The effect of the nugget on Gaussian process emulators of computer models

Ioannis Andrianakis*, Peter G. Challenor
National Oceanography Centre, Southampton, SO14 3ZH, UK

ARTICLE INFO

Article history:
Received 10 February 2011
Received in revised form 18 April 2012
Accepted 19 April 2012
Available online 14 May 2012

Keywords:
Computer experiments
Ill conditioning
Interpolation
Approximation
Kriging

ABSTRACT

The effect of a Gaussian process parameter known as the nugget, on the development of computer model emulators is investigated. The presence of the nugget results in an emulator that does not interpolate the data and attaches a non-zero uncertainty bound around them. The limits of this approximation are investigated theoretically, and it is shown that they can be as large as those of a least squares model with the same regression functions as the emulator, regardless of the nugget's value. The likelihood of the correlation function parameters is also studied and two mode types are identified. Type I modes are characterised by an approximation error that is a function of the nugget and can therefore become arbitrarily small, effectively yielding an interpolating emulator. Type II modes result in emulators with a constant approximation error. Apart from a theoretical investigation of the limits of the approximation error, a practical method for automatically imposing restrictions on its extent is introduced. This is achieved by means of a penalty term that is added to the likelihood function, and controls the amount of unexplainable variability in the computer model. The main findings are illustrated on data from an Energy Balance climate model.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

Computer model emulation is a statistical technique that allows the development of fast surrogates for computer models that are typically computationally expensive (Sacks et al., 1989; Kennedy and O'Hagan, 2000). Emulators offer the possibility of carrying out efficiently typical computer model related tasks, such as calibration (Kennedy and O'Hagan, 2001; Bayarri et al., 2007) and sensitivity analysis (Oakley and O'Hagan, 2004), which can be computationally very demanding otherwise. A common approach for building computer model emulators is based on fitting a Gaussian process (GP) to data obtained from running the simulator at a predefined set of inputs (design points), effectively trying to learn a mapping between the input and output spaces of the model.

Gaussian processes have found extensive use in machine learning (Rasmussen and Williams, 2006), while in the analysis of spatial data have been long known as kriging (Cressie, 1993; Stein, 1999). In the above disciplines, it is common to account for the existence of measurement error or noise in the observations. This can be achieved with the addition of a small number on the main diagonal of the design points correlation matrix, which is known as nugget or jitter (Cressie, 1993; Carmack et al., 2012). In the analysis of deterministic computer models however, the simulator's outputs are assumed to be 'noise-free', since repeated runs of the simulator for the same input configuration return the same output. For this reason many authors defend that a nugget should not be added.

On the other hand, it is not an uncommon practice to include a nugget to the correlation matrix, primarily as a means of alleviating numerical problems that often arise with its inversion, which is a common calculation involved in fitting a
The Gaussian process to data (Neal, 1997). The inclusion of the nugget is reconciled with the assumption of noise-free data, via the conventional wisdom that wants its effect on the Gaussian process to be negligible, as long as its value is kept small. In a recent work, Pepelyshev (2010) has put this common belief to the test, finding that the addition of the nugget does indeed affect the overall shape and modes of the Gaussian process likelihood.

Apart from the mitigation of computational problems, deterministic model emulators can include a nugget for a number of other reasons. For example, a nugget can be used to account for the variability that is attributed to inactive inputs, that is, inputs used for obtaining the model runs, but chosen to be ignored in building the emulator, typically because of their small effect in the output. Gramacy and Lee (2012) also advocate the use of the nugget for essentially modelling potential discrepancies between the GP model and the simulator, which may arise from inaccurate assumptions about stationarity, correlation structure etc.

The purpose of this work is to further investigate the effect of the nugget on the likelihood and more importantly on the predictions made with the Gaussian process. It is known (e.g., Cressie (1993) and Pepelyshev (2010)), that a Gaussian process that contains the nugget term does not interpolate the data and attaches a non-zero uncertainty bound around them. We derive expressions for the mean squared error (MSE) of this approximation and show that it can be as large as that of a least squares model with the same regression functions as the GP mean, regardless of the nugget’s value. We also investigate the relationship between the correlation function parameters (correlation lengths) and the nugget, explaining the negative correlation that is reported to exist between them (Seheult, pers. comm.).

Finally, we propose a method for automatically imposing restrictions on the MSE of the approximation error. Our approach is based on a penalty term that is added to the log likelihood and allows specifying the MSE that is considered acceptable, with a minimal computational overhead. A similar method based on the iterative Tikhonov regularisation is described in Ranjan et al. (2011). The method we propose is obviously useful when an interpolating emulator is desired. However, even when we are willing to accept some discrepancy between the emulator and the simulator, controlling the degree of the approximation can still be a sensible precaution, therefore the proposed method would still be relevant. For example, requiring a close match between the emulator and the simulator can reveal inadequate assumptions about the simulator’s prior distribution, or even issues with the simulator itself, such as discontinuities. Examples are given in Sections 6 and 7.

The structure of the paper is as follows: Section 2 introduces the GP model, formalising also the notation and Section 3 describes how the numerical problems are alleviated with the addition of the nugget. The effect of a fixed size nugget on the emulator is discussed in Sections 4 and 5 discusses treating the nugget as a variable parameter. The penalty term for controlling the MSE of the approximation is presented in Section 6. Section 7 presents a two dimensional example and Section 8 concludes this paper.

2. The model

Let the simulator be represented by a function \( f: \mathbb{R}^p \rightarrow \mathbb{R} \). Even though the simulator can be evaluated at any input \( x \), it is assumed that obtaining new runs is computationally expensive, and that only \( n \) of them are available. We denote the inputs at which the simulator is run as \( D = \{x_1, x_2, \ldots, x_n\} \) and the respective outputs as \( y = [f(x_1), f(x_2), \ldots, f(x_n)]' \). A central assumption we make is that the simulator’s output can be represented by a Gaussian process; that is, the values of the output obtained for different input points, can be modelled with a joint Gaussian distribution.

The Gaussian process prior for \( f(x) \) can be completely specified by a mean and a covariance function. We define the mean as \( \text{E}[f(x)] = h'(x)\beta \), where \( h(x) \) is a \( (q \times 1) \) vector of a priori known regression functions, and \( \beta \) is the vector of the corresponding regression coefficients. The covariance between two output points \( f(x) \) and \( f(x^*) \), will be denoted as \( \text{Cov}[f(x), f(x^*)] = \sigma^2c(x, x^*) \), where \( \sigma^2 \) represents the variance of the Gaussian process, and \( c(x, x^*) \) the correlation between outputs \( f(x) \) and \( f(x^*) \).

