ROYAL STATISTICAL SOCIETY
DATA | EVIDENCE | DECISIONS

Journal of the Royal Statistical Society
Applied Statistics
Series C

Appl. Statist. (2016)

Multivariate emulation of computer simulators: model selection and diagnostics with application to a humanitarian relief model

Antony M. Overstall

University of Glasgow, UK

and David C. Woods

University of Southampton, UK

[Received September 2014. Final revision November 2015]

Summary. We present a common framework for Bayesian emulation methodologies for multivariate output simulators, or computer models, that employ either parametric linear models or non-parametric Gaussian processes. Novel diagnostics suitable for multivariate covariance separable emulators are developed and techniques to improve the adequacy of an emulator are discussed and implemented. A variety of emulators are compared for a humanitarian relief simulator, modelling aid missions to Sicily after a volcanic eruption and earthquake, and a sensitivity analysis is conducted to determine the sensitivity of the simulator output to changes in the input variables. The results from parametric and non-parametric emulators are compared in terms of prediction accuracy, uncertainty quantification and scientific interpretability.

Keywords: Bayesian emulation; Computer experiment; Gaussian process; Lightweight emulator; Non-parametric regression

1. Introduction

There are many systems in the physical, social and engineering sciences for which physical experimentation is infeasible or unaffordable. Some examples include investigations on ecosystems, infectious diseases, climate change and galaxy formation (see Kennedy *et al.* (2006) for some case-studies). In such situations, it is now common for the scientist or engineer to develop a *simulator*, or computer model, that provides an approximation of the observed response from the physical system. In essence, the simulator is a deterministic or stochastic mathematical function that maps the inputs of a system to a prediction of its outputs.

A simulator that has been successfully calibrated and validated, perhaps by using physical data, can be employed for a number of tasks including prediction, optimization, and sensitivity and uncertainty analyses (Kennedy and O'Hagan, 2001). However, both calibrating and exploiting the simulator typically require very many simulator evaluations. For complex problems, the computational expense of the simulator means that brute force approaches to these problems are infeasible, taking many hours, days or even weeks. Therefore, a fundamental step in understanding and using simulators is often the construction of a statistical *emulator*, or metamodel, through a *computer experiment* (Sacks *et al.*, 1989). Here, the simulator is run at a

Address for correspondence: David C. Woods, Mathematical Sciences, University of Southampton, Highfield, Southampton, SO17 1BJ, UK.
E-mail: D.Woods@southampton.ac.uk

 0035–9254/16/65000
Published by John Wiley & Sons Ltd on behalf of the Royal Statistical Society.

carefully selected collection of combinations of the input variables and the resulting evaluations are treated as data to which a statistical model, the emulator, is fitted. The emulator can then be used to produce fast predictions of the output of the simulator for any values of the input variables, along with an associated measure of the prediction uncertainty. The emulator can then replace and supplement the simulator in both statistical calibration and scientific investigation. For more on computer experiments, see Santner *et al*. (2003), Fang *et al*. (2006) and Levy and Steinberg (2010).

A Bayesian approach is very natural when constructing statistical emulators (O'Hagan, 2006) with the chosen statistical model treated as a prior distribution on the simulator outputs and prediction, with associated uncertainty quantification, via the posterior predictive distribution (see Section 1.2). Typically, a non-parametric Gaussian process (GP) regression model (Rasmussen and Williams, 2006) is employed; its advantages include flexibly adapting to the simulator evaluations and, for deterministic simulators, interpolating between data points. However, for some simulators, these advantages may be more than offset by the computational expense of estimating the GP model, and simpler and more computationally efficient models, such as multivariate linear regression, may be effective and more interpretable. Whatever statistical approach is taken to constructing the emulator, an important step is assessing its adequacy through formal statistical diagnostics (Bastos and O'Hagan, 2009).

Frequently, each run of a simulator outputs a multivariate response, perhaps as a result of a time series or other dynamic process. The purpose of this paper is to present a Bayesian framework for covariance separable emulation of multivariate simulators using parametric and non-parametric models and to develop novel model diagnostic procedures that are appropriate for such emulators. As part of our presentation, we unify the *multivariate GP emulator* of Conti and O'Hagan (2010) and the *lightweight emulator* of Rougier (2007). Through an application to a simulator of a humanitarian relief mission, we demonstrate effective emulation, model selection and model checking for multivariate problems with a mixture of continuous and categorical input variables.

1.1. A humanitarian relief simulator with multivariate dynamic output

Simulators have a long history of use in military and civilian emergency planning (see, for example, Ingber *et al*. (1991)). The 'Diplomatic and military operations in a non-warfighting domain' (DIAMOND) simulator (Taylor and Lane, 2004) is an emergency planning simulator for modelling peace support operations such as humanitarian relief and peace keeping. DIAMOND is mission based, with high-level operational plans deconstructed into missions for individual units. It can model the actions and interactions between a wide range of agents, including military forces in non-warfighting roles, non-governmental organizations (NGOs), indigenous forces and civilians. A range of environmental and infrastructure features can also be varied.

Our application of DIAMOND provides a deterministic model of a humanitarian relief mission to Sicily after an earthquake and subsequent eruption of Mount Etna. Etna is an active stratovolcano on the east coast of Sicily near the cities of Catania and Giarre (Fig. 1). It has been designated a 'decade volcano' by the International Association of Volcanology and Chemistry of the Earth's Interior and the United Nations owing to its history of large eruptions and proximity to populated areas. Historically, more fatalities have been caused by earthquakes in the region, such as in 1693 when an earthquake of estimated magnitude 7.4 on the moment magnitude scale devastated the area and caused about 12000 deaths in Catania (about 63% of the population at the time; Guidoboni *et al*. (2007)).

Fig. 1. Map of Sicily, showing the locations of Mount Etna, Giarre, Catania, a possible humanitarian task force base and the capital city Palermo

The simulator models damage to the food supply, hospitals and housing (shelter) in Giarre and Catania resulting from the earthquake and eruption. An NGO launches a humanitarian relief operation which has two missions:

(a) *food aid mission*—to supply food to Catania and Giarre by helicopter from the NGO base;
(b) *repair mission*—to transport engineers from the NGO base to Giarre and Catania, where they repair the food supply infrastructure and/or the shelter.

We consider a scenario that was designed by the UK Defence Science and Technology Laboratory for the explicit and sole aim of model testing; the scenario is not intended to support any real world decisions. Here, the NGO has four helicopter teams, two engineering teams and a single food depot. Two helicopter teams are assigned to the food aid mission and the others to transporting the engineers for the repair mission.

The simulator has $p = 13$ input variables, which represent the scale of the disaster and features of the humanitarian relief operation (Table 1). Of these variables 11 are continuous, with the other two being categorical with each having two levels. Input variables x_1–x_6 determine the effect of the earthquake and eruption on the population of Giarre and Catania by specifying the capacity of hospitals, shelter and food supply immediately following the disaster. The specification of these input variables creates a shortfall between population and shelter and/or food supply, leading to casualties.

The remaining input variables (five continuous; two categorical) control certain features of the humanitarian relief mission. The continuous input variables are self-explanatory with the exception of x_7: the weighting of the engineer toolbox. This variable controls the relative importance given to repairing shelter and the food supply by the two engineering teams; $x_7 = 0$ and $x_7 = 1$ correspond to engineers only repairing the shelter or the food supply respectively.

The two levels for categorical variable x_{12} correspond to supplying food aid to both Giarre and Catania or to Catania alone. Although the second option is perhaps morally and politically unappealing, it may be practically relevant as there can be a much greater shortfall between the available and required food in Catania. Simulation modelling allows investigation of the

Table 1. Input variables for the humanitarian relief mission simulator†

Name	*Symbol*	*Range*	*Units*
Continuous input variables			
Giarre hospital capacity	x_1	(135, 270)	people day^{-1}
Giarre shelter capacity	x_2	(13500, 27000)	people day^{-1}
Giarre food supply capacity	x_3	(13500, 27000)	people day^{-1}
Catania hospital capacity	x_4	(2000, 3000)	people day^{-1}
Catania shelter capacity	x_5	(200000, 300000)	people day^{-1}
Catania food supply capacity	x_6	(200000, 300000)	people day^{-1}
Weighting of the engineer toolbox	x_7	(0, 1)	—
Planning time for the humanitarian mission	x_8	(36, 60)	h
Helicopter cruise speed	x_9	(220, 270)	km h^{-1}
Helicopter cargo capacity	x_{10}	(7000, 7500)	—
Engineer ground speed	x_{11}	(0, 10)	km h^{-1}
		Levels	
Categorical input variables			
Recipient of food aid	x_{12}	{Giarre and Catania, Catania only}	
Location of NGO base	x_{13}	{continental Europe, task force base}	

†The units of measurement for helicopter cargo capacity are specific to this simulator. Note that the initial populations in the simulator of Giarre and Catania are 27000 and 300000 respectively. Under normal circumstances, the simulator expects only 1% of the population per day to require hospital treatment.

effect of potentially unattractive options. For x_{13}, the two levels correspond to the NGO base being

(a) in continental Europe or
(b) part of a military task force located on a fleet of ships in the Strait of Messina between Italy and Sicily (see Fig. 1).

Each run of the simulator is defined by a setting for x_1–x_{13}. The output from each simulator run is the number of civilian casualties that have occurred on each of days 2, 3, 4, 5 and 6 following the disaster. Therefore, the output for each run is a five-dimensional vector.

