Computer Experiments: Prediction Accuracy, Sample Size and Model Complexity Revisited

Ofir Harari[*], Derek Bingham[*], Angela Dean [†]and Dave Higdon [‡]

Abstract

We revisit the problem of determining the sample size for a Gaussian process emulator and provide a data analytic tool for exact sample size calculations that goes beyond the $n = 10d$ rule of thumb and is based on an IMSPE-related criterion. This allows us to tie sample size and prediction accuracy to the anticipated roughness of the simulated data, and to propose an experimental process for computer experiments, with extension to a robust scheme.

1 Introduction

The Gaussian process model was proposed by Sacks et al. (1989b) as a statistical emulator for deterministic computer codes, and a large body of literature has subsequently been devoted to the exploration of its performance under various conditions. Experimental design for computer experiments has been extensively investigated, whether space-filling designs (see e.g. Johnson et al. 1990, Joseph et al. 2012) or optimal designs driven by different statistical criteria (e.g. Sacks et al. 1989a, Shewry and Wynn 1987, Harari and Steinberg 2014).

[*]Department of Statistics and Actuarial Science, Simon Fraser University

[†]Department of Statistics, The Ohio State University

[‡]Social and Decision Analytics Laboratory, Biocomplexity Institute of Virginia Tech

With many of the bases appearing to be covered, one would think that the design of experiments for deterministic computer model emulation would be a topic with little left to study. It is thus surprising that –

(a) Only a small amount of literature has been dedicated to the foundational topic of sample size in computer experiments, and

(b) unlike other areas of statistics, in which awareness to the curse of dimensionality has long been raised and model flexibility is sacrificed in favor of structure, the computer experiments practitioners often make the naive assumption that the sample size should grow linearly with the input dimension, known as the "$n = 10d$" rule of thumb, where d is the number of inputs to the computer model (Chapman et al. 1994, Jones et al. 1998).

A partial justification for the latter is given by Loeppky et al. (2009), who advocate the $n = 10d$ rule in very specific cases. Their message is that, for relatively uncomplicated surfaces and moderate d, good prediction accuracy can be obtained with $n = 10d$ observations in an initial experiment, and increasing n further can increase accuracy further. However, if $n = 10d$ observations results in poor accuracy, (which tends to happen in complicated or high dimensional codes with input factors having similar complexity), then the improvement in accuracy through adding more runs tends not to be helpful.

In this paper, rather than taking a dichotomous view of settings for successful computer model emulation, we aim to reveal a broader spectrum of choices using analytic methods. Our aim is not to show that the $n = 10d$ rule of thumb is bad. On the contrary, in many cases $n = 10d$ is suitable, but it is a very coarse rule. Instead, we go further than Loeppky et al. (2009) and take the view that the choice of experimental design ought to take into account the prior belief about the complexity of the response surface, the desired prediction accuracy and the available resources.

The underlying structural assumptions about the approximated response surface by embracing the $n = 10d$ rule are far-reaching, and are often not fully understood. For $d = 20$, for example, $n = 200$ would not even suffice to estimate a linear model with 20 main effects and all 190 two-factor interactions. Fitting a Gaussian process model using so few data points must then reflect the belief by experimenter that either the change in the response is purely additive in many of the factors or that several factors are entirely inert. Additionally, post hoc diagnostics may also be misleading, as the Kriging predictor interpolates the data. Thus a seemingly good fit may, in reality, be no more than over-smoothing and ignoring the complexity of the true response due to large areas of the input space remaining unsampled. That is akin to the aliasing of under-sampled signals (see e.g. Diniz et al. 2002).

Originally, small sample sizes were a consequence of lengthy simulation run-times. In addition, numerical singularity in the correlation matrix and computational issues, stemming from the need to store and repeatedly invert a large $n \times n$ covariance matrix, have contributed to the general reluctance to deal with large samples in computer experiments. However, with remedies already in place (see e.g. Ranjan et al. 2011, Kaufman et al. 2011), in conjunction with improving computing capabilities, it is expected that large-scale computer experiments will routinely take place in the near future. It is easy to imagine models with a large number of inputs (≥ 100) where factor sparsity (Box and Meyer 1986) implies that relatively few (≈ 20) of the inputs are important. In these cases, careful consideration of the model structure and the goals of the experiment are important.

Even without fixing ideas quantitatively, the sample size for an experiment, the complexity of the model and the prediction goals of the experiment are intimately related. In this paper we attempt to provide methodology for computer experiments addressing questions that scientists have been asking for quite some time for physical experiments:

1. For a given model complexity, what is the minimum sample size required to achieve a desired level of prediction accuracy?

2. For a given sampling budget and model complexity, what level of prediction accuracy can be expected?

3. For a given budget and desired prediction accuracy, what is the most complex model one can anticipate being able to estimate?

This paper is organized as follows. Section 2 provides a brief introduction of the Gaussian process model commonly used for computer model emulation. In addition, we take high prediction accuracy as an experimental objective and propose different interpretations for that goal. Section 3 ties the prediction accuracy to the sample size and the model hyperparameters that specify the response surface complexity, and discusses consequences for experimental design. In Section 4 we develop a methodical experimental process for computer experiments, and in Section 5 we demonstrate the proposed process on an experiment involving a piston simulator. We then provide a robust scheme in Section 6, to handle uncertainty with regard to the hyperparameters. Finally, Section 7 includes a discussion and future work.

For the reader's convenience, a web applications was created to accompany this paper and facilitate future analyses. For details, see the Supplementary Materials section.

2 Gaussian Processes Emulators for Deterministic Computer Models

2.1 Gaussian Process Regression

Building a computer model emulator can be viewed as nonparametric regression for deterministic simulators. The reasons for using the conventional specification for the Gaussian

process (Sacks et al., 1989b) lie with its ability to interpolate the model output and to quantify uncertainty at unsampled inputs.

In this setting, the computer model output, $y(\boldsymbol{x})$, is viewed as a realization of a stationary, zero mean, Gaussian process with covariance function $C(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 R_{\boldsymbol{\theta}}(\boldsymbol{x} - \boldsymbol{x}')$. The *correlation function*, $R_{\boldsymbol{\theta}}(\cdot)$, depends on the vector of hyperparameters $\boldsymbol{\theta}$ that govern the correlation between responses at separate locations. Denote the experimental design $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\} \subset \mathcal{X}$, where $\mathcal{X} \subset \mathbb{R}^d$ is the experimental region. We will assume, without loss of generality, that $\mathcal{X} = [0,1]^d$. Let $\boldsymbol{y} = [y(\boldsymbol{x}_1), \ldots, y(\boldsymbol{x}_n)]^\mathsf{T}$ be a vector of observations at the design points, and denote by $\boldsymbol{R}_{\boldsymbol{\theta}}$ the matrix whose entries are $R_{ij} = R_{\boldsymbol{\theta}}(\boldsymbol{x}_i - \boldsymbol{x}_j)$. Then for any $\boldsymbol{x} \in \mathcal{X}$, choosing the *Kriging* predictor

$$\widehat{y}(\boldsymbol{x}) = \mathbb{E}\left\{y(\boldsymbol{x}) \,\middle|\, \boldsymbol{y}\right\} = \boldsymbol{r}_{\boldsymbol{\theta}}(\boldsymbol{x})^\mathsf{T} \boldsymbol{R}_{\boldsymbol{\theta}}^{-1} \boldsymbol{y}, \tag{1}$$

to predict $y(\boldsymbol{x})$ would yield the *Mean Squared Prediction Error* (MSPE)

$$\mathbb{E}\left[\{\widehat{y}(\boldsymbol{x}) - y(\boldsymbol{x})\}^2 \,\middle|\, \boldsymbol{y}\right] = \mathbb{V}\mathrm{ar}\left\{y(\boldsymbol{x}) \,\middle|\, \boldsymbol{y}\right\} = \sigma^2 \left\{1 - \boldsymbol{r}_{\boldsymbol{\theta}}(\boldsymbol{x})^\mathsf{T} \boldsymbol{R}_{\boldsymbol{\theta}}^{-1} \boldsymbol{r}_{\boldsymbol{\theta}}(\boldsymbol{x})\right\}, \tag{2}$$

for $\boldsymbol{r}_{\boldsymbol{\theta}}(\boldsymbol{x}) = [R_{\boldsymbol{\theta}}(\boldsymbol{x} - \boldsymbol{x}_1), \ldots, R_{\boldsymbol{\theta}}(\boldsymbol{x} - \boldsymbol{x}_n)]^\mathsf{T}$. For the rest of this paper we shall suppress the $\boldsymbol{\theta}$ subscript, keeping in mind that the correlation between responses at different locations depends heavily on these hyperparameters.