The correlation function \( c(x, x^*) \) maps a pair of vectors \( x, x^* \) to the set of real numbers \( \mathbb{R} \). We restrict ourselves to continuous and stationary correlation functions, that is, functions that depend only on the Euclidean distance between their inputs. We define the distance between \( x \) and \( x^* \) along the input dimension \( d \) as \( d_i = |x(i) - x^*(i)| \), where \( x(i) \) is the \( i \)th element of \( x \). We also assume that \( c(\cdot, \cdot) \) has an unknown vector of parameters \( \delta \in (0, \infty)^d \), which determines the strength of the correlation between the input vectors. The components of \( \delta \) are referred to as correlation lengths. We finally assume that if \( \sum_{i=1}^d (d_i/\delta_i)^2 \rightarrow \infty \), then \( c(x, x^*) \rightarrow 0 \) and similarly if \( \sum_{i=1}^d (d_i/\delta_i)^2 \rightarrow 0 \) then \( c(x, x^*) \rightarrow 1 \).

A number of correlation functions satisfy the above criteria and among the most widely used are the squared exponential, the Matérn, and the rational quadratic (Rasmussen and Williams, 2006). In this work we will be using the squared exponential (SE) correlation function, although the results apply to other correlation functions that satisfy the above criteria. The SE correlation function is given by

\[
c(x, x^*) = \exp \left\{-\sum_{i=1}^d \frac{(d_i)^2}{\delta_i^2}\right\}. \tag{1}
\]

Conditional on its three parameters, the Gaussian process prior for \( f(x) \) can be written as

\[
p(f(x)|\beta, \sigma^2, \delta) \sim \mathcal{N}(h'(x)\beta, \sigma^2c(x, x)). \tag{2}
\]
The parameters are generally unknown, and therefore need to either be estimated or marginalised. An analytical marginalisation is possible for two out of the three parameters, namely $\beta$ and $\sigma^2$. Their marginalisation can be carried out using the standard non-informative prior $p(\beta, \sigma^2) \propto \sigma^{-2}$ (Berger et al., 2001). The marginal likelihood can then be written as

$$L(\delta|y) \propto |A|^{-\frac{1}{2}} |H^T A^{-1} H|^{-\frac{1}{2}} (\hat{\sigma}^2)^{-\frac{n-2}{2}},$$

with $\hat{\sigma}^2 = (y - H \hat{\beta})^T A^{-1} (y - H \hat{\beta})$ and $\hat{\beta} = (H^T A^{-1} H)^{-1} H^T A^{-1} y$. In the above equations we have also defined the design points regression matrix $H$, whose $i$th row is $(H)_i = h'(x_i)$, and the design points correlation matrix $A$, with $(A)_{ij} = c(x_i, x_j)$.

A fully Bayesian analysis would require the numerical marginalisation of $\delta$, typically with an MCMC algorithm. This approach accounts for the uncertainty in the value of $\delta$, but comes at a high computational cost. A less expensive alternative is to find a maximum likelihood estimate for $\delta$ and use this as if it were the true value (Kennedy and O’Hagan, 2001). This ‘plug-in’ approach can underestimate the simulator’s variability, but this is normally outweighed by savings in the computational effort, especially in the case of relatively smooth models (Bayarri et al., 2007). In this work, we will employ the plug-in approach, although the results are still relevant for the fully Bayesian solution.

An estimate for $\delta$ can be obtained by maximising the logarithm of $L(\delta|y)$, which we denote by $L(\delta)$, that is

$$\hat{\delta} = \arg\max_{\delta} \{L(\delta)\}.$$

Having obtained an estimate for $\delta$, we can use the Gaussian process to make predictions about the simulator’s output for an untried input $x$. It turns out that the predictive distribution is a Student’s $t$-distribution, with $n - q$ degrees of freedom (Bastos and O’Hagan, 2009). That is

$$p(f(x)|y, \delta) \sim t_{n-q}(m(x), V(x, x)), $$

with

$$m(x) = h'(x) \hat{\beta} + T(x) A^{-1} (y - H \hat{\beta})$$

and

$$V(x, x^*) = \frac{\hat{\sigma}^2}{n - q - 2} \left[ c(x, x^*) - T(x) A^{-1} T'(x^*) + P(x) (H' A^{-1} H)^{-1} P'(x^*) \right],$$

where $T(x)$ is the correlation vector between $x$ and $D$, i.e. $(T(x))_i = c(x, x_i)$ and $P(x) = h'(x) - T(x) A^{-1} H$.

### 3. Alleviating numerical problems using the nugget

#### 3.1. Ill-conditioning of the correlation matrix

The computation of Eqs. (3), (5) and (6) can be susceptible to numerical problems. A main source of such problems is the fact that the design points correlation matrix $A$ can often be ill-conditioned. As a result, its inversion may be inaccurate and sometimes not even feasible in double precision arithmetic. The conditioning of a matrix can be quantified by its condition number $\kappa$. For a symmetric matrix, such as $A$, the condition number is defined as the ratio of the largest (in magnitude) eigenvalue to the smallest, i.e. $\kappa(A) = \frac{|\lambda_{\text{max}}|}{|\lambda_{\text{min}}|}$ (Press et al., 1992). A matrix with an infinite condition number is singular, whereas a matrix whose condition number is close to the inverse of the machine precision is ill-conditioned.

For an $n \times n$ symmetric matrix with ones across its main diagonal, it can be shown that the sum of its eigenvalues equals $n$. For example, if $A = U A U'$, with $U$ being the orthonormal matrix of eigenvectors and $A$ the diagonal matrix of eigenvalues, we can write

$$\sum_{i=1}^{n} \lambda_i \equiv \text{tr}[A] = \text{tr}[U' U A]$$

$$= \text{tr}[U A U'] = \text{tr}[A] = n,$$

where $\text{tr}[\cdot]$ is the trace of a matrix.

As all elements of $\delta$ tend to zero ($\delta \rightarrow 0$) all $n$ eigenvalues of $A$ approach 1. When all the elements of $\delta$ approach infinity ($\delta \rightarrow \infty$), the matrix $A$ turns into a matrix of ones, and we can show using the eigenvector equation $A x = \lambda x$, that the eigenvalues of such a matrix are all zero, apart from one that equals $n$. Considering then that all the eigenvalues of $A$ are positive, because $A$ has to be positive definite, we see that as the correlation lengths increase, the largest eigenvalue of $A$ tends to $n$ and the smallest tends to zero. Therefore, the condition number of $A$, which tends to $1$ for $\delta \rightarrow 0$, increases with $\delta$ and eventually becomes infinite. In other words, the conditioning of $A$ deteriorates as the correlation lengths increase, and in the limit, $A$ becomes non invertible. An alternative interpretation of this behaviour, is that increasing the correlation lengths is equivalent to the design points getting closer with each other, and in the limit they offer no information to the random part of the GP, as the columns of $A$ become linearly dependent.
3.2. The nugget

A remedy for the ill-conditioning of $A$ is the addition of a small positive number $\nu$ on its main diagonal, which is called nugget or jitter. This is a technique related to Tikhonov regularisation and ridge regression. We define the correlation matrix that includes the nugget as

$$\tilde{A} = A + \nu I,$$  

(8)

where $I$ is the identity matrix. It is straightforward to show that the eigenvalues $\lambda_i$ of $\tilde{A}$ are related to the eigenvalues of $A$ by $\lambda_i = \lambda + \nu$. The two matrices also have the same eigenvectors. It can then be shown that $\kappa(\tilde{A}) \leq \kappa(A)$, implying that the matrix $\tilde{A}$ is better conditioned than $A$ for a given value of the correlation lengths. In the worst case scenario ($\delta \to \infty$), the condition number of $\tilde{A}$ is $\kappa(\tilde{A}_{\delta \to \infty}) = \frac{n + \nu}{\nu}$, which is finite, under the assumption $n < \infty$; hence, $\tilde{A}$ is always invertible. Similar conclusions on the effect of the nugget on the conditioning of the correlation matrix were also drawn in Ababou et al. (1994).