1.2. Bayesian emulators

A Bayesian approach will be taken to constructing an emulator for the DIAMOND simulator. Let $\mathbf{x} = (x_1, \ldots, x_p)^{\mathrm{T}} \in \mathcal{X} \subset \mathbb{R}^p$ denote the vector of p input variables, with $\mathcal{X}$ the p-dimensional input space. The simulator is assumed to be a black box function, $f : \mathcal{X} \to \mathcal{Y} \subset \mathbb{R}^k$, with $\mathcal{Y}$ the k-dimensional output space, i.e.

$$f(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_k(\mathbf{x}))^{\mathrm{T}}$$

is the $k \times 1$ output vector from the simulator at input combination $\mathbf{x}$. An emulator for $f(\cdot)$ is a prediction equation that provides a surrogate for $f(\mathbf{x}_0)$, where $\mathbf{x}_0$ is an input combination at which the simulator has not previously been evaluated.

For a collection of input combinations $\zeta = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, with $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^{\mathrm{T}}$, the simulator outputs are collated into an $n \times k$ output matrix

$$Y = \begin{pmatrix} f(\mathbf{x}_1)^{\mathrm{T}} \\ \vdots \\ f(\mathbf{x}_n)^{\mathrm{T}} \end{pmatrix}.$$

A priori, we assume that Y is a realization from a probability distribution, specified up to a $d \times 1$ vector of unknown parameters $\boldsymbol{\theta} \in \Theta$, with $\Theta \subset \mathbb{R}^d$ the parameter space. After running the simulator for the input combinations in ζ, the emulator is constructed as the posterior predictive distribution (see, for example, O'Hagan and Forster (2004), page 89) of $\mathbf{y}_0 = f(\mathbf{x}_0)$, given by

$$\pi(\mathbf{y}_0|Y) = \int_{\Theta} \pi(\mathbf{y}_0|\boldsymbol{\theta}, Y)\, \pi(\boldsymbol{\theta}|Y)\, \mathrm{d}\boldsymbol{\theta}. \tag{1}$$

Here, $\pi(\boldsymbol{\theta}|Y)$ is the posterior density function for $\boldsymbol{\theta}$, which is found by using Bayes theorem, and $\pi(\mathbf{y}_0|\boldsymbol{\theta}, Y)$ is the conditional posterior predictive density for $\mathbf{y}_0$.

In the remainder of this paper, methodology for multivariate Bayesian emulation is developed and applied. In Section 2, the detailed methodology that was used to obtain the posterior predictive distribution is described for both multivariate GPs and linear models. In Section 3, model selection and diagnostics for multivariate emulators are developed and discussed. In Section 4, results are presented from applying the methodology to emulating the DIAMOND simulator. Section 5 gives a brief discussion.

Code to fit the emulators that are described in this paper and the training and test data sets are available from

`http://wileyonlinelibrary.com/journal/rss-datasets`

2. Multivariate emulation via the posterior predictive distribution

In this section, the posterior predictive distribution is derived for a general class of multivariate linear models that includes GP models and linear regression models. As such, the multivariate GP emulator of Conti and O'Hagan (2010) and lightweight emulator of Rougier (2007) are special cases. We also demonstrate how the multivariate GP emulator can include categorical input variables by using the distance metrics of Qian *et al*. (2008).

Our basic modelling assumption is that any finite set of multivariate responses has a joint matrix normal distribution (Dawid, 1981) with mean function a linear combination of unknown model parameters and a separable covariance structure with, potentially, correlations between outputs from the same run and also between different runs of the simulator, i.e. for $n \times k$ response matrix Y

$$Y|B, \Sigma, A \sim \mathrm{MN}_{n,k}(HB, \Sigma, A), \tag{2}$$

where HB is the $n \times k$ mean matrix and Σ and A are respectively $k \times k$ and $n \times n$ positive definite column and row scale matrices. Note that

$$\mathrm{vec}(Y)|B, \Sigma, A \sim N_{nk}\{\mathrm{vec}(HB), \Sigma \otimes A\}$$

is a multivariate normal distribution, where $\mathrm{vec}(\cdot)$ denotes the vectorization function that stacks columns of a matrix and '$\otimes$' denotes the Kronecker product.

In distribution (2), the matrix H is the $n \times m$ model matrix with ith row given by $h(\mathbf{x}_i)^{\mathrm{T}}$, where $h : \mathcal{X} \to \mathcal{H} \subset \mathbb{R}^m$ is a known function of the simulator inputs ($i = 1, \ldots, n$). For example, if $h(\mathbf{x}) = (1, x_1)$, then the model contains an intercept and a linear term in x_1. If some input variables are categorical, then we define the appropriate elements of $h(\mathbf{x}_i)$ through the usual

constraints, e.g. corner point or sum to zero. The matrix B is an $m \times k$ matrix of unknown regression parameters.

The separability of the covariance structure that is implied by this matrix normal distribution results in a common scale matrix Σ for the k multivariate responses at each of the n simulator runs. An emulator with a separable covariance structure is easier both to implement and to interpret. If diagnostic measures (see Section 3.1) suggest inadequacy of the separable emulator, alternative methodologies could be employed (see, for example, Fricker *et al.* (2013), and references therein).

If homogeneity of variance across the simulator runs is assumed, i.e. $\text{var}\{f(\mathbf{x}_i)\} = \Sigma$ for all $i = 1, \ldots, n$, then A can be specified as a correlation matrix. For the multivariate GP emulator, we define A through a stationary correlation function, and we set the ijth entry equal to $a_{ij} = c(|\mathbf{x}_i - \mathbf{x}_j|; \mathbf{r})$, i.e. the correlation between any two rows of Y depends only on the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ $(i, j = 1, \ldots, n)$ and a vector of unknown correlation parameters $\mathbf{r}$. The lightweight emulator is defined as a special case with

$$c(\mathbf{x}_i, \mathbf{x}_j; \mathbf{r}) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if otherwise.} \end{cases}$$

Thus we can replace conditioning on A in distribution (2) by conditioning on $\mathbf{r}$.

We use the conditionally conjugate (given $\mathbf{r}$) matrix–normal–inverse Wishart prior distribution for B and Σ, denoted $\text{MNIW}_{m,k}(M, \Omega, S, \delta)$, where

$$B|\Sigma, \mathbf{r} \sim \text{MN}_{m,k}(M, \Sigma, \Omega), \tag{3}$$

$$\Sigma|\mathbf{r} \sim \text{IW}_k(S, \delta). \tag{4}$$

Here, IW_k denotes the inverse Wishart distribution for $k \times k$ positive definite matrices, M, Ω and S are the $m \times k$, $m \times m$ and $k \times k$ matrices of hyperparameters respectively and $\delta > 0$ is the prior degrees of freedom. The corresponding probability density function is given in section 1 of the on-line supplementary material, up to a normalizing constant; see also Rougier (2007).

Using this prior distribution the conditional posterior distribution, given $\mathbf{r}$, is

$$B, \Sigma|Y, \mathbf{r} \sim \text{MNIW}_{m,k}(\hat{M}, \hat{\Omega}, \hat{S}, \hat{\delta})$$

(see section 2 of the on-line supplementary material), where

$$\begin{aligned} \hat{\Omega} &= (H^{\mathrm{T}} A^{-1} H + \Omega^{-1})^{-1}, \\ \hat{M} &= \hat{\Omega}(H^{\mathrm{T}} A^{-1} Y + \Omega^{-1} M), \\ \hat{S} &= Y^{\mathrm{T}} A^{-1} Y + M^{\mathrm{T}} \Omega^{-1} M + S - \hat{M}^{\mathrm{T}} \hat{\Omega}^{-1} \hat{M}, \\ \hat{\delta} &= \delta + n. \end{aligned}$$

To predict the simulator output $Y_0 = (f(\mathbf{x}_{01}), \ldots, f(\mathbf{x}_{0n_0}))^{\mathrm{T}}$ at a set of n_0 test inputs, $\zeta_0 = \{\mathbf{x}_{01}, \ldots, \mathbf{x}_{0n_0}\}$, we first define the joint conditional distribution of Y and Y_0,

$$\begin{pmatrix} Y \\ Y_0 \end{pmatrix} \Big| B, \Sigma, \mathbf{r} \sim \text{MN}_{n+n_0,k}\left\{ \begin{pmatrix} H \\ H_0 \end{pmatrix} B, \Sigma, \begin{pmatrix} A & T \\ T^{\mathrm{T}} & A_0 \end{pmatrix} \right\}, \tag{5}$$

where H_0 is the $n_0 \times m$ matrix with uth row $h(\mathbf{x}_{0u})^{\mathrm{T}}$, A_0 is the $n_0 \times n_0$ matrix with uvth element given by $c(\mathbf{x}_{0u}, \mathbf{x}_{0v}; \mathbf{r})$ and T is the $n \times n_0$ matrix with iuth element given by $c(\mathbf{x}_i, \mathbf{x}_{0u}; \mathbf{r})$ $(u, v = 1, \ldots, n_0; i = 1, \ldots, n)$.

It can be shown (see section 3 of the on-line supplementary material) that the conditional distribution of Y_0 is

$$Y_0|Y, B, \Sigma, \mathbf{r} \sim \mathrm{MN}_{n_0,k}\{H_0B + T^{\mathrm{T}}A^{-1}(Y - HB), \Sigma, A_0 - T^{\mathrm{T}}A^{-1}T\}. \tag{6}$$

From distributions (5) and (6), we can see the fundamental difference between the GP and lightweight emulators; for the lightweight emulator, the output from different simulator runs is assumed independent given $\{B, \Sigma\}$ and hence the matrix T of correlations between the observed and unobserved simulator runs will be a zero matrix. Hence, conditionally on B and Σ, the distribution of Y_0 does not depend on Y. For the GP emulator, with non-zero correlations between simulator runs, the dependence between Y_0 and Y remains even after conditioning on B and Σ.

