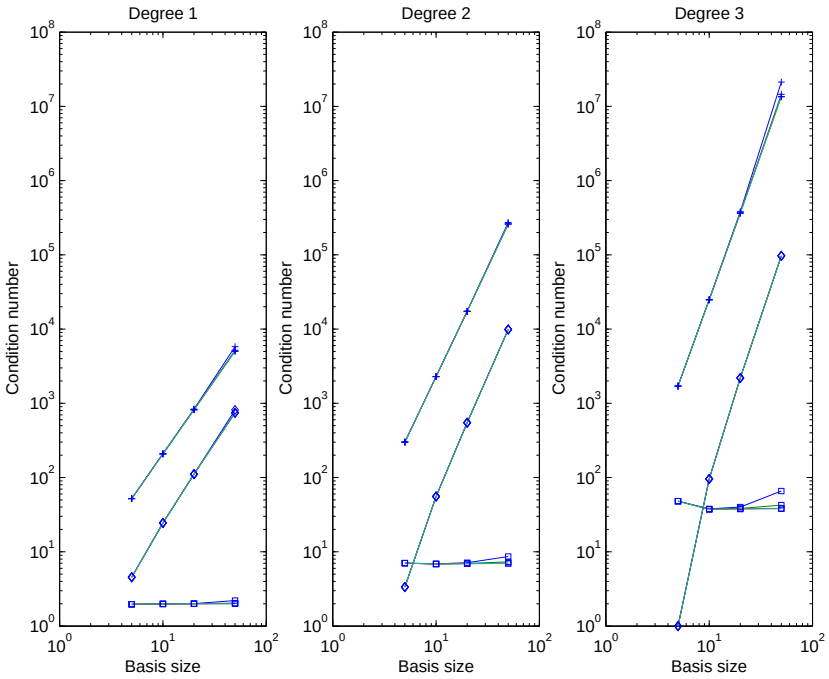

Figure 10: Condition numbers of three types of bases: truncated power functions (crosses), B-splines (squares) and Z-matrices for mixed models (diamonds). The sample sizes are 100, 200, 500 or 1000, but this difficult to see, because the lines and symbols overlap strongly.

8 Derivative estimation

Frequently one is not only interested in a fitted curve, but also in its derivatives. For example, in mechanics one might want to estimate velocity and acceleration from position measurements. Specifically in studies of human growth, one may be interested in growth spurts, which are characterized by first and second derivatives of the height of an individual.

Penalized splines allow easy calculation of derivatives. This is clear for TPF: one only needs to differentiate the polynomial branches and sum them, weighted by the estimated coefficients. For B-splines the situation is a little more complicated. However if the knots are equally-spaced, there is a simple explicit formula to compute the derivative of a weighted sum of B-splines (de Boor, 2001):

$$\frac{d}{dt}\sum_k B_k(t;p)\alpha_k = \sum_k (p-1)B_k(t;p-1)\Delta\alpha_k/h. \tag{13}$$

Suppose we are interested in a curve of the second derivative. This means that minimally we like to see a piecewise linear result. It follows that the TPF or B-spline basis has to be cubic. When substantial interpolation is involved extra care is needed. If the interpolating curve has degree $2d-1$, its second derivative has degree $2d-3$. For a better than linear result, $d=3$ should be the minimum.

As was discussed in the previous section, cubic TPF have a very poor condition number, so great attention is required with the implementation of computations for derivative estimation. No such problems occur with B-splines.

9 Discrete smoothing

In many cases one has no need to interpolate with B-splines, because the data are a discrete series, sampled at equal distances, and only a smoothed discrete series is needed. Time series and spectra are typical examples. In such a setting a B-spline basis of degree zero, with a knot at every observation, may be an attractive choice. The basis then is the identity matrix, the system of equations becomes $(I + \lambda D_d' D_d)\hat{\alpha} = y$ and the coefficients α and the smooth series μ coincide. This brings us back full circle to Whittaker (1923), who used this approach to smooth life tables.

The discrete smoother is very attractive for long data series, provided that one has access to sparse matrix software. For example, a series of a hundred thousand observations can be smoothed in a few seconds, including leave-one-out cross-validation. See Eilers (2003) for details. Of course, sparseness is essential in such large-scale applications and TPF will not work in such settings.

