Monograph

Author: Ksenia N. Kyzyurova

Analysis of scientific computer models

Methodology in numerical simulator data analysis

September 21, 2019

This monograph provides methodological foundation for analysis of data from scientific computationally challenging mathematical models. Mathematical topics include statistical modeling, Bayesian inference, stochastic processes, and decision-theoretic model assessment.

Prerequisites

Science attempts to describe complex natural or engineering phenomena by use of mathematical processes. These computer models are usually either computationally slow and/or too resources demanding for operation of the model and sometimes even for storage of the data produced by the model. This makes the computer model output at any *desirable input not available. However, fast approximations to such an output may be obtained, once an* emulator *of the model (its* statistical approximation*), using only a handful of input-output data points, is constructed. Construction of the 'default' emulator within its objective implementation is outlined below.*

Statistical approximation to a computer model

Suppose that a computer model is represented by a smooth function $f(\cdot)$, which takes a d-dimensional input $\boldsymbol{\zeta} \in Z \subseteq \mathcal{R}^d, d \geq 1$ and produces a d^*-dimensional output $\mathbf{f}(\boldsymbol{\zeta}) \subseteq R^{d^*}, d^* \geq 1$. Both d and d^* are positive integers. In the first two chapters of the present monograph, computer models with one-dimensional output $d^* = 1$ are considered. The problem of construction of statistical approximations to computer models with multivariate output is considered in chapter .

Suppose m inputs to the simulator $\boldsymbol{\zeta} = (\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, \ldots, \boldsymbol{\zeta}_m)$ are chosen, and the computer model f has been evaluated at those inputs, resulting in outputs $\boldsymbol{\Xi}(\boldsymbol{\zeta}) = (f(\boldsymbol{\zeta}_1), f(\boldsymbol{\zeta}_2), \ldots, f(\boldsymbol{\zeta}_m))$. Using these pairs of inputs and outputs $\{\boldsymbol{\zeta}_i, f(\boldsymbol{\zeta}_i)\}_{i=1}^m$ and assuming a Gaussian process prior $\Xi(\cdot) = \mathcal{GP}(\mu(\cdot), \mathrm{cov}(\cdot,\cdot))$ (specified by some mean and covariance function) on the computer model output $f(\cdot)$, we aim at constructing a probabilistic representation of the model at any new input in its domain Z.

A Gaussian process is defined as a stochastic process, whose every finite sample of variables follows (all compatible) multivariate normal distribution. Since any multivariate normal distribution is specified by its mean and covariance matrix, a Gaussian process may be specified by the mean $\mathrm{E}\boldsymbol{\Xi}(\boldsymbol{\zeta})$ and covariance $\mathrm{Cov}(\Xi(\boldsymbol{\zeta}_k), \Xi(\boldsymbol{\zeta}_\ell))$ for any two inputs $\boldsymbol{\zeta}_k$ and $\boldsymbol{\zeta}_\ell$. For infinite number of random variables, the mean and covariance are convenient to specify parametrically (to satisfy the compatibility requirements). For example, customarily is to specify the mean of the Gaussian process at point $\boldsymbol{\zeta}$ as *linear* with two parameters, intercept β_0 and slope β_1

$$\mathrm{E}\boldsymbol{\Xi}(\boldsymbol{\zeta}) = \beta_0 + \beta_1 \zeta_e \,, \tag{1}$$

where $e \in 1, 2, \ldots, d$ is one of the input dimensions.

Equivalently, one may write $\mathrm{E}\boldsymbol{\Xi}(\boldsymbol{\zeta}) = \mathbf{H}\boldsymbol{\beta} = \begin{pmatrix} 1 & \zeta_e \end{pmatrix} \boldsymbol{\beta}$, where $\boldsymbol{\beta} = \begin{pmatrix} \beta_0 & \beta_1 \end{pmatrix}^{\mathrm{T}}$ is a vector of linear model regression coefficients. $\mathbf{H}^{\mathrm{T}}$ is a vector of regression functions evaluated at $\boldsymbol{\zeta}$. In this case, two regression functions are the constant $\psi_0(\boldsymbol{\zeta}) = 1$ and $\psi_1(\boldsymbol{\zeta}) = \zeta_e$. These are the first and second element of the vector $\mathbf{H}^{\mathrm{T}}$.

In theoretical development easy to extend the mean specification to any number of regression functions. If the number of regression functions is s, then one must specify each of the functions $h_0(\cdot), h_1(\cdot), \ldots, h_{s-1}(\cdot)$; then $\mathbf{H} = \begin{pmatrix} h_0(\boldsymbol{\zeta}) & h_1(\boldsymbol{\zeta}) & \ldots & h_{s-1}(\boldsymbol{\zeta}) \end{pmatrix}$.

Specification of covariance for infinite number of random variables is a more complicated matter, since it must be *valid*. Typically covariance between outputs of the computer model at corresponding inputs is specified as the product of variance parameter σ^2 and correlation function $c(\cdot,\cdot)$ evaluated at those inputs, i.e. for any two inputs $\boldsymbol{\zeta}_k$ and $\boldsymbol{\zeta}_\ell$

$$\mathrm{Cov}(\Xi(\boldsymbol{\zeta}_k), \Xi(\boldsymbol{\zeta}_\ell)) = \sigma^2 c(\boldsymbol{\zeta}_k, \boldsymbol{\zeta}_\ell) \,. \tag{2}$$

If an input to the model is one-dimensional, i.e. $d = 1$, a convenient and useful choice for the purpose of emulation of a computer model is squared-exponential correlation function, which is specified by the parameter ω

$$c(\zeta_k, \zeta_\ell) = \exp\left\{ -\exp(\omega) (\zeta_k - \zeta_\ell)^2 \right\}. \tag{3}$$

If an input to the model is multi-dimensional with $d \geq 2$, then convenient is to specify correlation function in the form of a product of correlation functions along each of the d input dimensions, that is $c(\boldsymbol{\zeta}_k, \boldsymbol{\zeta}_\ell) = \prod_{j=1}^d c(\zeta_{kj}, \zeta_{\ell j})$.

Lemma. The product of correlation functions is a valid correlation function.

Parameters of the mean and covariance form a vector of parameters $\boldsymbol{\theta} = (\boldsymbol{\beta}, \sigma^2, \omega)$ which altogether specify the Gaussian process. While β_0 and β_1 define a linear mean trend, parameter σ^2 is responsible for variability of the process, and parameter ω determines the range scale at which correlations among Gaussian process random variables remain strong.