To obtain the posterior predictive distribution of Y_0, given $\mathbf{r}$, we integrate expression (6) with respect to the posterior distribution of B and Σ (see section 4 of the on-line supplementary material):

$$Y_0|Y, \mathbf{r} \sim \mathrm{MT}_{n_0,k}(Q, \hat{S}, R, \hat{\delta}), \tag{7}$$

where

$$Q = H_0\hat{M} + T^{\mathrm{T}}A^{-1}(Y - H\hat{M}),$$
$$R = A_0 - T^{\mathrm{T}}A^{-1}T + (H_0 - T^{\mathrm{T}}A^{-1}H)\hat{\Omega}(H_0 - T^{\mathrm{T}}A^{-1}H)^{\mathrm{T}},$$

and $\mathrm{MT}_{n_0,k}(Q, \hat{S}, R, \hat{\delta})$ denotes the matrix t-distribution (Javier and Gupta, 1985) with location matrix Q, column scale matrix $\hat{S}$, row scale matrix R and degrees of freedom $\hat{\delta}$. Marginal posterior predictive distributions for the uth simulator run, $\mathbf{y}_{0u} = f(\mathbf{x}_{0u})$, and the sth output, $y_{0,us} = f_s(\mathbf{x}_{0u})$, are multivariate and univariate t-distributions respectively:

$$\begin{aligned} \mathbf{y}_{0u}|Y, \mathbf{r} &\sim t_k\left(\mathbf{q}_u^{\mathrm{T}}, \frac{R_{uu}\hat{S}}{\hat{\delta}}, \hat{\delta}\right); \\ y_{0,us}|Y, \mathbf{r} &\sim t\left(q_{us}, \frac{R_{uu}\hat{S}_{ss}}{\hat{\delta}}, \hat{\delta}\right). \end{aligned} \tag{8}$$

Here, $\mathbf{q}_u$ is the uth row of Q and q_{us} is the usth element of Q, R_{uu} is the uth diagonal element of R and $\hat{S}_{ss}$ is the sth diagonal element of $\hat{S}$.

For the lightweight emulator, where $A = I_n$, which is an $n \times n$ identity matrix, distribution (7) provides closed form posterior predictive distributions. For the multivariate GP emulator, and the most commonly used correlation functions $c(\cdot, \cdot; \mathbf{r})$, there is no prior distribution for $\mathbf{r}$ such that a closed form expression can be obtained for the marginal posterior predictive distribution of Y_0. Typically, one of two approaches is taken:

(a) $\mathbf{r}$ is replaced by a 'plug-in' estimate $\hat{\mathbf{r}}$, a representative value with respect to the marginal posterior distribution of $\mathbf{r}$, or
(b) Markov chain Monte Carlo (MCMC) methods are used to sample from the marginal posterior distribution of $\mathbf{r}$ and then, for each sampled value of $\mathbf{r}$, a value is drawn from the conditional posterior predictive distribution (7).

The plug-in approach is less computationally expensive than the fully Bayesian approach and provides a closed form emulator. We adopt the plug-in approach for prediction using the marginal posterior mode of $\mathbf{r}$, obtained by maximizing the unnormalized marginal posterior density

$$\pi(\mathbf{r}|Y) \propto \pi_{\mathbf{r}}(\mathbf{r})|A|^{-k/2}|\hat{\Omega}|^{k/2}|\hat{S}|^{-(\hat{\delta}+k-1)/2},$$

where $\pi_{\mathbf{r}}(\mathbf{r})$ is the prior probability density function for $\mathbf{r}$.

The final step in building the multivariate GP emulator is the choice of the correlation function $c(\cdot,\cdot;\mathbf{r})$. The most commonly used function is the power exponential function, which was extended by Qian *et al.* (2008) to incorporate both quantitative and qualitative variables. Assuming without loss of generality that the variables are ordered, so that the first p_1 variables in $\mathbf{x}$ are quantitative and the next $p-p_1$ are qualitative variables, a correlation function that is exchangeable in the levels of the qualitative variables has the form

$$c(\mathbf{x}_1,\mathbf{x}_2;\mathbf{r})=\exp\left\{-\sum_{l=1}^{p_1} r_l|x_{1l}-x_{2l}|^{g_l}-\sum_{l=p_1+1}^{p} r_l\mathbf{I}(x_{1l}\neq x_{2l})\right\}. \tag{9}$$

Qian *et al.* (2008) suggested various correlation functions for qualitative variables, each reducing to the common form (9) for two-level qualitative variables. Throughout this paper, we fix $g_l=2$ for all l.

3. Emulator diagnostics and improvement

In this section, we address diagnostics for emulator adequacy and methods for improving emulator performance, including variable selection and the addition of a nugget term for the multivariate GP.

3.1. Emulator diagnostics

We start by developing generalizations to multivariate emulators of the diagnostics that were provided by Bastos and O'Hagan (2009) for univariate GP emulators. These diagnostics assess the assumption underlying expression (2), that the responses conditionally follow a matrix normal distribution with specified mean and correlation functions. Their evaluation requires an additional validation set of simulator runs, ζ_0 and Y_0, to be available.

3.1.1. Individual prediction errors

As suggested by Bastos and O'Hagan (2009), standardized prediction errors can be explored graphically or used to construct nominal level predictive probability intervals. If the emulator is an adequate model of the simulator, from distribution (8), the standardized individual prediction error

$$D^{\mathrm{I}}_{us}(Y_0)=\sqrt{\left(\frac{\hat{\delta}}{R_{uu}\hat{S}_{ss}}\right)}(y_{0,us}-q_{us})$$

has a standard t-distribution, conditional on Y with $\hat{\delta}$ degrees of freedom ($u=1,\ldots,n_0; s=1,\ldots,k$). A large number of outlying standardized prediction errors, with respect to the reference distribution, indicates serious inadequacy of the emulator. Bastos and O'Hagan (2009) suggested various graphical methods for identifying patterns in outliers and, subsequently, causes for emulator inadequacy, e.g. plots of the individual prediction errors against each input variable or the predictive mean.

Individual $100(1-\alpha)\%$ predictive probability intervals for each element of Y_0 can be constructed as

$$q_{us}\pm c_\alpha\sqrt{\left(\frac{R_{uu}\hat{S}_{ss}}{\hat{\delta}}\right)},$$

where c_α is the $(1-\alpha/2)$th quantile of the standard t-distribution with $\hat{\delta}$ degrees of freedom. The obtained coverage of these intervals can be compared against $1-\alpha$, with low coverage suggesting that the emulator is underestimating the prediction uncertainty.

3.1.2. Omnibus diagnostic

We now develop a summary statistic for overall emulator adequacy, which is analogous to the Mahalanobis distance diagnostic of Bastos and O'Hagan (2009). Define E as the $n_0 \times k$ matrix of standardized predictions

$$E = G_R^{-1}(Y_0 - Q)G_S^{-1},$$

where $R = G_R G_R^{\mathrm{T}}$ and $\hat{S} = G_S^{\mathrm{T}} G_S$. Following Javier and Gupta (1985), for an adequate emulator, the conditional posterior distribution of E is

$$E|Y, \mathbf{r} \sim \mathrm{MT}_{n_0,k}(\mathbf{0}_{n_0\times k}, I_k, I_{n_0}, \hat{\delta}).$$

We now define the diagnostic

$$U = |I_k + E^{\mathrm{T}}E|^{-1}, \tag{10}$$

with extreme (large or small) values of U, relative to the reference distribution, indicating emulator inadequacy. Following Dickey (1967), the reference distribution for U is a $U_{k,n_0,k+\hat{\delta}-1}$-distribution (conditional on Y and $\mathbf{r}$). Anderson (2003), page 307, showed that the $U_{k,n_0,k+\hat{\delta}-1}$-distribution has the same distribution as a product of k independent beta random variables, i.e.

$$\prod_{s=1}^{k} X_s \sim U_{k,n_0,k+\hat{\delta}-1},$$

where $X_s \sim \mathrm{beta}\{(k+\hat{\delta}-s)/2, n_0/2\}$. Summaries of this distribution can be calculated by simulation.

The matrices G_R and G_S are not unique and depend on the chosen decomposition of R and $\hat{S}$ respectively, e.g. the eigendecomposition or Cholesky decomposition. However,

$$\begin{aligned} U &= |I_k + (G_S^{-1})^{\mathrm{T}}(Y_0 - Q)^{\mathrm{T}} R^{-1}(Y_0 - Q)G_S^{-1}|^{-1} \\ &= |I_k + \hat{S}^{-1}(Y_0 - Q)^{\mathrm{T}} R^{-1}(Y_0 - Q)|^{-1}, \end{aligned}$$

and therefore the value of the diagnostic U is invariant to the choice of decomposition.

Assuming distribution (2), also note that

$$\mathrm{cov}\{\mathrm{vec}(E)\} = \frac{1}{\hat{\delta}-2} I_{kn_0},$$

and hence the elements of $\hat{\delta}^{1/2}E$ form an uncorrelated sample from the t-distribution with $\hat{\delta}$ degrees of freedom. Quantile–quantile (QQ-) plots of these elements can be used as an additional check on emulator adequacy. The elements of E are dependent on the decomposition that is used to obtain G_R and G_S. However, as noted by Bastos and O'Hagan (2009), any choice of decomposition method is appropriate for use in a QQ-plot, and we use the Cholesky decomposition.

For univariate simulator output ($k = 1$), the omnibus statistic reverts to the Mahalanobis distance that was suggested by Bastos and O'Hagan (2009). Now, E is an $n_0 \times 1$ vector following a $t_{n_0}\{\mathbf{0}, (1/\hat{\delta})I_{n_0}, \hat{\delta}\}$ distribution, $E^{\mathrm{T}}E$ is scalar and $1 - U \sim \mathrm{beta}(n_0/2, \hat{\delta}/2)$ with

$$\frac{\hat{\delta}(1-U)}{n_0 U} = \frac{\hat{\delta}}{n_0} E^{\mathrm{T}}E \sim F(n_0, \hat{\delta}).$$

The quantity $E^{\mathrm{T}}E/(\hat{\delta}-2)$ is the Mahalanobis distance and $F(a, b)$ denotes an F-distribution with a and b degrees of freedom.

