Smooth supersaturated models

Ron A. Bates [a], Hugo Maruri-Aguilar [b,*], Henry P. Wynn [a]

[a] *Department of Statistics, London School of Economics, London WC2A 2AE, UK*

[b] *School of Mathematical Sciences, Queen Mary University, Mile End E1 4NS, UK*

Abstract

In areas such as kernel smoothing and non-parametric regression there is emphasis on smooth interpolation and smooth statistical models. Here we concentrate on pure interpolation. Splines are known to have optimal smoothness properties in one and higher dimensions. It is shown that smooth polynomial interpolators can be constructed by first extending the monomial (polynomial) basis and then minimising a measure of roughness with respect to the free parameters in the extended basis. Algebraic methods are a help in choosing the extended basis, which can also be found as a saturated basis for an extended experimental design with dummy design points. One can get arbitrarily close to optimal smoothing for any dimension and over an arbitrary region, giving simple alternative models arbitrarily close to splines. Examples show that the interpolators do much better than straight polynomial fits and for small sample size perform better than kriging-type methods. The tractability of their polynomial forms points to fruitful areas of research.

Key words: regression, splines, kernel smoothing, non-parametric regression, computer experiments, algebraic statistics.

1 Introduction

There is a considerable literature on smooth interpolation and its statistical counterpart, for example in non-parametric regression. The optimal smoothness properties of splines have a substantial literature. The main optimality result for one dimension is attributed to Holladay [1957] and for two dimensions, where thin-plate splines are optimal, to Duchon [1976]; see Kimeldorf and Wahba [1970] and Micula [2002] for reviews of spline optimality.

* Corresponding author.
Email address: H.Maruri-Aguilar@qmul.ac.uk (Hugo Maruri-Aguilar).

2 September 2010

In computer experiments, Bayesian kriging using Gaussian kernel stochastic process models has been preferred to splines, Sacks et al. [1989], Kennedy and O'Hagan [2001], Kleijnen [2009] and have also become popular in machine learning, see Rasmussen and Williams [2005]. Of course, the connection between kriging and splines is thoroughly researched and, for example, splines can arise as kriging (conditional expectation) interpolators for special Gaussian stochastic processes, see Kimeldorf and Wahba [1970].

Raw polynomial interpolation is known in general not to have optimal rates of interpolation unless special sampling (design) points are used such as in Tchebychev approximation. On the other hand the *existence* of polynomial interpolators over an arbitrary design is at the core of the newer theory of "algebraic statistics": for any arbitrary design in d dimensions there is always a monomial basis out of which we can build a polynomial interpolator. This was introduced into statistics by Pistone and Wynn [1996], covered at length in the monograph Pistone et al. [2001] and was also the basis for Bates et al. [2003] which can be seen as the forerunner of the present paper.

The basic idea of this paper may seem at first to be somewhat contradictory. We start with a given polynomial interpolator and by extending the basis make the interpolator smoother. Although there is a tendency to associate higher order polynomial terms with lack of smoothness, we can, in fact, extend the basis and use the freedom this gives to *increase* smoothness. It should be pointed out that the use of polynomials to build kernels with pre-specified properties is familiar in signal processing, see Lin et al. [2004]. The algebraic method referred to above, is used here to help clarify the extension of the basis to higher degree monomials.

1.1 An introductory example

The Lagrange interpolator of the three points $(x, y) = (0, 1), (\frac{1}{2}, 3), (1, 2)$ is the quadratic:

$$y(x) = 1 + 7x - 6x^2.$$

The (average) roughness of $y(x)$ over $[0, 1]$ is, according to the criteria we shall use in the paper,

$$\Psi_2 = \int_0^1 \left(\frac{d^2 y(x)}{dx^2} \right)^2 dt = 144.$$

Now, consider a quartic interpolator which interpolates the same points but also two additional points $(2, s), (3, t)$. We may call s, t "dummy" values. The quartic interpolator is a function of (s, t) and so, therefore, is the roughness Ψ_2. In fact, Ψ_2 a is quadratic function of (s, t) and we may minimise it precisely. The minimal value is $\frac{768}{7} = 109.714 < 144$, which is achieved at $(s, t) = (\frac{117}{7}, \frac{1276}{7})$. This gives the following quartic interpolator which is smoother than $y(x)$:

$$\tilde{y}(x) = 1 + \frac{39}{7}x + \frac{8}{7}x^2 - \frac{80}{7}x^3 + \frac{40}{7}x^4.$$

We note that if we replace the extra points $x = 2, 3$ by any other points (distinct from $\{0, 1/2, 1\}$) we obtain the same interpolator. This is because it is the extension of the *basis*, which determines the method. We shall see that for larger problems we obtain very substantial increases in smoothness by increasing the basis and typically achieve a large decrease in integrated squared error.

1.2 Monomial and extended bases

Recent work in the area of algebraic statistics shows how to construct estimable (identifiable) monomial bases for polynomial regression and we start with a very short description. The motivation is that we shall need an extended basis with certain conditions and the algebra is one way of achieving this.

We start with a set of factors $x = (x_1, \ldots, x_d)$. For a set of nonnegative integers $\alpha = (\alpha_1, \ldots, \alpha_d)$, a monomial, such as $x_1^2 x_2$, is written $x^\alpha = x_1^{\alpha_1} \cdots x_d^{\alpha_d}$, and a polynomial is a linear combination of monomials. A design D_n is a set of n distinct points in d dimensions, $D_n = \{x^{(1)}, \ldots, x^{(n)}\}$, $x^{(i)} \in \mathbb{R}^d, i = 1, \ldots, n$. This rather general definition of a design is familiar in computer experiments and spatial sampling, where good designs are chosen to cover the input space in some desirable fashion.

The algebraic methods give us the following: *given an experimental design*, D_n, *it is always possible to find a saturated non-singular monomial basis* $B_L = \{x^\alpha, \alpha \in L\}$. Thus, the size of the basis is equal to the size of the design $|L| = |D_n| = n$ and the $n \times n$ X-matrix, from the saturated regression model $X = \{x^\alpha\}_{x \in D_n, \alpha \in L}$ is non-singular. We call such a basis a *good saturated basis* for the design. The intuition behind the algebraic method is straightforward: terms are included in the good saturated basis according to a term ordering and a rank inclusion criterion. For details on term orderings see Cox et al. [1997], and for description of the algebraic method see Pistone et al. [2001].

Example 1 Let D_{24} to be the first 24 points of a bidimensional Sobol's space filling sequence. Sobol' sequence is a (multivariate) binary sequence, bitwise constructed with the aid of special binary generators called "direction numbers". We do not pursue here a detailed explanation of the construction of Sobol' sequence, which can be found in Bratley and Fox [1988]. This sequence is implemented in the `R` package `fOptions` through the function `runif.sobol`, see Ihaka and Gentleman [1996]. By selecting terms with a degree lexicographic term order $x_1 \succ x_2$, a good saturated basis with 24 monomials is identified for D_{24}. This model includes the monomials $x_2^6, x_1x_2^5, x_1^2x_2^4$ plus all the terms of a model of total degree five. This basis will be extended in the example of Section 3.3.

