6th St.Petersburg Workshop on Simulation (2009) 1-3

Managing Uncertainty in Complex Stochastic Models: Design and Emulation of a Rabies Model[1]

Alexis Boukouvalas[2], Dan Cornford[3], Alexander Singer[4]

Abstract

In this paper we present a novel method for emulating a stochastic, or random output, computer model and show its application to a complex rabies model. The method is evaluated both in terms of accuracy and computational efficiency on synthetic data and the rabies model. We address the issue of experimental design and provide empirical evidence on the effectiveness of utilizing replicate model evaluations compared to a space-filling design. We employ the Mahalanobis error measure to validate the heteroscedastic Gaussian process based emulator predictions for both the mean and (co)variance. The emulator allows efficient screening to identify important model inputs and better understanding of the complex behaviour of the rabies model.

1 Introduction

In many scientific and engineering problems complex simulators, based on mechanistic and physical process driven models, are routinely used to solve complex problems. Such simulators are often computationally expensive, and full uncertainty analysis, sensitivity analysis or other probabilistic analysis becomes extremely time consuming, effectively being computationally intractable. The most commonly applied solution is to create a meta-model for the simulator [5], often referred to as an *emulator* [3]. The role of the emulator can be seen to be approximating the simulator. In most existing work emulator methods are applied to deterministic models, of the form $\mathbf{y} = \mathbf{f}(\mathbf{x})$ where $\mathbf{x}$ represents the inputs to the simulator, $\mathbf{y}$ represents the outputs of the simulator, or some summary of these, and $\mathbf{f}$ represents the mapping imposed by the simulator evaluation. The probabilistic nature of the emulator, which is typically modelled as a Gaussian Process (GP) [3], arises from the *approximation* of the simulator due to having a finite number of simulator runs. In this paper we develop novel methods for the emulation of a stochastic simulator, a relatively new field [5].

[1]This research was funded as part of the Managing Uncertainty in Complex Models project by EPSRC grant D048893/1.

[2]Aston University, E-mail: boukouva@aston.ac.uk

[3]Aston University, E-mail: D.Cornford@aston.ac.uk

[4]Central Science Laboratories, E-mail: alexssinger@googlemail.com

A GP is defined as a collection of random variables, any finite subset of which has a joint Gaussian distribution [8]. It is completely defined by a mean and a covariance function, the specification of which allows the incorporation of prior knowledge in the emulation analysis such as the smoothness and differentiability of the approximated function, that is the simulator.

Another issue commonly occurring in the context of complex datasets is that of experimental design [7]. We assess the efficiency of different designs, examining the effect of replicate model evaluations, where the simulator is evaluated repeatedly for a single design point, against a more traditional space filling design. Utilizing the moments of the replicate evaluations allows for computationally efficient inference, and we empirically show that it also increases the accuracy of the heteroscedastic emulator, especially the (co)variance estimates.

2 Stochastic emulation

Relatively little work has addressed the question of the emulation of stochastic simulators. In this work we consider a stochastic simulator to be a mapping that produces random output given a fixed set of inputs. A recent review of the application of 'Kriging' (or GP regression) to emulation can be found in [5].

Kleijnen and co-workers [5] have studied the problem of stochastic emulation closely, investigating queuing models. In the work of Kleijnen the emulator of stochastic simulators uses m repetitions of the simulator at each of the i design points. From this the mean response $\bar{y}_i = \frac{1}{m}\sum_{j=1}^{m} \mathbf{y}_{i,j}$ and the variance of the response $\mathbf{S}_i^2 = \frac{1}{m-1}\sum_{j=1}^{m}(\bar{y}_i - \mathbf{y}_{i,j})^2$ are computed, where $\mathbf{y}_{i,j}$ is the j'th realisation from the stochastic simulator, at the i'th design point. The main concern in [6] is modelling the mean response of the stochastic simulator. The variance estimates, $\mathbf{S}_i^2$ are used to 'Studentize' the output with the transformation $\tilde{\mathbf{y}}_i = \bar{y}_i / \sqrt{\mathbf{S}_i^2/m^2}$, where they assume $\mathbf{y}$ has had any 'large scale' trend removed. A standard GP regression of the transformed output, $\tilde{\mathbf{y}}_i$, is then applied. The allowance for heteroscedastic, i.e. input dependent, variance is limited to a small number of simple parametric models. In all the work on stochastic emulation very little attention is paid to the treatment of heterogeneity of the output variance. In this paper we extend the recent work of [4] to enable improved stochastic emulation of more complex models and test it on a rabies disease simulator.