A smaller-scale application in which the discrete approach may be appropriate is histogram smoothing. To estimate a density one constructs a histogram with many, say 200, narrow bins. Such a plotted histogram will look unappealing and uninformative, but generalized linear smoothing with penalized splines completely changes presentation. In a Poisson regression setting, Eilers and Marx (1996) presented histogram smoothing with PB-splines and showed that narrow bins are no problem. The number of knots is essentially immaterial, as long as it is large enough (Ruppert, 2002), hence in the limiting case of one knot per bin we still get a nicely smoothed histogram. To conserve the variance, it is advisable to use a third order penalty.

10 Multidimensional smoothing

Tensor products of B-splines are a natural and attractive choice for smoothing in two dimensions and higher (de Boor, 2001; Dierckx, 1998). However, without a penalty unpleasant surprises can occur when the data are not rather evenly distributed on the domain of the independent variables. Even in two dimensions one frequently encounters empty corners. In such a situation, there is little or no support for some of the tensor products and a singular or ill-conditioned system of normal equations will result. The fitted surface then will show wild fluctuations at the borders, or cannot be estimated at all. With penalties these problems disappear, like they did for interpolation and extrapolation in one dimension.

If the data are triples (x_i, y_i, z_i) for $i = 1, \ldots, m$ and the tensor products are written as $B_j(x_i)\tilde{B}_k(y_i)$, then the fitted surface can be expressed by

$$\hat{y}_i = \sum_j \sum_k B_j(x_i)\tilde{B}_k(y_i)\alpha_{jk},$$

and the coefficient matrix $A = [\alpha_{jk}]$ summarizes the fit ($j = 1, \ldots, J$ and $k = 1, \ldots, K$). A natural choice for the difference penalties is to have two sets of them: one set "vertical", working on each column of A, the other set "horizontal", working on each row. The weights of the two sets can be quite different, especially when the two independent variables have different meanings. An example, in the context of two-dimensional signal regression is presented by Eilers and Marx (2003). They estimate a regression coefficient surface along the dimensions temperature

and wavelength. Along temperature a heavy penalty is optimal, giving a nearly linear variation. Yet the penalty allows for general interaction and the linear slopes varied considerably across the wavelength index. For wavelength a much lighter penalty is indicated, to allow a complex variation along this variable. A ridge penalty does not allow anisotropy: it has the same weight for both dimensions. Tensor product PB-splines with a Poisson response have been used successfully to smooth and extrapolate large mortality tables (Durbán, Currie and Eilers, 2002; Currie, Durbán and Eilers, 2003, 2004).

Successful application to large real-world problems, like image smoothing, has shown us that tensor products of B-splines and difference penalties are a practical and effective tool. In contrast, RWC report insurmountable problems with tensor products of PT-splines, using TPF. As a remedy they proposed a rather complicated algorithm with space-filling knots and radial basis functions. Of course, radial basis functions are very similar to B-spline tensor products.

For two-dimensional smoothing of scattered observations the data have to be strung out into vectors to implement the computations in efficient matrix-vector operations. For very large data sets this may lead to memory problems. If the data come on a grid, like images, a very fast, small-footprint, algorithm is available, that avoids the vectorization step (Eilers, Currie and Durbán, 2006). This algorithm allows arbitrary weights and so can handle GLM-like iterative fitting and "holes" in the data (if they are given zero weights). It can also be used for scattered data if one is willing to accept discretization of the independent variables to a fine grid — which is reasonable in the context of smoothing. The algorithm can be extended straightforwardly to grids in higher dimensions.

11 Optimal smoothing, mixed models and Bayes

Given a weight of the penalty, the solution of the penalized least squares problem is straightforward. Iterations are needed with a generalized linear component, but these tend to be well-behaved. This may be enough, because the penalty weight has been set beforehand, or some trial and error with visual examination can be sufficient. In many cases however, one will want to use the data to determine λ or κ. Three general strategies are available: 1) optimize a performance criterion, such as cross-validation or an information criterion; 2) apply a mixed model setting; or 3) use Bayesian technology.

EM exclusively advocate cross-validation or the use of AIC, and this in a rather primitive way: change the penalty parameters on a "nice" grid and, depending on the criterion, search for the minimum or maximum measure of performance. The grid is usually linear on a logarithmic scale. In one dimension this recipe is quite effective, because in practice there is no need to determine λ up to many decimal places. In more dimensions the amount of work increases rapidly (although still reasonably in the light of many thousands of model fits that are accepted routinely in MCMC or similar Bayesian methods). There is much room for improvement here, either using more advanced search methods, like the simplex, or with Newton algorithms, like those Wood (2000) uses for thin-plate splines.