2.2 A Measure for Prediction Accuracy

Using (2), the Integrated MSPE (IMSPE) of the Kriging predictor (1) is given by

$$\mathcal{J}(\widehat{y}, \mathcal{D}; \boldsymbol{\theta}) = \int_{[0,1]^d} \mathbb{E}\left[\{\widehat{y}(\boldsymbol{x}) - y(\boldsymbol{x})\}^2 \,\middle|\, \boldsymbol{y}\right] \mathrm{d}\boldsymbol{x} = \sigma^2 - \sigma^2 \mathrm{tr}\left\{\boldsymbol{R}^{-1} \int_{[0,1]^d} \boldsymbol{r}(\boldsymbol{x})\, \boldsymbol{r}(\boldsymbol{x})^\mathsf{T} \mathrm{d}\boldsymbol{x}\right\}. \tag{3}$$

Weighted versions of (3) have been proposed to emphasize prediction in certain areas of the design region (see e.g. Sacks et al. 1989b).

Recalling that $y(\boldsymbol{x})$ is assumed to be stationary (i.e., $\mathbb{V}\mathrm{ar}\{y(\boldsymbol{x})\} = \sigma^2$), we may then think of the normalized quantity

$$\frac{\mathcal{J}(\widehat{y}, \mathcal{D}; \boldsymbol{\theta})}{\sigma^2} = \int_{[0,1]^d} \frac{\mathbb{V}\mathrm{ar}\left\{y(\boldsymbol{x}) \middle| \mathcal{D}\right\}}{\mathbb{V}\mathrm{ar}\{y(\boldsymbol{x})\}} \mathrm{d}\boldsymbol{x} \tag{4}$$

as the average proportion of the variability of $y(\boldsymbol{x})$ that remains unexplained by design $\mathcal{D}$. This is reminiscent of the proportion of unexplained variability in linear regression models, typically calculated as the ratio of the sum of squares for error and the total sum of squares. In this case. however, (4) is for out of sample observations. Following this analogy, we can form the counterpart of the squared multiple correlation coefficient in regression, as in the following proposition.

Proposition 1. *Let* $y(\boldsymbol{x}) \sim \mathrm{GP}(0, \sigma^2 \mathrm{R})$ *and let* $\widehat{y}(\boldsymbol{x}) = \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \boldsymbol{R}^{-1} \boldsymbol{y}$ *be the Kriging predictor of* $y(\boldsymbol{x})$*. Then*

$$1 - \frac{\mathcal{J}(\widehat{y}, \mathcal{D}; \boldsymbol{\theta})}{\sigma^2} = \overline{\rho}^2(y, \widehat{y}) \coloneqq \int_{[0,1]^d} \rho^2(y(\boldsymbol{x}), \widehat{y}(\boldsymbol{x}))\, \mathrm{d}\boldsymbol{x}, \tag{5}$$

where $\rho(\cdot, \cdot)$ *denotes the correlation coefficient.*

Proof. First note that, from (1)

$$\mathrm{cov}(y(\boldsymbol{x}), \widehat{y}(\boldsymbol{x})) = \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \boldsymbol{R}^{-1} \mathrm{cov}(y(\boldsymbol{x}), \boldsymbol{y}) = \sigma^2 \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}).$$

Now,

$$\mathbb{V}\mathrm{ar}\{\widehat{y}(\boldsymbol{x})\} = \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \boldsymbol{R}^{-1} \left(\sigma^2 \boldsymbol{R}\right) \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) = \sigma^2 \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}),$$

and since $\mathbb{V}\mathrm{ar}\left\{y\left(\boldsymbol{x}\right)\right\} = \sigma^2$ we have

$$\rho^2\left(y\left(\boldsymbol{x}\right), \widehat{y}\left(\boldsymbol{x}\right)\right) = \frac{\left\{\sigma^2 \boldsymbol{r}\left(\boldsymbol{x}\right)^{\mathsf{T}} \boldsymbol{R}^{-1} \boldsymbol{r}\left(\boldsymbol{x}\right)\right\}^2}{\sigma^2 \cdot \sigma^2 \boldsymbol{r}\left(\boldsymbol{x}\right)^{\mathsf{T}} \boldsymbol{R}^{-1} \boldsymbol{r}\left(\boldsymbol{x}\right)} = \boldsymbol{r}\left(\boldsymbol{x}\right)^{\mathsf{T}} \boldsymbol{R}^{-1} \boldsymbol{r}\left(\boldsymbol{x}\right) \quad ,$$

and the result follows from (2) and (4). □

We can interpret $\overline{\rho}^2\left(y, \widehat{y}\right)$ in (5) as the average squared correlation among the simulator responses and predicted responses at unsampled inputs. More importantly, Proposition 1 sheds new light on the interpretation of IMSPE-optimal designs (see e.g. Sacks et al. 1989b). Clearly, by minimizing the IMSPE we can expect to improve the predictive ability of the Kriging predictor. The proposition demonstrates that minimizing the IMSPE is equivalent to maximizing the average, squared, out-of-sample correlation between $y\left(\boldsymbol{x}\right)$ and $\widehat{y}\left(\boldsymbol{x}\right)$.

Definition 1.

The *Root Average Unexplained Variability* (RAUV) of predictor $\widehat{y}$, evaluated at design $\mathcal{D} \subset [0,1]^d$ is

$$\mathrm{RAUV}\left(\widehat{y}; \mathcal{D}, \boldsymbol{\theta}\right) = \left(\frac{\mathcal{J}\left(\widehat{y}, \mathcal{D}; \boldsymbol{\theta}\right)}{\sigma^2}\right)^{1/2} = \left(\int_{[0,1]^d} \frac{\mathbb{V}\mathrm{ar}\left\{y\left(\boldsymbol{x}\right) \middle| \mathcal{D}\right\}}{\mathbb{V}\mathrm{ar}\left\{y\left(\boldsymbol{x}\right)\right\}} \mathrm{d}\boldsymbol{x}\right)^{1/2} .$$

We propose the RAUV as a measure of expected prediction error on designing a computer experiment. Its indicated magnitude is relative to the signal strength (i.e. the prior standard deviation), and the choice of the square root scale is in line with similar measures used by both Loeppky et al. (2009) and Chen et al. (2016). In general, it is common practice to measure model performance by its Empirical Root Mean Square Error (ERMSE) computed on a holdout set, normalized by some measure of the variation in the data measured in the original units, such as the range or the empirical standard deviation. We have found sample sizes that warrant low a priori RAUV to be consistent with good empirical prediction accuracy,

much more so than the average unexplained variability without the square root. It can also be justified equivalently in terms of uncertainty quantification: requiring RAUV ≤ 0.05 means that we want the square root of the average squared length of our prediction intervals to shrink by 95% once data is observed. Note that from Proposition 1

$$\text{RAUV}\left(\widehat{y}; \mathcal{D}, \boldsymbol{\theta}\right) = \sqrt{1 - \overline{\rho}^2\left(y, \widehat{y}\right)}.$$

To summarize, the following design objectives are equivalent for Gaussian process model fitting in the context of computer model emulation:

1. Ensuring RAUV $\left(\widehat{y}\right) \leq \varepsilon$.
2. Explaining at least $100\left(1 - \varepsilon^2\right)\%$ of the variability in $y\left(\boldsymbol{x}\right)$ by the model.
3. Achieving $\left|\overline{\rho}\left(\widehat{y}, y\right)\right| \geq \sqrt{1 - \varepsilon^2}$.

In the next section, the RAUV will play a key role in evaluating specific design settings.

3 Model Complexity, Sample Size and Prediction Accuracy

The aim of this paper is to create a link between the complexity of the model being estimated, prediction accuracy and sample size. Micchelli and Wahba (1981) and Harari and Steinberg (2014) derived the following result that can shed light on this goal.

Theorem 1. *Let*

$$R\left(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}\right) = \sum_{k=1}^{\infty} \lambda_k \varphi_k\left(\boldsymbol{x}\right) \varphi_k\left(\boldsymbol{x}'\right)$$

be the Mercer expansion of $R(\cdot; \boldsymbol{\theta})$, where the $\mathcal{L}_2$-orthogonal eigenfunctions $\{\varphi_k\left(\boldsymbol{x}\right)\}$ and the non-negative, real valued eigenvalues $\{\lambda_k\}$ are the solutions of the Fredholm integral equation

of the second kind, namely

$$\int_{[0,1]^d} R\left(\boldsymbol{x},\boldsymbol{u};\boldsymbol{\theta}\right)\varphi_k\left(\boldsymbol{u}\right)\mathrm{d}\boldsymbol{u}=\lambda_k\varphi_k\left(\boldsymbol{x}\right),\ \ k=1,2,\ldots,\ \ \lambda_1\geq\lambda_2\geq\cdots, \tag{6}$$

then

$$\inf_{\mathcal{D}}\left\{\frac{\mathcal{J}\left(\widehat{y};\mathcal{D},\boldsymbol{\theta}\right)}{\sigma^2}\right\}\geq\sum_{k\geq n+1}\lambda_k. \tag{7}$$

Over any rectangular region, the integral equation (6) can be handily solved numerically; in particular, for separable correlation functions, the problem reduces to a series of univariate eigendecompositions. Details of the numerical procedure are provided in Harari and Steinberg (2014).