It will be critical in the development that we may extend a basis. By this we mean we keep the design D_n fixed but take a larger set of $N > n$ monomials, hence the term "supersaturated" in the title of the paper. We first require a condition contained in the following definition.

Definition 1 *(1) A finite set of monomials B is called a hierarchical basis if, for any monomial x^α in B, all its divisors are in B.*

(2) Given a design D_n, with sample size n, a good supersaturated basis *is a basis $B_M = \{x^\alpha, \alpha \in M\}$ with $|B| = N > n$ such that there is a hierarchical non-singular sub-basis of size n.*

As an example start with a rather poor design in two dimensions: $D_4 = \{(0,0),(1,1),(2,2),(3,3)\}$. It is straightforward to see that there are only two good saturated model bases $\{1, x_1, x_1^2, x_1^3\}$ or $\{1, x_2, x_2^2, x_2^3\}$. From this we can see that the extended basis $\{1, x_1, x_1^2, x_2, x_2^2\}$ with five terms is not useful as it has no good sub-basis of size four.

If we start with a non-singular hierarchical basis for a design D_n and extend it, in any way, then we always obtain a good supersaturated basis. But there is a revealing way of generating a good supersaturated basis, namely by extending the design D_n to a design D_N with N points and finding a good saturated basis for the larger design, which contains the good basis for D_n. The algebra method guarantees that this is always possible. This leads to a second, and equivalent, way of producing the smooth models which will be called the "dummy design" method, covered in sub-section 2.2. This is precisely the method we used in the introductory example.

2 Smooth interpolators

Let the experimental design be D_n and $y_1, \ldots, y_n$ be real values (observations) taken at the design points $x^{(i)} \in D_n, i = 1, \ldots, n$, respectively. Let B_M be a good supersaturated basis for the design D_n and let

$$y(x) = \sum_{\alpha \in M} \theta_\alpha x^\alpha \tag{1}$$

be a polynomial model in that basis. A good supersaturated model will be sought using a measure of roughness.

In one dimension ($d = 1$) we shall adopt the following measure of roughness based on the second derivative

$$\Psi_2 = \int_{\mathcal{X}} (y''(x))^2 dx, \tag{2}$$

where the integration is carried out over a desired region $\mathcal{X} \subset \mathbb{R}$. For higher dimensions the Hessian is

$$H(y(x)) = \left\{ \frac{\partial^2 y(x)}{\partial x_i \partial x_j} \right\},$$

and

$$\sum_{ij} \left(\frac{\partial^2 y(x)}{\partial x_i \partial x_j} \right)^2 = ||H(y(x))||^2 = \text{trace}\left(H(y(x))^2 \right). \tag{3}$$

Then define

$$\Psi_2 = \int_{\mathcal{X}} ||H(y(x))||^2 dx, \tag{4}$$

for some desired region $\mathcal{X} \subset \mathbb{R}^d$.

Smooth here means "having minimal roughness", so that a smooth interpolator is $\hat{y}(x) = \sum_{\alpha \in M} \hat{\theta}_\alpha x^\alpha$, where the coefficients $\hat{\theta}_\alpha$ are selected to minimise roughness subject to the interpolation condition, i.e. solving the constrained optimisation problem

$$\min_\theta \Psi_2(y(x)) \ \text{ subject to } \ y_i = \hat{y}(x^{(i)}),\ i = 1, \ldots, n \tag{5}$$

In the next subsection we give the solution of this constrained problem and in the second subsection the dummy design method. It is revealing that these two methods are equivalent.

2.1 The constrained problem

The main technical difficulty arises from the fact that linear parts of the model make no difference to the criterion Ψ_2 but nonetheless affect the interpolation. It is necessary to partition the X-matrix to take account of this.

Let $f(x)$ and θ respectively be the vectors which hold the good supersaturated basis and the parameters so that we can write (1) as $y(x) = \theta^T f(x)$. Denote $f^{(ij)} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}$ and define

$$K = \int_{\mathcal{X}} \left(\sum_{i,j=1}^{k} f^{(ij)} f^{(ij)^T} \right) dx. \tag{6}$$

Then we see that

$$\Psi_2(y(x)) = \theta^T K \theta. \tag{7}$$

The difficulty with linear terms mentioned above has the effect that K may not be full rank. In particular the constant and any linear term in the models basis will give zero entries. Call these entries *structural zeros*. Permute the rows and columns of K so that the structural zeros are adjacent:

$$K = \begin{bmatrix} 0 & 0 \\ 0 & \tilde{K} \end{bmatrix} \tag{8}$$

Let $X = [X_0, X_1]$, $f = (f_0^T : f_1^T)^T$ and $\theta = (\theta_0^T : \theta_1^T)^T$ be the corresponding rearranged and partitioned versions of X_n, f and θ, respectively. The matrix X has n rows and as many columns as terms in f. Let y be the column vector with n observations and note that $\Psi_2 = \theta_1^T \tilde{K} \theta_1$.

With this partitioning the constrained quadratic problem (5) is:

$$\min_{\theta} \ \theta_1^T \tilde{K} \theta_1 \quad \text{subject to} \quad X_0\theta_0 + X_1\theta_1 = y \tag{9}$$

Let 2λ be an $n \times 1$ vector of Lagrange multipliers (2 is for convenience) so that the Lagrangian is

$$\theta_1^T \tilde{K} \theta_1 - 2\lambda(X_0\theta_0 + X_1\theta_1).$$

After differentiation the full set of equations for θ_0, θ_1 and λ can be written in block form

$$\begin{bmatrix} X_0 & X_1 & 0 \\ 0 & \tilde{K} & -X_1^T \\ 0 & 0 & X_0^T \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \lambda \end{bmatrix} = \begin{bmatrix} y \\ 0 \\ 0 \end{bmatrix} \tag{10}$$

If the matrix on the left hand side of Equation (10) is nonsingular we obtain a unique solution $\hat{\theta}_0, \hat{\theta}_1, \hat{\lambda}$. The following three conditions are together sufficient for this.

(i) The full basis is a good supersaturated basis for D_n, so that X is full rank.

(ii) X_0 is full rank.

(iii) $\tilde{K}$ is full rank and thus invertible.

The full matrix inverse with solutions $\hat{\theta}_0, \hat{\theta}_1, \hat{\lambda}$ are given in Appendix 1. Finally, using these results, we express the smooth estimator as

$$\hat{y}(x) = \hat{\theta}_0 f_0 + \hat{\theta}_1 f_1 = \hat{\theta} f(x)$$

and the optimal Ψ_2 as

$$\Psi_2^* = \hat{\theta}_1^T \tilde{K} \hat{\theta}_1.$$

In applications, as is common with quadratic programming, we simply invert the matrix on the right hand side of (9) using a fast numerical method. Thus, given the design D_n, the good supersaturated basis and $\tilde{K}$, the method is fairly straightforward to implement.