EM ignore the connection between penalties and mixed models. RWC explicitly discuss this. If we write (5) as $y = X\beta + Fb + e$, we recognize a mixed model with fixed part $X\beta$, random part Fb and error e. The variance components are $\sigma^2 = \text{var}(e)$ and $\tau^2 = \text{var}(b)$ and in (7) we recognize the mixed model equations, with $\kappa = \sigma^2/\tau^2$.

The beauty of this approach is that one can use existing mixed model software for estimation. In practice extra work is needed, as the poor numerical condition of the TPF basis can lead to instabilities. One solution is to compute the singular value decomposition of F, and deflate the singular vectors that correspond to very small singular values.

We remark that mixed models are attractive when the response is Gaussian. This is not necessarily the case with non-Gaussian data, for which robust mixed model software is scarce. See the introduction of Zhao et al. (2006). Software for generalized linear mixed models is less generally available and relies on approximations that do not work well with small numbers of observation (Poisson counts or binomial denominators). AIC and a grid search might be a competitive choice. Both P-splines and TPF are confronted with this problem, so we do not investigate it further here.

We like to emphasize that there is nothing natural or sacred about mixed models in the context of smoothing, but the existence of robust mixed model software on several platforms makes it an excellent choice. Good technology for estimating variance components is available, so one does not have to bother writing new algorithms.

TPF are a natural basis for mixed models, but not the only possible choice. Eilers (1999) proposed to use

$$\mu = B\alpha = X\gamma + Za, \tag{14}$$

with $Z = BD_d'(D_dD_d')^{-1}$, a transformation of the B-spline basis B. As Figure 10 shows, the matrix Z has a somewhat better numerical condition than the TPF basis F.

RWC give a clear account of an effective Bayesian approach if the response is Gaussian. One cycles between 1) sampling new coefficients from a multivariate normal distribution, given variances σ^2 and τ^2; 2) sampling σ^2 and τ^2 from univariate inverse Gamma distributions, respectively involving the number residuals and the number of the TPF coefficients. This approach is directly transferable to B-splines and a difference penalty, with the second inverse Gamma distribution determined by $||D_d\alpha||^2$. Lang and Brezger (2004) adopted an alternative approach, using (integrated) random walk priors and implemented it in the *BayesX* software.

Here too, a non-Gaussian response can complicate matters appreciably, because simulation of new proposals for the coefficients is no longer straightforward. Zhao et al. (2006) report tests of several sampling algorithms.

Our experience shows that a simple hybrid between the Bayesian approach and mixed models can be quite effective. Consider smoothing of Gaussian data. In the mixed model sense, λ can be interpreted as the ratio between the variance of $y - B\alpha$, σ^2, and the variance of $D\alpha$, τ^2. The first of these can be estimated as $\hat{\sigma}^2 = ||y - B\hat{\alpha}||^2/(m - ED)$ and the second as $\hat{\tau^2} = ||D\hat{\alpha}||^2/ED$, where the effective dimension is taken to be (Hastie and Tibshirani, 1990)

$$ED = \mathrm{tr}[(B'B + \lambda D'D)^{-1}B'B].$$

One repeats smoothing, estimating the variances and recomputing λ until convergence. This generally occurs surprisingly fast, often in less than 10 iterations. To avoid divergence in cases where strong smoothing is indicated, we suggest to use $\lambda = \hat{\sigma}^2/(\hat{\tau}^2 + \epsilon\hat{\sigma}^2)$, with ϵ a small number like 10^{-8}. This approach is related to