Although it may seem unclear at first glance, and not designed for this goal, inequality (7) encapsulates the complex relationship between sample size, model complexity (in the form of the Gaussian process hyperparameters) and prediction accuracy. However, to the best of our knowledge, no use has been made of Theorem 1 in this way. One immediate result arising from Theorem 1 is the following.

Corollary 1.

Let $\{\lambda_k\}$ be the set of eigenvalues of $R\left(\boldsymbol{x},\boldsymbol{x}';\boldsymbol{\theta}\right)$ obtained by solving (6), and denote by n^c the critical sample size required to achieve RAUV $\leq\varepsilon$ for some $\varepsilon>0$. Then

$$n^c\geq\min\left\{n:\sqrt{\sum_{k\geq n+1}\lambda_k}\leq\varepsilon\right\}=\min\left\{n:\sum_{k=1}^{n}\lambda_k\geq1-\varepsilon^2\right\}. \tag{8}$$

Corollary 1 reduces the need for guesses or rules of thumb regarding the sample size needed

for a computer experiment. Instead, given some idea about the way in which the different inputs act, one can (at least approximately) derive analytically the required sample size for a given average level of prediction accuracy. For example, if we would like to explain some proportion of the variability in the response surface (or, equivalently, achieve an acceptable amount of unexplained variability), this leads us to a choice for ε. So what is a "good" value of ε in practice? Obviously, that will depend on the particular application and experimenter goals.

To gain some insight on the choice of ε, consider the piston example of Section 5, which will be described in detail later. Figure 7 displays actual vs. predicted values for a large holdout set, where the predicted values are obtained from (1). The four panels show sample sizes of 30, 50, 70 and 120, respectively, which, for the hyperparameters $\boldsymbol{\theta}$ selected for the piston example, correspond to ε values of 0.191, 0.120, 0.086 and 0.045. Looking at the top left panel of the figure which has sample size of $n = 30$ and $\varepsilon = 0.191$, the quality of the fit is far from perfect. Surprisingly, this corresponds to 96% of the average explained variability, which would be high in terms of the measure r^2 in linear regression for noisy data. In our experience, agreement such as that indicated in top right panel of Figure 7 (with $n = 50$ and $\varepsilon = 0.1$) reflects, for deterministic simulators, an acceptable fit, i.e.: one should require that at least 99% of the variability (on average) be explained. An even better fit such as the one presented in the bottom right corner of Figure 7 corresponds to $\varepsilon = 0.05$ (and $n = 120$).

In principle, exact sample size calculations can be achieved by finding IMSPE-optimal designs for various run sizes and RAUV specifications (by trial and error, the critical n is found), the run times for such a hard optimization problem deem it impractical – especially when there is uncertainty regarding parameter values (see Section 6) – while for any reasonable number of inputs solving (6) is almost instantaneous.

Remark 1.

Ideally, we would love to have derived an upper bound of the form

$$\inf_{\mathcal{D}} \left\{ \frac{\mathcal{J}\left(\widehat{y}; \mathcal{D}, \boldsymbol{\theta}\right)}{\sigma^2} \right\} \leq \sum_{k \geq n+1} \lambda_k + \alpha^2 \left(n, \boldsymbol{\theta}\right)$$

and consequently bound the critical sample size $n_{_L} \leq n^c \leq n_{_U}$, with $n_{_U}$ guaranteeing an RAUV below the desired threshold. However, no meaningful upper bound exists at this point for this noiseless case (in fact, whether or not the lower bound can be achieved remains an open question). We therefore treat inequality (7) roughly as an equality throughout this paper, with the convention that we need to choose a slightly larger sample size than the one recommended by (8). This approach is supported by Figure 2 and the findings of the simulation study in Section 5.

We now proceed to use these ideas in a practical setting by fixing any two of the three vertices of the triangle appearing in Figure 1 and observing the impact on the third vertex. First, in Section 3.1, we will demonstrate how to put Corollary 1 into practice by deriving the required sample size for a desired level of prediction accuracy and fixed correlation parameters, then, in Section 3.2, we will show how to assess prediction accuracy when sample size limitations are strict and the correlation parameters are specified, and, finally in Section 3.3, we will provide partial answer to the question of the most complicated function that can be learned to a satisfactory level (in terms of the RAUV) for a given sample size.

3.1 Sample size for a desired level of prediction accuracy and fixed $\boldsymbol{\theta}$

For fixed Gaussian process hyperparameters (model complexity) and a given level of prediction accuracy (RAUV), determination of the required sample size follows a simple application

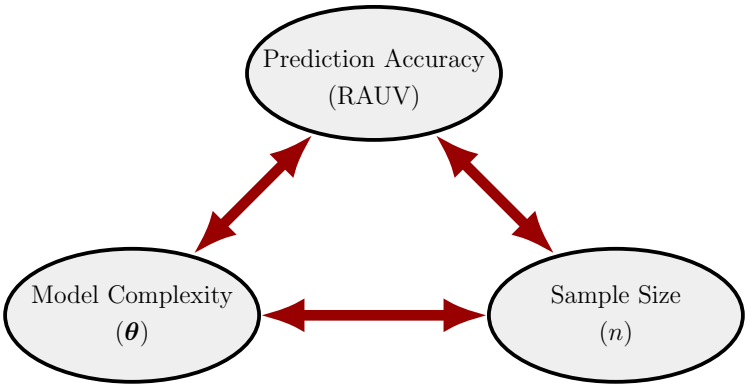


Figure 1: The three interacting elements of a (computer) experiment.

of Corollary 1. This approach is illustrated using a few straightforward examples.

Example 3.1.1.

To illustrate the use of (8), we consider the computer code used by Yi et al. (2005) to simulate ligand activation of G-protein in yeast. Loeppky et al. (2009) fixed five of nine factors and used a 4-dimensional Gaussian process emulator for the response, using the *squared-exponential* correlation function

$$R\left(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}\right) = \exp\left\{-\sum_{i=1}^{4} \frac{|x_i - x_i'|^2}{\theta_i}\right\}. \tag{9}$$

Here $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3, \theta_4)$ is a vector of *correlation length* parameters, where smaller θ_i values make for more complex ("bumpy") realizations along the x_i direction.

Based on a design with $n = 80$ runs, they found the maximum likelihood estimates for the emulator to be $\widehat{\boldsymbol{\theta}} = (9.09, 1.59, 1.79, 0.56)$. Treating these for the moment as the actual parameters for the data-generating process, we can solve the eigenvalue problem (6). Figure 2 shows the lower bound $\sqrt{\sum_{k \geq n+1} \lambda_k}$ for the RAUV versus n. If we set a threshold $\varepsilon = 0.05$, the smallest sample size for which the threshold is crossed is $n = 21$. Also plotted are the RAUV values for IMSPE-optimal designs of various sample sizes and the given $\boldsymbol{\theta}$. While the

theoretical lower bound is closely approached by the empirical values, caution needs to be taken and a few more runs (in this example 5 or 6) may be needed to guarantee that the desired precision level is achieved. More conservatism is called for when the design to be used is not IMSPE-optimal. In this example both the lower bound curve and the empirical IMSPE values indicate that a sample size of $n = 10d = 40$ should be more than enough for an adequate fit (if parameter estimates are to be trusted), which is consistent with the findings of Loeppky et al. (2009).