Function $f(\zeta) = \exp(-\zeta) + \sin(4\zeta)$ in the range $\zeta \in [-1, 1]$ is taken as a prototype of a computer model. Suppose this model has been evaluated at eight input points $\boldsymbol{\zeta} = (\zeta_1, \zeta_2, \ldots, \zeta_8)$ which are equadistantly placed between -0.94 and 0.94 resulting in outputs $\mathbf{f}(\boldsymbol{\zeta}) = (f(\zeta_1), f(\zeta_2), \ldots, f(\zeta_8))$ respectively.

Suppose for now that the parameters of the Gaussian process $\boldsymbol{\theta}$ are known and given. Let these parameters be $\beta_0 = 1.5$, $\beta_1 = 2$, $\sigma^2 = 2$ and $\omega = 2$. Once these are set, the construction of the Gaussian process emulator with parameters of the process being given is possible: assuming that this *training data*[1] $\{\boldsymbol{\zeta}, \mathbf{f}(\boldsymbol{\zeta})\}$ belong to its to-be constructed approximation Ξ, such that $\boldsymbol{\Xi}(\boldsymbol{\zeta}) = \mathbf{f}(\boldsymbol{\zeta})$ and that Ξ has the Gaussian process prior, the model for the data is given by its likelihood

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}) \sim \mathcal{N}(\mathbf{H}\boldsymbol{\beta}, \sigma^2 \mathbf{C}_\omega)\,, \tag{4}$$

where $\mathbf{C}_\omega$ is the correlation matrix given by a correlation function $c(\cdot, \cdot)$ such that the element of the matrix at row ι and column ι' equals to $c(\boldsymbol{\zeta}_\iota, \boldsymbol{\zeta}_{\iota'})$.

Let $\boldsymbol{\zeta}'$ be a vector of input points at which the approximated solution is desired to be found. The joint distribution for inputs $\boldsymbol{\zeta}$ and $\boldsymbol{\zeta}'$ is multivariate normal

$$\begin{pmatrix} \boldsymbol{\Xi}(\boldsymbol{\zeta}) \\ \boldsymbol{\Xi}(\boldsymbol{\zeta}') \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mathbf{H}\boldsymbol{\beta} \\ \mathbf{H}'\boldsymbol{\beta} \end{pmatrix}, \sigma^2 \begin{pmatrix} \mathbf{C}_\omega & c(\boldsymbol{\zeta}, \boldsymbol{\zeta}')_\omega \\ c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega & \mathbf{C}'_\omega \end{pmatrix} \right)\,. \tag{5}$$

The approximation Ξ at points $\boldsymbol{\zeta}'$ conditional on the training data $\{\boldsymbol{\zeta}, \mathbf{f}(\boldsymbol{\zeta})\}$ is given by the predictive multivariate normal distribution

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \boldsymbol{\theta} \sim \mathcal{N}\left(\mathbf{H}'\boldsymbol{\beta} + c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega \mathbf{C}_\omega^{-1}(\boldsymbol{\Xi}(\boldsymbol{\zeta}) - \mathbf{H}\boldsymbol{\beta}), \sigma^2(\mathbf{C}'_\omega - c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega \mathbf{C}_\omega^{-1} c(\boldsymbol{\zeta}, \boldsymbol{\zeta}')_\omega)\right)\,. \tag{6}$$

This predictive distribution **defines** the statistical *emulator*, approximation to a computer model f at inputs $\boldsymbol{\zeta}'$. Figure 1 demonstrates the approximation constructed for the simulator f.[a]

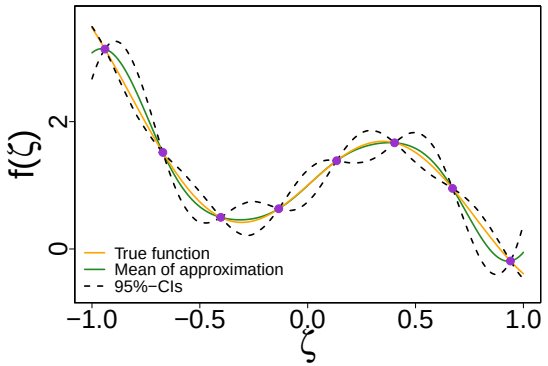

Figure 1: Constructed statistical approximation to the computer model f. The orange curve is the true function; the green — the mean of the emulator; the black dashed lines are those of the central 95%-credible area. The purple points are design points.

Theorem. $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \boldsymbol{\theta}$ is the stochastic process.

Proof. This is indeed so by construction. All joint disributions of any sequence of random variables are defined and are compatible. ■

Definition. $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \boldsymbol{\theta}$ is called the Gaussian process stochastic emulator (or approximation) to a computer model f.

$$\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \boldsymbol{\theta} \sim \mathcal{GASP}(\mu^*(\cdot), \sigma^{*2}(\cdot, \cdot))\,, \tag{7}$$

where GASP stands for Gaussian stochastic process, $\mu^*(\cdot)$ is the mean of the predictive process and $\sigma^{*2}(\cdot, \cdot)$ is its variance-covariance, which for any vector of inputs follow the distribution 6.

[a] In literature words "emulator", "surrogate", "metamodel" are used interchangeably meaning "a probabilistic statistical approximation to a computer model".

Within objective implementation, suppose that parameter ω is fixed, then the prior for the rest of parameters $\boldsymbol{\beta}$ and σ^2 for this normal model (4) is

$$p(\boldsymbol{\beta}, \sigma^2) \propto \frac{1}{\sigma^2}\,. \tag{8}$$

The likelihood (4) for the model parameters $L(\omega, \boldsymbol{\beta}, \sigma^2; \boldsymbol{\Xi}(\boldsymbol{\zeta})) = p(\boldsymbol{\Xi}(\boldsymbol{\zeta}) \mid \omega, \boldsymbol{\beta}, \sigma^2)$, which is, while omitting all expression which do not contain parameters $\boldsymbol{\beta}, \sigma^2, \omega$, is proportional to

$$L(\omega, \boldsymbol{\beta}, \sigma^2; \boldsymbol{\Xi}(\boldsymbol{\zeta})) \propto |\sigma^2 \mathbf{C}_\omega|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\boldsymbol{\Xi}(\boldsymbol{\zeta}) - \mathbf{H}\boldsymbol{\beta})^{\mathrm{T}} \frac{1}{\sigma^2} \mathbf{C}_\omega^{-1} (\boldsymbol{\Xi}(\boldsymbol{\zeta}) - \mathbf{H}\boldsymbol{\beta})\right\} = (\sigma^2)^{-\frac{m}{2}} |\mathbf{C}_\omega|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\boldsymbol{\Xi}(\boldsymbol{\zeta}) - \mathbf{H}\boldsymbol{\beta})^{\mathrm{T}} \frac{1}{\sigma^2} \mathbf{C}_\omega^{-1} (\boldsymbol{\Xi}(\boldsymbol{\zeta}) - \mathbf{H}\boldsymbol{\beta})\right\}. \tag{9}$$

[1] Also called *design* data points.