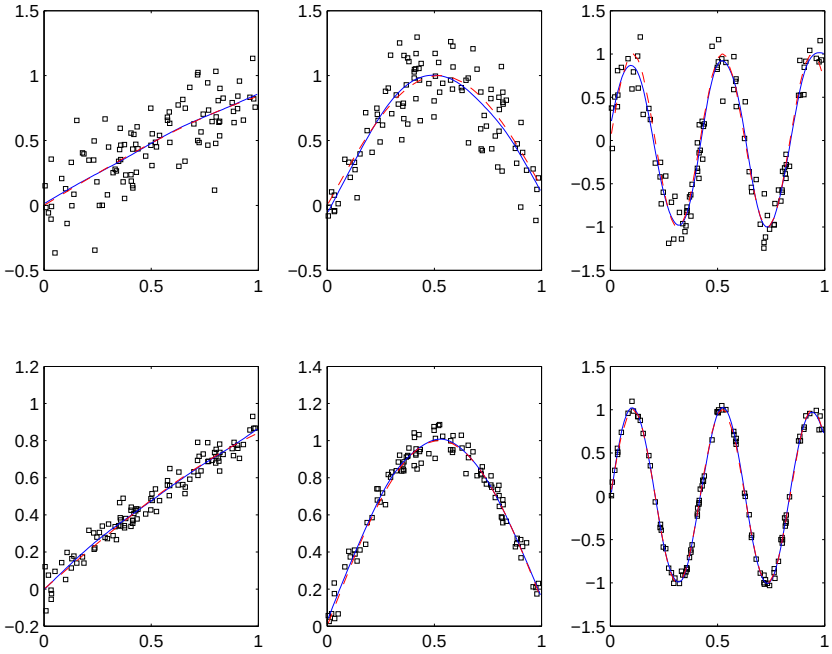

Figure 11: Automatic choice of the smoothing parameter with the hybrid algorithm for variance estimation. Simulated data. Broken line: true curve; full line: automatically estimated smooth.

the work of Schall (1991). Figure 11 illustrates this procedure for estimation of the smoothing parameter.

An intriguing property of this approach is that it avoids the introduction of "fixed" components as well as problems with singular priors. Through a pure random component model, the variance of the well-defined contrast $D\alpha$ plays the central role.

This procedure also applies to non-Gaussian smoothing with the Poisson or binomial distribution: simply use $\lambda = (||D\hat{\alpha}||^2/ED)^{-1}$, where now, following EM,

$$ED = \mathrm{tr}[(B'WB + \lambda D'D)^{-1}B'WB],$$

where the weights in W follow from the iterative re-weighted penalized least squares algorithm for smoothing of non-normal data with PB-splines.

12 Periodic smoothing

Periodic data are common, because of natural (daily, yearly, lunar) or social (weekly, monthly) or cycles, or because of harmonic vibrations (periodic stars, radio signals). When smoothing such data on a linear axis, it may happen that both ends do not join smoothly. This can be avoided by using a suitable periodic basis. With B-splines this is quite easy: simply wrap around the basis functions at the "end" to the "front". More specifically: a "linear" B-splines basis of degree p has $n + p$ basis functions. Keep the first n columns in a new matrix, say C, and add the last p columns of B to the first p columns of C. This procedure is illustrated by the perspective view in Figure 12.

The difference penalty has to be changed too. A simple solution is to wrap it around in the same way as for the B-spline basis. An improvement is to use

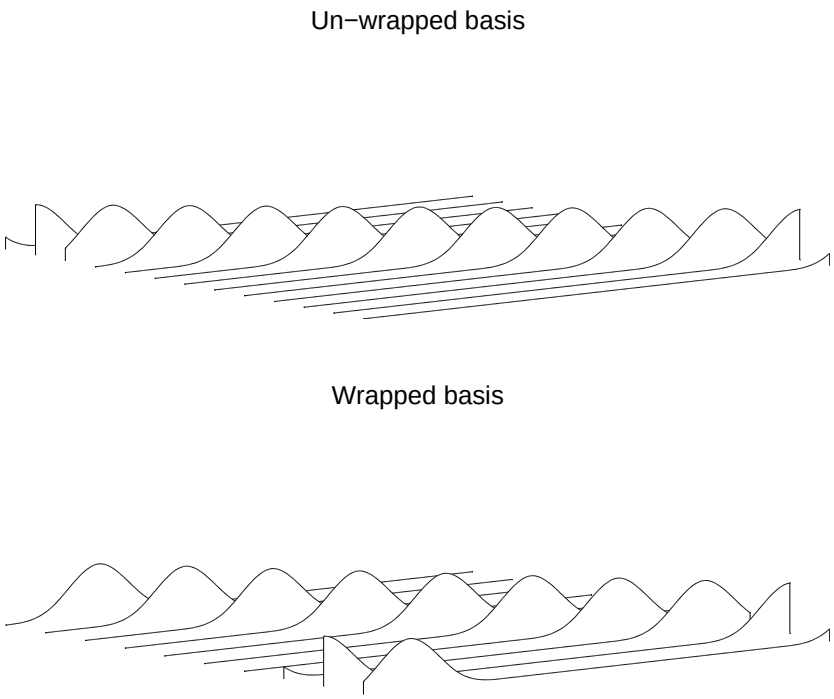

Figure 12: A standard B-spline basis (top) and the corresponding wrapped basis.

$\sum(\alpha_j - 2\phi\alpha_{j-1} + \alpha_{j-2})$, also with proper wrapping at the ends and $\phi = \cos(2\pi/n)$, where n is the size of the B-spline basis. The limit for strong smoothing will then be $\hat{\alpha}_j = c_1 \cos(2\pi j/n) + c_2 \sin(2\pi j/n)$, with c_1 and c_2 determined from the least squares fit to the data.