It is revealing to consider the case where K is nonsingular. Then we do not need the partition of Equation (8) and instead can write Equation (10) as

$$\begin{bmatrix} X & 0 \\ \tilde{K} & -X \end{bmatrix} \begin{bmatrix} \theta \\ \lambda \end{bmatrix} = \begin{bmatrix} y \\ 0 \end{bmatrix},$$

which has the solution:

$$\hat{\theta} = (X^T X + K(I - P)K)^{-1} X^T y$$

where $P = X^T(XX^T)^{-1}X$ is the projector onto the row space of X. Thus, although X^TX is not invertible, because we have a supersaturated model, the second term $K(I - P)K$ on the left hand side can be seen as a smoothness induced regularisation of the problem which compensates for this singularity.

2.2 The dummy design method

For simplicity of development we assume that K is non-singular in the present case. Let D_N be a large design, with $N > n$ distinct points, which contains the original design D_n and write

$$D_N = D_n \cup D_q,$$

where $q = N - n$. Let $h(x)$ be a good saturated basis for D_n, and let $f(x)$ be an (extended) good saturated basis for D_N, $f(x) = (h(x)^T, g(x)^T)^T$. Also extend the observation vector to $z = (y^T, z^T)^T$ where, as before y holds the "true" observations taken at points in D_n, and z can be thought of as dummy observations on the design D_q, as in the introductory example. The extended model is written

$$y(x) = f(x)^T\theta = h^T(x)\beta + g^T(x)\gamma \tag{11}$$

and we assume, as in the last section, that $y(x)$ interpolates the observations y over D_n.

We now minimize Ψ_2 over the the choice of dummy observations z which is now an unconstrained optimization problem, but with a reduced set of free parameters, namely z. This is the procedure we used in the introductory example. The constrained optimization (9) and this unconstrained optimization (12) are equivalent in the case that the full basis is good for the full design,

D_N. This is because of the one-to-one correspondence between observations and parameters and the fact that the interpolation constraint is the same in both cases.

The unconstrained problem is:

$$\min_z \,(y^T : z^T) X_N^{-1^T} K X_N^{-1} \binom{y}{z}. \tag{12}$$

Where X_N is the X-matrix for the full large model $f(x)$. First, let the following matrix be partitioned according to the model bases $f(x) = (h(x)^T, g(x)^T)^T$:

$$A = X_N^{-1^T} K X_N^{-1} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}.$$

Then after expanding (12) and differentiating, the optimal z is $\hat{z} = -A_{22}^{-1} A_{21} y$ and the minimum roughness value is

$$\Psi_2^* = y^T Q \, y,$$

where $Q = A_{11} - A_{12} A_{22}^{-1} A_{21}$. The smooth interpolator is

$$\hat{y}(x) = f^T(x) X_N^{-1} \binom{y}{\hat{z}} = f^T(x) X_N^{-1} \binom{I}{-A_{22}^{-1} A_{21}} y = f^T(x) K^{-1} (X_{11} : X_{12}) Q y \tag{13}$$

where

$$X_N = \begin{pmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{pmatrix}$$

is the appropriate partitioning of X_N, i.e. the rows of X_N are indexed by D_n and D_q, while the columns are indexed by $h(x)$ and $g(x)$.

Both the last equality and the equivalence to the solution in Subsection 2.1 is shown for the case that K is non-singular. The equivalence in general holds under conditions (i), (ii) and (iii) in that section. We note, as for the introductory example, that the solution does do not depend on the dummy design D_q, except in so far as it is involved in guaranteeing that we have a good supersaturated basis.

2.3 Computing effects

An important step of the analysis of computer models involves assessing the importance of effects. This is usually computed by the functional Analysis of Variance (functional ANOVA), see Sobol′ [2001]. It has been used extensively in sensitivity analysis: see Saltelli et al. [2000].

Due to its polynomial nature, the smooth supersaturated model has important advantages which makes this analysis very simple. For instance, main effects and interactions can be computed analytically, and the sensitivity indexes can also be directly computed. Moreover, those effects remain polynomial and can be easily plotted, which is specially useful for double interactions. As an example, the general form for $f_i(x_i)$, the main effect for the i-th factor, is computed integrating Equation (1) over the region $[0,1]^d$ and over all factors except x_i and substracting the average value of $y(x)$, yielding the closed formula

$$f_i(x_i) = \int y(x) \prod_{k \neq i} dx_k - \int y(x) dx = \sum_{\alpha \in M} \theta_\alpha \left(\frac{x_i^{\alpha_i}(\alpha_i + 1) - 1}{\prod_{j=1}^d (\alpha_j + 1)} \right). \qquad (14)$$

The variation of $f_i(x_i)$, termed D_i, has a simple form $D_i = \int_0^1 f_i(x_i)^2 dx_i = \theta^T A \theta$, where θ is the vector of model parameters $\theta = (\theta_\alpha)_{\alpha \in M}$ and A is the square matrix $A = (a_{\alpha,\beta})_{\alpha,\beta \in M}$ with entries

$$a_{\alpha,\beta} = \frac{1}{\prod_{j=1}^d (\alpha_j + 1)(\beta_j + 1)} \cdot \frac{\alpha_i \beta_i}{\alpha_i + \beta_i + 1}.$$

Similar expressions can be derived for other interactions and their variations. Note that the above results are general and do not require knowledge of the fitted model parameters $\hat{\theta}_\alpha$. The estimates $\hat{\theta}_\alpha$ can be substituted once they are available.

2.4 Towards splines

We make the claim that as the supersaturated model order increases we get closer to the most smooth interpolating function. For our criteria it is well known, see references in the introduction, that cubic splines are optimal in one dimension, thin-plate splines in two dimensions and their generalisations in higher dimensions. However, the published analytic results are where the region of integration $\mathcal{X}$ has a standard shape (eg hyper-rectangle, ball etc) and typically contains the knots. On the other hand, except for numerical stability and our sufficient conditions, our methods apply to any region $\mathcal{X}$. Although we do not present here a formal proof of the convergence to splines the intuitive explanation is that as the model order increases and if the bases are suitably nested the optimal Ψ_2 decreases monotonically as the size of the model basis increases. Thus the Ψ_2 will converge to a minimum. Point-wise convergence of the interpolators to a limiting function can then be shown, and these limiting functions can be interpreted as a spline.

Recent references from the literature covering smoothing over irregular bivariate regions are Ramsay [2002] and Wood et al. [2008].

3 Examples

3.1 A one dimensional example: spline-like behavior

In this example, smooth saturated models are used for interpolating a known univariate function. The function considered is the sine cardinal $m(x) = \mathrm{sinc}(ax + b)$ with $a = 15\pi/2$ and $b = -10\pi/2$. The region over which the interpolators will be smoothed is $\mathcal{X} = [0, 1]$.