Posterior distributions of parameters. Joint posterior distribution of parameters $\boldsymbol{\beta}$ and σ^2 is convenient to write in conditional form

$$p(\boldsymbol{\beta}, \sigma^2 \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega) = p(\boldsymbol{\beta} \mid \sigma^2, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega) p(\sigma^2 \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega)\,. \tag{10}$$

These distributions are

$$\boldsymbol{\beta} \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2, \omega \sim \mathcal{N}_{\beta}\left((\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\right)\,, \tag{11}$$

$$\sigma^2 \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega \sim \mathcal{IG}_{\sigma^2}\left(\frac{m-s}{2}, \frac{\boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}}(\mathbf{C}_{\omega}^{-1} - \mathbf{C}_{\omega}^{-1}\mathbf{H}(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1})\boldsymbol{\Xi}(\boldsymbol{\zeta})}{2}\right)\,. \tag{12}$$

Instead of using predefined *ad hoc* values of $\boldsymbol{\beta}$ and σ^2, the maximum *a posteriori* estimates obtained *from the data* are readily available from the joint posterior. Modes of posterior distributions 11 and 12 are

$$\hat{\boldsymbol{\beta}} = (\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\boldsymbol{\Xi}(\boldsymbol{\zeta})\,, \tag{13}$$

$$\hat{\sigma^2} = \frac{\boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}}(\mathbf{C}_{\omega}^{-1} - \mathbf{C}_{\omega}^{-1}\mathbf{H}(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1})\boldsymbol{\Xi}(\boldsymbol{\zeta})}{m-s+2}\,. \tag{14}$$

Point estimates of these modes if the training data $\{\boldsymbol{\zeta}, \boldsymbol{\Xi}(\boldsymbol{\zeta})\}$ is plugged in are

$$\hat{\boldsymbol{\beta}} = \begin{pmatrix} 1.8344 \\ -0.1047 \end{pmatrix},\ \hat{\sigma^2} = 11.906\,. \tag{15}$$

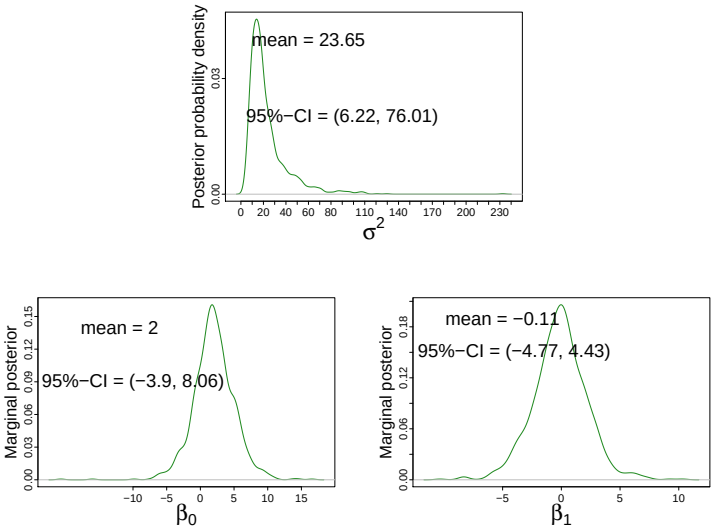

Figure 2: Plots of marginal posterior distributions for parameters σ^2, β_0 and β_1 of the Gaussian process.

As with respect to parameter ω, in this theoretical exposition, the maximum likelihood estimate of this parameter is used, while parameters $\boldsymbol{\beta}$ and σ^2 being integrated out. (In practice, the objective implementation considered in [1] and [2] is preferable over the MLE estimate.) That is, the likelihood function $L(\omega; \boldsymbol{\Xi}(\boldsymbol{\zeta}))$ is

$$L(\omega; \boldsymbol{\Xi}(\boldsymbol{\zeta})) = p(\boldsymbol{\Xi}(\boldsymbol{\zeta}) \mid \omega) = \int\int p(\boldsymbol{\Xi}(\boldsymbol{\zeta}) \mid \omega, \boldsymbol{\beta}, \sigma^2) p(\boldsymbol{\beta}, \sigma^2) \mathrm{d}\boldsymbol{\beta}\mathrm{d}\sigma^2 \tag{16}$$

$$\propto \int\int (\sigma^2)^{-\frac{m}{2}-1} |\mathbf{C}_{\omega}|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}}\frac{1}{\sigma^2}\mathbf{C}_{\omega}^{-1}\boldsymbol{\Xi}(\boldsymbol{\zeta})\right\} \exp\left\{-\frac{1}{2\sigma^2}\boldsymbol{\beta}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H}\boldsymbol{\beta}\right\} \exp\left\{-\frac{1}{2\sigma^2}\left(-2\boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H}(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})\boldsymbol{\beta}\right)\right\} \mathrm{d}\boldsymbol{\beta}\mathrm{d}\sigma^2 = \tag{17}$$

$$= \int\int (\sigma^2)^{-\frac{m}{2}-1} |\mathbf{C}_{\omega}|^{-\frac{1}{2}} \mid \sigma^2(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}|^{\frac{1}{2}} \exp\left\{-\frac{1}{2\sigma^2}\boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}}(\mathbf{C}_{\omega}^{-1} - \mathbf{C}_{\omega}^{-1}\mathbf{H}(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1})\boldsymbol{\Xi}(\boldsymbol{\zeta})\right\} \mathcal{N}_{\beta}\left((\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\right) \mathrm{d}\boldsymbol{\beta}\mathrm{d}\sigma^2$$

$$\propto \int\int |\mathbf{C}_{\omega}|^{-\frac{1}{2}} |\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H}|^{-\frac{1}{2}} \mathcal{N}_{\beta}\left((\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\right) \mathcal{IG}_{\sigma^2}\left(\frac{m-s}{2}, \frac{\boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}}(\mathbf{C}_{\omega}^{-1} - \mathbf{C}_{\omega}^{-1}\mathbf{H}(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1})\boldsymbol{\Xi}(\boldsymbol{\zeta})}{2}\right) \mathrm{d}\boldsymbol{\beta}\mathrm{d}\sigma^2$$

$$\propto (\boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}}(\mathbf{C}_{\omega}^{-1} - \mathbf{C}_{\omega}^{-1}\mathbf{H}(\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1})\boldsymbol{\Xi}(\boldsymbol{\zeta}))^{-\frac{m-s}{2}} |\mathbf{C}_{\omega}|^{-\frac{1}{2}} |\mathbf{H}^{\mathrm{T}}\mathbf{C}_{\omega}^{-1}\mathbf{H}|^{-\frac{1}{2}}\,,$$

where s is the number of parameters in vector $\boldsymbol{\beta}$.