A matrix is considered ill-conditioned if the reciprocal of its condition number approaches the machine’s precision. For double precision, a matrix can be considered ill-conditioned if its condition number is larger than $10^{12}$ (Press et al., 1992). Solving the equation $\frac{n + \nu}{\nu} = 10^{12}$ w.r.t. $\nu$ yields a lower limit for the nugget that guarantees that $\tilde{A}$ is well-conditioned for all values of $\delta$ even for $\delta \to \infty$. In the following, we will see that since it is unlikely for the likelihood mode $\hat{\delta}$ to occur for arbitrarily large $\delta$, the value of the nugget can be reduced even further, or even set to zero, without making $\tilde{A}_{\delta = \hat{\delta}}$ ill-conditioned. Guidelines for the selection of a lower limit for the nugget were also given in Ranjan et al. (2011).

4. The effect of the nugget on the emulator

The inclusion of the nugget in the Gaussian process model has an impact that is not always negligible. Pepelyshev (2010) showed that the shape of the likelihood $L(\delta | y)$ is affected by the addition of the nugget and its modes might change. Similar conclusions about the effect of the nugget on the likelihood are drawn in this section. Furthermore, we investigate the implications on the emulator’s predictions, which are also shown to be significant. In the current section, the nugget is considered to be a small but fixed number. The next section treats the nugget as a free parameter that can be jointly optimised with $\delta$.

4.1. Adding the nugget to the GP model

In the ‘nugget-free’ model of Section 2, the posterior mean of the GP at the design points, $m(D)$, is exactly equal to the observations $y$ and the respective variance $V(D, D)$ is exactly zero. This is readily deduced from Eqs. (5) and (6), considering that $c(D, D) = T(D) = A$ and $h^2(D) = H$. Therefore, in the absence of the nugget the emulator interpolates the simulator’s runs. This is no longer true as soon as a nugget is added.

The addition of the nugget to the correlation matrix $A$ implies that the prior distribution for the simulator $f(x)$ is

$$p(f(x) | \beta, \sigma^2, \delta, \nu) \sim \mathcal{N}(H\beta, \sigma^2(A + \nu I)).$$

(9)

An interpretation of the above distribution is that the emulator contains some variability $\sigma^2 \nu$ that is not explained by the inputs $x$. An alternative parameterisation of the covariance structure as $\sigma^2(A/\nu + I)$ (De Oliveira, 2007) can be shown to be identical to the one in Eq. (9). Another equivalent parameterisation is $\sigma^2A + \nu I$, which has however the drawback of making the marginalisation of $\sigma^2$ analytically intractable, implying that $\sigma^2$ would have to be estimated jointly with $\delta$ or even marginalised numerically.

In De Oliveira (2007) it was shown that the non-informative prior for $\beta$ and $\sigma^2$ after the addition of the nugget is also $p(\beta, \sigma^2) \propto \sigma^{-2}$. Using the same procedure as in Section 2 we can show that the likelihood $L(\delta, \nu, y)$ and the posterior mean of $p(f(x) | y, \delta, \nu)$ are the same as those given in Eqs. (3) and (5) with the exception that $A$ is now replaced by $\tilde{A}$. The posterior variance conceals a subtle point, which is detailed below.

Consider the interpolation of the noisy data $f(x) = g(x) + \eta$, where $g(x)$ is the process of interest and $\eta$ is uncorrelated noise. In this case, we are normally interested in making inferences about the structured process $g(x)$ and not about the noisy process $f(x)$. The addition of the nugget to the GP model resembles the above setup, where the observed variability in $f(x)$ is split to a part that can be explained by $x$ and to an unexplained part $\eta$ that is attributed to the nugget. In the emulation of deterministic (and possibly stochastic) models however, we are interested in making inferences about the simulator $f(x)$ and not about an underlying, unknown process $g(x)$. Therefore, the unexplainable variability that is represented by the nugget should be included in the posterior variance. This leads to the following expression for the posterior variance

$$V(x, x') = \frac{\hat{\sigma}^2}{n-q-2} \left[ c(x, x') + \nu I - T(x)\tilde{A}^{-1}T'(x') + P(x)(H'\tilde{A}^{-1}H)^{-1}P'(x') \right],$$

(10)

The above argument can become clearer if we consider validating our simulator in the presence of some inactive inputs, that is, when we know a priori that the model runs contain some unexplainable variability. This variability will also be
present in the validation data set, and our posterior variance should account for it. The same should also be true, when we allow the simulator not to interpolate the data, for reasons suggested in Gramacy and Lee (2012), such as violations of the stationarity or correlation assumptions. These assumptions will be violated by the validation data as well, and our posterior variance should also account for it.

As mentioned previously, the addition of the nugget implies that the emulator no longer interpolates the training data $y$, but approximates them. A measure of this approximation is the Mean Squared Error (MSE) between the training points $y$ and the emulator’s posterior mean at the design points, $m(D)$, which we define as

$$\tilde{M}(\delta, \nu) \equiv \frac{1}{n} (y - m(D))' (y - m(D)).$$

Replacing $m(D)$ from Eq. (5) and noting that $I - \bar{A}^{-1} = \nu^{-1}$ we get

$$\tilde{M}(\delta, \nu) = \frac{\nu^2}{n} (y - H\hat{\beta})' \bar{A}^{-2} (y - H\hat{\beta}),$$

(11)

which is generally larger than zero. We can similarly show that the average posterior variance at the design points $D$ is also greater than zero, that is, $\tilde{V}(\delta, \nu) \equiv \text{tr}(V(D, D))/n > 0$.

Eq. (11) shows that the approximation error depends not only on the nugget but also on the values of the correlation lengths. The implication is that selecting a small $\nu$ does not guarantee a close fit of the emulator to the model runs, because this also depends on the value of $\delta$. This observation raises a question about the limits of this approximation. We investigate these limits by considering two special cases, $\delta \to \infty$ and $\delta \to 0$, which will help us appreciate what level of approximation accuracy is feasible for a fixed nugget size. The dependence of $M$ and $V$ on $\nu$ is dropped for the remainder of Section 4, as $\nu$ is considered fixed.

4.2. Behaviour of the GP for extreme correlation length values

The model’s behaviour for $\delta \to \infty$ depends on whether the regression functions contain a constant term or not. For this reason, they will be considered in turn. We examine first the regression functions that contain a constant term, a case we denote by $1 \in H$ because the column space of $H$ contains a vector of ones. The results of this section are derived in the Appendix.