3.2. Emulator improvement

The diagnostics in Section 3.1 can be used to suggest improvements to a multivariate emulator. For example, graphical assessment of standardized errors may suggest different mean functions $h(\mathbf{x})$, transformations of inputs or regions of $\mathcal{X}$ where new simulator runs should be performed; see Bastos and O'Hagan (2009). We focus on selection of an appropriate mean function and improvement of GP emulators via the addition of a nugget.

3.2.1. Mean function selection via model comparison

It is common in the application of GP emulators usually to assume a simple form for the mean function such as $h(\mathbf{x}) = 1$ or $h(\mathbf{x}) = c(1, \mathbf{x})$ (see, for example, Bayarri *et al.* (2007)). Clearly, for the lightweight emulator, with uncorrelated errors, such a simple assumption will usually be inappropriate. We demonstrate in Section 4 that using an overly complex mean function (i.e. overfitting) can also be detrimental to the accuracy of the emulator on an independent test data set, as with the more usual applications of the linear model. This motivates the use of Bayesian model comparison as a vehicle for the selection of an appropriate mean function.

Let each unique choice of $h(\mathbf{x})$ be indexed by v, i.e. we label mean functions as $h_v(\mathbf{x})$, with $v \in \mathcal{V}$ and $\mathcal{V}$ denoting the set of possible models. Then, following equations (2) and (7),

$$Y|B_v, \Sigma_v, v, \mathbf{r}_v \sim \mathrm{MN}_{n,k}(H_v B_v, \Sigma_v, A_v)$$

and

$$Y_0|Y, v, \mathbf{r}_v \sim \mathrm{MT}_{n_0,k}(Q_v, \hat{S}_v, R_v, \hat{\delta}_v), \tag{11}$$

where

$$\begin{gathered}
Q_v = H_{v,0}\hat{M}_v + T_v^{\mathrm{T}} A_v^{-1}(Y - H_v \hat{M}_v), \\
R_v = A_{v,0} - T_v^{\mathrm{T}} A_v^{-1} T_v + (H_{v,0} - T_v^{\mathrm{T}} A_v^{-1} H_v)\hat{\Omega}_v (H_{v,0} - T_v^{\mathrm{T}} A_v^{-1} H_v)^{\mathrm{T}}, \\
\hat{\Omega}_v = (H_v^{\mathrm{T}} A_v^{-1} H_v + \Omega_v^{-1})^{-1}, \\
\hat{M}_v = \hat{\Omega}_v (H_v^{\mathrm{T}} A_v^{-1} Y + \Omega_v^{-1} M_v), \\
\hat{S}_v = Y^{\mathrm{T}} A_v^{-1} Y + M_v^{\mathrm{T}} \Omega_v^{-1} M_v + S_v - \hat{M}_v^{\mathrm{T}} \hat{\Omega}_v^{-1} \hat{M}_v, \\
\hat{\delta}_v = \delta_v + n,
\end{gathered}$$

M_v, Ω_v, S_v and δ_v are hyperparameters for the vth model, $\mathbf{r}_v$ holds the correlation parameters for the vth model and $H_{v,0}$, H_v, A_v, $A_{v,0}$, T_v and B_v for model v are analogous to matrices defined in Section 2.

A fully Bayesian approach would average equation (11) with respect to the posterior distribution of the correlation parameters, $\mathbf{r}_v$, and the posterior model probabilities to provide a model-averaged posterior predictive distribution. Alternatively, Bayesian model comparison can be used to identify a model $\hat{v}$, based on the posterior model probabilities, and $Y_0|Y, \hat{\mathbf{r}}_{\hat{v}}, \hat{v}$ can be employed as an emulator. The obvious choice for $\hat{v}$ is the posterior modal model with highest posterior model probability. We adopt this approach, both for computational convenience and also to provide interpretable emulators that aid scientific understanding of the simulator.

The posterior model probability for model v is given by

$$\pi(v|Y) = \frac{\pi(v) \displaystyle\int \pi(Y|\mathbf{r}_v, v)\pi(\mathbf{r}_v|v)\,\mathrm{d}\mathbf{r}_v}{\displaystyle\sum_{v \in \mathcal{V}} \pi(v) \int \pi(Y|\mathbf{r}_v, v)\pi(\mathbf{r}_v|v)\,\mathrm{d}\mathbf{r}_v},$$

where $\pi(v)$ is the prior model probability of v such that $\Sigma_{v\in\mathcal{V}}\pi(v)=1$,

$$\pi(Y|\mathbf{r}_v,v)=\frac{\Gamma_k\left(\dfrac{k+\hat{\delta}_v-1}{2}\right)}{\pi^{nk/2}\Gamma_k\left(\dfrac{k+\delta_v-1}{2}\right)|A_v|^{k/2}}\frac{|\hat{\Omega}_v|^{k/2}}{|\Omega_v|^{k/2}}\frac{|S_v|^{(\hat{\delta}_v+k-1)/2}}{|\hat{S}_v|^{(\hat{\delta}_v+k-1)/2}},$$

and $\Gamma_k(\cdot)$ is the multivariate gamma function (Javier and Gupta, 1985)

$$\Gamma_k(x)=\pi^{k(k-1)/4}\prod_{s=1}^{k}\Gamma\{x-(s-1)/2\}.$$

The term $\int\pi(Y|\mathbf{r}_v,v)\pi(\mathbf{r}_v|v)\,\mathrm{d}\mathbf{r}_v$ which features in the posterior model probability is known as the marginal likelihood. For the GP emulator, the integration that is required to evaluate the marginal likelihood will not be analytically tractable. For the lightweight emulator, where $A_v=I_n$ and does not depend on $\mathbf{r}_v$, the marginal likelihood is available in closed form. However, if the number of models, $|\mathcal{V}|$, is large then calculating the marginal likelihood for every model will be computationally infeasible. Instead we generate a sample from the posterior distribution of the model index v, using MCMC methods. For a GP emulator, each iteration of the MCMC method has two phases.

Phase 1 uses the MCMC model composition algorithm (Raftery *et al.*, 1997) to update the model index conditionally on the current value of the correlation parameters. Suppose that the current model is v and a move to a model w is proposed with probability $\rho(v,w)$ where the correlation parameters remain unchanged, i.e. $\mathbf{r}_w=\mathbf{r}_v$. The move is accepted with probability

$$\alpha=\frac{\pi(Y|\mathbf{r}_v,w)\pi(w)}{\pi(Y|\mathbf{r}_v,v)\pi(v)}\frac{\rho(w,v)}{\rho(v,w)}. \tag{12}$$

Phase 2 updates the correlation parameters $\mathbf{r}_v$, conditionally on the current model v by using a suitable MCMC method. We employ a random-walk Metropolis–Hastings algorithm.

For the lightweight emulator, phase 2 is not required. After a large number of iterations, when the chain has reached a stationary distribution, the proportion of iterations that visit model v provides an approximation to $\pi(v|Y)$. We choose $\rho(v,w)$ such that

(a) proposed models can add or remove only a single term from the current model, adhering to marginality, and
(b) all possible models that obey these conditions are equally likely to be proposed.

3.2.2. Non-zero nugget

Gramacy and Lee (2012) discussed improving the adequacy of univariate GP emulators via the inclusion of a non-zero nugget parameter, principally to mitigate the effects of incorrect model assumptions. Use of a nugget changes the (i,j)th element of A,

$$a_{ij}=c(\mathbf{x}_i,\mathbf{x}_j;\mathbf{r})+\eta I(i=j),$$

where $\eta\geqslant 0$ is the nugget parameter and $I(i=j)$ is the indicator function. For prediction, we again adopt a plug-in approach for the nugget parameter and replace η by a representative value $\hat{\eta}$ (the posterior mode). For model selection, the value of the nugget is sampled in phase 2 of the MCMC algorithm. The prior for η that is used in this paper is given by $\pi(\eta)=(1+\eta^2)^{-1}$, which has previously been used by Conti and O'Hagan (2010) for correlation parameters.

4. Application to the DIAMOND simulator

In this section, the methodology from Sections 2 and 3 is employed to construct and check multivariate GP and lightweight emulators for the DIAMOND simulator. Recall that the scenario under investigation has been solely designed for model testing purposes. Hence, when, for example, we refer to the importance of specific input variables, we do so only in that context. In particular, we do not intend these observations to be applied to other situations. For the construction of each emulator, we scale the continuous input variables to $[0, 1]$ and denote the levels of the categorical variables as $\{0, 1\}$.

4.1. Prior information

When constructing individual GP and lightweight emulators, we assume weak prior information for the model parameters B, Σ and $\mathbf{r}$, following Conti and O'Hagan (2010):

$$M = \mathbf{0}_{m\times k},$$
$$\Omega^{-1} = \mathbf{0}_{m\times m},$$
$$S = \mathbf{0}_{k\times k},$$
$$\delta = -k+1.$$

The correlation parameters $\mathbf{r}$ are assumed independent, with prior distributions specified by using the approach of Linkletter *et al.* (2006). We rewrite $c(\mathbf{x}_1, \mathbf{x}_2; \mathbf{r})$, from expression (9), as

$$c(\mathbf{x}_1, \mathbf{x}_2; \mathbf{r}) = \prod_{l=1}^{p_1} \rho_l^{|x_{1l}-x_{2l}|^2} \prod_{l=p_1+1}^{p} \rho_l^{I(x_{1l}\neq x_{2l})},$$

where $\rho_l = \exp(-r_l) \in (0, 1)$ for $r_l > 0$ $(l = 1, \ldots, p)$. We assume a uniform prior distribution for ρ_l, leading to the induced prior for r_l being an exponential distribution with $E(r_l) = 1$.