The corresponding log-likelihood function is

$$\ell(\omega; \boldsymbol{\Xi}(\boldsymbol{\zeta})) = \log(L(\omega; \boldsymbol{\Xi}(\boldsymbol{\zeta}))) = \log(p(\boldsymbol{\Xi}(\boldsymbol{\zeta}) \mid \omega)) = \text{const} - \frac{m-s}{2}\log(\boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}}(\mathbf{C}_\omega^{-1} - \mathbf{C}_\omega^{-1}\mathbf{H}(\mathbf{H}^{\mathrm{T}}\mathbf{C}_\omega^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_\omega^{-1})\boldsymbol{\Xi}(\boldsymbol{\zeta})) - \frac{1}{2}\log(\det(\mathbf{C}_\omega)) - \frac{1}{2}\log(\det(\mathbf{H}^{\mathrm{T}}\mathbf{C}_\omega^{-1}\mathbf{H}))\,. \quad (18)$$

The plot of the log-likelihood is given in Figure 3. The emulator then can be constructed using the *estimated* parameters $\hat{\boldsymbol{\beta}}, \hat{\sigma^2}, \hat{\omega}$.

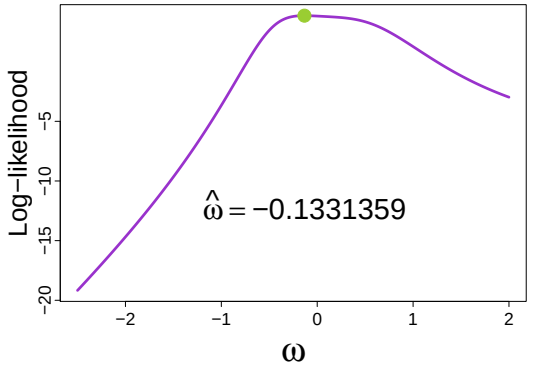

Figure 3: Log-likelihood $\ell(\omega; \boldsymbol{\Xi}(\boldsymbol{\zeta}))$. $\hat{\omega}$ is its maximum.

Accounting for uncertainty in parameters. Incorporation of *uncertainty* in parameters is **defined** as marginalizing over (or integrating out) the parameters. The attempt is to integrate out as many parameters as possible from the corresponding approximation 6. The reason for doing this is two-fold: (a) to formally account for the uncertainty in parameters expressed in their posterior distributions of these, and (b) to reduce the number of parameters which can not be integrated out and, therefore, must be estimated as, for instance, maximum likelihood or maximum *a posteriori*.

Marginalizing over $\boldsymbol{\beta}$, the predictive distribution, emulator, becomes:

$$p(\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2, \omega) = \int p(\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \boldsymbol{\beta}, \sigma^2, \omega) p(\boldsymbol{\beta} \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2, \omega) \mathrm{d}\boldsymbol{\beta}\,.$$

$\boldsymbol{\beta}$ is integrated out following the lemma from [5]. That is, rewrite

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \boldsymbol{\beta}, \sigma^2, \omega = c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega \mathbf{C}_\omega^{-1}\boldsymbol{\Xi}(\boldsymbol{\zeta}) + (\mathbf{H}' - c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega \mathbf{C}_\omega^{-1}\mathbf{H})\boldsymbol{\beta} + \mathcal{N}\left(\vec{\mathbf{0}}, \sigma^2(\mathbf{C}'_\omega - c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega \mathbf{C}_\omega^{-1} c(\boldsymbol{\zeta}, \boldsymbol{\zeta}')_\omega)\right) \quad (19)$$

$$\boldsymbol{\beta} \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2, \omega = (\mathbf{H}^{\mathrm{T}}\mathbf{C}_\omega^{-1}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{C}_\omega^{-1}\boldsymbol{\Xi}(\boldsymbol{\zeta}) + \mathcal{N}(\vec{\mathbf{0}}, \sigma^2(\mathbf{H}^{\mathrm{T}}\mathbf{C}_\omega^{-1}\mathbf{H})^{-1}) \quad (20)$$

Summing everything up, we get the following predictive distribution with $\boldsymbol{\beta}$ marginalized over

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2, \omega \sim \mathcal{N}\left(\tilde{\boldsymbol{\mu}}, \sigma^2\tilde{\boldsymbol{\Sigma}}\right)\,, \quad (21)$$

where

$$\tilde{\boldsymbol{\mu}} = \mathbf{H}'\hat{\boldsymbol{\beta}} + c(\boldsymbol{\zeta}', \boldsymbol{\zeta})\mathbf{C}_\omega^{-1}(\boldsymbol{\Xi}(\boldsymbol{\zeta}) - \mathbf{H}\hat{\boldsymbol{\beta}})\,, \quad (22)$$

$$\tilde{\boldsymbol{\Sigma}} = \mathbf{C}'_\omega - c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega \mathbf{C}_\omega^{-1} c(\boldsymbol{\zeta}, \boldsymbol{\zeta}')_\omega + (\mathbf{H}' - c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega \mathbf{C}_\omega^{-1}\mathbf{H})(\mathbf{H}^{\mathrm{T}}\mathbf{C}_\omega^{-1}\mathbf{H})^{-1}(\mathbf{H}' - c(\boldsymbol{\zeta}', \boldsymbol{\zeta})_\omega \mathbf{C}_\omega^{-1}\mathbf{H})^{\mathrm{T}}\,. \quad (23)$$

By plugging-in estimates of parameters σ^2 and ω we acquire a new emulator — the one which has the uncertainty from parameters $\boldsymbol{\beta}$ incorporated. The expression for $\tilde{\boldsymbol{\mu}}$ contains $\hat{\boldsymbol{\beta}}$ — the mode (and the mean) of the posterior distribution of $\boldsymbol{\beta}$ given in equation (11). While we may easily plug in the estimate $\hat{\boldsymbol{\beta}}$, this is left so for the convenience and clearance in the expression for $\tilde{\boldsymbol{\mu}}$; and to highlight that the mean of the (21) coincides exactly with that of the emulator for which posterior modes of parameters $\boldsymbol{\beta}$ and σ^2 are used.

Theorem. $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2, \omega$ is the stochastic process.
Corollary. $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2, \omega$ is the Gaussian stochastic process emulator.