If the data have a mean that is not near zero, the sine/cosine limit might not fit well. A simple change to the penalty will solve this. Denote the corresponding "differencing" matrix by Φ and let D_1 be the (wrapped) matrix that forms first differences. Use the penalty $\lambda||D_1\Phi\alpha||^2$.

13 Designer penalties

In the preceding section we have already seen a "designer penalty", a change from the usual finite difference scheme to a more general sum of adjacent coefficients. Interesting variations on this theme are possible. In their rejoinder, EM introduced the "periodic penalty" for smoothing of a time series of observations on a periodic star. The limiting behavior for strong smoothing is of special interest. A similar example is the penalty $\sum(\phi\alpha_j - \alpha_{j-1})^2$, which in the limit gives $\alpha_j = c\phi^j$, with c a constant determined by (the best least squares fit to) the data.

An area that deserves much more research is weighting of the differences in the penalty, to get $\alpha' D' V D \alpha$, with $V = \text{diag}(v)$ and v having (unknown) non-negative elements. Ruppert and Carroll (2000) pioneered this in a general way for the ridge penalty, giving a recipe to estimate a general v from the data. Their approach is directly applicable to B-splines with difference penalties.

Less ambitious goals, or a pre-specified v can be quite useful too. Take as an example second order differences and a vector v of all ones, except for a zero in

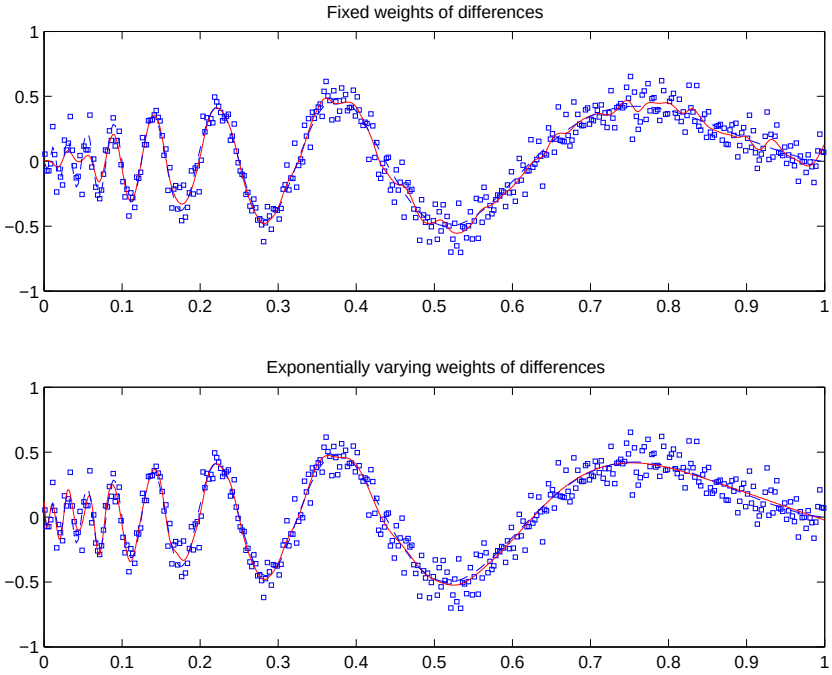

Figure 13: Smoothing of simulated data (dots) with and without exponentially varying weights on the differences in the penalty. Upper: uniform weights; lower: varying weights. Parameters optimized with grid search and leave-one-out cross-validation. Full line: fitted curve (100 cubic B-splines, second order penalty); broken line: true curve.

position k. It follows, for large λ, $\hat{\alpha}$ will be a smooth series, except for a kink at position k. If both v_k and v_{k-1} are zero, $\hat{\alpha}$ will be smooth, except for a jump at α_k. Depending on the number of knots, the kink or jump will show up in a more or less smoothed way in the fitted curve. Of course, combinations of multiple kinks and jumps can be introduced this way.