When performing model comparison for the selection of the mean function with only weak prior information available for the parameters of each model, we adopt prior hyperparameters $S_v = 0_{k\times k}$ and $\delta_v = -k+1$ for Σ_v, which is present in all models, and unit information prior distributions for B_v, with $M_v = 0_{p\times p}$ and

$$\Omega_v = n(H_v^{\mathrm{T}} A_v^{-1} H_v)^{-1},$$

as proposed by Kass and Wasserman (1995). The use of proper prior distributions for B_v avoids Lindley's paradox (see Bernardo and Smith (1994), page 394) which states that the posterior model probabilities are sensitive to the scale of the prior variance (see also O'Hagan and Forster (2004), pages 322–324, Raftery *et al.* (1997) and Fernandez *et al.* (2001)). We assume the same exponential prior (see above) for each element of $\mathbf{r}_v$ for each model, i.e. $\pi(\mathbf{r}_v|v) = \pi(\mathbf{r}_v)$. A uniform prior over the model space is chosen, i.e. $\pi(v) = |\mathcal{V}|^{-1}$, where $\mathcal{V}$ is the set of all submodels of the maximal model that respect marginality. The maximal model has a mean function consisting of the intercept, all linear, two-way interaction and, for the continuous inputs, quadratic terms. The resulting model matrix H has $m = 103$ columns.

For this weak prior information, α from equation (12) reduces to

$$\alpha = (n+1)^{k(m_v-m_w)/2} \frac{|\hat{S}_v|^{n/2}}{|\hat{S}_w|^{n/2}} \frac{\rho(w, v)}{\rho(v, w)},$$

where

$$\hat{S}_v = Y^{\mathrm{T}} A_v^{-1} \left(I_n - \frac{n}{n+1} H_v (H_v^{\mathrm{T}} A_v^{-1} H_v)^{-1} H_v^{\mathrm{T}} A_v^{-1} \right) Y.$$

4.2. Design of the computer experiment

We employed a space filling design that would enable the estimation of both the Gaussian process and lightweight emulators. The most common design that is used for computer experiments is the Latin hypercube (McKay *et al.*, 1979) and its extensions (see, for example, Tang (1993) and Morris and Mitchell (1995)). Such designs provide low dimensional uniformity in the input variables, hence achieving good projection properties, and allow the estimation of non-parametric regression models. They are also an attractive choice for lightweight emulation, as the exact form of the emulator will be unknown in advance of the data collection and a flexible design that allows the fitting of many different parametric models may be required (see Section 3.2).

The design, $\zeta = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, for this study needed to combine both continuous and categorical input variables. We used a sliced space filling design as proposed by Qian and Wu (2009) with $n = 120$ runs. Such a design, constructed from an orthogonal array, has not only good space filling properties overall but also for the projection into the continuous variables for each combination of values of the categorical input variables.

4.3. Construction of adequate emulators

We constructed both lightweight and multivariate GP emulators for the DIAMOND simulator using the $n = 120$ simulator runs, each outputting $k = 5$ responses, from the sliced space filling design as training data. For model validation and diagnostics, we use a second design $\zeta_0 = \{\mathbf{x}_{01}, \ldots, \mathbf{x}_{0n_0}\}$, with associated $n_0 \times k$ simulator output matrix Y_0. This design is also a sliced space filling design with $n_0 = 120$ runs and was constructed by using an orthogonal array which was different from that used to construct ζ.

We chose, assessed and compared emulators by using the diagnostics from Section 3. We calculated the root-mean-squared error RMSE for Y_0,

$$\mathrm{RMSE} = \left\{ \frac{1}{n_0 k} \sum_{u=1}^{n_0} \sum_{s=1}^{k} (Y_{0,us} - q_{us})^2 \right\}^{1/2},$$

where $Y_{0,us}$ is the simulator output from the uth validation run for response s. We also calculated the root relative mean-squared error RRMSE,

$$\mathrm{RRMSE} = \left\{ \frac{1}{n_0 k} \sum_{u=1}^{n_0} \sum_{s=1}^{k} \frac{(Y_{0,us} - \gamma_{us})^2}{Y_{0,us}^2} \right\}^{1/2},$$

where the point estimate $\gamma_{us} = E(Y_{0,us}^{-1} | Y, \hat{\mathbf{r}}) / E(Y_{0,us}^{-2} | Y, \hat{\mathbf{r}})$ minimizes the relative squared error loss function.

4.3.1. Lightweight emulators

Our first lightweight emulator was the maximal model. The value of the omnibus test statistic U and coverage of the 95% predictive probability intervals are given in Table 2. Note that the reference distribution for U has an expected value of 0.030, and 2.5% and 97.5% quantiles of 0.019 and 0.044 respectively. The diagnostics indicate that there is a discrepancy between the simulator and this emulator, with the observed value of U and the coverage achieved both being low. Further evidence of this discrepancy is the QQ-plot of the uncorrelated errors against a

Table 2. Observed values (to three decimal places) of the omnibus diagnostic U, coverage of the 95% predictive probability intervals, RMSE and RRMSE for the various emulators considered

Emulator	*Mean function*	*Nugget*	*U*†	*Coverage*	*RMSE*	*RRMSE*
Lightweight	Maximal	—	0.000	0.478	2728.791	6.975
	Modal	—	0.025	0.953	988.729	0.528
Multivariate GP	Intercept	Zero	0.001	0.958	415.030	0.457
	Linear	Zero	0.015	0.965	344.234	0.397
	Modal	Zero	0.012	0.958	341.859	0.396
	Maximal	Zero	0.000	0.477	2701.149	6.791
Multivariate GP	Intercept	Non-zero	0.033	0.975	363.014	0.387
	Linear	Non-zero	0.019	0.948	1264.094	0.539
	Modal	Non-zero	0.034	0.963	334.597	0.403
	Maximal	Non-zero	0.000	0.478	2728.383	6.973

†The reference distribution for U has expected value of 0.030 and 2.5% and 97.5% quantiles of 0.019 and 0.044 respectively.

Table 3. Marginal posterior probabilities (up to three decimal places) of the terms in the modal mean functions

Term		*Lightweight emulator probability*	*GP emulator (zero nugget) probability*	*GP emulator (non-zero nugget) probability*
Linear effects				
Food capacity (Giarre)	x_3	0.999	1.000	1.000
Food capacity (Catania)	x_6	1.000	1.000	1.000
Planning time	x_8	0.970	1.000	1.000
Recipient of food aid	x_{12}	1.000	—	—
Location of NGO base	x_{13}	1.000	1.000	1.000
Quadratic effects				
Planning time		0.764	0.999	1.000
Interactions				
Food capacity (Giarre) × recipient of food aid		0.811	—	—
Food capacity (Catania) × recipient of food aid		1.000	—	—
Food capacity (Catania) × location of NGO base		1.000	0.983	0.828
Recipient of food aid × location of NGO base		0.914	—	—

reference t-distribution (Fig. 2(a)); the points form a line with slope greater than 1, indicating that the variance that is associated with the emulator predictions has been underestimated.

To attempt to alleviate the obvious inadequacy of this emulator, alternative mean functions $h(\mathbf{x})$ were compared by using Bayesian model comparison (Section 3.2). The posterior modal model was found from 10^5 iterations of the MCMC algorithm (discarding the first 10% of iterations as burn-in). The algorithm took 2.5 min on a computer with a 3.20 GHz processor and 8 Gbytes of random-access memory, and the average acceptance rate for the proposed

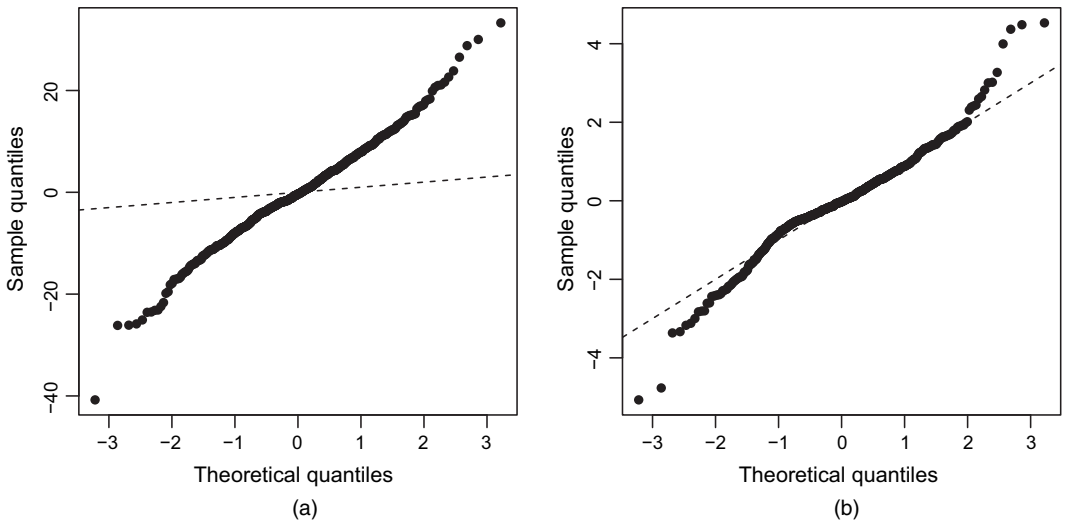

Fig. 2. *QQ*-plots of the uncorrelated errors against a reference *t*-distribution for lightweight emulators: (a) maximal model; (b) modal model

moves in phase 1 was 4.7%, reflecting the concentration of the posterior model probabilities on a small number of models. Table 3 displays the terms in the posterior modal model and gives the associated posterior marginal inclusion probabilities (i.e. the proportion of models visited that included that term). The model matrix H for the posterior modal model has $m = 11$ columns. The value of U and the coverage for the emulator with this alternative mean function are shown in Table 2. These values suggest that there is no evidence of a discrepancy between the simulator and the emulator. This conclusion is supported by the QQ-plot of the uncorrelated errors in Fig. 2(b). Also shown in Table 2 are the RMSE and the RRMSE of the maximal and modal model emulators. Note how the simpler form of emulator has smaller values for RMSE and RRMSE, indicating that the modal model has significantly improved predictive accuracy.