Accounting for variability in σ^2, the following predictive distribution appears

$$p(\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega) = \int p(\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \sigma^2, \omega) p(\sigma^2 \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega) \mathrm{d}\sigma^2 = \int \mathcal{N}_{\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \sigma^2, \omega}(\tilde{\boldsymbol{\mu}}, \sigma^2 \tilde{\boldsymbol{\Sigma}}) \mathcal{IG}_{\sigma^2 \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega}(\tilde{\alpha}/2, \tilde{\beta}/2) \mathrm{d}\sigma^2 \,, \tag{24}$$

where $\tilde{\alpha} = m - s$ and $\tilde{\beta} = \boldsymbol{\Xi}(\boldsymbol{\zeta})^{\mathrm{T}} (\mathbf{C}_\omega^{-1} - \mathbf{C}_\omega^{-1} \mathbf{H} (\mathbf{H}^{\mathrm{T}} \mathbf{C}_\omega^{-1} \mathbf{H})^{-1} \mathbf{H}^{\mathrm{T}} \mathbf{C}_\omega^{-1}) \boldsymbol{\Xi}(\boldsymbol{\zeta})$.

$$p(\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega) \propto \int \exp\left(-\frac{(\boldsymbol{\Xi}(\boldsymbol{\zeta}') - \tilde{\boldsymbol{\mu}})^{\mathrm{T}} \tilde{\boldsymbol{\Sigma}}^{-1} (\boldsymbol{\Xi}(\boldsymbol{\zeta}') - \tilde{\boldsymbol{\mu}})}{2\sigma^2} - \frac{\tilde{\beta}}{2\sigma^2} \right) (\sigma^2)^{-\frac{m}{2} - \frac{\tilde{\alpha}}{2} - 1} \mathrm{d}\sigma^2 \propto \left(\tilde{\beta} + (\boldsymbol{\Xi}(\boldsymbol{\zeta}') - \tilde{\boldsymbol{\mu}})^{\mathrm{T}} \tilde{\boldsymbol{\Sigma}}^{-1} (\boldsymbol{\Xi}(\boldsymbol{\zeta}') - \tilde{\boldsymbol{\mu}}) \right)^{-\frac{m+\tilde{\alpha}}{2}}$$

$$\propto \left(1 + \frac{1}{\tilde{\alpha}} \frac{(\boldsymbol{\Xi}(\boldsymbol{\zeta}') - \tilde{\boldsymbol{\mu}})^{\mathrm{T}} \tilde{\alpha} \tilde{\boldsymbol{\Sigma}}^{-1} (\boldsymbol{\Xi}(\boldsymbol{\zeta}') - \tilde{\boldsymbol{\mu}})}{\tilde{\beta}} \right)^{-\frac{m+\tilde{\alpha}}{2}} . \tag{25}$$

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega \sim \mathcal{T}_n(\tilde{\alpha}, \tilde{\boldsymbol{\mu}}, \gamma^2 \tilde{\boldsymbol{\Sigma}}) \tag{26}$$

is the multivariate (n-dimensional, because n is the length of the vector $\boldsymbol{\Xi}(\boldsymbol{\zeta}')$) Student's t-distribution. Once the estimate of ω is plugged-in, this distribution is an approximation to the computer model f. This distribution is parameterized by *degrees of freedom* $\tilde{\alpha}$, *location* parameter $\tilde{\boldsymbol{\mu}}$ and *shape matrix* $\gamma^2 \tilde{\boldsymbol{\Sigma}}$, where $\gamma^2 = \tilde{\beta}/\tilde{\alpha}$. If the training data are plugged in, then the estimate of $\gamma^2 = 4.6113$.

Theorem. $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega$ is the stochastic process. We call $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega$ - the T-process. This process is not Gaussian, but it can also serve as an emulator of f.
Corollary. A Gaussian process whose mean is $\mathrm{E}(\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega)$ and covariance is $\mathrm{Cov}(\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}), \omega)$ is the GASP emulator of function f.

Alternative way to write out the multivariate t-distribution is $\mathcal{T}_{\tilde{\alpha}}(\tilde{\boldsymbol{\mu}}, \gamma^2 \tilde{\boldsymbol{\Sigma}})$ which puts emphasis only on the parameters of this distribution. As with respect to properties of the multivariate Student's t-distribution, the mean of the distribution is its location parameter $\tilde{\boldsymbol{\mu}}$, while variance-covariance equals to $\frac{\tilde{\alpha}}{\tilde{\alpha}-2} \gamma^2 \tilde{\boldsymbol{\Sigma}}$.

The final approximation to the computer model that we have constructed in this chapter is shown in Figure 4. Perhaps, surprisingly, but this emulator, which within objective implementation accounts for uncertainty in all of its parameters except for just one, is a much better emulator than the one we started with, which had somewhat *ad hoc* parameters, with no acknowledging for uncertainty in their estimates. Visually, the approximation is much tighter while closely following the function itself. The first emulator, shown in Figure 1 is not as good, providing somewhat large *uncertainty* by its 95% credible area compared to very small uncertainty given by the last emulator. As we can see, the quality of the approximation highly depends on the choice of the procedure about estimates of parameters of the approximation.

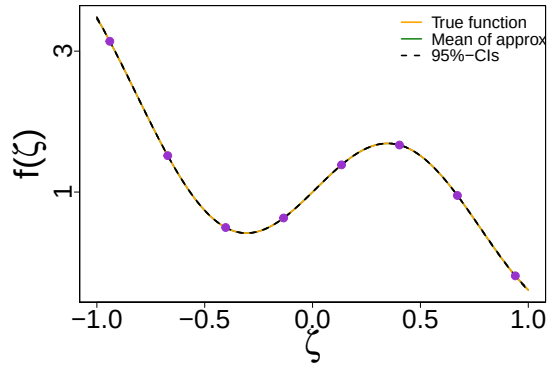

Figure 4: Gaussian process emulator which accounts for uncertainty in parameters $\boldsymbol{\beta}$ and σ^2 within its objective implementation.

Next chapter discusses formal ways to assessing the quality of a constructed statistical approximation of a computer model (or any other statistical model). While in this exposition we have compared the performance of the emulator as an approximation to the true function visually, in practice, assessing the quality of the constructed approximation is not as trivial, because the number of computer model *testing* data points available for assessment of the emulator is limited.

Ksenia N. Kyzyurova

Chapter 1. Assessment of a statistical model

Protagoras argued that a man is a measure of all things. Within the decision-theoretic approach this is mathematically shown to be indeed so: scoring rules *calculated for predictive model evaluation and comparison are* SUBJECTIVE. *This means that the choice of a scoring rule for model comparison affects the results of the comparison and, therefore, the decision on a model choice. What's more, the scoring rules are hardly interpretable. Recommendation is to, instead, employ three independent* frequency estimates *of the quality of model predictions: (1) empirical frequency coverage, (2) predictive bias, and (3) uncertainty (variability) in predictions.*

In order to assess quality of the predictive performance of an emulator, in addition to training data, one must obtain *testing* data. Since the computer model is computationally slow, a small number of additional runs n has been obtained at inputs $\boldsymbol{\zeta}^*$ resulting in outputs $\mathbf{f}(\boldsymbol{\zeta}^*)$. As for the illustrative example $n = 7$ testing inputs are shown in Figure 5. These are chosen right in-between the pairs of the neighbouring training inputs. In this example (1) all testing outputs are captured within the 95% central credible area provided by the emulator, indicating good performance, (2) the bias — discrepancy between true values and the mean (on average or the maximum) of the stochastic approximation is rather small, (3) the average (or maximum) length of credible intervals is short, so that the emulator provides tight approximations for testing points as well as along the whole domain of its inputs.