The likelihood function can be shown to have the same non-zero value for both $\delta \to 0$ and $\delta \to \infty$, that is,

$$p(y|\delta \to \infty) = p(y|\delta \to 0) \text{ if } 1 \in H.$$  \hspace{1cm} (12)

This implies that the data are equally likely to have been generated by either values of $\delta$, and that the resulting likelihood is not integrable.

The emulator’s posterior mean for $\delta \to 0$ is

$$m(x) = h'(x)\hat{\beta}_0 \text{ for } x \notin D,$$

$$m(D) = \frac{1}{1 + \nu} \bar{y} + \frac{\nu}{1 + \nu} H\hat{\beta}_0$$

(13)

and for $\delta \to \infty$ is

$$m(x) \approx h'(x)\hat{\beta}_0, \hspace{1cm} \forall x \in \mathbb{R}^p \text{ if } 1 \in H,$$

(14)

where $\hat{\beta}_0 = (H'H)^{-1}H'y$ is the least squares estimate of $\beta$. We therefore see that for both $\delta \to 0$ and $\delta \to \infty$, the emulator’s output mainly coincides with that of a least square error model. The difference is that for $\delta \to 0$, the uncorrelated part of the GP allows the model to deviate locally at $x \in D$ and approximate the simulator’s output $y$ at the design points with a very small error.

Specifically, $M(\infty)$ is approximately equal to the MSE of a least squares model that uses the same regression functions $h(x)$, as it can be seen from Eq. (14). We can also show that for $\delta \to 0$

$$\tilde{M}(0) = \frac{\nu^2}{(1 + \nu)^2} \frac{(y - H\hat{\beta}_0)' (y - H\hat{\beta}_0)}{n},$$

(15)

which considering that typically $\nu \ll 1$, implies that it is smaller than $\tilde{M}(\infty)$ by a factor of $\nu^2$.

In conclusion, for both $\delta \to 0$ and $\delta \to \infty$, the emulator’s predictions largely coincide with those of the respective least squares model. However, for $\delta \to 0$, the uncorrelated part of the GP allows a local deviation of the output for $x \in D$, resulting in an approximation error that is smaller by a factor of $\nu^2$ than that obtained with $\delta \to \infty$.

When the regression functions do not include a constant term ($1 \notin H$), the analysis for $\delta \to 0$ is identical to that of the previous case. For infinite correlation lengths, the resulting model is closely related to an augmented model that has regression functions $h^*(x) = [1; h(x)]$ and $\delta \to \infty$. This can be seen as if the infinitely correlated random part of the GP takes up the role of the constant term that is missing from the regression functions. In the Appendix it is shown that $p(y|\delta \to \infty) > 0$, implying that the likelihood is again not integrable, and that for $\delta \to \infty$ the mean response of the emulator equals that of the augmented model, i.e. $m(x) = m^*(x)|\delta \to \infty$. $\tilde{M}(0)$ is also in the order of $\nu^2\tilde{M}(\infty)$. 
Fig. 1. Log likelihoods for the first example. Thick continuous line: \( \nu = 0 \), double precision. Dotted line: \( \nu = 0 \) 100 digits precision calculated with Maple. Thin continuous line: \( \nu = 10^{-12} \), double precision. The locations of the modes are \( \delta_1 = 0.12 \) and \( \delta_2 = 330 \).

**Table 1** Characteristic values of the 4 different emulators (1 \( \in \mathcal{H} \)).

<table>
<thead>
<tr>
<th>( \delta )</th>
<th>( \mathcal{L}(\delta) )</th>
<th>( \hat{\beta} )</th>
<th>( \hat{\sigma}^2 )</th>
<th>( \bar{M} )</th>
<th>( \bar{V} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \delta \to 0 )</td>
<td>-1.77</td>
<td>[36.1, -9.5]</td>
<td>1.17</td>
<td>9.3 ( \cdot 10^{-25} )</td>
<td>3.1 ( \cdot 10^{-12} )</td>
</tr>
<tr>
<td>( \delta = \delta_1 = 0.12 )</td>
<td>-1.06</td>
<td>[35.9, -9.4]</td>
<td>1.48</td>
<td>2.4 ( \cdot 10^{-24} )</td>
<td>4.0 ( \cdot 10^{-12} )</td>
</tr>
<tr>
<td>( \delta = \delta_2 = 330 )</td>
<td>1.59</td>
<td>[-3.2 ( \cdot 10^5 ), -8.9]</td>
<td>3.4 ( \cdot 10^{11} )</td>
<td>0.24</td>
<td>0.59</td>
</tr>
<tr>
<td>( \delta \to \infty )</td>
<td>-1.77</td>
<td>[36.1, -9.5]</td>
<td>1.17 ( \cdot 10^{12} )</td>
<td>1.77</td>
<td>0.93</td>
</tr>
</tbody>
</table>

4.3. **Examples**

We now present two examples that illustrate the results of the previous section. The results are based on an emulator built for surfebmn, which is a numerical implementation by Prof. John Shepherd of an Energy Balance Climate Model as described in North et al. (1981). For our example we vary 1 input, the albedo, and observe the mean surface temperature. The GP model is fitted on \( n = 10 \) runs of the simulator; for the first part of the example the GP mean function has a constant term, that is, \( h^\prime(x) = [1, x] \).

4.3.1. **Mean function with a constant term**

Fig. 1 shows the log likelihood of the correlation lengths, \( \mathcal{L}(\delta) \), with and without the nugget, calculated using different numerical precisions. Without the nugget, the likelihood drops as the value of \( \delta \) tends to infinity. Using the standard double precision arithmetic, the likelihood is computable only up to \( \delta \approx 1 \). For larger values of \( \delta \), the inversion of \( A \) initially becomes numerically unstable, and is eventually not possible in double precision arithmetic; thus the likelihood cannot be calculated. On the other hand, after the inclusion of the nugget, the likelihood is computable for all \( \delta \in [0, \infty) \), but it deviates significantly from the likelihood of the original model for large values of \( \delta \). Furthermore, an additional mode appears, which in this case is taller than the mode of the zero-nugget likelihood. This example demonstrates that the inclusion of a nugget even as small as \( 10^{-12} \) can alter the likelihood by introducing new modes, which may appear more likely than those obtained with \( \nu = 0 \).

Fig. 2 shows the emulators obtained using \( \nu = 10^{-12} \) and four different values of \( \delta \). Table 1 shows some characteristic values. Starting with the extreme values \( \delta \to 0 \) and \( \delta \to \infty \), we note in both cases the posterior means coincide with the least squares solution \( h^\prime(x)\hat{\beta}_0 \). The difference is that when \( \delta \to 0 \) the model is capable of deviating from this solution locally at \( x \in D \), effectively interpolating the design points. Table 1 shows that the log likelihood for \( \delta \to 0 \) equals the log likelihood for \( \delta \to \infty \) and that the estimates of \( \beta \) are in both cases the same. Additionally, \( M(0) \) is in the order of \( \nu^2 \bar{M}(\infty) \), and \( V(0) \) is in the order of \( \nu \bar{V}(\infty) \). Panels (b) and (c) of Fig. 2 show the emulators obtained using the two modes. The first emulator essentially interpolates the data, yielding very small MSE and mean variance at \( x \in D \), as it can be verified by Table 1. The emulator in panel (c) has a very smooth posterior mean and interprets most of the local variability as observational noise, with accordingly large \( \bar{M} \) and \( \bar{V} \). Note however, that the second mode arises due to the addition of the nugget, while the first is present even when \( \nu = 0 \).