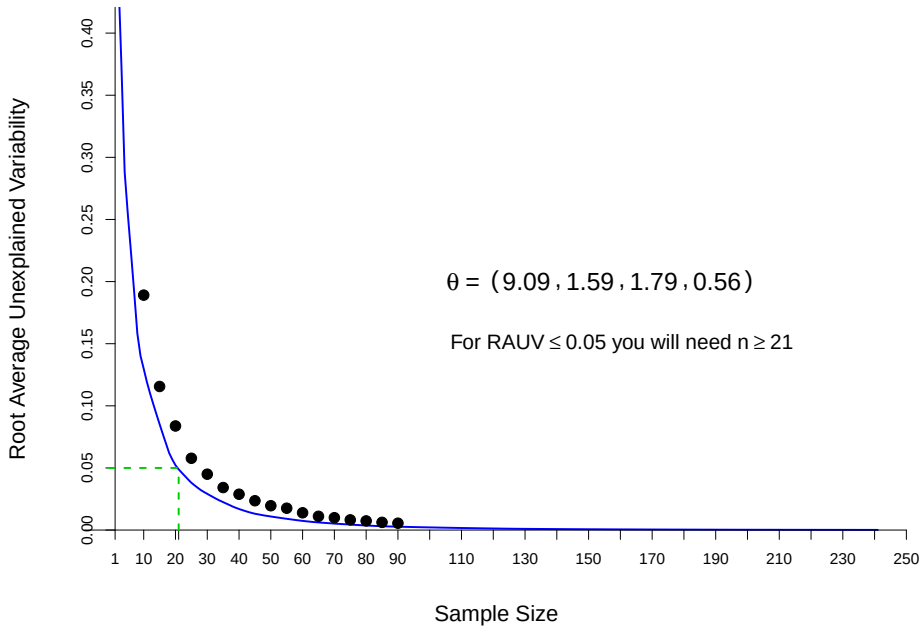


Figure 2: RAUV lower bound curve for the squared exponential correlation function, with the estimated correlation length parameters for the G-protein example from Loeppky et al. (2009). The dots in the figure denote RAUV values calculated for IMSPE-optimal designs.

Example 3.1.2.

As a second example, we wish to find the required n that will give us an RAUV of $\varepsilon = 0.05$ for a Gaussian process model with the product Matérn correlation function

$$R(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\phi}, \boldsymbol{\nu}) = \prod_{i=1}^{d} \frac{1}{\Gamma(\nu_i) 2^{\nu_i - 1}} \left(\frac{2\sqrt{\nu_i}\,|x_i - x_i'|}{\phi_i} \right)^{\nu_i} \mathcal{K}_{\nu_i} \left(\frac{2\sqrt{\nu_i}\,|x_i - x_i'|}{\phi_i} \right),$$

where ϕ_i and ν_i are the correlation length and smoothness parameters along the ith direction, respectively, and $\mathcal{K}_\nu(\cdot)$ is the modified Bessel function of order ν. Here we focus on an isotropic 4-dimensional process with $\phi_1 = \cdots \phi_4 = 1$ and $\nu_1 = \cdots = \nu_4 = 5/2$ (to guarantee twice differentiable realizations). In this case the lower bound curve (see Figure 3) indicates that a sample size of $n \geq 112$ will be required for the precision target we set for ourselves, and the $n = 10d$ rule with $d = 4$ is inadequate. It is not surprising that a process whose realizations are harder to predict (due to more limited smoothness, compared to the $\mathcal{C}^\infty$ realizations of a process that is based on the squared exponential correlation function, see e.g. (Santner et al., 2003)) requires more observations for the same level of prediction accuracy.

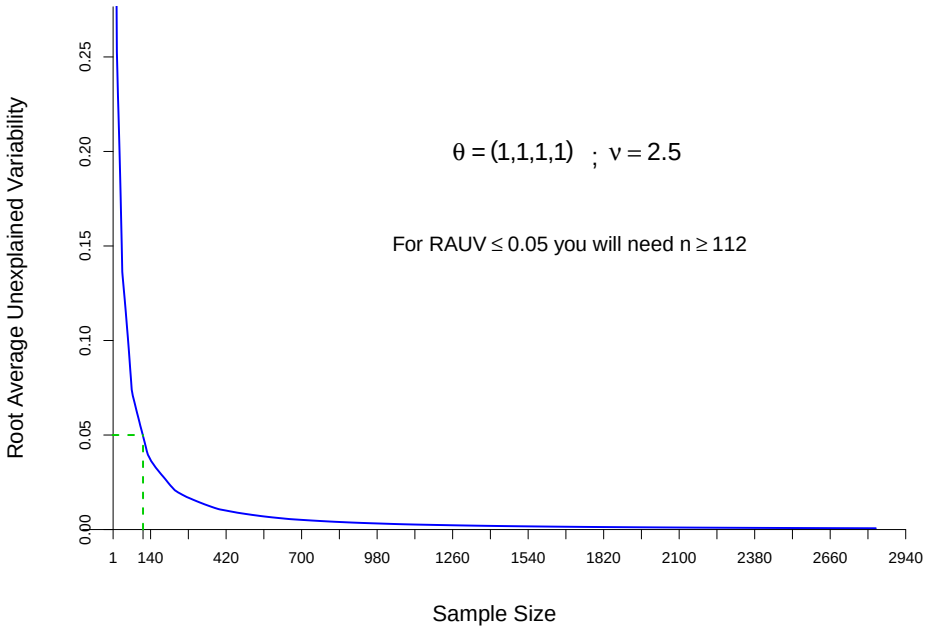


Figure 3: RAUV lower bound curve for the product Matérn correlation function of an isotropic process with $\nu = 5/2$ and $\theta = 1$.

3.2 Assessing prediction accuracy for fixed n and $\boldsymbol{\theta}$

Consider the setting where an experimenter has a limited computation budget - this may be due, for example, to a limited allocation of computer time on a cluster or super-computer. In this case, the experimenter is interested in anticipated quality of the predictions that can be achieved for the emulator. For instance, solving (6) and taking advantage of (7), we learn that for the product Matérn correlation function specified in Example 3.1.2, with $n = 40$ runs, would lead to RAUV ≥ 0.128. Should this level of inaccuracy be deemed excessive, the experimenter may consider instead exploring only a sub-set of the inputs or somehow increasing the computational budget. In Section 4 we will elaborate on such contingencies.

3.3 Maximum model complexity for fixed n and prediction accuracy

Inequality (7) does not define a one-to-one relationship between the Gaussian process hyperparameters and the sample size required for a desired level of prediction accuracy. If, however, one only considers isotropic models, (8) can be inverted. Suppose, for example, that one wishes determine the most complicated isotropic Gaussian process model that can be investigated with prediction accuracy RAUV ≤ 0.05 and the product Matérn correlation function with $\nu = 5/2$, with a computational budget of $n = 40$. It can be verified that correlation length of $\phi = 1.46$ is the minimum value that results in $\sqrt{\sum_{k \geq 41} \lambda_k} \leq 0.05$. With this parameter determined, the experimenter can then produce Functional Analysis of Variance (FANOVA) plots (see Saltelli et al. 2008) of realizations from an isotropic Gaussian process with $\phi = 1.46$.

Figure 4 displays 20 realizations from a univariate Gaussian process based on the product Matérn correlation function with $\phi = 1.46$ and $\nu = 5/2$. If the realizations demonstrate a complexity that is less than the experimenter's belief in the computer model then this ϕ is

unlikely to achieve the desired level of precision. In that case, one may either find a way to increase the sample size or lower the a priori expectations with regard to prediction accuracy.

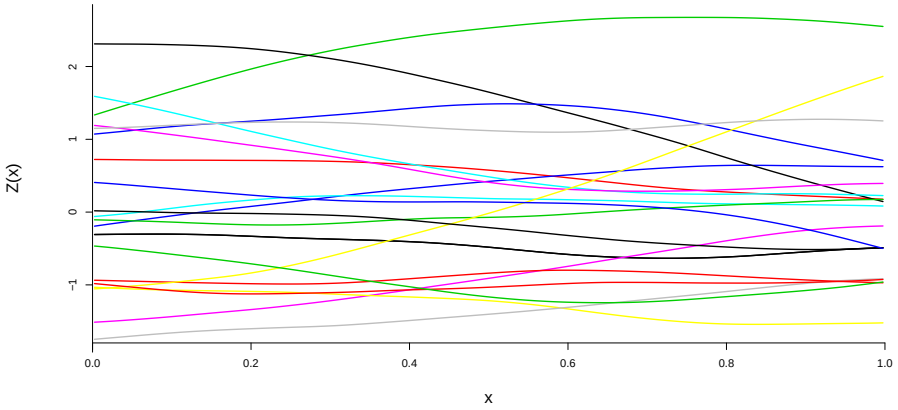


Figure 4: 20 realizations drawn from a univariate Gaussian process based on the product Matérn correlation function with $\phi = 1.46$ and $\nu = 5/2$.

4 Stepping Through Design Process

Ultimately, the design of computer experiments shares many of the same features as the design of physical experiments. The experimenter is usually faced with having to propose 2 of the 3 values in Figure 1. At this point we would like to take a moment to gather the various ideas put forward in Section 3. In general, we recommend roughly following the proposed scheme, below, when designing a computer experiment.