In some application a gradually changing smoothness may be sufficient. This can be accomplished by taking $v_k = e^{\gamma k}$. Both λ and γ are optimized by cross-validation or AIC. Of course, this applies equally well to TPF. An example of smoothing with an exponential change of the weights in the penalty is shown in Figure 13, using simulated data: a sine function with changing frequency and amplitude. If we use uniform weights and optimize λ with leave-one-out cross-validation (which gives optimal $\lambda^\star = 0.1$), we get a result that gives rather strong fluctuations of the fitted curve in the low-frequency part and misses the data in the high-frequency part. If we introduce weights $e^{\gamma k}$ and optimize both γ and λ, we get a more reasonable result. A grid search gave (approximate) optimal values $\gamma^\star = 0.2$ and $\lambda^\star = 3 \times 10^{-4}$. This means that, with the 100 knots used here, the largest weight is about 5×10^8 times larger than the smallest.

Sometimes it is fruitful to have multiple difference penalties, of different orders, or to add an extra ridge penalty. Marx and Eilers (2002) found, in the context of multivariate calibration by penalized signal regression, markedly improved cross-validation behavior. Aldrin (2006) investigated the use of both first and second order penalties in additive models based on P-splines, and found improved prediction.

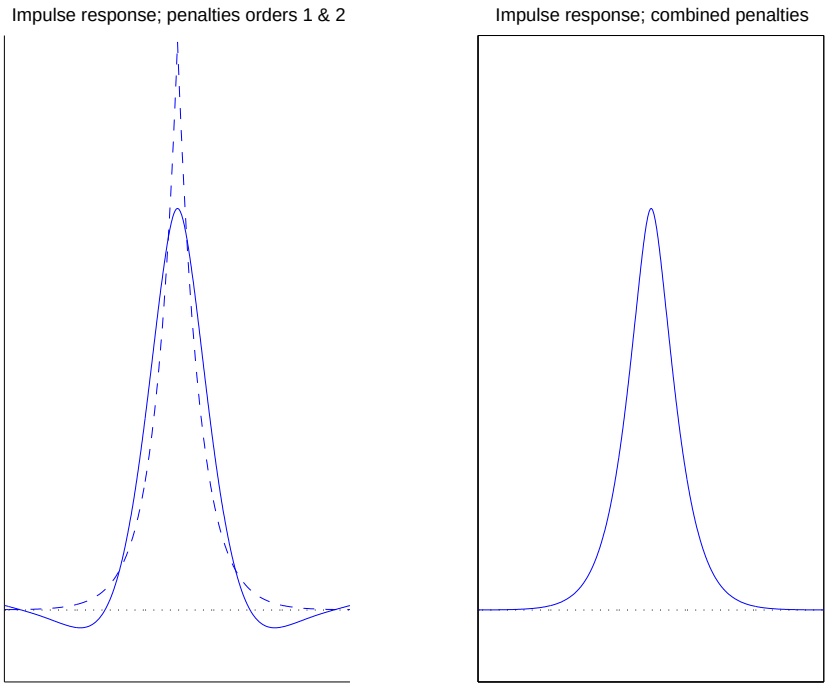

Figure 14: Left: impulse response of a P-spline smoother with a only a first or second order difference penalty; right: impulse response of a P-spline smoother with second *and* first order penalties: pen $= \lambda||D_2\alpha||^2 + 2\sqrt{\lambda}||D_1\alpha||^2$.

A combination of second and first order can prevent the impulse response of the smoother from becoming negative. This can be useful when smoothing signals that consist of a rather flat baseline and sharp pulses (plus noise), or when smoothing a histogram (without going to the generalized linear approach). With only a second order penalty negative excursions can occur, which are unattractive when the context demands positive results. If the impulse response is positive everywhere, the results of smoothing non-negative data can never become negative, by virtue of the linearity of the PB-spline smoother. Figure 14 shows examples of the impulse response with penalty $\lambda||D_2\alpha||^2$ and with the penalty $\lambda||D_2\alpha||^2+2\sqrt{\lambda}||D_1\alpha||^2$. The latter combines the rounded top from the second order penalty with the positive tails from the first order penalty.

Other useful application of combined first- and second-order penalties are found in interpolation and extrapolation. With only a second order penalty, under certain circumstances the interpolating curve can make rather enthusiastic sweeps, as Figure 15 shows. With the penalty matrix $\lambda||D_2\alpha||^2 + \gamma\sqrt{\lambda}||D_1\alpha||^2$ we can tune this down by giving γ a (small) positive value. The interpolating curve now is essentially a sum of linear and exponential functions, and extrapolation is by an exponential function to a constant level. The large γ, the faster the exponentials decay. This is the PB-spline analog of exponential splines.