Suppose that the design D_6 is a uniform design (evenly spaced) in $[0, 1]$, and that the response vector y contains the values of $m(x)$ at points in D_6. The choice of a good saturated and supersaturated models can be driven by algebraic methods. For the present case, an obvious candidate is $h(x) = (1, x, \dots, x^5)^T$. Call $\hat{y}_0$ the interpolator fitted solely with $h(x)$. Now a process of smoothing is carried out by adding dummy points, one at a time. While adding dummy points, $h(x)$ remains unchanged. With only one dummy point, a clear candidate for $g(x)$ is $g(x) = (x^6)$, while for q dummy points, $g(x) = (x^6, \dots, x^{6+q-1})$. Call $\hat{y}_q$ the smooth interpolator obtained by adding q dummy points, $q = 1, \dots, 5$. The value of roughness for $\hat{y}_q$ quickly drops down so that a similar roughness to that of a spline is achieved with $\hat{y}_4$ (only four extra terms), see Table 1.

Model	$\hat{y}_0$	$\hat{y}_1$	$\hat{y}_2$	$\hat{y}_3$	$\hat{y}_4$	$\hat{y}_5$	Spline
Ψ_2^*	76.543	74.698	33.153	33.020	27.767	27.745	26.744

Table 1
Convergence of Ψ_2^* to spline for the univariate example of Section 3.1.

With a uniform design, the polynomial interpolator $\hat{y}_0$ exhibits the undesirable oscillating feature called *Runge phenomenon*, see Trefethen and Weideman [1991]. However, the smooth supersaturated models tended to remove the oscillations. The progressive smoothing achieved with extra terms can be seen in Figure 1 which shows the interpolator and smooth saturated models.

A comparison between the smooth supersaturated method and cubic splines, which are optimally smooth, was carried out as follows. First, for a uniform design D_n on $[0, 1]$, a saturated model $\hat{y}_0$ was fitted to the values of $m(x)$ at the design points. Call $\Psi_2^*(0)$ the value of smoothness for $\hat{y}_0$. Then, using q extra basis terms, a smooth supersaturated model $\hat{y}_q$ was fitted. Call $\Psi_2^*(q)$ the corresponding value of smoothness. A cubic interpolating spline was also fitted to the same data and call $\Psi_2^*(\mathrm{sp})$ its smoothness value. We observe experimentally that values $\Psi_2^*(0), \Psi_2^*(1), \dots$ form a decreasing sequence which converges surprisingly quick to $\Psi_2^*(\mathrm{sp})$, see the discussion in subsection 2.4.

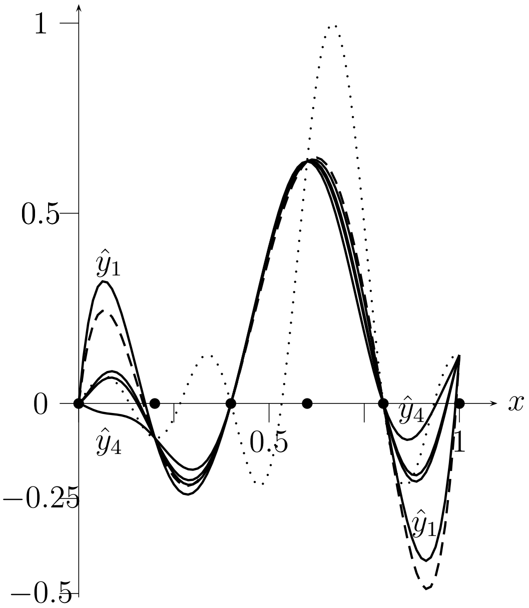

Fig. 1. Sequence of smooth saturated models: $\hat{y}_0$ is a polynomial of fifth degree (- -), $\hat{y}_1, \ldots, \hat{y}_4$ (—) are supersaturated models. The true model $m(x)$ $(\cdots)$ and design points are also shown.

This behavior can be quantified by plotting the ratio $\sqrt{\Psi_2^*(q)/\Psi_2^*(\text{sp})}$ against the number of terms added to smooth the model. Figure 2 shows such comparison when D_n are uniform designs of size $n = 5, 10, 15, 20$. The line for $n = 20$ is indistinguishable from $R(q) = 1$.

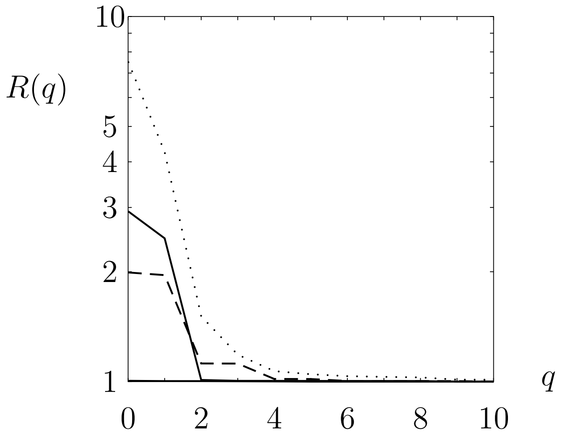

Fig. 2. Logarithm of the smoothness ratio $R(q) = \sqrt{\Psi_2^*(q)/\Psi_2^*(\text{sp})}$ against the number of smoothing terms added q: sample sizes $n = 5, 10, 15$ (- -,$\cdots$,—).

3.2 Smoothing and kriging: unidimensional comparison

An important feature of smooth supersaturated interpolators is that, even for small sample sizes, an interpolator can be fitted to data. This feature can be an advantage over other methods such as kriging, which requires a initial stage of parameter estimation. If the sample size is small, and no prior information for kriging parameters is available, then smooth supersaturated models can be used as an alternative to kriging interpolators.

A comparison was performed between smooth supersaturated models and kriging. The aim was to judge the performance of both interpolating systems to produce good fits to data using extra validation points. The design region for the study was $[0, 1]$ and call $D_n, n = 5, \ldots, 17$ a design of n points constructed with the first $n - 2$ points of the standard univariate Sobol' sequence implemented in `R`, together with 0 and 1. The designs are nested, for example D_6 can be obtained by adding the point 0.375 to $D_5 = \{0, 1, 0.5, 0.75, 0.25\}$.

The following four univariate functions were used as true (but assumed unknown) simulators:

$g_1(x) = \mathrm{sinc}(23x-15.7)$; $g_2(x) = 1+\sin(13.9x)$; $g_3(x) = \sin(12x^2)$ and $g_4(x) = (1 + \sin(13.9x))u(x - 0.34)$ where $u(x)$ is the Heaviside step function, *i.e.* $u(x) = 1$ if $x \geq 0$ and $u(x) = 0$ otherwise.

The selected functions were chosen to include features which are difficult to model with polynomials. For instance, g_2 is periodic; g_1 features damping oscillations; g_3 has frequency that changes with variable x and g_4 has a flat region and a periodic region.

For each function $g_1, \ldots, g_4$, training data was generated at the design points D_n, and both smooth supersaturated models and a kriging model were fitted to the data. The analysis was performed independently for every function. The smooth model was computed using nine smoothing terms, while the kriging model used an exponential correlation function $\mathrm{corr}(Y(s), Y(t)) = \exp(-\theta|s-t|^p)$, with parameters θ, p carefully estimated by maximum likelihood, see Sacks et al. [1989].

Finally, a validation design was constructed taking 30 further points from Sobol's sequence. Empirical root mean square error (RMSE) was computed using the models fitted and the true function. The comparison is made using the ratio of RMSE value for kriging against that for smooth supersaturated models $RMSE_{kr}/RMSE_{ssm}$, which is shown in Table 2 and plotted in Figure 3.