1. **Eliciting a prior**

   Choose a suitable correlation family and hyperparameters. This amounts to choosing the response surface model and related complexity. This is the most subtle part of the process and should be carefully considered by an expert (most likely the person who coded the simulator). For each individual input, the question that needs to be asked is – "if all other inputs are fixed at their respective midpoints, and letting only this input vary, what would you expect the level of smoothness/wiggliness of the response to

be?". We recommend presenting the experimenter with several of plots or realizations similar to Figure 4 as guidance because most will not be able to link the correlation parameters directly to the model complexity. Additionally, we would recommend erring on the conservative side (choosing a prior process that is slightly more wiggly than the true response) as a matter of robustness. Careful consideration similar to this of the likely complexity of a response surface is routinely done for physical experiments for polynomial regression models and related power calculations, for example.

2. **Calculating the required sample size/expected accuracy**

   Now that a target response surface has been fully specified, the experimenter may proceed to obtain a lower bound for the required number of runs for a desired level of prediction accuracy (as in Subsection 3.1). Alternatively, if the researcher operates on a fixed budget of runs, the expected prediction accuracy of the Gaussian process model (see Subsection 3.2) can be assessed. As a third alternative, one can examine the complexity level of the response that the budget and prediction accuracy allow, as in Section 3.3.

3. **Making operational decisions**

   If the first strategy of step 2 is taken, and if the calculated required number of runs is operationally feasible, one can then proceed with the computer experiment using that number of runs (preferably more to compensate for the inequality in (8)). If, however, for a feasible number of runs, the calculated RAUV appears to be greater than a tolerable level, one of the following mitigating measures may be considered. For example, one of:

   (a) **Reducing dimensionality by eliminating inputs:** Obviously, every input coded into the simulator likely matters to some degree. Some inputs, however, may be thought to be less influential than others. If an expert can identify inputs whose absence from the model may have little bearing on the response, omitting

those from the model (while holding them fixed at, say, their midpoint during computer runs) can significantly decrease the required number of runs.

(b) **Altering the emulation model:** The interpolation property, combined with the uncertainty attached to predictions (through the predictive distribution) and zero posterior variance at the data, have made Kriging an appealing emulation method for deterministic computer models. It is also well-known, though, for regressing towards the mean fairly quickly when departing from the observed data, especially in high dimensions. Excessive RAUV is an indication that lack of training data would lead to predicting a constant everywhere (except for spikes at the observed simulator outputs), in which case one may consider sacrificing interpolation in favor of some added structure (e.g., a regression model for the mean of the simulator response surface using specified basis functions). In other words: retreat to "traditional", parametric statistical models.

5 An Illustration

For illustration purposes only, the piston simulation appearing in Kenett and Zacks (1998) is now considered. Here, a piston's linear motion is transformed into circular motion of a rod connected to a disk. The measured response is the time it takes to complete one cycle, given by

$$C\left(M, S, V_0, k, P_0, T_a, T_0\right) = 2\pi \left(\frac{M}{k + S^2 \frac{P_0 V_0}{T_0} \frac{T_a}{V^2}} \right)^{1/2} \tag{10}$$

$$\text{for} \quad V = \frac{S}{2k} \left\{ \left(A^2 + 4k \frac{P_0 V_0}{T_0} T_a \right)^{1/2} - A \right\} \quad \text{and} \quad A = P_0 S + 19.62 M - \frac{k V_0}{S},$$

where

- $M \in [30, 60]$ is the piston weight (kg),
- $S \in [0.005, 0.020]$ is the piston surface area (m^2),
- $V_0 \in [0.002, 0.010]$ is the initial gas volume (m^3),
- $k \in [1000, 5000]$ is the spring coefficient (N/m),
- $P_0 \in [9 \times 10^4, 11 \times 10^4]$ is the atmospheric pressure (N/m^2),
- $T_a \in [290, 296]$ is the ambient temperature (K), and
- $T_0 \in [340 - 360]$ is the filling gas temperature (K).

More documentation (and code) for this model can be found at http://www.sfu.ca/~ssurjano/emulat.html.

Suppose now that the underlying model is unknown, but an expert has provided the following information:

- With all other factors being held fixed, letting M increase will increase the cycle time moderately in a nonlinear fashion. The same is true for k, although a reverse trend is expected.
- Likewise, increasing S will result in a sharp, nonlinear decrease in cycle time, while the opposite will happen when V_0 alone varies.
- Within the experimental region the average effect of P_0 is very limited.
- Varying T_a has an unnoticeable effect on the cycle time. The same is true for T_0.

In practice, we would use realizations drawn from univariate Gaussian processes, with different values of the correlation lengths, to help specify $\boldsymbol{\theta}$. In the absence of a domain expert for this illustration, our assessments were based on the main effect plots from the FANOVA

of the cycle time $C(\boldsymbol{x})$, appearing in Figure 5. In light of the sensitivity plots, a Gaussian process prior with a squared exponential correlation function (9) was selected with hyper-parameters $\boldsymbol{\theta} = (1, 0.4, 0.4, 1, 3, 10, 10)$, leading to a sample size of $n \geq 210$ when aiming at RAUV ≤ 0.05 in (8). Instead, we chose to ignore T_0 and T_a altogether and treated the model as 5 dimensional, leading to a critical sample size of $n \geq 111$. We proceeded to compare the performance of different sample sizes.

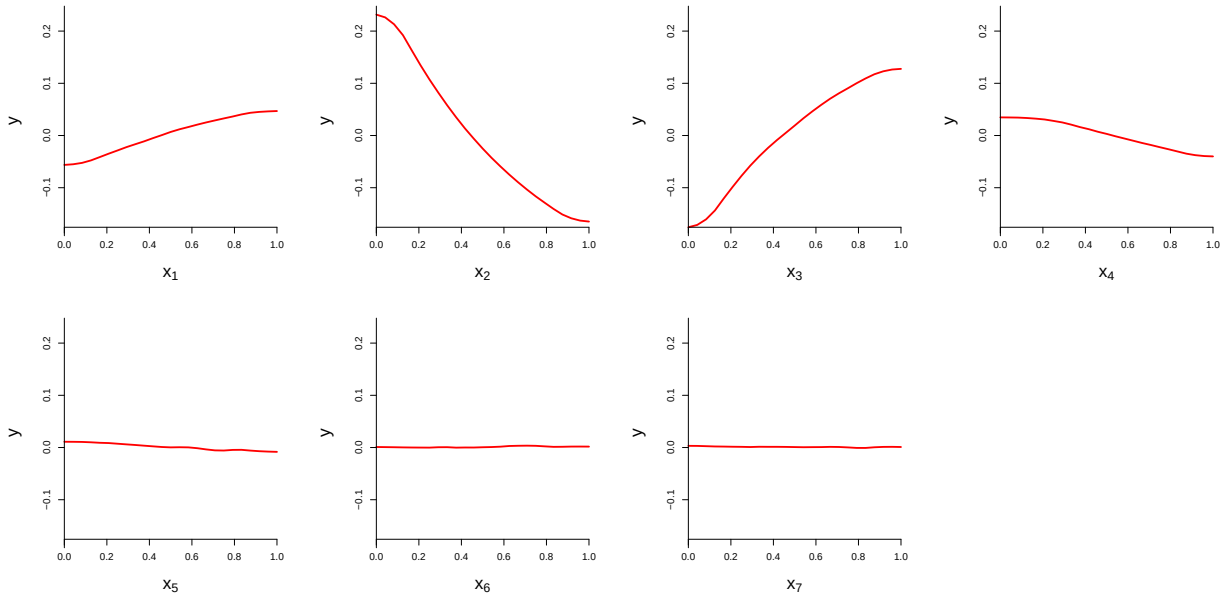


Figure 5: Main effect plots for the piston simulator, provided by the `tgp` **R** package (Gramacy and Taddy 2010), where $x_1, \ldots, x_7$ represent the input variables M, S, V_0, k, P_0, T_a, T_0, respectively.

Since randomness in the response can only be incorporated via randomness in the design, we generated 50 random 5-dimensional Latin hypercube samples of various sample sizes. First, we took a conservative approach to the sample size and chose $n = 120$ instead of $n = 111$ suggested by the inequality (8). In order to compare the $10d$ rule of thumb for both $d = 5$ or $d = 7$, we also considered $n = 50$ and $n = 70$. The corresponding RAUV is shown in column 2 of Table 1.

To measure performance using simulated data, we evaluated the *Empirical RAUV*

$$\text{ERAUV}\left(\widehat{y};\mathcal{D}\right)=\left(\frac{\sum_{i=1}^{n_{\text{ho}}}\left(\widehat{y}_i^{\text{ho}}-y_i^{\text{ho}}\right)^2}{n_{\text{ho}}\widehat{\sigma}^2}\right)^{1/2}$$

at a size $n_{\text{ho}}=100,000$ holdout set in the 7-dimensional space, where the estimate $\widehat{\sigma}^2=0.022$ was obtained by fitting a one time 5 dimensional model to a size 1000 dataset and remained fixed throughout the simulation study.