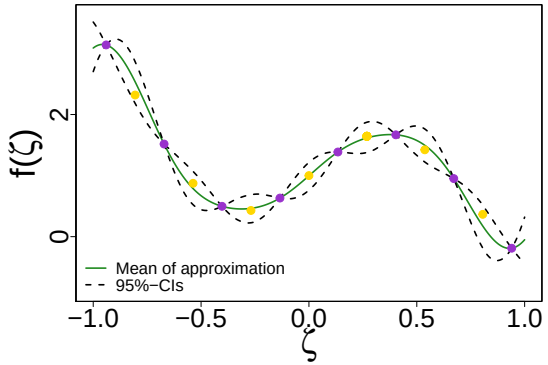

Figure 5: The first emulator constructed in the previous chapter. Training points are those coloured in purple. The emulator is an interpolator at these points. Testing points are coloured in yellow.

Denote the marginal predictive distribution for the ιth input $\boldsymbol{\zeta}^*_\iota$ as p_ι. Let the mean of this distribution be μ_ι and variance of the distribution σ^2_ι. Formally the *frequency estimates* are: *empirical frequency coverage* (EFC), the proportion of true values captured by the corresponding predictive distribution within its central 95%-credible interval[2]

$$\text{EFC} = \left(\frac{\sum_{\iota=1}^{n} I_{f(\boldsymbol{\zeta}^*_\iota) \in \text{CI}_\iota}}{n} \right) 100\% \in (0, 100)\,, \tag{27}$$

where CI_ι is the 95% credible area; *root-mean-square predictive error* (RMSPE) — the estimate of the bias[3] —

$$\text{RMSPE} = \sqrt{\frac{\sum_{\iota=1}^{n} (f(\boldsymbol{\zeta}^*_\iota) - \mu_\iota)^2}{n}} \in (0, \infty)\,, \tag{28}$$

and the average length of the 95% credible intervals over n points

$$\overline{\text{L}_{\text{CI}}} = \frac{\sum_{\iota=1}^{n} \text{CI}_\iota}{n} \in (0, \infty)\,. \tag{29}$$

For normal predictive distributions the last estimate is defined as $\overline{\text{L}_{\text{CI}}} = \frac{\sum_{\iota=1}^{n} 2 \times 1.96 \sigma_\iota}{n}$. Since $\overline{\text{L}_{\text{CI}}}$ does not have true values involved in its calculation, one may also obtain this estimate over greater number of possible inputs — to investigate the overall emulator's uncertainty in predictions. Useful is to verify that the maximal values among

[2] In my experience 95% threshold for a credible area has served as a useful nominal value. This choice is subjective. Other nominal values may be employed.

[3] Alternatively, one may calculate the average absolute mean bias as $\frac{\sum_{\iota=1}^{n} |f(\boldsymbol{\zeta}^*_\iota) - \mu_\iota|}{n} \in (0, \infty)$.

predictive credible intervals are acceptable, and are not substantially greater than average ones. Empirical values of EFC which correspond to the *nominal* (theoretical) value of 95% are desired to be close to each other. However, one needs to subjectively assess if, say, 80% empirical coverage versus corresponding 95% area is good enough. For other two frequency estimates, it is desirable that RMSPE and $\overline{\mathrm{L_{CI}}}$ are small.

Three estimates provide clear and intuitive *interpretation* of what they mean in the evaluation of the predictive performance of a model. Altogether, they *attempt* to provide *objective* assessment because these are *frequency* estimates. These estimates must be considered independently. The attempt to collapsing them into one leads to an *improper* criterion (an improper scoring rule)[4], that is the rule that favours a wrong forecast rather than the true one. If one emulator (or any statistical model) is substantially better than another one, then comparison with the frequency estimates reveal this fact. If two statistical models are *on par* in their performance, then comparison between the models with any estimate or criterion is challenging.

There have been attempts to find a one single criterion which would make predictive model comparison possible. The methodology of *scoring rules* have been proposed. Turns out, this task is not achievable. Two main properties of why this is so are: scoring rules are subjective; and being evaluated, the scores are hardly interpretable.

Definition. A *scoring rule* S is a function which takes a probabilistic forecast, that is, a random variable ζ whose distribution is F, and a true forecast, random variable ω, producing a real-valued *score* $S(\zeta, \omega)$.

Definition. A scoring rule is said to be *proper* if its maximum is reached at the true random variable, that is $\max(S(\zeta,\omega)) = S(\omega,\omega)$.[5]

Theorem. Every scoring rule is *subjective*, that is no scoring rule is equivalent to any other scoring rule.

Proof. Consider two scoring rules S_1 and S_2 that are distinct (that is functions $S_1(\cdot,\cdot) \neq S_2(\cdot,\cdot)$) and two forecasts $\zeta_1 \neq \zeta_2$ in distribution, and the true variable ω, then $f(S_1(\zeta_1,\omega), S_1(\zeta_2,\omega)) \neq f(S_2(\zeta_1,\omega), S_2(\zeta_2,\omega))$, where $f(x_1,x_2)$ is a function of two arguments used to compare scores, that is, for model comparison $f(x_1,x_2)$ is either the difference $x_1 - x_2$ or the ratio of corresponding scores x_1/x_2. ■

This theorem holds, even if S_2 is a monotone increasing transformation of S_1. Therefore, the theorem asserts that conclusion on *how much better* one model is compared to another is not possible to make.

Illustrative example. The illustrative example demonstrates that within a scoring rule the assessment may be inadequate. In particular, a popular logarithmic scoring rule is chosen to demonstrate its deficiencies. Let p denote a predictive density of a forecast, real-valued random variable ζ. Let the random variable ω have a degenerate distribution, that is $P(\omega = v) = 1$. Therefore, v is the true realization. Consider a *logarithmic scoring rule*

$$\mathrm{logS}(\zeta, \omega = v) = \log p(v)\,. \tag{30}$$

Consider a class of normal predictive distributions with mean μ and standard deviation σ. Then for $\zeta \sim \mathcal{N}(\mu, \sigma^2)$ with predictive density p, the logarithmic scoring rule with respect to v is

$$\mathrm{logS}(\zeta, \omega = v) = -\frac{\log(2\pi\sigma^2)}{2} - \frac{(v-\mu)^2}{2\sigma^2}\,. \tag{31}$$

Assuming that the true value v equals to zero, a plot with contourlines of scores which in this case depends simply on the mean and standard deviation of the random variable ζ is shown in Figure 6. The contourlines show that since there are infinitely many points along a particular contourline, infinitely many distributions exist which have exactly the same value of a log-score. The same value of a score makes these distributions to be considered equally good predictive distributions, but these distributions may be very different from each other.