For design sizes $n < 10$, the smooth supersaturated model compares rather

Design	Simulator used			
size n	g_1	g_2	g_3	g_4
5	1.308	1.229	0.993	0.882
6	1.399	0.550	0.969	0.320
7	0.524	0.566	0.987	0.319
8	0.497	0.573	1.043	1.176
9	0.751	0.267	1.369	3.315
10	6.679	5.318	1.314	3.752
11	17.591	41.458	3.022	22.984
12	19.897	59.092	9.981	9.345
13	39.301	255.953	17.570	9.743
14	239.687	6431.865	41.209	25.047
15	479.360	5722.610	176.935	25.989
16	218.640	133.324	74.937	15.767
17	611.246	36.982	178.632	47.473

Table 2
Ratio $RMSE_{kr}/RMSE_{ssm}$ for the univariate study.

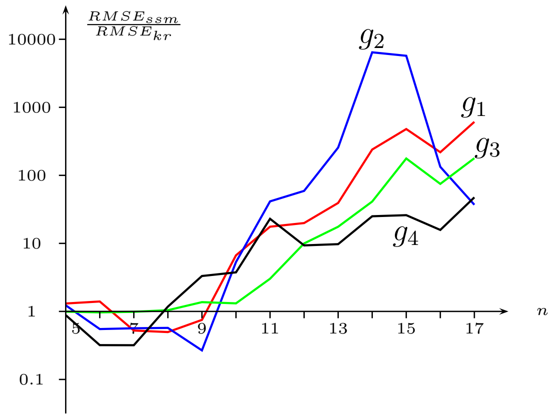

Fig. 3. The ratio $RMSE_{kr}/RMSE_{ssm}$ for the univariate study.

favorably with kriging. As n size increases, the value of $RMSE$ for kriging becomes much smaller, relative to the smooth supersaturated model. This phenomena of smooth supersaturated model consistently being better RMSE than kriging for small sample sizes was observed for different numbers of smoothing terms, ranging from three to thirty.

3.3 Smoothing and kriging: bidimensional comparison

Our second comparison was performed using bidimensional functions. The settings were similar to the unidimensional study. The design region was $[0,1]^2$; the design $D_n, n = 5, \ldots, 17$ was composed of $n-2$ points of a bidimensional Sobol' sequence, together with the origin and the point $(1,1)$. Four bivariate functions were used as simulators:

$$\begin{aligned}
g_1(x_1,x_2) &= \sin((x_1-0.5)^2 + (x_2-0.5)^2 + 7x_1(x_2-0.5)) \\
g_2(x_1,x_2) &= (x_2+1/2)^4/(x_1+1/2)^2 \\
g_3(x_1,x_2) &= 3(1-u)^2 \exp\left(-u^2-(v+1)^2\right) - 10(u/5-u^3-v^5)\exp\left(-u^2-v^2\right) \\
&\quad -1/3\exp\left(-(u+1)^2-v^2\right) \\
g_4(x_1,x_2) &= 100(v-u^2)^2 + (1-u)^2
\end{aligned}$$

The function g_3 is the `peaks` function from MATLAB®, while g_4 is the Rosenbrock function; both were rescaled to the design region $[0,1]^2$ with $u = 4x_1 - 2$ and $v = 4x_2 - 2$. As in the unidimensional study of Section 3.2, the functions were selected to include features which are difficult to model with polynomials, such as flat regions with sharp peaks or oscillations with changing frequency.

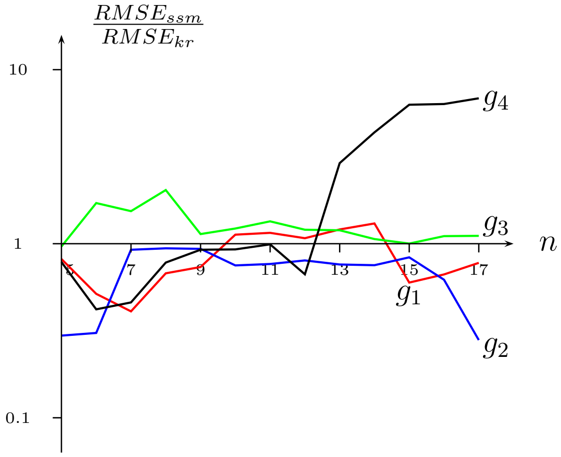

Fig. 4. Ratio $RMSE_{kr}/RMSE_{ssm}$ for the simulated bivariate study.

A smooth supersaturated model with 20 additional smoothing terms was fitted to the simulated values. The smoothing terms consist of the next 20 terms in the same degree lexicographic term order used for the saturated basis. This smooth model was compared with a kriging model with exponential correlation function $\mathrm{Corr}(Y(s_1,s_2), Y(t_1,t_2)) = \exp(-\sum_{i=1}^{2}\theta_i|s_i-t_i|^{p_i})$. The parameters $\theta_i, p_i, i = 1,2$ were fitted using maximum likelihood. The RMSE values were computed for both fits using a set of 30 extra bivariate Sobol' design points. Table 3 contains values of the ratio $RMSE_{kr}/RMSE_{ssm}$, which are also plotted in Figure 4.

The results observed are similar to those of Section 3.2. The RMSE of smooth supersaturated models compare favourably with that of kriging for small sample values. Moreover, in two cases (g_1, g_2) the RMSE remains smaller for smooth supersaturated model up to sample size is 17. For g_4 we observe a similar phenomena to the unidimensional situation: from a certain sample size $(n = 13)$, kriging starts performing better.

We do not claim superiority of smooth supersaturated models for small sample sizes in all circumstances. But we are clear that smooth supersaturated models are a resource for modellers that can perform better than kriging for small sample sizes. The usual care should always be taken in the form of validation and diagnostics of the models.

Design	Function			
size n	g_1	g_2	g_3	g_4
5	0.817	0.297	0.964	0.785
6	0.515	0.307	1.712	0.420
7	0.409	0.923	1.538	0.460
8	0.676	0.941	2.034	0.779
9	0.735	0.934	1.136	0.923
10	1.127	0.750	1.223	0.928
11	1.155	0.765	1.344	0.992
12	1.076	0.802	1.203	0.667
13	1.208	0.760	1.195	2.904
14	1.307	0.753	1.063	4.363
15	0.598	0.835	1.002	6.288
16	0.666	0.621	1.106	6.347
17	0.776	0.280	1.110	6.846

Table 3
Ratio $RMSE_{kr}/RMSE_{ssm}$ for the bivariate study.

3.4 A case study: Engine Emissions Data

The performance of a smooth supersaturated model was evaluated against a kriging model using the engine emissions data set analysed in Bates et al. [2003]. This data set comes from a computer experiment without noise and comprises 48 observations in five factors N, C, A, B and M. An extra set of 49 observations is available for validation purposes. The smooth supersaturated model, termed $\hat{y}$, was constructed with 100 terms fitted to the set of 48 observations. For this model, 48 terms correspond to the good saturated basis proposed in [Bates et al., 2003, Section 6.3], and this forms $h(x)$. A set of 22 terms were added to complement missing terms of total degree three and then a set of extra 30 terms of total degree four were added. All the extra 52 terms described form $g(x)$ and were added using a degree lexicographic order.