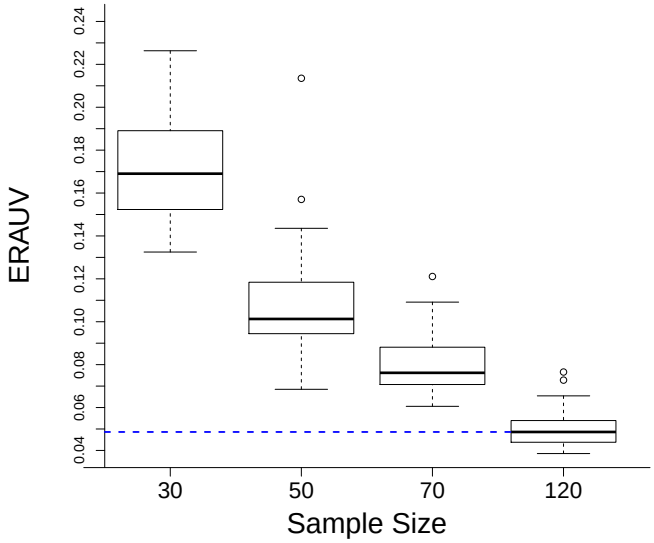


Figure 6: Empirical RAUV values over 50 repetitions for the piston example. Each repetition stands for a randomly chosen Latin hypercube (for each sample size).

Results of the simulation study appear in Table 1 and Figure 6. The variability in the results shows how the randomness of the choice of the design (Latin hypercube in this case) propagates into the model. With an additional design optimality criterion (maximin distance for example) one should expect to see clear separation between the different sample sizes.

The maximum likelihood estimates, $\widehat{\boldsymbol{\theta}}=(1.60,0.30,0.50,0.44,1.99)$, turned out to be very different from our early assessment. It is interesting to note that in spite of that, our proce-

Sample Size	Theoretical RAUV	ERAUV
30	≥ 0.191	$0.172\,(\pm 0.025)$
50	≥ 0.120	$0.107\,(\pm 0.024)$
70	≥ 0.086	$0.080\,(\pm 0.013)$
120	≥ 0.045	$0.050\,(\pm 0.008)$

Table 1: Summary of simulation results for the piston cycle time model. Average ERAUV results appear with $\pm$ one standard deviation.

dure seems to have captured the overall complexity of the model to a good degree, judging by the proximity of the theoretical RAUV values (for the specified parameter values) to the Empirical ones in Table 1. Section 6 discusses a robust procedure that allows the experimenter to provide a range of values for each correlation length parameter, rather than giving a single guess.

Finally, Figure 7 provides a visualization of the improvement of the RAUV (for randomly chosen designs) from 0.167 to 0.106, 0.079 and finally 0.048 as the sample size increases from 30 to 50, 70 and 120, respectively (for a single fit, each). This should give the reader an idea of the accuracy levels that might be attained by choosing different ε values. Indeed, following these steps gives a fair amount of insight beyond just hoping that the $10d$ rule is sufficient.

6 Robust Sample Size Calculations

The task of choosing values for the correlation parameters is challenging, and specifying a range of values may be easier in practice. It is therefore natural to consider incorporating some uncertainty with respect to these chosen values to enhance robustness.

Denote by $g : \mathbb{R}^d \to \mathbb{N}^+$ the function that maps each vector of correlation parameters $\boldsymbol{\theta}$ to a critical sample size n^c through (8). If we now assign a distribution $\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta})$, g will induce a probability measure on n^c. Drawing a random sample $\{\boldsymbol{\theta}_i\}$ from $\pi(\boldsymbol{\theta})$ will then result in

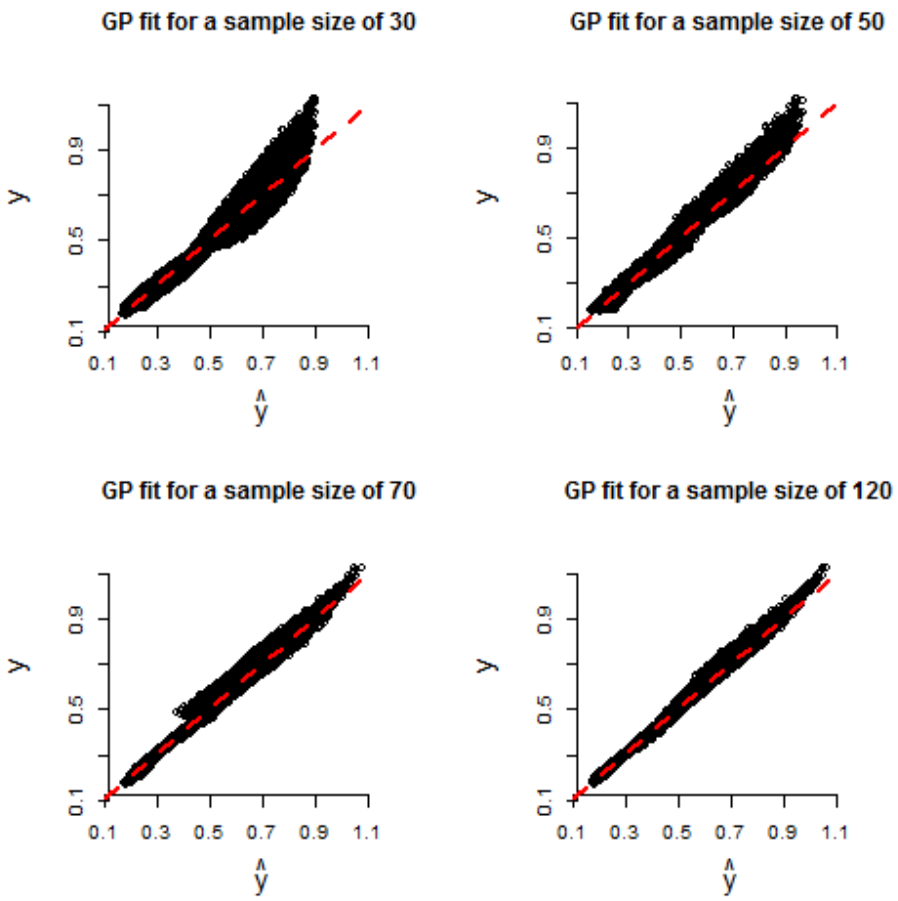


Figure 7: True response vs. fitted plots for the piston example, based on sample sizes of 30, 50, 70 and 120, respectively, and randomly constructed Latin hypercube designs.

a Monte Carlo sample $\{g(\boldsymbol{\theta}_i)\}$ from a distribution $\pi(n^c)$ of sample sizes.

As an example, consider again the piston simulation of Section 5 and assign $\theta_1,\ldots,\theta_5$ independent uniform priors on $[0.8, 1.2]$, $[0.25, 0.55]$, $[0.25, 0.55]$, $[0.8, 1.2]$ and $[2, 4]$, respectively (see the "Moderate uncertainty" scenario in Table 2). We drew a random sample of size $10,000$ from $\pi(\boldsymbol{\theta})$ and produced the corresponding random sample from $\pi(n^c)$ through solving the integral equation (6) and calculating the critical sample size (8) for each drawn vector. Figure 8 shows the histogram for the sample sizes. Given the uncertainty in the response surface specification, a choice must be made in order to run the experiment. One could, of course, choose the maximum sample size from those observed. Doing so would be extreme in our

view (and also would require more random samples to appropriately estimate the maximum sample size). Looking at the plot, a line at $n = 140$, marking the 95th percentile of the sample sizes, is added. We view this as representing a safe choice for a sample size that accounts for uncertainty in $\boldsymbol{\theta}$. Note that although the eventual recommended sample size is somewhat larger than that from Section 5, it is still rather economical compared to the worst case scenario sample of 187.

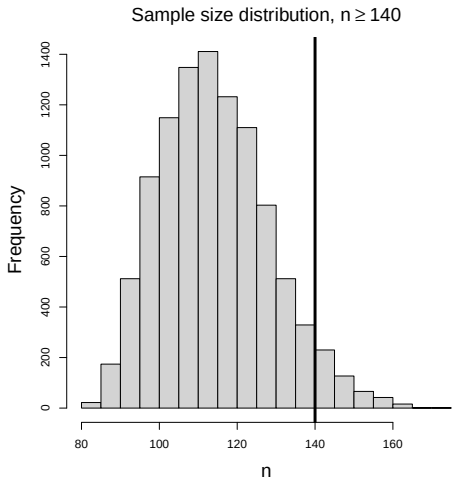


Figure 8: A histogram of a Monte Carlo sample from the sample size distribution $\pi(n^c)$, induced by assigning a distribution $\pi(\boldsymbol{\theta})$. The cutoff at $n \geq 140$ marks the 95th percentile.