4.3.2. Multivariate Gaussian process emulators

We construct GP emulators with four different forms for the mean function $h(\mathbf{x})$:

(a) intercept only ($m = 1$);
(b) linear terms only ($m = 8$);
(c) the modal model found by the model comparison procedure ($m = 7$; see Table 3);
(d) the maximal model ($m = 103$).

We initially fix the nugget at zero. As a comparison with Section 4.3.1, the model comparison procedure took 30 min and had an acceptance rate of 0.5%.

Table 2 shows the values of U and the coverage for these four GP emulators. Figs 3(a)–3(d) show QQ-plots of the uncorrelated errors for these emulators. Clearly, the values in Table 2 and the QQ-plots show that there are serious discrepancies between all four emulators and the simulator. Similarly to the maximal lightweight emulator plot, the QQ-plot shows that the variances that are associated with the GP emulator predictions are underestimated.

To remedy these inadequacies, we included a non-zero nugget in emulators using all four forms of the mean function. The model comparison algorithm took 30 min and had an acceptance rate of 1.6%. The modal mean function for both types of GP emulator (with and without a

nugget) are identical (see Table 3). The values of U and the coverage for the four non-zero nugget GP emulators are also shown in Table 2. The corresponding QQ-plots are shown in Figs 3(e)–3(h). There still are discrepancies between the emulator and simulator for the maximal and linear forms of the mean function. However, for the intercept and modal forms, the values in Table 2 and the QQ-plots provide no evidence of inadequacy, with the diagnostics being highly plausible under their reference distributions. The values of RMSE and RRMSE for all eight GP emulators are also given in Table 2. Note the high values of these errors under the maximal models. The intercept and modal GP emulators (with non-zero nugget) have significantly higher predictive accuracy than the lightweight emulators. There appears to be little difference between the intercept and modal model for the GP emulators (with non-zero nugget) in terms of predictive accuracy.

4.4. Sensitivity analysis

An important application of statistical emulators is sensitivity analyses to identify important input variables and their effect on the responses. For the lightweight emulator, the model comparison algorithm in Section 3.2.1 has the advantage of automatically identifying the most important input variables. When product terms are included in the mean function, it can also identify important interactions. For the DIAMOND simulator, there are interactions between the food capacity at Catania and both the location of the NGO base and the recipient of the food aid. There are also interactions between the food capacity at Giarre and the recipient of food aid and location of NGO base and recipient of food aid. There is evidence that planning time has a non-linear effect.

For the multivariate GP emulator, input variables can impact the response through both the mean function and the correlation structure. Hence, the model selection algorithm in Section 3.2.1 may not identify all the important variables. For an intercept-only GP, the relative importance of the input variables is only determined by the relative magnitude of the corresponding correlation parameters $\mathbf{r}$. In general, the output is more sensitive to those input variables with large correlation parameters. As calibrating the size of correlation parameters can be difficult, Linkletter *et al.* (2006) proposed a more formal variable selection method for univariate GPs: reference distribution variable selection (RDVS). Values of an inert input variable x^* are randomly generated from the input space $\mathcal{X}$. An MCMC sample is generated from the marginal posterior distribution of $\mathbf{r}$ and r^*, where r^* is the correlation parameter of the inert input variable. This procedure is repeated B times with different randomly generated values of inert input variables. The posterior median of each element of $\mathbf{r}$, approximated from the union of the MCMC samples from all randomly generated sets of inert input variables, is compared with the null reference distribution of the posterior medians of r^* (obtained from the B sets of values for the inert input variable). For more details see Linkletter *et al.* (2006).

Application of RDVS to multivariate GP emulators is straightforward. Our simulator has both continuous and categorical input variables, and hence we adapt RDVS by at each iteration randomly generating values for two inert input variables, x_1^* and x_2^*, where $x_1^* \in [0, 1]$ and $x_2^* \in \{0, 1\}$, with $\{0, 1\}$ indicating the two levels for a categorical variable. The posterior median of the elements of $\mathbf{r}$ corresponding to continuous input variables is then compared with the null reference distribution of the posterior medians of r_1^*, and similarly for the categorical input variables and r_2^*.

We applied RDVS with the GP emulator (with non-zero nugget and the intercept mean function), using $B = 1000$. Fig. 4 displays the null reference distributions for the correlation parameters (on the log-scale) of Fig. 4(a) the continuous and Fig. 4(b) the categorical inert

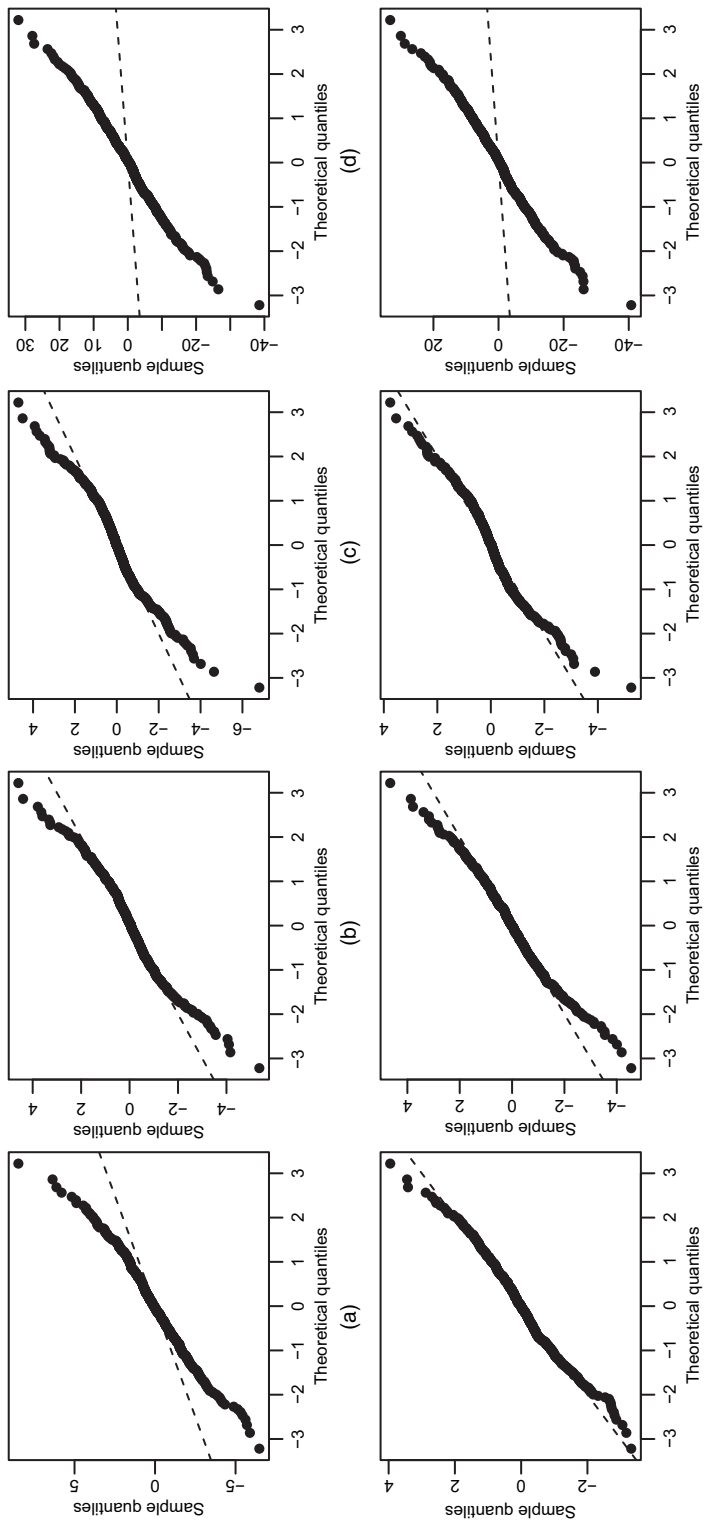

Fig. 3. *QQ*-plots of the uncorrelated errors against a reference *t*-distribution for the zero nugget GP emulator with (a) the intercept model, (b) the linear model, (c) the modal model and (d) the maximal model, and for the non-zero nugget GP emulator with (e) the intercept model, (f) the linear model, (g) the modal model and (h) the maximal model