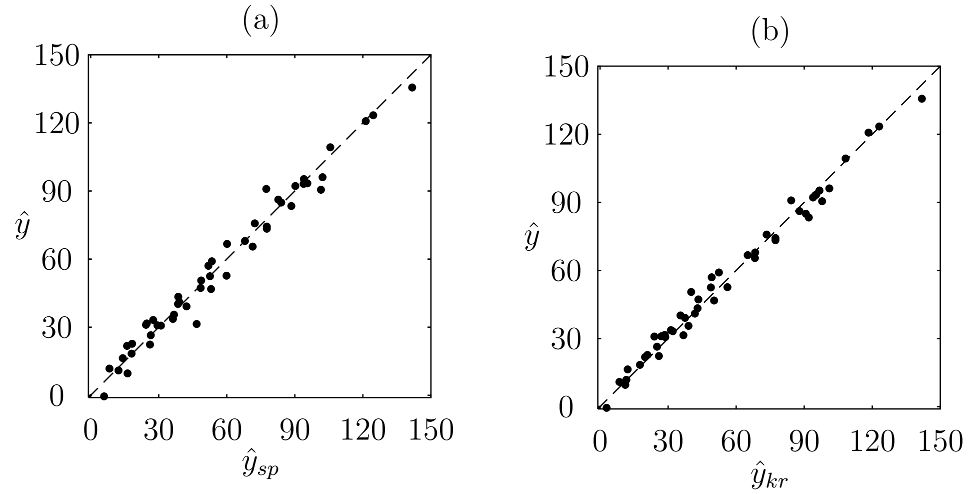

Fig. 5. Smooth supersaturated predictions ($\hat{y}$) against spline ($\hat{y}_{sp}$) and kriging predictions ($\hat{y}_{kr}$) for the validation data set of Section 3.4.

Kriging and spline models were constructed with the first data set for comparison purposes. The kriging model, termed $\hat{y}_{kr}$, was built with a five dimensional exponential covariance structure, with parameters estimated by maximum likelihood. The spline model, named $\hat{y}_{sp}$, was constructed with the `tpaps` function from Matlab®.

In the validation stage, predictions at the extra 49 design points were built using the three models $\hat{y}, \hat{y}_{sp}$ and $\hat{y}_{kr}$. Existing observations at extra design points allow computation of RMSE. The values of RMSE for $\hat{y}, \hat{y}_{sp}$ and $\hat{y}_{kr}$ are $5.844, 5.896$ and 4.450, which represent $4.4\%, 4.5\%$ and 3.4% respectively of the range of the response values. The smooth supersaturated model $\hat{y}$ compares well with both spline and kriging, being close to the spline model.

Scatterplots were generated using the validation and predicted model data. Figure 5 shows that predictions with the smooth supersaturated model are highly correlated with those obtained with spline and kriging models. Figure 6 shows the smooth supersaturated model to be a good predictor of the true response.

Finally, as described in Section 2.3, the fitted smooth supersaturated model was decomposed using the functional ANOVA and effects and sensitivity indices were analytically computed. The sensitivity indices allocate 99.51% of the total variability to all main effects and two factor interactions; of which the

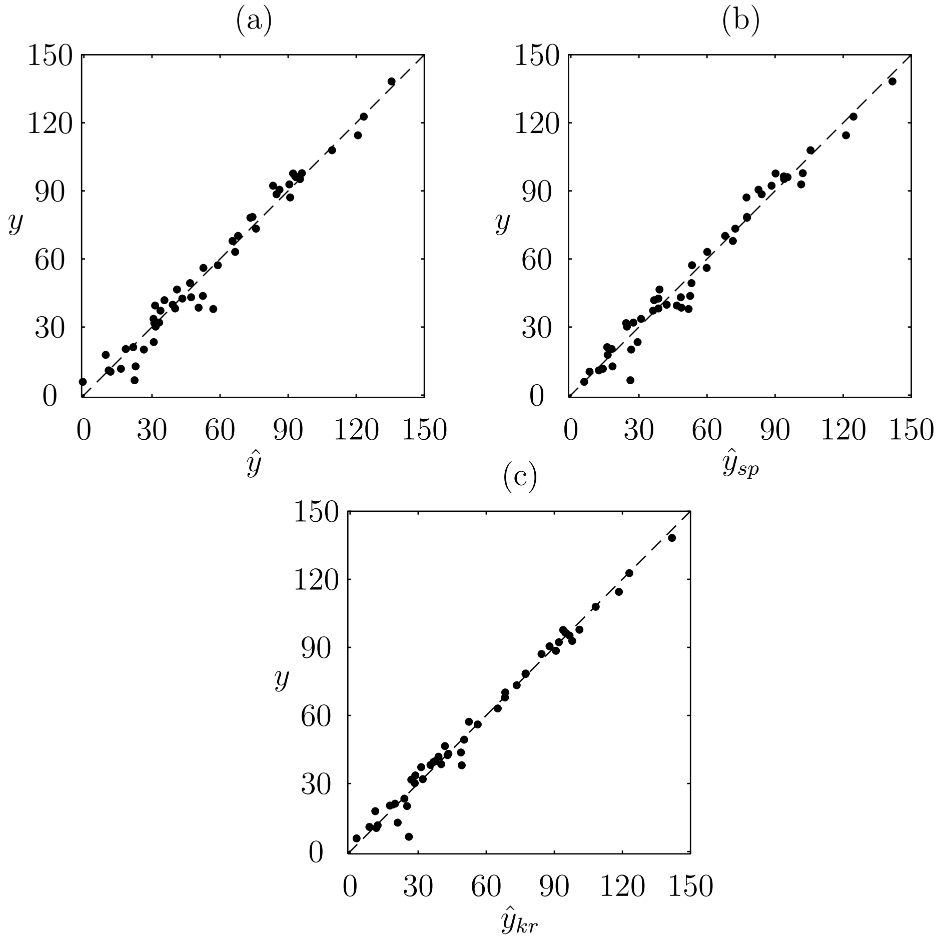

Fig. 6. True values (y) against smooth supersaturated predictions ($\hat{y}$), spline ($\hat{y}_{sp}$) and kriging predictions ($\hat{y}_{kr}$) for the validation data set of Section 3.4.

most important are the main effects for factor M (63.90% of total variability) and N (32.68%). The two factor interaction MN has a small effect (0.56%). The two main effects M and N stand out in the effects plot shown in Figure 7. The interaction MN is also shown in Figure 7.