The choice of prior distribution for the correlation parameters impacts the sample size. One might be tempted to think that more uncertainty in the complexity requires more samples, but this is not necessarily the case. To study how sample size calculations are impacted by the choice of $\pi(\boldsymbol{\theta})$, we assigned $\theta_1,\ldots,\theta_5$ independent uniform priors on respective intervals about their conjectured values. Table 2 summarizes the results of a small scale simulation that we carried out We considered three scenarios: "Very high uncertainty", "High uncertainty" and "Moderate uncertainty", pertaining to very long, fairly long and medium length intervals, respectively. For each scenario we drew a random sample of size $10,000$ from $\pi(\boldsymbol{\theta})$ and produced the corresponding random sample from $\pi(n^c)$ through solving the integral equation (6) and calculating the critical sample size (8) for each sampled vector.

Very high uncertainty		High uncertainty		Moderate uncertainty	
Range	n (95%)	Range	n (95%)	Range	n (95%)
$\theta_1 : [0.001, 5]$ $\theta_2 : [0.001, 5]$ $\theta_3 : [0.001, 5]$ $\theta_4 : [0.001, 5]$ $\theta_5 : [0.001, 5]$	≥ 157	$\theta_1 : [0.5, 1.5]$ $\theta_2 : [0.1, 0.7]$ $\theta_3 : [0.1, 0.7]$ $\theta_4 : [0.5, 1.5]$ $\theta_5 : [1, 5]$	≥ 214	$\theta_1 : [0.8, 1.2]$ $\theta_2 : [0.25, 0.55]$ $\theta_3 : [0.25, 0.55]$ $\theta_4 : [0.8, 1.2]$ $\theta_5 : [2, 4]$	≥ 140

Table 2: Robust sample size calculations for the piston simulation, based on different levels of uncertainty.

Looking at Table 2, we see that increased uncertainty (i.e., width of the uniform prior distributions) does not automatically translate into increased sample size. The wide intervals in the leftmost column resulted in some inactive dimensions for many of the randomly drawn vectors , and in turn to a smaller sample size than the one recommended for the more focused "High uncertainty" scenario. This is but an illustration of the sensitivity of the chosen sample size to both the width of the uncertainty interval and the location of its center. (The choice of independent uniform distributions was one of convenience, and need not be viewed as a recommendation.)

7 Discussion

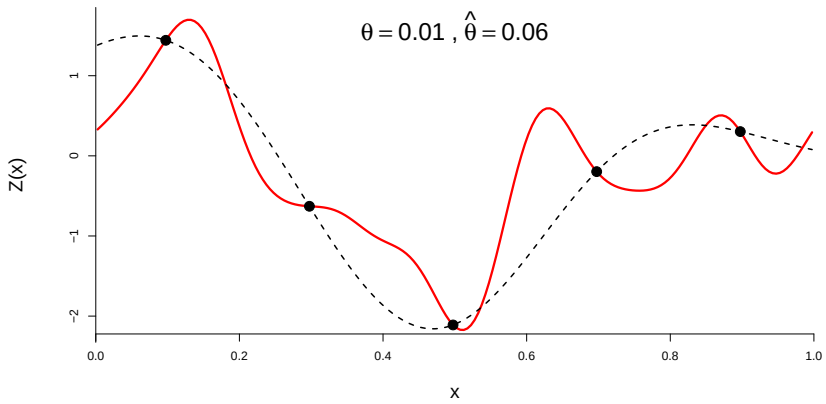


Figure 9: Kriging a realization from a GP based on the squared exponential correlation function with $\theta = 0.01$. Undersampling leads to overestimation of the correlation length.

The problem of choosing the sample size for a computer experiment has so far only partially been addressed in the literature. We identify the lack of a well defined objective as a root cause, and, in this paper, propose the average explained variability or equivalently, the average out of sample correlation between prediction and response as a natural experimental goal.

The machine learning community refers to the IMSPE curve versus n as the "learning curve" (see e.g. Williams and Vivarelli 2000) and, over the years, tighter lower bounds than (7) for the case of noisy data – as well as upper bounds – have been derived (see e.g. Sollich 1999). However, in the interpolating setting we are considering, these bounds do not apply. Thus the bound in (7) is used in this paper. It has been found to be fast to calculate and, as Figure 2 implies, fairly tight in practice.

We would also like to offer a word of caution. As we mentioned in Section 1, Kriging models often flatter to deceive in that, in the case of undersampling, the estimated response will be far smoother than the true one, and using the estimated correlation parameters in (7) can inevitably result in an underestimated sample size in (8). The circular process of basing sample size calculations on estimated parameters should therefore be avoided, unless a validation set is at hand to assess goodness of fit. An example illustrating this is given in Figure 9, where a realization from a GP based on the squared exponential correlation function (9) with correlation length $\theta = 0.01$ is drawn. The maximum likelihood estimate based on a size 5 sample turns out to be $\widehat{\theta} = 0.06$. Consequently, for a target RAUV of 0.05, our calculations would yield a required sample size of $n \geq 7$, as opposed to the sample size of $n \geq 15$ that using the true value of θ would yield.

As a possible topic for future research, one may be interested in establishing an explicit expression governing the trade-off between the correlation length $\boldsymbol{\theta}$ (equivalently, function smoothness) and the required sample size for a given ε. We performed a simulation study

for the isotropic 4 dimensional product Matérn kernel with smoothness parameter $\nu = 2.5$ by calculating n^c for $\theta = 0.05, 0.1, 0.15, \ldots, 2.5$ and $\varepsilon = 0.05$. Figure 10 shows the results in the log-log scale, along with the fitted linear regression line. The estimated slope is very close to -3, suggesting the critical sample size decays at a rate of $\mathcal{O}(\theta^{-3})$ for this example.

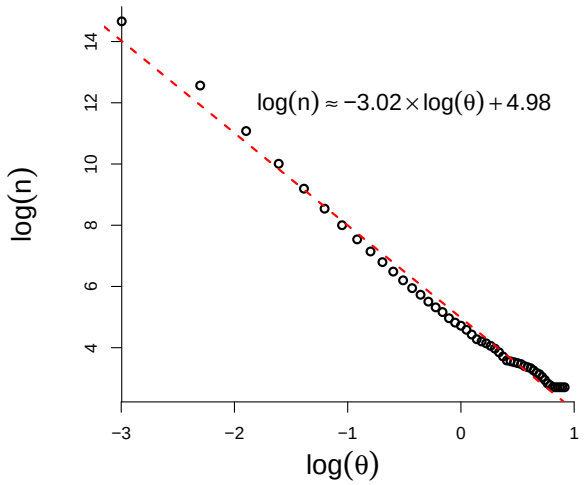


Figure 10: The minimum sample size (8) required to achieve RAUV ≤ 0.05 vs. correlation length for a 4 dimensional isotropic Gaussian process based on the product Matérn correlation function with smoothness $\nu = 2.5$.

We have attempted to provided the groundwork for careful consideration of the design and analysis of a computer experiment. However, we are in agreement with Loeppky et al. (2009) in noting that in high dimensions, the number of samples may be onerous. As an illustration, we considered a setting with $d = 26$ inputs and varied the number of active factors from 1 to 26. For each number of active inputs, and sample sizes of $n = 50$, 100, 200, 400 and 500, we generated 50 realizations of a Gaussian process with the product power-exponential covariance and $\theta = 1$ for the active factors, using a randomly drawn Latin hypercube design along with a hold-out set of 100 randomly chosen trials. The Kriging model, with the product power-exponential covariance, was fit to each simulated dataset and the predictive

performance was evaluated on the corresponding hold-out set using the empirical value of

$$\int_{[0,1]^d} \frac{\mathbb{V}\mathrm{ar}\left\{y\left(\boldsymbol{x}\right)\left|\mathcal{D}\right.\right\}}{\mathbb{V}\mathrm{ar}\left\{y\left(\boldsymbol{x}\right)\right\}}\mathrm{d}\boldsymbol{x}.$$

That is, we take the ratio of the empirical mean-square predictive error for the hold-out sets, for each combination of the number of active factors and sample size, and the variance of the hold-out set. The results are plotted in Figure 11a. A quick glance at the figure reveals some interesting results. First, for a fixed sample size, the average unexplained variability grows fairly rapidly as the number of active factors increases. Indeed, even when $n = 500$, we see that, when the number of active factors is about 15, the Gaussian process emulator has a difficult time predicting the response surface accurately. In reality, one needs many samples to adequately fill, say, a 20-dimensional hypercube. Overall, the more complex the response surface and the more active inputs influencing the response, the larger the sample size required.