[4] Predictive model choice criterion (PMCC) is improper.

[5] If in this definition the scoring rule is maximized, then the scoring rule is said to be *positively oriented*. Alternatively, one can choose to minimize a scoring rule, then the rule is said to be *negatively oriented*.

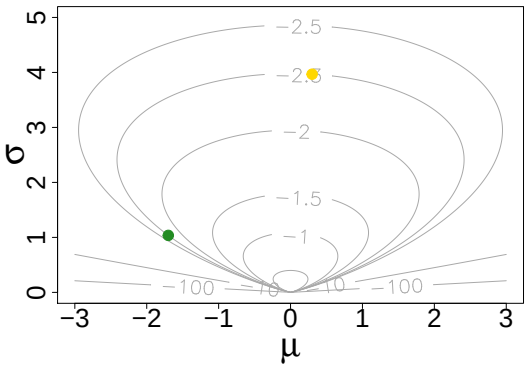

Figure 6: Contourlines of the logarithmic score given for a class of normal distributions characterized by two parameters: mean μ and standard deviation σ. Two coloured points correspond to two distributions.

As example, consider two points on the plot: the green point corresponds to the distribution $\mathcal{N}(\mu = -1.7, \sigma = 1.038363)$ and orange point corresponds to $\mathcal{N}(\mu = 0.3, \sigma = 3.967767)$. Indeed, the distributions are very different: the first distribution has a large bias — the mean of the distribution is far from the true value of zero. The orange distribution has bias more than 5 times smaller of that of the green one. However, the orange distribution has standard deviation more than three times greater than that of the green one, which translates to variance being more than 14 times greater. Yet, two distributions are assigned exactly the same value of a logarithmic score.

Consider another example of two similar forecasts which could have resulted from two other models, namely $\mathcal{N}(\mu = 0.9, \sigma = 1)$ and $\mathcal{N}(\mu = 1, \sigma = 0.9)$. Providing virtually the same information on the predictive distribution in practice, the scores, as compared to the true value of zero, are -1.323939 and -1.430862 respectively, thus differentiating between the two distributions much more than one desires.

These examples demonstrate that logarithmic score can not distinguish between very different distributions, assigning the same scores to considerably different shapes; and at the same time assigning much more different value to very similar distributions. It may be demonstrated that other scores provide similar but different contourplots for the class of normal distributions, behaving in analogous way as have been observed with the logarithmic scoring rule.

Chapter 2. Computer model with multivariate output

Computer models often produce multivariate output for every single run of the model. There have been attempts to account for correlation among outputs in the construction of a Gaussian process emulator of such a model with the goal of achieving a more accurate emulator. Both, theoretical evidence and simulations are presented here which demonstrate that multivariate emulator does not lead to "better" (that is, more accurate, precise or less uncertain) emulation results compared to independent modeling of each component of the output.

Suppose a computer model produces bivariate output, which may be described as two smooth functions $f_1(\cdot)$ and $f_2(\cdot)$. Suppose the simulator data is obtained at n d-dimensional points $\boldsymbol{\zeta} = (\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_n)$ from the input space $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_d$. In other words, at the ith input $\boldsymbol{\zeta}_i$, the output of a simulator is a two-dimensional vector $(f_1(\boldsymbol{\zeta}_i), f_2(\boldsymbol{\zeta}_i))^{\mathrm{T}}$.

The simplest emulator of such bivariate output of a computer model may be constructed choosing independent emulators of each output. That is, if ιth function is assumed to come from a Gaussian process characterized by parameters $\boldsymbol{\theta}_\iota = (\boldsymbol{\beta}_\iota, \sigma^2_\iota, \boldsymbol{\omega}_\iota)$

$$\Phi_\iota(\cdot) \mid \boldsymbol{\theta}_\iota \sim \mathcal{GP}(\mu_\iota(\cdot), \mathrm{cov}_\iota(\cdot,\cdot)) \,. \tag{32}$$

A computer model bivariate output from the simulator by $\boldsymbol{\Phi}_1(\boldsymbol{\zeta}) = (f_1(\boldsymbol{\zeta}_1), \ldots, f_1(\boldsymbol{\zeta}_n))$ and $\boldsymbol{\Phi}_2(\boldsymbol{\zeta}) = (f_2(\boldsymbol{\zeta}_1), \ldots, f_2(\boldsymbol{\zeta}_n))$.

$$\Phi_\iota(\cdot) \mid \cdot, \boldsymbol{\theta}_\iota, \boldsymbol{\Phi}_\iota(\boldsymbol{\zeta}) \sim \mathcal{GASP}(\mu^*_\iota(\cdot), \sigma^{*2}_\iota(\cdot,\cdot)) \,. \tag{33}$$

The resulting independent emulators are the posterior predictive distributions which are independent normal distributions $\mathcal{N}(\boldsymbol{\mu}^*_\iota(\boldsymbol{\zeta}'), \sigma^{*2}_\iota(\boldsymbol{\zeta}', \boldsymbol{\zeta}'))$, $\iota = 1, 2$.

We want to compare such emulators to an emulator of a joint bivariate Gaussian process $(f_1(\cdot), f_2(\cdot))^{\mathrm{T}} \mid \boldsymbol{\zeta}$ in terms of their predictive distributions. This is especially of interest if $f_1(\cdot)$ and $f_2(\cdot)$ are suspected to be highly correlated. The answer if joint modeling is beneficial compared to independent modeling of multivariate output for emulation purpose is no.

Suppose $f_2(\cdot) = \lambda f_1(\cdot) + \gamma$, where λ and γ are unknown constants. This is a situation of two perfectly correlated functions, for which one might expect to achieve the most value from multivariate modeling. (Example of such functions is given in Figure 7.) The following proposition establishes this is false.

Proposition. Suppose the bivariate simulator f_1 and f_2 is observed at the input points $\boldsymbol{\zeta}$ and that $f_2 = \lambda f_1 + \gamma$. GASP approximations that would result from emulating each output independently are given by (33). Let $c_1(\cdot,\cdot)$ and $c_2(\cdot,\cdot)$ have the same functional form, but with possibly different parameters $\boldsymbol{\omega}_1$ and $\boldsymbol{\omega}_2$. Let each Gaussian process mean be linear with the same vector of regression covariates $\mathbf{h}(\cdot)^{\mathrm{T}}$, such that $h_0(\cdot) = 1$ defining an intercept, i.e., $\mu_\iota(\cdot) = h(\cdot)\boldsymbol{\beta}^{(\iota)}$. Then the maximum likelihood estimates of the model parameters have the following properties:

The estimates of the correlation parameters are the same, i.e. $\hat{\boldsymbol{\omega}}_1 = \hat{\boldsymbol{\omega}}_2$.