3 Heteroscedastic Modelling

In this section we briefly describe our method. The reader is referred to [2] for a detailed description. Following [4], we define a GP on the mean model output $\mathbf{G}_\mu$ and a second GP on the log variance of the model output, $\mathbf{G}_{\mathbf{\Sigma}}$. We do not present the full GP inference framework here but note that in all experiments maximum marginal likelihood estimation was used for the covariance hyper-parameters. The notation used is: N the number of design points used during inference, $D = \{\mathbf{x}_i, y_i\}$ the training dataset, n_i the number of replicate model evaluations at each design point location $\mathbf{x}_i$ $i \in [1, \ldots, N]$ and $diag$ signifies a diagonal matrix.

The algorithm is initialized by estimating a homoscedastic GP which is fitted on the empirical mean values. This is treated as our initial estimate of $\mathbf{G}_\mu$. We proceed by estimating the variance GP $\mathbf{G}_{\mathbf{\Sigma}}$. Where no replicate model evaluations are available for a design point $\mathbf{x}_i$, the predictive distribution of the mean GP $\mathbf{G}_\mu$ is sampled to estimate the noise levels of the data [4]. In the case of replicate evaluations at $\mathbf{x}_i$ the empirical variance $\mathbf{S}_i^2$ is estimated directly. To correct for the biased estimate of the variance due to the log transformation we apply the correction: $r_i = log(\mathbf{S}_i^2) + (d_i + d_i \log(2) - \Psi(d_i/2))^{-1}$, where r_i is the true log variance, $d_i = n_i - 1$, and Ψ the digamma function.

Finally the heteroscedastic GP $\mathbf{G}_\mu$ is estimated to jointly predict the mean and variance. The predictive distribution equations for $\mathbf{G}_\mu$ for M test points $\mathbf{x}_*$ are:

$$\begin{aligned} E[\mathbf{y}_*|\mathbf{x}_*, D] &= K^*(K + RP^{-1})^{-1}\mathbf{y} + E^T\bar{\beta}, \\ Var[\mathbf{y}_*|\mathbf{x}_*, D] &= K^{**} + R^* - K^{*^T}(K + R)^{-1}K^* + E^T(H(K + R)^{-1}H^T)^{-1}E, \end{aligned}$$

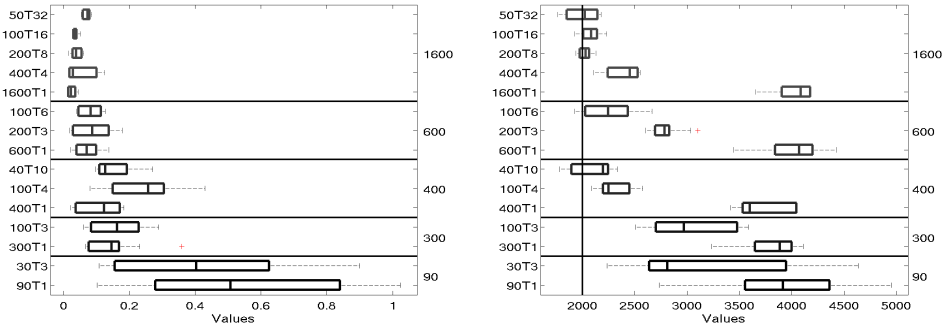
where $\mathbf{y} = [y_1 \dots y_N]$ is the vector of outputs in the training set D, K is the covariance of training points, K^* the cross-covariance between training and test points, K^{**} the covariance of test points, H a set of fixed basis functions, $\bar{\beta} = (H(K + R)^{-1}H^T)^{-1}H(K + R)^{-1}\mathbf{y}$ the regression coefficients, $E = H_* - H(K + R)^{-1}K^*$, $P = diag(n_1 \dots n_N)$ the number of replicates at each training point, $R = diag[r(x_1) \dots r(x_N)]$ and $R^* = diag[r(x_{*1}) \dots r(x_{*M})]$ the variance estimate from $\mathbf{G}_{\mathbf{\Sigma}}$ at the training and test points respectively. We note that the non-standard RP^{-1} term in the predictive mean arises from the use of replicate evaluations. The algorithm is repeated until convergence.