4.3.2. **Mean function without a constant term**

We now build an emulator without including the constant term in the mean function. The mean function we use is \( h^\prime(x) = x \), which implies that the model of the previous section is the ‘augmented’ model mentioned in Section 4.2. The likelihoods with and without the nugget are virtually identical in shape with the likelihoods presented in Fig. 1. Fig. 3 shows
(a) $\delta \to 0$. (b) $\delta = \delta_1$. (c) $\delta = \delta_2$. (d) $\delta \to \infty$.

Fig. 2. Emulators obtained for different $\delta$ and $\nu = 10^{-12}$ ($1 \in H$). The thin line is the simulator’s output and the crosses represent the training data. The thick continuous line is the posterior mean of each emulator and the shaded area its posterior variance (95%).

Table 2

<table>
<thead>
<tr>
<th>Characteristic values of the 4 different emulators ($1 \notin H$).</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\delta \to 0$</td>
</tr>
<tr>
<td>$\mathcal{L}(\delta)$</td>
</tr>
<tr>
<td>$\hat{\beta}$</td>
</tr>
<tr>
<td>$\hat{\sigma}^2$</td>
</tr>
<tr>
<td>$\bar{M}$</td>
</tr>
<tr>
<td>$\bar{V}$</td>
</tr>
</tbody>
</table>

the emulators obtained for correlation lengths that correspond to the two modes, $\delta_1$ and $\delta_2$ as well as $\delta \to 0$ and $\delta \to \infty$ and a fixed nugget $\nu = 10^{-12}$. Table 2 shows some characteristic values.

It is notable that for $\delta \to \infty$, the emulator is almost identical to that obtained in Section 4.3.1 for $\delta \to \infty$, as it can be verified from panels (d) of Figs. 2 and 3 as well as from the values of $\bar{M}$ and $\hat{\beta}$ in the last columns of Tables 1 and 2. This shows, that in the absence of a constant term in the mean function, this role is taken up by the random part of the GP for $\delta \to \infty$. The emulator obtained for $\delta = \delta_2$ is also virtually identical to the respective emulator of Section 4.3.1, as it can be verified by panels (c) of Figs. 2 and 3 as well as from the values of $\bar{M}$ and $\hat{V}$ in Tables 1 and 2. On the other hand, the emulator obtained for $\delta = \delta_1$, although it essentially interpolates the model runs, has very narrow uncertainty intervals compared to the respective emulator of Section 4.3.1.

In conclusion, this section has demonstrated that the addition of a nugget to the GP model, regardless of how small its value is, can alter significantly the model’s likelihood and can produce emulators that have an approximation error in the order of the corresponding least squares model. This directly contradicts a common belief that wants the addition of the nugget only to stabilise the calculations without affecting the model itself.

5. Varying the nugget size

In this section we consider the nugget as a variable parameter of the model, which can be optimised jointly with the correlation lengths. This will reveal the existence of two different types of likelihood modes and will attempt to explain a negative correlation that has been reported to exist between $\delta$ and $\nu$. 
Fig. 3. Emulators obtained for different $\delta$ and $\nu = 10^{-12}$ ($1 \not\in H$). The thin line is the simulator's output and the crosses represent the training data. The thick continuous line is the posterior mean of each emulator and the shaded area its posterior variance (95%).

Fig. 4. Joint log likelihood $L(\delta, \nu)$, plotted as a function of (a) $\delta$ and $\nu$ and (b) $\delta$ and $\nu^2$. 

Fig. 4(a) shows the log likelihood of the model used in Section 4.3.1, which is now a function of both $\delta$ and $\nu$, i.e. $L(\delta, \nu)$. The contours denote the paths across which the log likelihood takes the values $-1.1, 0$, and $1.4$. The two most discernible characteristics of this surface are the two ridges, one parallel to the axis of $\nu$ and a second one that is diagonal.

The horizontal ridge indicates that the likelihood is virtually constant for a large range of $\nu$ when $\delta$ equals 0.12, which is mode $\delta_1$ in Fig. 1. In other words, as long as the nugget has a small value, it has almost no effect on the likelihood when $\delta = \delta_1$. The diagonal ridge on the other hand, shows a negative correlation between $\delta$ and $\nu$ and appears for large $\delta$. In this case, the model tries to maintain a constant product of $\nu \hat{\sigma}^2$ and achieves this by increasing the correlation lengths, and subsequently the value of $\hat{\sigma}^2$, as the value of $\nu$ decreases. Plotting the value of the log likelihood as a function of $\delta$ and $\nu^2$ shows the diagonal ridge aligned with the $\delta$-axis (Fig. 4(b)).

Fig. 5 shows the two approximation errors $\bar{M}$ and $\bar{V}$ as a function of $\delta$ and $\nu$. Setting the correlation length to $\delta_1$, the nugget controls the degree of the approximation error, which can become arbitrarily small, or even zero for $\nu = 0$, assuming that
the calculations are feasible in the given arithmetic precision. The second likelihood ridge on the other hand, lies across paths of approximately constant values of $\bar{M}$ and $\bar{V}$.

The above analysis shows that for a given nugget size, two different mode types might appear in the likelihood $L(\delta | y)$. The first is almost invariant to the value of the nugget, as long as this is relatively small. For this mode type, the approximation errors $M$ and $V$ can become very small by decreasing the value of $\nu$, or even zero if the numerical precision allows. We will refer to these modes as interpolating or Type I modes.

The second mode type is characterised by a roughly constant approximation error. For this reason we will refer to these modes as approximating or Type II modes. Decreasing the value of $\nu$ can push the mode to higher values of $\delta$, giving rise to the diagonal ridge in Fig. 4(a); the approximation errors however, remain relatively constant. It is possible that Type II modes appear only for a particular set of $\delta$ and $\nu$ values, which in Fig. 4(a) would resemble a peak instead of a ridge. In our experience, this variant of a Type II mode is more often encountered in models that have some intrinsic unexplainable variability, such as stochastic models or models with inactive inputs. In conclusion, the fundamental difference between Types I and II modes is that the latter arise only in the presence of a nugget, whereas the former can exist even for $\nu = 0$, if the numerical precision of the machine allows.

Recall that in Section 3.2 we suggested a method for finding a lower limit for $\nu$ and hinted that in some occasions its value can be decreased even further. Once a Type I mode is found, we can fix $\delta$ and set $\nu$ to the smallest value that yields a well-conditioned $\tilde{A}$ (this value can often be $\nu = 0$). Fixing $\nu$ at its new value, we can re-optimise $L(\delta)$, although this is optional, as decreasing $\nu$ has a negligible effect on the location of Type I modes. The above method can be used to find a set of $\delta, \nu$ that minimises the approximation errors $\bar{M}$ and $\bar{V}$.