Interestingly, we repeated the same procedure, but with a simpler data generating model. Instead of using the product power-exponential covariance model, we generated data for the same scenarios using a sum of 1-d independent Gaussian processes (one for each active dimension) with $\theta = 1$ for the active factors. However, the data analysis was performed using the same Kriging model as before with the product power-exponential covariance. The results are summarized in Figure 11b. Looking at the plot, we see that, for the same sample sizes and numbers of active dimensions, the standard GP explains almost all of the variability on average, except for the relatively small setting of $n = 50$. The take-away message here is that, when the model is simple, the Gaussian process does an admirable job at computer model emulation.

The second case that we considered is unrealistically simple. However, our belief is that

the complexity of many computer models lies somewhere between these two extremes, and thus the methodology proposed in this paper is a conservative approach. The illustrations point to a need for future work where one specifies a class of simpler random functions that represent the space in which the computer model response surface may lie for both design and analysis.

Supplementary Materials

A web application that performs all the analyzes presented in this paper at a click is available at https://harario.shinyapps.io/Sample_Size_Shiny. In addition, **R** code for the study of Section 5 is available online as supplementary material.

Acknowledgments

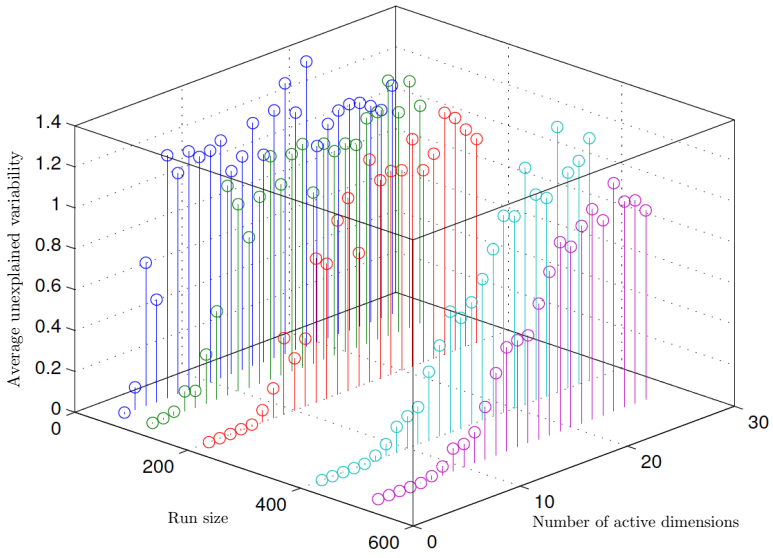
This work was supported by grants from the Natural Sciences and Engineering Research Council of Canada and the Canadian Statistical Sciences Institute. The authors would like to thank the Referees for their insightful comments.

References

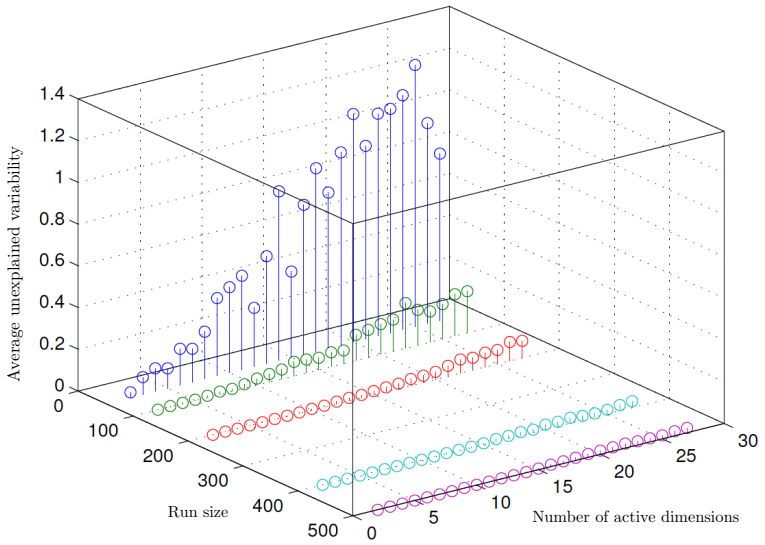
Box, G. and Meyer, R. (1986), "An analysis for unreplicated fractional factorials," *Technometrics*, 28, 11–18.

Chapman, W. L., Welch, W. J., Bowman, K. P., Sacks, J., and Walsh, J. E. (1994), "Arctic Sea Ice Variability: Model Sensitivities and a Multidecadal Simulation," *Journal of Geophysical Research: Oceans*, 99, 919–935.

Chen, H., Loeppky, J. L., Sacks, J., and Welch, W. J. (2016), "Analysis Methods for Computer Experiments: How to Assess and What Counts?" *Statistical Science*, to appear.



(a)



(b)

Figure 11: Empirical average unexplained variability values for data generated by multiplicative (11a) and additive (11b) univariate Gaussian processes vs. the number of active inputs and sample size.

Diniz, P., Netto, S., and Silva, E. D. (2002), *Digital Signal Processing: System Analysis and Design*, Cambridge University Press.

Gramacy, R. B. and Taddy, M. (2010), "Categorical Inputs, Sensitivity Analysis, Optimization and Importance Tempering with tgp Version 2, an R Package for Treed Gaussian Process Models," *Journal of Statistical Software*, 33, 1–48.

Harari, O. and Steinberg, D. (2014), "Optimal Designs for Gaussian Process Models via Spectral Decomposition," *Journal of Statistical Planning and Inference*, 154, 87–101.

Johnson, M., Moore, L., and Ylvisaker, D. (1990), "Minimax and Maximin Distance Designs," *Journal of Statistical Planning and Inference*, 26, 131 – 148.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998), "Efficient Global Optimization of Expensive Black-Box Functions," *Journal of Global Optimization*, 13, 455–492.

Joseph, V., Dasgupta, T., and Wu, C. (2012), "Minimum energy designs: from nanostructure synthesis to sequential optimization," *under revision*.

Kaufman, C. G., Bingham, D., Habib, S., Heitmann, K., and Frieman, J. A. (2011), "Efficient Emulators of Computer Experiments Using Compactly Supported Correlation Functions, with an Application to Cosmology," *The Annals of Applied Statistics*, 5, 2470–2492.

Kenett, R. S. and Zacks, S. (1998), *Modern Industrial Statistics: Design and Control of Quality and Reliability*, Pacific Grove: Duxbury Press.

Loeppky, J., Sacks, J., and Welch, W. (2009), "Choosing the Sample Size of a Computer Experiment: A Practical Guide," *Technometrics*, 51, 366–376.

Micchelli, C. and Wahba, G. (1981), "Design Problems for Optimal Surface Interpolation," in *Approximation Theory and Applications*, ed. Ziegler, Z., New York: Academic Press, pp. 329–347.

Ranjan, P., Haynes, R., and Karsten, R. (2011), "A Computationally Stable Approach to Gaussian Process Interpolation of Deterministic Computer Simulation Data," *Technometrics*, 53, 366–378.

Sacks, J., Schiller, S., and Welch, W. (1989a), "Designs for Computer Experiments," *Technometrics*, 31, 41–47.

Sacks, J., Welch, W., Mitchell, T., and Wynn, H. (1989b), "Design and Analysis of Computer Experiments," *Statistical Science*, 4, 409–423.

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Trantola, S. (2008), *Global Sensitivity Analysis: the Primer*, John Wiley.

Santner, T., Williams, B., and Notz, W. (2003), *The Design and Analysis of Computer Experiments*, Springer.

Shewry, M. and Wynn, H. (1987), "Maximum Entropy Sampling," *Journal of Applied Statistics*, 14, 165–170.

Sollich, P. (1999), "Learning Curves for Gaussian Processes," in *Advances in Neural Information Processing Systems 11*, eds. Kearns, M., Solla, S., and Cohn, D., MIT Press, pp. 344–350.

Williams, C. K. and Vivarelli, F. (2000), "Upper and Lower Bounds on the Learning Curve for Gaussian Processes," *Machine Learning*, 40, 77–102.

Yi, T.-m., Fazel, M., Liu, X., Otitoju, T., Goncalves, J., Papachristodoulou, A., Prajna, S., and Doyle, J. (2005), "Application of Robust Model Validation Using SOSTOOLS to the Study of G-protein Signaling in Yeast," in *Proceedings of the First Conference on Foundations of Systems Biology in Engineering*, Santa Barbara, CA.