4 Experimental design analysis using synthetic data

In this section we utilize our framework to assess the efficacy of different experimental design towards emulation accuracy on a synthetic dataset [10]. Our chief validation measure is the Mahalanobis error $D_{MD} = (\mathbf{y} - \mathbf{t})'\Sigma^{-1}(\mathbf{y} - \mathbf{t})$, where $\mathbf{t}$ the vector of model outputs, $\mathbf{y}$ and Σ the predictive GP mean and covariance respectively. The Mahalanobis error assesses the goodness of the joint fit, both of the mean and covariance prediction [1].

In this experiment the total number of model evaluations is kept fixed and we contrast a space-filling design with only single model evaluations against a more widely-spaced replicate design that has the same number of evaluations for all design points.

The benefits of a replicate design can be seen in Figure 1 where the Mean Squared Error (MSE) and Mahalanobis error are shown for the different designs. There is little difference in terms of MSE signifying similar performance with regards to the prediction of the mean. The Mahalanobis error however reveals significant gains when replicate designs are used, reflecting an improvement in variance prediction. The replicate designs are also substantially faster to use from a computational perspective, i.e. inference time.



(a) Mean Squared Error

(b) Mahalanobis

Figure 1: Comparison of emulator fit where the total number of model evaluations is fixed at different levels. Notation is: 30T3 = 30 design points each with 3 replicates. Results shown for a total of 90, 300, 400, 600 and 1600 total number of model evaluations.

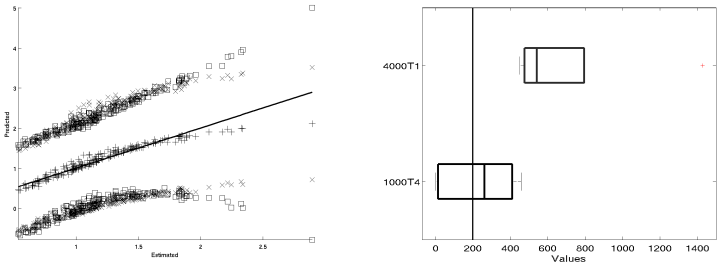
4.1 Stochastic Rabies Model

Although wildlife rabies was eradicated from large parts of Europe, there is a remaining risk of disease re-introduction. The situation is aggravated by an invasive species, the raccoon dog (Nyctereutes procyonoides) that can act as a second rabies vector in addition to the red fox (Vulpes vulpes). The purpose of our rabies model is to analyse the risk of rabies spread in this new type of vector community [9]. The individual-based, non-spatial, time-discrete model incorporates population and disease dynamical processes such as host reproduction and mortality as well as disease transmission. These processes are modelled stochastically to reflect natural variability (e.g. demographic stochasticity). Thus model analysis (e.g. sensitivity analysis) has to deal with stochastic, indeed heteroscedastic, model output.

The model output investigated in this study is the number of time steps to disease extinction. This output is important in deciding on the response to a potential rabies outbreak. This output has a rather complex, non-Gaussian, distribution for a fixed input; in this paper we emulate the first two moments of the log extinction time, which is more approximately Gaussian, as evidenced from visual inspection of Q-Q plots.

In Figure 2 we show the validation results of a single instance of our GP framework. The GPs were trained using a 1000 point Latin Hypercube design with a mixture of single and replicate model evaluations. A total of 4000 rabies model evaluations were used. In Figure 2(a), estimates of the 'correct' mean and standard deviation response (using 1000 repetitions) are plotted against the corresponding predicted values from $\mathbf{G}_\mu$.