14 Discussion

We have compared two approaches to penalized spline smoothing: PB-splines (B-splines with a difference penalty) and PT-splines (truncated power functions (TPF)

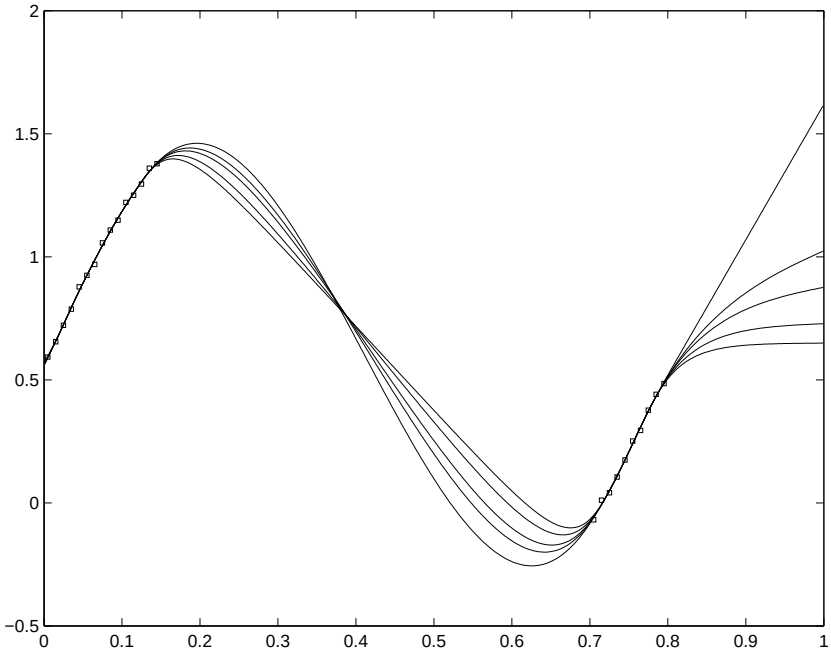

Figure 15: Interpolation and extrapolation with a combination of penalties of first and second order: pen $= \lambda||D_2\alpha||^2 + \gamma\sqrt{\lambda}||D_1\alpha||^2$. The values of γ are 0, 0.01, 0.02 0.05 and 0.1, A larger γ, gives a tighter curve.

with a ridge penalty). We found that:

- In our practical experience, equally-spaced knots are always to be preferred.
- A ridge penalty on TPF is equivalent to a difference penalty of (fixed order) on B-splines; B-splines allow a flexible choice of the order of the penalty.
- B-splines can be computed almost trivially from TPF.
- Both bases allow a mixed model approach. In addition, PB-splines can be written as a pure random component model, simplifying the computations.
- Equally-spaced knots allow easy smooth interpolation with both TPF and B-splines;
- TPF bases have poor numerical properties, while B-splines have excellent numerical properties.
- B-splines allow informative visualization.
- B-spline bases are sparse and lend themselves well to large-scale problems.
- B-spline tensor products and difference penalties are powerful tools for multi-dimensional smoothing. In contrast to a ridge penalty they allow a different weights for each dimension (anisotropic smoothness).
- B-splines and difference penalties are easily adapted to smoothing of periodic data.

- Designer penalties, generalizations of difference penalties that make the smooth curve approach special (exponential or periodic) limits, are easily implemented in the B-spline framework.

In light of the above list, we are not aware of any advantage of PT-splines, or TPF (with non-uniform knots) and the ridge penalty, over the original P-splines or PB-splines (B-splines with difference penalties). However, if truncated power functions are to be chosen, then we recommend to use equally-spaced knots.

Acknowledgements Research supported in part for Brian Marx by NSF Grant DMS-0102131, and for Paul Eilers in part by the Spanish Ministry of Science and Innovation (project MTM 2008-02901).

References

Aldrin, M. (2006). Improved predictions by penalizing both slopes and curvature in additive models. *Computational Statistics and Data Analysis* **50**, 267–284.

Currie, I., Durbán, M., and Eilers, P.H.C. (2003). Using P-splines to extrapolate two-dimensional Poisson data. In: Proceedings of the 18th International Workshop on Statistical Modelling. Leuven, Belgium. Eds. G. Verbeke, G. Molenberghs, A. Aerts, and S. Fieuws, 97-102.

Currie, I., Durbán, M., and Eilers, P.H.C. (2004). Smoothing and forecasting mortality rates. *Statistical Modelling* **4**, 279–298.

de Boor, C. (2001). *A Practical Guide to Splines.* Revised edition. Applied Mathematical Sciences **27**. Springer-Verlag, New York.

Dierckx, P. (1993). *Curve and Surface Fitting with Splines.* Clarendon Press, Oxford.