6. Controlling the approximation

We have established that even though after the inclusion of the nugget the emulator always approximates the data, some likelihood modes yield essentially interpolating emulators. In the emulation of deterministic models this is often desirable, therefore, one might be interested in avoiding Type II (approximating) likelihood modes. This is also very relevant in case we are interested in sampling the posterior distribution of $\delta, \nu$, as an intermediate step to a fully Bayesian MCMC emulator.

Even when we are willing to accept some approximation error between the emulator and the simulator, we still believe that controlling the approximation level is a sensible precaution. Take for example Section 4.3. Had we used the mean function of Section 4.3.2 we could have ended up with the emulator of Fig. 3(c), which would most likely validate, since the simulator lies within the 95% interval of the posterior variance. Requiring a closer match between the emulator and the simulator, would have led us to choose the emulator of Fig. 3(b), which would most certainly not validate, as its posterior variance is rather narrow. This could have prompted us to revisit our prior model, possibly by updating the structure of the mean function, which could have led us to the emulator of Fig. 2(b) that we believe represents more accurately the underlying simulator.

A practical method for automatically controlling the approximation error $\bar{M}$ is presented in the following. In Section 4.2 we showed that the maximum value of $\bar{M}$ equals that of a least squares model, is achieved as $\delta \to \infty$ and is independent of $\nu$. Using this observation we formulate the following penalty term

$$\pi(\delta, \nu) = \exp \left\{ -2 \frac{M(\delta, \nu)}{\epsilon M(\infty)} \right\}.$$  \hspace{1cm} (16)

The scalar $\epsilon$ defines the fraction of the maximum error $\bar{M}(\infty)$ that is considered acceptable. For $M(\delta, \nu) < \epsilon M(\infty)$ the log penalty takes values in the interval $[0, -2]$, which was chosen because of its association with the 95% mass of a distribution.
Fig. 6. Penalised log likelihood for three different values of $\epsilon$. The reference contour lines are the same as in Fig. 4.

Due to the range of $\bar{M}$, the value of the log penalty drops very rapidly for errors larger than $\epsilon \bar{M}(\infty)$. Multiplying the likelihood $L(\delta, \nu | y)$ with $\pi(\delta, \nu)$ penalises heavily the regions of the $\delta, \nu$ parameter space that yield emulators with approximation error larger than $\epsilon \bar{M}(\infty)$, while leaving the rest unaffected. The expression for the penalised log likelihood is

$$L_p(\delta, \nu) = L(\delta, \nu) - 2 \frac{\bar{M}(\delta, \nu)}{\epsilon \bar{M}(\infty)},$$

where $L(\delta, \nu)$ is given by Eq. (3) after replacing $A$ with $\tilde{A}$.

Fig. 6 shows the penalised log likelihood of the model in Section 4.3.1 for three different values of the parameter $\epsilon$. For $\epsilon = 1$ (Fig. 6(a)), the penalised log likelihood is virtually identical to that of Fig. 4(a), except for the slight drop for large values of $\delta$ and $\nu$. Setting $\epsilon = 10^{-3}$ (Fig. 6(b)) results in discarding the approximating mode and retaining only the interpolating mode at $\delta_1$. Finally, imposing a much stricter limit using $\epsilon = 10^{-12}$ (Fig. 6(c)), has the additional effect of limiting the acceptable values of $\nu$, so as to achieve a further reduction in $\bar{M}$.

7. A 2-dimensional example

In this section we present a 2-dimensional example with data runs taken from the model of Section 4.3, this time varying two inputs, the albedo and the solar radiation. We use $n = 20$ runs to build our emulator and $np = 10$ runs to validate it. The design points are all drawn from a Latin hypercube, and are mapped to the $[0,1]$ interval. The simulator’s output with the training and validation points are shown in Fig. 7. The nugget is initially fixed to $\nu = 10^{-6}$.

The log likelihood, depicted in Fig. 8(a), has two distinct modes, $\delta_I = [2.25, 0.14]$ and $\delta_II = [27.3, 8.8]$. The first mode is a Type I (interpolating) mode, and as we will see, its location remains unchanged as the value of the nugget decreases. The second mode is a Type II (approximating) mode, which moves to higher values of $\delta$ as the nugget goes to zero. The posterior mean of the emulator built with $\delta_II$ is a smooth surface – similar to the emulator in Fig. 2(c) – which does not interpolate the training data. The emulator built with $\delta_I$ is an interpolating emulator, which tries to capture the simulator’s local variability. Fig. 8(b) shows the penalised log likelihood with $\epsilon = 10^{-3}$, where it can be seen that mode $\delta_II$ has been removed but $\delta_I$ has remained intact.
Fig. 8. (a) Log likelihood $\mathcal{L}(\delta)$ and (b) penalised log likelihood $\mathcal{L}_p(\delta)$ for the two dimensional example. The nugget was fixed at $\nu = 10^{-6}$.

Table 3
Characteristic values of the emulators obtained for 4 different values of $\delta$.

<table>
<thead>
<tr>
<th>$\delta$</th>
<th>$\mathcal{L}(\delta)$</th>
<th>$\bar{M}$</th>
<th>$\bar{V}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\delta \rightarrow 0$</td>
<td>$-1.09$</td>
<td>$8 \cdot 10^{-13}$</td>
<td>$2 \cdot 10^{-6}$</td>
</tr>
<tr>
<td>$\delta = \delta_I$</td>
<td>$9.13$</td>
<td>$7 \cdot 10^{-11}$</td>
<td>$6 \cdot 10^{-6}$</td>
</tr>
<tr>
<td>$\delta = \delta_{II}$</td>
<td>$9.46$</td>
<td>$0.13$</td>
<td>$0.25$</td>
</tr>
<tr>
<td>$\delta \rightarrow \infty$</td>
<td>$-1.09$</td>
<td>$0.76$</td>
<td>$1.17$</td>
</tr>
</tbody>
</table>

Table 4
Order of approximation errors for the emulator obtained with $\delta_I$ and decreasing values of nugget.

<table>
<thead>
<tr>
<th>$\nu$</th>
<th>$\bar{M}$</th>
<th>$\bar{V}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-10}$</td>
<td>$10^{-18}$</td>
<td>$10^{-10}$</td>
</tr>
<tr>
<td>$10^{-14}$</td>
<td>$10^{-26}$</td>
<td>$10^{-14}$</td>
</tr>
<tr>
<td>$0$</td>
<td>$10^{-29}$</td>
<td>$10^{-16}$</td>
</tr>
</tbody>
</table>

(a) Emulator built with $\delta_I$. (b) Emulator built with $\delta_{II}$.

Fig. 9. Individual prediction errors for the emulators obtained with $\delta_I$ and $\delta_{II}$. The horizontal lines indicate the $[-2, 2]$ interval. Errors outside this interval indicate a conflict between the emulator and the simulator.

Table 3 shows some characteristic values of the emulators obtained using the correlation lengths that correspond to the two modes as well as using the extreme values $\delta \rightarrow 0$ and $\delta \rightarrow \infty$. As in the one dimensional example, the mode $\delta_I$ yields values of $\bar{M}$ and $\bar{V}$ that are close to those obtained with zero correlation lengths and the mode $\delta_{II}$ yields errors close to those of a least squares model. Note that $\mathcal{L}(0)$ and $\mathcal{L}(\infty)$ are also equal.