4 Further developpments

4.1 Smooth polynomial kernels and optimal knots

The smooth interpolators of this paper are of the form

$$\hat{y}(x) = \hat{\theta}^T f(x) = y^T B f(x).$$

If we set the data vector data y to be the basis vector $e^{(i)}$ which has unity in the i-th entry and zero elsewhere we retrieve an indicator function $k_i(x) = {e^{(i)}}^T B f(x)$

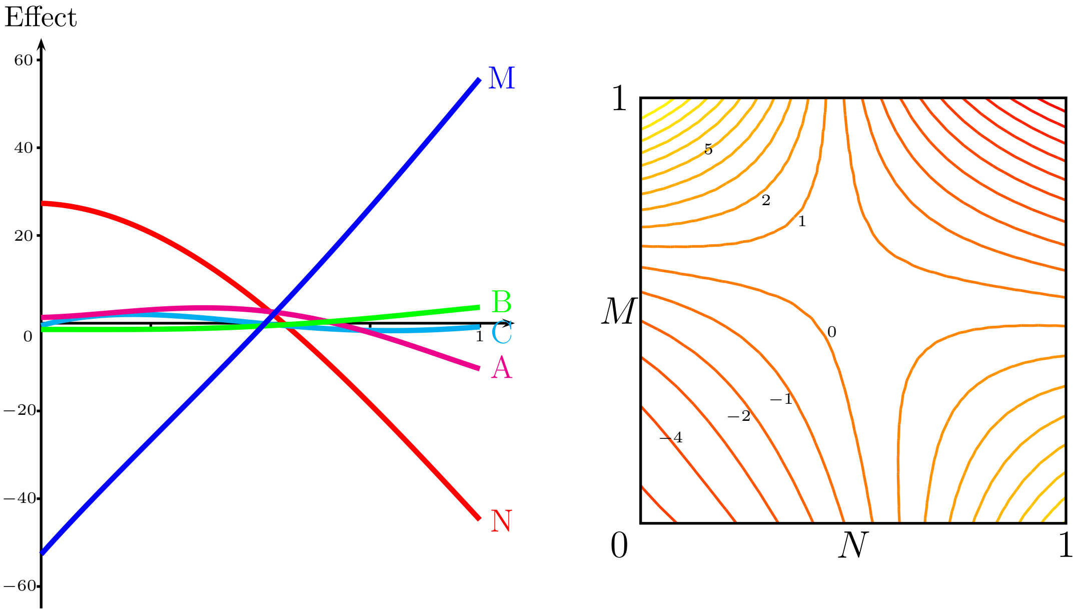

Fig. 7. Main effects plot and contour plot for interaction MN.

in the sense that if x_j is a design point

$$k_i(x_j) = \delta_{ij},$$

the Kronecker delta. We can then write:

$$\hat{y}(x) = \sum_{i=1}^{n} y_i k_i(x)$$

Because the optimal smoothness is a quadratic form in the data y: $\Psi_2^* = y^T Q y$ (2.2) we have the sure knowledge that each individual $k_i(x)$ is an optimally smooth interpolator under the same conditions that a general $\hat{y}(x)$, is optimally smooth. Moreover the optimal smoothness for $k_i(x)$ is the i-th diagonal element of Q, namely $\Psi_{2,i}^* = Q_{ii} = {e^{(i)}}^t Q e^{(i)}$.

This simple point suggests a method of selecting the design points optimally. In the language of splines, the design points are *knots*. The design D_n affects the value of the smoothness via the matrix X. Given that we have to choose the design *before* we observe the data $\{y_i\}$ we may consider choosing it to minimize, over all designs in our preferred design region, some measure of the size of the matrix Q, which does not depend on y. Following the above discussion, the most obvious criterion is to minimize

$$\text{trace}(Q) = \sum_{i=1}^{n} \Psi_{2,i}^*.$$

However, we could minimise other criteria such as $\max_i \lambda_i(Q)$ the largest eigenvalue of Q. Since Q is singular we cannot use the full determinant, but we could minimise the "positive" determinant, namely the product of the non-zero eigen-values. A general criteria would be to minimise the s-norm: $(\sum_i (\lambda_i(Q))^s)^{\frac{1}{s}}$ for some $s > 0$, where the sum is over the non-zero eigenvalues. The trace is for $s = 1$ and the positive determinant and the maximum eigenvalue are the limiting cases as s tends to zero and infinity respectively.

4.2 SSM regression and optimal design

If we consider the design points in D_n as knots we are free to use them to construct the optimal kernels $k_i(x)$ without necessary observing the $\{y_i\}$ at the knots. This leaves us free to consider the optimal experimental design problem based on using $k_i(x)$ as a set of regression functions. This is the analogy of optimal design for spline regression: given knots we can construct a spline basis, but, to repeat, we do not have to observe at the knots. In that case it may well be the case that the actual optimal design points do not correspond to the knots: see Woods and Lewis [2006]. The spline optimal design problem has proved hard because of the difficulty of obtaining expressions for splines, see Kaishev [1989], Dette et al. [2008]. In so far as taking larger supersaturated bases, gets closer to spline regression, the smooth polynomial methods of this paper combined with optimal design algorithms and provide platform of approximating spline optimal design over arbitrary regions $\mathcal{X}$, a currently unsolved problem. This is the subject of further research by the authors.

In summary, this discussion points to a new technology for high dimensional function fitting over arbitrary regions which sets up a double optimization problem: choosing knots to maximize smoothness and design points to optimize some statistical criterion (such as D-optimality) , or one could use combined criteria; with or without taking observation at knots. Maintaining the models as polynomials promises to be more tractable than working directly with splines.

4.3 Other smoothness criteria

There are a number of ways in which one can generalize or adapt our methods. A similar analysis will go through for a weighted criterion

$$\Psi_2 = \int_{\mathcal{X}} ||H(y(x))||^2 w(x) dx,$$

where $w(x)$ is a non-negative weight function. This simply changes the definition of K and $\tilde{K}$, in our analysis. Also, the smoothness criteria we adopted is one of a number in a wider quadratic class, which includes

$$\Psi_1 = \int_{\mathcal{X}} || \bigtriangleup (y(x))||^2 dx,$$

where $\bigtriangleup(y(x))$ is the gradient vector; and a measure of deviation from a target function can be used

$$\Psi_3 = \int_{\mathcal{X}} |y(x) - t(x)|^2 dx.$$

5 Appendix

5.1 Appendix 1: solution for $\hat{\theta}_0$ and $\hat{\theta}_1$

It is possible to use block matrix inverse methods, but they are a little cumbersome. We first find $\hat{\theta}_0$. Writing out Equation (10) we have

$$\begin{aligned} X_0\theta_0 + X_1\theta_1 &= y \\ K\theta_1 - X_1^T\lambda &= 0 \\ X_0\lambda &= 0 \end{aligned}$$

Solving for λ from the second two equations we have

$$\lambda = (X_1K^{-1}X_1^T + X_0X_0^T)^{-1}X_1\theta_1$$

Using this to eliminate θ_1 from the first equation we have

$$X_0^T(X_1K^{-1}X_1^T + X_0X_0^T)^{-1}X_0\theta_0 = X_0^T(X_1K^{-1}X_1^T + X_0X_0^T)^{-1}y,$$

giving

$$\hat{\theta}_0 = (X_0^T(X_1K^{-1}X_1^T + X_0X_0^T)^{-1}X_0)^{-1}X_0^T(X_1K^{-1}X_1^T + X_0X_0^T)^{-1}y,$$