Durbán, Currie, I., and Eilers, P.H.C. (2002). Using P-splines to smooth two-dimensional Poisson data. In: Proceedings of the 17th International Workshop on Statistical Modelling. Chania, Greece. Eds. M. Stasinopoulos and G. Touloumi, 207-214.

Eilers, P.H.C. (1999). Discussion of: Kempton R., Mead R., Engel B., et al. The analysis of designed experiments and longitudinal data by using smoothing splines. *Journal of the Royal Statistical Society Series C* **48**, 300–311.

Eilers, P.H.C. (2003). A perfect smoother. *Analytical Chemistry*, **75**, 3631–3636.

Eilers, P.H.C., Currie, I.D. and Durbán, M. (2006) Fast and compact smoothing on multi-dimensional grids. *Computational Statistics and Data Analysis* **50**, 61–76.

Eilers, P.H.C. and Marx, B.D. (1992). Generalized linear models with P-splines. In: Proceedings of GLIM 92 and 7th International Workshop on Statistical Modelling, Munich, Germany. Lecture Notes in Statistics, Vol. 78, Advances in GLIM and Statistical Modelling, Eds. L. Fahrmeir, B. Francis, R. Gilchrist, G. Tutz. Springer-Verlag, New York, 72–77.

Eilers, P.H.C. and Marx, B.D. (1996). Flexible smoothing using B-splines and penalized likelihood (with Comments and Rejoinder). *Statistical Science* **11**(2), 89–121.

Eilers, P.H.C. and Marx, B.D. (2002). Generalized linear additive smooth structures. *Journal of Computational and Graphical Statistics* **11**(4), 758-783.

Eilers, P.H.C. and Marx, B.D. (2003). Multivariate calibration with temperature interaction using two-dimensional penalized signal regression. *Chemometrics and Intelligent Laboratory Systems*, **66**, 159-174.

Gasson, P.C (1983) *Geometry of Spatial Forms.* Ellis Horwood.

Golub. G.H. and Van Loan, C.F. (1989) *Matrix Computations.* The Johns Hopkins Press, Baltimore.

Hastie, T. and Tibshirani, R. (1990) *Generalized Additive Models.* Chapman and Hall, London.

Lang, S. and Brezger, A. (2004) Bayesian P-Splines. *Journal of Computational and Graphical Statistics* **13**, 183–212.

Marx, B.D. and Eilers, P.H.C. (1998). Direct generalized additive modeling with penalized likelihood. *Computational Statistics and Data Analysis* **28**, 193-209.

Marx, B.D. and Eilers, P.H.C. (1999). Generalized Linear Regression on Sampled Signals and Curves: A P-Spline Approach. *Technometrics* **41**, 1-13.

Marx, B.D. and Eilers, P.H.C. (2002). Multivariate calibration stability: a comparison of methods. *Journal of Chemometrics* **16**, 129–140.

Marx, B.D. and Eilers, P.H.C. (2005). Multidimensional penalized signal regression. *Technometrics* **47**, 13–22.

O'Sullivan, F. (1986). A Statistical Perspective on Ill-Posed Inverse Problems (with Discussion). *Statistical Science* **1**, 505-527.

Ruppert, D. (2002). Selecting the number of knots for penalized splines. *Journal of Computational and Graphical Statistics* **11**(4), 735-757.

Ruppert, D. and Carroll, R.J. (2000). Spatially-adaptive penalties for spline fitting. *Australian and New Zealand Journal of Statistics* **42**, 205-223.

Ruppert, D., Wand, M.P., and Carroll, R.J. (2003). *Semiparametric Regression*. Cambridge University Press, New York.

Ruppert, D., Wand, M.P., and Carroll, R.J. (2009). Semiparametric regression during 2003–2007. The Electronic Journal of Statistics. **3**, 1193–1256.

Schall, R. (1991). Estimation in generalized linear models with random effects. *Biometrika*, **78**, 719–727.

Wand, M.P. and Ormerod, J.T. (2008). On semiparametric regression with OSullivan penalized splines. *Australian and New Zealand Journal of Statistics* **50**, 179-198.

Whittaker, E.T. (1923). On a new method of graduation. Proc. Edinburgh Math. Soc. **41**, 63-75.

Wood, S.N. (2000) Modelling and smoothing parameter estimation with multiple quadratic penalties. *Journal of the Royal Statistical Society B* **62**, 413–428.

Zhao Y., Staudenmayer J., Coull B.A. and Wand M.P. (2006). General design Bayesian generalized linear mixed models. *Statistical Science* **21**, 35–51.