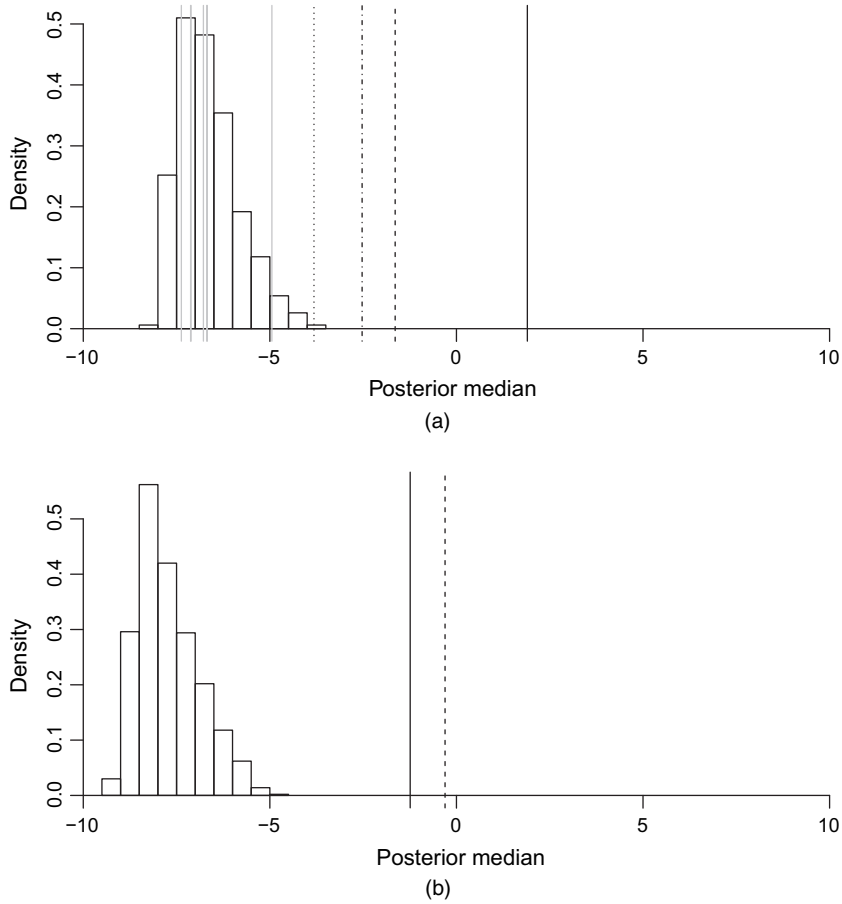

Fig. 4. Histograms of the null reference distributions from the RDVS method for the correlation parameters of (a) the continuous input variables and (b) the categorical input variables: the posterior medians of the input variables are shown as vertical lines (in (a), ·-·-·-, food capacity (Giarre); ———, food capacity (Catania); ------, planning time; · · · · · ·, helicopter speed; ———, others; in (b), ———, recipient of food aid; ------, location of NGO base)

input variables, i.e. the 1000 posterior medians of the correlation parameters r_1^* and r_2^* from the MCMC samples. Also indicated in Fig. 4 are the posterior medians of the actual input variables as vertical lines. Clearly the most important continuous input variables are the food capacities at both Giarre and Catania, planning time and helicopter speed. Both of the categorical input variables are deemed to be important. This agrees with the conclusions from the modal lightweight emulator, except for the inclusion of helicopter speed.

RDVS with a GP emulator having mean function including only an intercept is unable to identify interactions explicitly. A probabilistic sensitivity analysis (see, for example, Santner *et al.* (2003), chapter 7) can be used to understand and visualize the functional form of the individual and joint effects of the variables.

The variation in the simulator output induced by variation in the input variables can be decomposed into main effects and interactions. Assume that interest is in the total number of casualties across days 2–6 of the disaster, given by $g(\mathbf{x}) = \Sigma_{i=1}^{k} f_i(\mathbf{x})$. Letting E denote expectation with respect to an assumed joint distribution for the input variables $\mathbf{x}$, we can then define

Table 4. Estimated first- and second-order sensitivity indices (multiplied by 1000 and displayed up to three decimal places) of the input variables under the lightweight and multivariate GP emulators

Term		*Lightweight emulator index*	*Multivariate GP index*
First order			
Food capacity (Giarre)	x_3	9.978	7.854
Food capacity (Catania)	x_6	887.818	895.176
Planning time	x_8	2.589	1.881
Helicopter speed	x_9	0.000	0.312
Recipient of food aid	x_{12}	2.566	2.264
Location of NGO base	x_{13}	63.739	64.067
Sum of others		0.000	0.023
Second order			
Food capacity (Giarre) × recipient of food aid		1.184	0.474
Food capacity (Giarre) × location of NGO base		0.000	0.365
Food capacity (Catania) × recipient of food aid		1.620	2.121
Food capacity (Catania) × location of NGO base		3.599	6.750
Planning time × location of NGO base		0.000	0.572
Planning time × food capacity (Catania)		0.000	0.173
Recipient of food aid × location of NGO base		1.099	0.906
Sum of others		0.000	0.178

the following main effects and first-order interactions:

$$g_i(x_i) = E\{g(\mathbf{x})|x_i\} - g_0, \tag{13}$$

$$g_{ij}(x_i, x_j) = E\{g(\mathbf{x})|x_i, x_j\} - g_0 - g_i(x_i) - g_j(x_j), \tag{14}$$

where $g_0 = E[g(\mathbf{x})]$. Corresponding partial variances are given by

$$V_i = E\{g_i(x_i)^2\},$$
$$V_{ij} = E\{g_{ij}(x_i, x_j)^2\}, \qquad i, j = 1, \ldots, p.$$

Following Oakley and O'Hagan (2004), these variances can be estimated by their expectation, denoted E^*, with respect to the posterior predictive distribution of $g(\mathbf{x})$, which is a non-standard t-distribution; see section 5 of the on-line supplementary material. Hence, the following estimated sensitivity indices can be defined:

$$\hat{\mathcal{S}}_i = E^*(V_i)/E^*(V)$$

(first order) and

$$\hat{\mathcal{S}}_{ij} = E^*(V_{ij})/E^*(V)$$

(second order), where $V = \mathrm{var}\{g(\mathbf{x})\}$ with respect to the distribution of the input variables. Explicit formulae for $E^*(V)$, $E^*(V_i)$ and $E^*(V_{ij})$ can be derived in terms of the expectation with respect to the distribution of the input variables and are given in section 6 of the supplementary material.

We assume that the input variables are independent, that the continuous variables are uniformly distributed over their corresponding ranges and the categorical input variables have probability 0.5 on each of their two levels. We compute the estimated sensitivity indices under

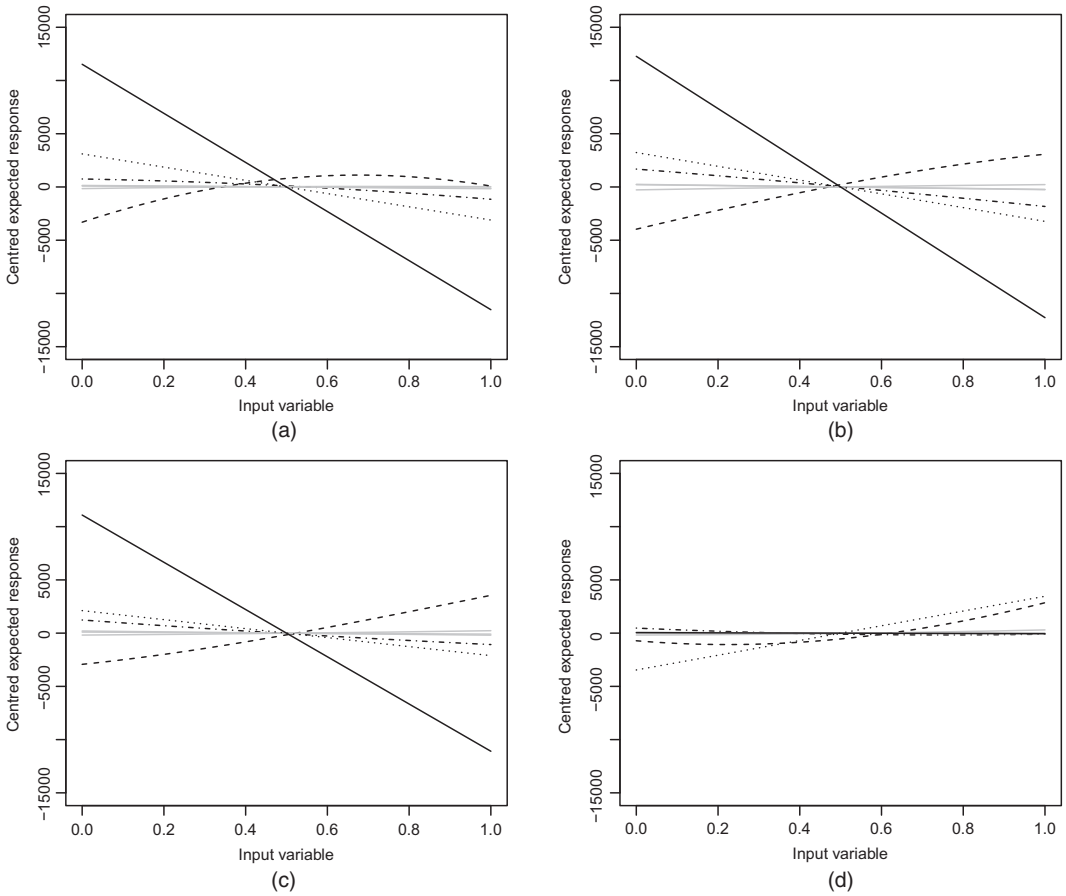

Fig. 5. Plots of expected conditional main effects (15) from the multivariate GP emulator (intercept mean function and non-zero nugget) for four settings for the food capacity at Catania, x_6 (- - - - - -, planning time; -·-·-·-, helicopter speed; · · · · · ·, recipient of food aid; ———, location of NGO base; ———, others): (a) $x_6 = 0$; (b) $x_6 = \frac{1}{3}$; (c) $x_6 = \frac{2}{3}$; (d) $x_6 = 1$

both the multivariate GP emulator (intercept mean function and non-zero nugget) and, for comparison, the lightweight emulator (modal mean function). For the lightweight emulator, the estimated sensitivity indices are available in closed form (Rougier, 2007) and can only be non-zero for those main effects and interactions that feature in the, selected, modal model. Under the multivariate GP emulator, the expectations with respect to the distribution of the input variables require approximation, which is achieved here by using Monte Carlo integration.