Writing $y^* = y - X_0\hat{\theta}_0$ we obtain reduced matrix equation:

$$\begin{bmatrix} X_1 & 0 \\ \tilde{K} & -X_1^T \\ 0 & X_0^T \end{bmatrix} \begin{bmatrix} \theta_1 \\ \lambda \end{bmatrix} = \begin{bmatrix} y^* \\ 0 \\ 0 \end{bmatrix}$$

Left multiplying by the transpose of the matrix on the left and inverting we have

$$\hat{\theta}_1 = (X_1^TX_1 + \tilde{K}(I - X_1^T(XX^T)^{-1}X_1)\tilde{K})^{-1}X_1y^* \qquad (15)$$

Note that in the case that X_0 and X_1 have orthogonal columns we reduce to the standard form $\hat{\theta}_0 = (X_0^T X_0)^{-1} X_0^T y$. This result can be achieved by rewriting the supersaturated basis so that the terms with degree higher than linear (degree one) are orthogonal to the linear terms with respect to the design. Of course, the definition of $\tilde{K}$ should be changed accordingly.

5.2 Equivalence of forms in the case K nonsingular

The following three forms for $\hat{\theta} = By$ are equivalent, where B is one of:

(i) $B_1 = (X_1^T X_1 + K(I-P)K)^{-1} X^T y$

(i) $B_2 = K^{-1}(X_{11}, X_{12})^T Q y$

(ii) $B_3 = X^{-1}\left(\begin{smallmatrix} I \\ -A_{22}^{-1}A_{21} \end{smallmatrix}\right)$

To show that $B_1 = B_2$ multiply both by $X_1^T X_1 + K(I-P)K$ and note that $PX^T = X^T$ to obtain respectively X^T and $X^T X K^{-1} X^T Q$. But from the definition of Q and using block the partition inverse formula we see that that $XK^{-1}X^T = Q^{-1}$ and we are done (reversing the steps).

To show that $B_2 = B_3$ we multiply both by ${X^{-1}}^T K$. Then B_2 gives

$$ {X^{-1}}^T K K^{-1} (X_{11}, X_{12})^T Q Q^{-1} = {X^{-1}}^T (X_{11}, X_{12})^T = \begin{pmatrix} I \\ 0 \end{pmatrix}, $$

while B_3 gives

$$ {X^{-1}}^T K X^{-1} \left(\begin{smallmatrix} I \\ -A_{22}^{-1}A_{21} \end{smallmatrix}\right) Q^{-1} = A \left(\begin{smallmatrix} I \\ -A_{22}^{-1}A_{21} \end{smallmatrix}\right) Q^{-1} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \left(\begin{smallmatrix} I \\ -A_{22}^{-1}A_{21} \end{smallmatrix}\right) Q^{-1} $$

$$ = \left(\begin{smallmatrix} A_{11} - A_{12}A_{22}^{-1}A_{21} \\ A_{21} - A_{22}A_{22}^{-1}A_{21} \end{smallmatrix}\right) Q^{-1} = \left(\begin{smallmatrix} A_{11} - A_{12}A_{22}^{-1}A_{21} \\ 0 \end{smallmatrix}\right) Q^{-1} = \left(\begin{smallmatrix} I \\ 0 \end{smallmatrix}\right). $$

Again, reversing the steps we obtain our result.

Acknowledgments

The first and third authors acknowledge the EPSRC grant GR/S63502/01, while the second and third authors acknowledge the EPSRC grant EP/D048893/1 (MUCM project).

References

Bates, R., Giglio, B., and Wynn, H. (2003). A global selection procedure for polynomial interpolators. *Techno.*, 45(3):246–255.

Bratley, P. and Fox, B. L. (1988). ALGORITHM 659 Implementing Sobol's quasirandom sequence generator. *ACM Trans. Math. Soft.*, 14(1):88–100.

Cox, D., Little, J., and O'Shea, D. (1997). *Ideals, Varieties, and Algorithms.* Springer-Verlag, New York. Second Edition.

Dette, H., Melas, V. B., and Pepelyshev, A. (2008). Optimal designs for free knot least squares splines. *Statist. Sinica*, 18(3):1047–1062.

Duchon, J. (1976). Interpolation des functions de deux variables suivant le principle de la flexion des plaques minces. *R.A.I.R. Analyses Numrique*, 10(3):5–12.

Holladay, J. (1957). A smoothest curve approximation. *Maths. Tables Aids Compute.*, 11(3):233–243.

Ihaka, R. and Gentleman, R. (1996). R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299–314.

Kaishev, V. K. (1989). Optimal experimental designs for the B-spline regression. *Comput. Statist. Data Anal.*, 8(1):39–47.

Kennedy, M. and O'Hagan, A. (2001). Bayesian calibration of computer models. *J. Roy. Statist. Soc. B.*, 63(3):425–2001.

Kimeldorf, G. and Wahba, G. (1970). A correspondance between bayesian estimation of stochastic processes and smoothing by splines. *Ann. Statist.*, 41:495–502.

Kleijnen, J. P. C. (2009). Kriging metamodeling in simulation: A review. *Eur. Jour. Op. Res.*, 192(3):707–716.

Lin, Z., Xu, L., and Wu, W. (2004). Applications of Gröbner bases to signal processing: a survey. *Lin. Alg. Appl.*, 391(3):169–202.

Micula, G. (2002). A variational approach to spline functions theory. *General Mathematics*, 10(1-2):21–50.

Pistone, G., Riccomagno, E., and Wynn, H. P. (2001). *Algebraic Statistics*, volume 89 of *Monographs on Statistics and Applied Probability.* Chapman & Hall/CRC, Boca Raton.

Pistone, G. and Wynn, H. (1996). Generalised confounding with Gröbner bases. *Biometrika*, 83(3):653–666.

Ramsay, T. (2002). Spline smoothing over difficult regions. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 64(2):307–319.

Rasmussen, C. and Williams, C. (2005). *Gaussian processes for machine learning.* MIT Press, Cambridge, Mass.

Sacks, J., Welch, W., Mitchell, T., and Wynn, H. (1989). The design and analysis of computer experiments. *Statistical Science*, 4(4):409–439.

Saltelli, A., Chan, K., and Scott, E. (2000). *Sensitivity Analysis: Gauging the Worth of Scientific Models.* Wiley, Chichester.

Sobol′, I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Math. Comput. Simulation*, 55(1-

3):271–280. The Second IMACS Seminar on Monte Carlo Methods (Varna, 1999).

Trefethen, L. N. and Weideman, J. A. C. (1991). Two results on polynomial interpolation in equally spaced points. *J. Approx. Theory*, 65(3):247–260.

Wood, S., Bravington, M., and Hedley, S. (2008). Soap film smoothing. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 70(5):931–955.

Woods, D. and Lewis, S. (2006). All-bias designs for polynomial spline regression models. *Aust. N. Z. J. Stat.*, 48(1):49–58.