Table 4 shows the reduction in the approximation errors $\bar{M}$ and $\bar{V}$ as the nugget decreases to 0, using the Type I mode $\delta_I$. We should note that the values of $\delta$ that maximised the likelihood for $\nu = 10^{-6}$ and $\nu = 0$ were different by less than $10^{-3}$. This was also the case for the value of the log likelihood. For $\nu = 0$, both $\bar{M}$ and $\bar{V}$ have a theoretical value of zero, and the values shown in the last column of Table 4 are due to round off error.

Fig. 9 shows some validation results for the emulators obtained with $\delta_I$ and $\delta_{II}$. The Individual Prediction Errors (Bastos and O’Hagan, 2009) are the differences between the emulator’s posterior mean and the simulator’s output divided by the emulator’s posterior standard deviation at the validation points. Individual prediction errors with an absolute value larger than 2, indicate a conflict between the emulator and the simulator. The diagnostic shows a potential conflict between the simulator and the emulator built with $\delta_I$, but this is not apparent for the emulator built with $\delta_{II}$. 

This example illustrates the benefits of assessing the fit of an emulator that uses a nugget. Had we used the nugget purely for ‘computational reasons’ we could have built an emulator with \( \delta \) perhaps being unaware of the fact that it does not interpolate the data, since the nugget was as small as \( 10^{-6} \). If on the other hand, we required a closer match between the emulator and the training data, either using the penalty term of Section 6, or by manually checking the approximation error, we would end up with mode \( \Delta \), and the resulting emulator would fail the diagnostic. A diagnostic failure would prompt us to revisit our GP prior model and/or obtain more training points. Both of these actions would be likely to result in a more accurate emulator. Even if our efforts to produce a valid Type I emulator failed, we could settle for an emulator with a Type II mode, but this would be a conscious decision, which would also flag the discrepancy between the emulator and the simulator, that could otherwise go unnoticed. Further investigation into this discrepancy could reveal features of the simulator such as non-stationarity or discontinuities.

8. Conclusion

The purpose of this work was to investigate the effect of the nugget on Gaussian process emulators. We have established that regardlessofits value, the addition of the nugget can result in emulators that approximatethe data with an MSE that equalsthat of an least squares model. The presence of a nugget is equivalent to the assumption that the simulator contains some variability that is not explainable by its inputs. We argue that this ‘unexplainable variability’ should be taken into account when we make inferences about the simulator. We have also shown that for a fixed nugget size, two different modes can appear in the likelihood of the correlation lengths. Type I modes are fairly insensitive to the value of the nugget, and can result in emulators with almost arbitrarily small approximation errors. Type II modes arise only in the presence of a nugget and result in emulators with a relatively fixed approximation error. Finally, we introduced a penalty term, that when added to the log likelihood of \( \delta \) and \( \nu \), it penalises solutions that have an approximation error greater than a specified level. This term automatically imposes restrictions on the amount of unexplainable variability that may be present in the simulator and can be useful both in optimisation and in numerical marginalisation of the model parameters.

Acknowledgements

We acknowledge the support of the RCUK basic technology initiative as part of the Managing Uncertainty in Complex Model (MUCM) project. We wish to thank the MUCM team members and in particular Prof. A. O’Hagan for several valuable and insightful discussions. We are also grateful to the anonymous referees and the associate editor, whose comments helped improve this paper.

Appendix. Model’s behaviour for infinite correlation lengths

A.1. Likelihood

For \( \delta \to \infty \) the correlation matrix \( \tilde{A} \) can be written as \( \tilde{A} = \nu I + 11' \), with 1 denoting a \((n \times 1)\) vector of ones. We give two identities from Berger et al. (2001) that will be needed in the following. For a non singular matrix \( U \) and a vector \( u \) it holds that \( U + uu' \) is also non singular and

\[
|U + uu'| = |U| (1 + u' U^{-1} u) \tag{A.1}
\]

and

\[
(U + uu')^{-1} = U^{-1} - \frac{U^{-1} uu' U^{-1}}{1 + u' U^{-1} u}. \tag{A.2}
\]

We can then show that for \( \delta \to \infty \)

\[
|\tilde{A}| = \nu^n (1 + n/\nu) \tag{A.3}
\]

and

\[
\tilde{A}^{-1} = \nu^{-1} \left[ I - \frac{11'}{v + n} \right]. \tag{A.4}
\]

Therefore, \( H' \tilde{A}^{-1} H \) can be written as

\[
H' \tilde{A}^{-1} H = \nu^{-1} \left[ H' H - \frac{H' 11' H}{v + n} \right]. \tag{A.5}
\]

Using the identity (A.1) with \( U = H' H \) and \( u = H' 1/\sqrt{v + n} \), we can write \( |H' \tilde{A}^{-1} H| \) as

\[
|H' \tilde{A}^{-1} H| = \nu^{-q} |H' H| \left[ 1 - \frac{1' (H' H)^{-1} H' 1}{v + n} \right], \quad \text{if } 1 \not\in H \tag{A.6}
\]
If $H$ has a column of ones, it is shown in Berger et al. (2001) that $\mathbf{1}'H(H'H)^{-1}H' = n$; hence the above expression simplifies to

$$|H'\tilde{A}^{-1}H| = v^{-1} \frac{|H'H|}{v + n}, \quad \text{if } \mathbf{1} \in H.$$  \hfill (A.7)

The expression for $\hat{\sigma}^2$ can be derived by rewriting $\hat{\sigma}^2$ as

$$\hat{\sigma}^2 = \mathbf{y}' \left[ \tilde{A}^{-1} - \tilde{A}^{-1}H(H'\tilde{A}^{-1}H)^{-1}H'\tilde{A}^{-1} \right] \mathbf{y}.$$  \hfill (A.8)

Substituting $\tilde{A}^{-1}$ from (A.4), which we rewrite as $\tilde{A}^{-1} = v^{-1}Q$, with an obvious definition for $Q$, we get

$$\hat{\sigma}^2 = v^{-1} \mathbf{y}' \left[ (Q - HQ(H'QH)^{-1}H')Q \right] \mathbf{y}, \quad \text{if } \mathbf{1} \not\in H.$$  \hfill (A.9)

If $\mathbf{1} \in H$, the above expression can be simplified, using Lemma 4 from Berger et al. (2001), which states that for two non-singular matrices $U$ and $V$, such that $U = V - \mathbf{1}'$, it holds

$$V^{-1} - V^{-1}H(H'V^{-1}H)^{-1}H'V^{-1} = U^{-1} - U^{-1}H(H'U^{-1}H)^{-1}H'U^{-1}.$$  

Using the above result with $U = \nu I$ and $V = \tilde{A}$, we can write $\sigma^2$ as

$$\hat{\sigma}^2 = v^{-1} \mathbf{y}' \left[ I - H(H'H)^{-1}H' \right] \mathbf{y}, \quad \text{if } \mathbf{1} \in H.$$  \hfill (A.10)

Note that this is equal to $\hat{\sigma}^2$, obtained for $\delta \to 0$, divided by $v$. Combining the results from Eqs. (A.3), (A.7) and (A.9) it turns out that $p(\mathbf{y}|\delta \to \infty) = p(\mathbf{y}|\delta \to 0)$, when $\mathbf{1} \in H$. In the case $\mathbf{1} \not\in H$, Eqs. (A.3), (A.6) and (A.8) give the expression for $p(\mathbf{y}|\delta \to \infty)$, which is also greater than zero, although there is no simple connection to $p(\mathbf{y}|\delta \to 0)$.