Table 4 shows the estimated sensitivity indices under both emulators. For the multivariate GP, we present first-order estimated sensitivity indices for each of the variables identified by the RDVS method. We also present the seven largest second-order sensitivity indices; four of the corresponding interactions were selected in the modal lightweight emulator. The dominance of input variable x_6, controlling the food capacity at Catania, is clear; variation in x_6 induces nearly 90% of the total output variation for both emulators. However, this input variable, in common with x_1–x_5, is essentially a noise variable and clearly could not be controlled in a real disaster. Hence, of particular interest are the interactions between x_6 and the control variables x_7–x_{13}. To investigate these effects for the GP emulator graphically, in Fig. 5 we display the expected

conditional main effects

$$E^*[E\{g(\mathbf{x})|x_i, x_6 = l\} - g_0], \tag{15}$$

for $i = 8, 9, 12, 13$ (as identified by RDVS) and $l = 0, \frac{1}{3}, \frac{2}{3}, 1$. For $x_6 \neq 1$, there are strong negative conditional effects for both categorical variables x_{12} and x_{13}, with lower casualties resulting from providing food aid only to Catania and, especially, locating the NGO base with the task force. However, for $x_6 = 1$, variable x_{13} no longer has a substantive effect and x_{12} now has a positive effect (lower casualties result from providing food aid to both cities). Planning time x_8 always has a positive effect, although the degree of non-linearity changes with the value of x_6.

5. Discussion

Statistical emulation of multivariate simulators is an important problem in various application areas and presents challenging methodological issues. We have presented a unified Bayesian approach to the construction of both parametric (lightweight, linear model) and non-parametric (GP) emulators, including model selection, diagnostics and sensitivity analyses. Our application, emulating a humanitarian relief simulator applied to an artificial scenario involving an earthquake and volcanic eruption in Sicily, demonstrated the utility and versatility of the methodology. We could identify the most important input variables, and their interactions, by using the lightweight emulator. Although the GP emulator was more accurate, the lightweight emulator was more scientifically intuitive and informative. The technology in this paper provides the capacity for our collaborators to explore efficiently 'what if?' questions and to make faster 'in-theatre' decisions.

Extensions of the methodology to allow the construction and model checking of different emulators are possible. In Section 4, only weakly informative prior distributions were assumed. If more informative prior information was available, this could be incorporated in both lightweight and GP emulators, e.g. via the prior distribution for the regression parameters $B|\Sigma$. It is likely that the use of such information would lead to a smaller difference in predictive accuracy between the two emulators, provided that there was not a conflict between the prior information and the simulator.

Diagnostics for multivariate emulators were also employed by Fricker *et al.* (2013) in case-studies using models with a general class of non-separable covariance structure. These diagnostics were similar in spirit to those of Bastos and O'Hagan (2009) but, for example, the non-separability prevents analytic marginalization across any of the scale parameters when calculating the equivalent to the omnibus statistic (10). An alternative non-separable model may be constructed as the full posterior distribution under model uncertainty; see Section 3.2.1. The model-averaged posterior predictive distribution is then a mixture of matrix t-distributions; see also Rougier (2007), who proposed a mixture of matrix normal–inverse Wishart joint prior distributions for B and Σ. The diagnostics that were described in Section 3.1 are straightforward to extend to mixture distributions by averaging over the components of the mixture by using simulation.

Acknowledgements

This work was funded by a US Defence Threat Reduction Agency basic research grant and the Defence Science and Technology Laboratory. D. C. Woods was supported by Fellowship EP/J018317/1 from the UK Engineering and Physical Sciences Research Council. The authors

thank Robin Ashmore from the Defence Science and Technology Laboratory for providing the simulator and data, and related invaluable conversations. The paper was improved by helpful comments from an Associate Editor and two referees.

References

Anderson, T. W. (2003) *An Introduction to Multivariate Statistical Analysis*, 3rd edn. New York: Wiley.
Bastos, L. S. and O'Hagan, A. (2009) Diagnostics for Gaussian process emulators. *Technometrics*, **51**, 425–438.
Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C. and Tu, J. (2007) A framework for validation of computer models. *Technometrics*, **49**, 138–154.
Bernardo, J. M. and Smith, A. F. M. (1994) *Bayesian Theory*. Chichester: Wiley.
Conti, S. and O'Hagan, A. (2010) Bayesian emulation of complex multi-output and dynamic computer models. *J. Statist. Planng Inf.*, **140**, 640–651.
Dawid, A. P. (1981) Some matrix-variate distribution theory: notational considerations and a Bayesian application. *Biometrika*, **68**, 265–274.
Dickey, J. M. (1967) Matricvariate generalisations of the multivariate t-distribution and the inverted multivariate t-distribution. *Ann. Math. Statist.*, **38**, 511–518.
Fang, K., Li, R. and Sudjianto, A. (2006) *Design and Modelling for Computer Experiments*. Boca Raton: Chapman and Hall.
Fernandez, C., Ley, E. and Steel, M. F. J. (2001) Benchmark priors for Bayesian model averaging. *J. Econmetr.*, **100**, 381–427.
Fricker, T. E., Oakley, J. E. and Urban, N. M. (2013) Multivariate emulators with nonseparable covariance structures. *Technometrics*, **55**, 47–56.
Gramacy, R. B. and Lee, H. K. H. (2012) Cases for the nugget in modeling computer experiments. *Statist. Comput.*, **22**, 713–722.
Guidoboni, E., Ferrari, G., Mariotti, D., Comastri, A., Tarabusi, G. and Valensise, G. (2007) CFTI4Med: catalogue of strong earthquakes in Italy (461 B.C. - 1997) and the Mediterranean area (760 B.C. - 1500). Istituto Nazionale di Geofisica e Vulcanologia, Bologna. (Available from `http://storing.ingv.it/cfti4med/`.)
Ingber, L., Fujio, H. and Wehner, M. F. (1991) Mathematical comparison of combat computer models to exercise data. *Math. Comput. Modllng*, **15**, 65–90.
Javier, W. R. and Gupta, A. K. (1985) On matric variate t distributions. *Communs Statist. Theor. Meth.*, **14**, 1413–1425.
Kass, R. E. and Wasserman, L. (1995) A reference Bayesian test for nested hypotheses and its relationship to the Schwarz criterion. *J. Am. Statist. Ass.*, **90**, 928–934.
Kennedy, M. C., Anderson, C. W., Conti, S. and O'Hagan, A. (2006) Case studies in Gaussian process modelling of computer codes. *Reliab. Engng Syst. Safty*, **91**, 1301–1309.
Kennedy, M. C. and O'Hagan, A. (2001) Bayesian calibration of computer models (with discussion). *J. R. Statist. Soc.* B, **63**, 425–464.
Levy, S. and Steinberg, D. M. (2010) Computer experiments: a review. *Adv. Statist. Anal.*, **4**, 311–324.
Linkletter, C., Bingham, D., Hengartner, N. and Ye, K. (2006) Variable selection of Gaussian process models in computer experiments. *Technometrics*, **48**, 478–490.
McKay, M. D., Beckman, R. J. and Conover, W. J. (1979) A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, **21**, 239–245.
Morris, M. D. and Mitchell, T. J. (1995) Exploratory designs for computer experiments. *J. Statist. Planng Inf.*, **43**, 381–402.
Oakley, J. E. and O'Hagan, A. (2004) Probabilistic sensitivity analysis of complex models: a Bayesian approach. *J. R. Statist. Soc.* B, **66**, 751–769.
O'Hagan, A. (2006) Bayesian analysis of computer code outputs: a tutorial. *Reliab. Engng Syst. Safty*, **91**, 1290–1300.
O'Hagan, A. and Forster, J. J. (2004) *Kendall's Advanced Theory of Statistics*, vol. 2B, *Bayesian Inference*, 2nd edn. London: Arnold.
Qian, P. Z. G. and Wu, C. F. J. (2009) Sliced space-filling designs. *Biometrika*, **96**, 945–956.
Qian, P. Z. G., Wu, H. and Wu, C. F. J. (2008) Gaussian process models for computer experiments with qualitative and quantitative factors. *Technometrics*, **50**, 383–396.
Raftery, A. E., Madigan, D. and Hoeting, J. A. (1997) Bayesian model averaging for linear regression models. *J. Am. Statist. Ass.*, **92**, 179–191.
Rasmussen, C. E. and Williams, C. K. I. (2006) *Gaussian Processes for Machine Learning*. Cambridge: MIT Press.
Rougier, J. C. (2007) Lightweight emulators for multivariate deterministic functions. *Technical Report 07/02*. Managing Uncertainty in Complex Models, University of Durham, Durham.
Sacks, J., Welch, W. J., Mitchell, T. J. and Wynn, H. P. (1989) Design and analysis of computer experiments (with discussion). *Statist. Sci.*, **4**, 409–435.

Santner, T. J., Williams, B. J. and Notz, W. I. (2003) *The Design and Analysis of Computer Experiments*. New York: Springer.
Tang, B. (1993) Orthogonal array-based Latin hypercubes. *J. Am. Statist. Ass.*, **88**, 1392–1397.
Taylor, B. and Lane, A. (2004) Development of a novel family of military campaign simulation models. *J. Oper. Res. Soc.*, **55**, 333–339.

Supporting information
Additional 'supporting information' may be found in the on-line version of this article:

'Supplementary material for "Multivariate emulation of computer simulators: model selection and diagnostics with application to a humanitarian relief model"'.