We finally explore the question of how the replicate framework compares to approximations often applied within GP inference. The projected process method



(a) Estimated (square) vs Predicted (x) deviation and mean (+).

(b) Mahalanobis Error

Figure 2: (a) Emulating the rabies model using 1000 design points with a replicate design. (b) Projected process 'Kersting' (4000) vs replicated design (1000).

utilizes all N training points but it only represents $m < N$ latent function values, called support points, as an approximation to the full GP posterior [8]. In Figure 2(b) the Mahalanobis error of applying the approximation on [4] using a 4000 point space-filling design with $m = 1000$ support points is contrasted against the replicate method on a 1000 point space-filling design with 4 replicate observations at each design point. Both methods require approximately the same amount of computational resource, but the replicate observation method gives substantially better results, over 10 repetitions.

4.1.1 Screening of the rabies model

Lastly we consider using the replicate framework to perform screening which is often used as a preliminary stage in sensitivity analysis to remove clearly unimportant factors. In our framework, screening can be accomplished quite intuitively by looking at the posterior values of regression coefficients and correlation length scales. Furthermore these effects can be decomposed for the mean process ($\mathbf{G}_\mu$) and variance process ($\mathbf{G}_{\mathbf{\Sigma}}$).

The three dominant factors (out of 14 model inputs) on the variance response of the rabies model in terms of linear effects and correlation length scales are shown in Table 1. We observe that density and mortality rates of raccoon dogs have strong linear effects (significantly higher regression coefficients than other parameters). With regards to correlation length scales which reveal non-linear and interaction effects, factors related to disease in the vector species appear influential.

Table 1: Interpreting the variance emulator ($\mathbf{G}_{\mathbf{\Sigma}}$) by looking at the regression coefficients (Coeff) and correlation length scales (Scale).

Factor	Coeff	Factor	Scale
Rac Density	0.1608	Rac Rabid	1.4281
Rac Death	0.0633	Fox Inf	1.4594
Rac Birth	0.0200	Fox Rabid	1.5047

5 Conclusions

In this paper we have presented a new approach to the emulation of stochastic models which improves upon existing methods both in terms of accuracy and computational efficiency. Our framework allows further analysis to be carried out in a straight-forward and efficient manner using the emulator as a proxy for the simulator. Examples of such analyses include screening and uncertainty analysis, where we have included a demonstration of the former on a rabies model. Furthermore the computer model parameter space can be explored without the necessity of a large number of (computationally demanding) simulator runs. In combination with a discrepancy model and real-world observations, this method could facilitate the efficient statistical calibration of stochastic models.

References

[1] L. S. Bastos and A. O'Hagan. Diagnostics for Gaussian process emulators. Technical report, University of Sheffield, 2008.

[2] A. Boukouvalas and D. Cornford. Learning heteroscedastic Gaussian processes for complex datasets. Technical report, NCRG, Aston University, Aston University, Aston Triangle, Birmingham, B4 7ET, 2009.

[3] M.C. Kennedy and A. O'Hagan. Bayesian calibration of computer models (with discussion). *Journal of the Royal Statistical Society*, B63:425–464, 2001.

[4] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard. Most likely heteroscedastic Gaussian process regression. In Zoubin Ghahramani, editor, *Proc. 24th International Conf. on Machine Learning*, pages 393–400. Omnipress, 2007.

[5] J P C Kleijnen. Kriging metamodeling in simulation: a review. *European Journal of Operational Research*, 2007.

[6] J.P.C. Kleijnen and W.C.M. van Beers. Robustness of Kriging when interpolating in random simulation with heterogeneous variances: Some experiments. *European Journal of Operational Research*, 165(3):826–834, 2005.

[7] Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *J. Mach. Learn. Res.*, 9:235–284, 2008.

[8] C E Rasmussen and C K I Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[9] A. Singer, F. Kauhala, K. Holmala, and G.C. Smith. Rabies risk in raccoon dogs and foxes. *Developments in Biologicals*, 131:213–222, 2008.

[10] M. Yuan and G. Wahba. Doubly penalized likelihood estimator in heteroscedastic regression. *Statistics and Probability Letters*, 69:11–20, 2004.