**A.2. Posterior mean**

For $\delta \to \infty$, using the expression for $H'\tilde{A}^{-1}H$ from Eq. (A.5) and the identity (A.2) with $U = H'H$ and $\mathbf{u} = H'\mathbf{1}/\sqrt{\nu + n}$ we can write

$$\begin{align*}
(H'\tilde{A}^{-1}H)^{-1} & = v \left[ (H'H)^{-1} + \frac{(H'H)^{-1}H' \mathbf{1}' \sqrt{\nu + n} (H'H)^{-1}}{1 - \frac{H'H^{-1}H' \mathbf{1}'}{\nu + v}} \right] \\
& = v \left[ (H'H)^{-1} + \frac{(H'H)^{-1}H'H'}{\nu + v} \right].
\end{align*}$$

Right multiplying with $H'\tilde{A}^{-1} \mathbf{y}$, and substituting $\tilde{A}^{-1}$ from Eq. (A.4), it can be shown after some algebra that

$$\hat{\beta}_\infty = (H'\tilde{A}^{-1}H)^{-1}H'\tilde{A}^{-1}\mathbf{y} = (H'H)^{-1}H' \left[ \mathbf{1} + \frac{H'H^{-1}H'}{\nu + v - \mathbf{1}'H(H'H)^{-1}H'} \right] \mathbf{y}, \quad \text{as } n \to \infty.$$  \hfill (A.10)

In the special case that $\mathbf{1} \in H$, it is shown in Berger et al. (2001) that $\mathbf{1}'H(H'H)^{-1}H' = \mathbf{1}'$. Using this identity in the above expression we get

$$\hat{\beta}_\infty = (H'H)^{-1}H' \mathbf{y}, \quad \text{if } \mathbf{1} \in H$$

which shows that $\hat{\beta}_\infty = \hat{\beta}_0$, if $\mathbf{1} \in H$.

Noting that $T(\mathbf{x}) = \mathbf{1}'$ and replacing $\tilde{A}^{-1}$ in Eq. (5) with Eq. (A.4), we can write the posterior mean as

$$m(\mathbf{x}) = h'(\mathbf{x}) \hat{\beta}_0 + \frac{\mathbf{1}'}{v + n} \left( \mathbf{y} - H\hat{\beta}_0 \right).$$

In the case $\mathbf{1} \in H$, $\mathbf{y} - H\hat{\beta}_0$ is the least squares residual and $\mathbf{1}'/(v + n)$ is an approximate mean operator. Therefore, the second part of the right-hand side in the above equation is approximately zero; hence $m(\mathbf{x}) \approx h'(\mathbf{x})\hat{\beta}_0$.

When the regression coefficients do not include a constant term, for $\delta \to \infty$, the model is closely related to an ‘augmented’ model with regression coefficients $h^*(\mathbf{x}) = [1; h(\mathbf{x})]$. The least squares estimate of the latter model’s regression coefficients is

$$\hat{\beta}_0 = \hat{\beta}_\infty = (H^*H^*)^{-1}H^*\mathbf{y}.$$  

The matrix $H^*H^*$ can be written as

$$H^*H^* = \begin{bmatrix} \mathbf{1}'\mathbf{1} & \mathbf{1}'H \\ H'\mathbf{1} & H'\mathbf{H} \end{bmatrix}.$$
Using the formula for the inversion of partitioned matrices (Press et al., 1992), we can show that
\[
(H'H')^{-1} = \begin{bmatrix}
P & \tilde{Q} \\
R & \tilde{S}
\end{bmatrix}
\]
with
\[
P = (11' - 1'H(H'H)^{-1}H')^{-1},
\]
\[
\tilde{Q} = -\tilde{P}'H(H'H)^{-1},
\]
\[
R = -(H'H)^{-1}H'\tilde{P}
\]
and
\[
\tilde{S} = (H'H)^{-1} + (H'H)^{-1}H'\tilde{P}'H(H'H)^{-1}.
\]

The \(\hat{\beta}_n^*\) coefficients can be written as
\[
\hat{\beta}_n^* = \begin{bmatrix} \tilde{P}' + \tilde{Q}H' \\ \tilde{R}' + \tilde{S}H' \end{bmatrix} y.
\]

The q last coefficients of \(\hat{\beta}_\infty^n\) can be written as
\[
[\tilde{R}' + \tilde{S}H']y = (H'H)^{-1}H' \left[ I + \frac{1}{n} \frac{11'(H'H)^{-1}H' - 11'}{1 - 1'H(H'H)^{-1}H'} \right] y
\]
which is approximately equal to \(\hat{\beta}_\infty^n\) (Eq. (A.10)) for \(1 \not\in H\), considering that \(n + v \approx n\).

We now show that the posterior mean of the two models is the same. For the augmented model we can write
\[
m^*(x) = \begin{bmatrix} 1 \\ h(x) \end{bmatrix} \begin{bmatrix} \tilde{P}' + \tilde{Q}H' \\ \tilde{R}' + \tilde{S}H' \end{bmatrix} y + 1\tilde{A}^{-1} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} \tilde{P}' + \tilde{Q}H' \\ \tilde{R}' + \tilde{S}H' \end{bmatrix} y - H \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} \tilde{P}' + \tilde{Q}H' \\ \tilde{R}' + \tilde{S}H' \end{bmatrix} y.
\]
which can be expanded as
\[
m^*(x) = \begin{bmatrix} \tilde{P}' + \tilde{Q}H' \\ \tilde{R}' + \tilde{S}H' \end{bmatrix} y + h'(x) \begin{bmatrix} \tilde{R}' + \tilde{S}H' \end{bmatrix} y + 1\tilde{A}^{-1} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} \tilde{P}' + \tilde{Q}H' \\ \tilde{R}' + \tilde{S}H' \end{bmatrix} y - H \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} \tilde{P}' + \tilde{Q}H' \\ \tilde{R}' + \tilde{S}H' \end{bmatrix} y.
\]
Considering that \(1\tilde{A}^{-1}1 = \frac{n}{n + v} \approx 1\) (from Eq. (A.4)), we see that the terms including \(\tilde{P}' + \tilde{Q}H\) cancel out, so the expression reduces to
\[
m^*(x) = h'(x)\hat{\beta}_\infty^n y + 1\tilde{A}^{-1} \begin{bmatrix} y - H\hat{\beta}_\infty^n y \end{bmatrix}.
\]
which equals the posterior mean of the original model for \(\delta \to \infty\).

References