The estimates of the variances satisfy $\hat{\sigma}_2 = \lambda\hat{\sigma}_1$.

The estimates of the intercepts satisfy $\hat{\boldsymbol{\beta}}_0^{(2)} = \lambda\hat{\boldsymbol{\beta}}_0^{(1)} + \gamma$ and, for the other regression coefficients, $\hat{\boldsymbol{\beta}}_j^{(2)} = \lambda\hat{\boldsymbol{\beta}}_j^{(1)}$.

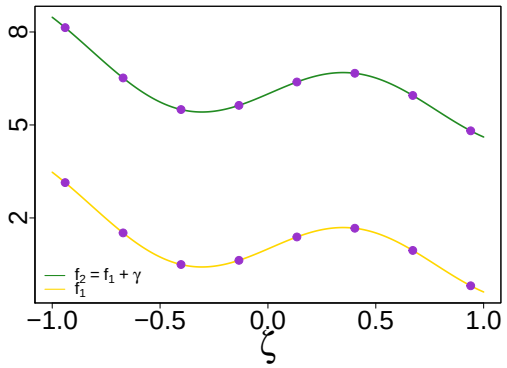

Figure 7: Two curves represent two output functions $f_1(\boldsymbol{\zeta})$ and $f_2(\boldsymbol{\zeta})$ of a computer model. The circles correspond to the design input-output points that are used to construct the joint emulator.

Corollary. Let the predictive distribution of the ι's output at a new point $\boldsymbol{\zeta}'$ be denoted as

$$\Phi_\iota(\boldsymbol{\zeta}') \sim \mathcal{N}(\mu_\iota^*(\boldsymbol{\zeta}'), \sigma_\iota^*(\boldsymbol{\zeta}', \boldsymbol{\zeta}'))\,. \tag{34}$$

At a new point $\boldsymbol{\zeta}'$ the predictive means and variances of the two independent emulators $\Phi_1(\cdot)$ and $\Phi_2(\cdot)$ (with maximum likelihood estimates of parameters) are related as

$$\mu_2^*(\boldsymbol{\zeta}') = \gamma + \lambda\mu_1^*(\boldsymbol{\zeta}')\,, \tag{35}$$

$$\sigma_2^{*2}(\boldsymbol{\zeta}', \boldsymbol{\zeta}') = \lambda^2\sigma_1^{*2}(\boldsymbol{\zeta}', \boldsymbol{\zeta}')\,. \tag{36}$$

Theorem. Let the joint predictive distribution of the two functions be specified conditionally, i.e.,

$$p(\Phi_2(\boldsymbol{\zeta}'), \Phi_1(\boldsymbol{\zeta}') \mid \boldsymbol{\zeta}') = p(\Phi_2(\boldsymbol{\zeta}') \mid \Phi_1(\boldsymbol{\zeta}'), \boldsymbol{\zeta}')p(\Phi_1(\boldsymbol{\zeta}') \mid \boldsymbol{\zeta}'). \tag{37}$$

Then marginal emulators $\Phi_1(\cdot)$ and $\Phi_2(\cdot)$ of each output $f_1(\cdot)$ and $f_2(\cdot)$ coincide in either their independent or joint modeling if maximum likelihood or maximum a posteriori estimates of parameters of the processes are employed.

Therefore, no benefit is due to construction of a multivariate emulator of a multivariate output computer model compared to independent emulation of each output. Proofs of the mathematical statements in this chapter are given in my thesis.

Chapter 3. Approximation to a system of computer models

Direct approximation of a system which consists of several computer models is difficult for computational and logistical reasons. The methodology of the linked Gaussian approximation has been demonstrated to be a successful alternative. This is outlined in this chapter.

Consider two computer models, f_1 and f_2, whose inputs and outputs are real-valued; and for which the corresponding Gaussian process approximations (introduced in chapter) may be constructed. Let Ξ and Υ be the corresponding emulators. The emulator $\Xi(\cdot)$ at any new input, given pairs of model runs $\{\boldsymbol{\zeta}, \boldsymbol{\Xi(\zeta)} = \mathbf{f_1}(\boldsymbol{\zeta})\}$ and

this emulator's vector of parameters $\boldsymbol{\theta}_\Xi$, is $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi(\zeta)}, \boldsymbol{\theta}_\Xi \sim \mathcal{GASP}(\mu^*_\Xi(\cdot), \sigma^{*2}_\Xi(\cdot,\cdot))$. Likewise, the emulator $\Upsilon(\cdot)$of the model f_2, given $\{\boldsymbol{\kappa}, \boldsymbol{\Upsilon(\kappa)} = \mathbf{f_2}(\boldsymbol{\kappa})\}$ and $\boldsymbol{\theta}_\Upsilon$, is $\Upsilon(\cdot) \mid \cdot, \boldsymbol{\Upsilon(\kappa)}, \boldsymbol{\theta}_\Upsilon \sim \mathcal{GASP}(\mu^*_\Upsilon(\cdot), \sigma^{*2}_\Upsilon(\cdot,\cdot))$. The parameters $\boldsymbol{\theta}_\Xi$ and $\boldsymbol{\theta}_\Upsilon$ here are assumed known. In practice, they are estimated.

Having introduced the framework for an emulator of a single computer model I consider a question how to approximate a system of two computer models $f_1 \circ f_2$ defined as the composite model if only the two submodels $f_1(\cdot)$ and $f_2(\cdot)$ may be independently computed at a few data points, while the composition $f_1 \circ f_2(\cdot)$ can not be observed. For any new input u to the system of computer models $f_1 \circ f_2$, the approximation is defined as $\Xi \circ \Upsilon$, which is denoted as Φ, i.e. $p(\Phi(u) \mid \boldsymbol{\Xi(\zeta)}, \boldsymbol{\Upsilon(\kappa)}, \boldsymbol{\theta}_\Xi, \boldsymbol{\theta}_\Upsilon, u) = \int p(\Xi(\Upsilon(u)) \mid \boldsymbol{\Xi(\zeta)}, \Upsilon(u), \boldsymbol{\theta}_\Upsilon) p(\Upsilon(u) \mid \boldsymbol{\Upsilon(\kappa)}, \boldsymbol{\theta}_\Xi) \mathrm{d}\Upsilon(u)$.

Definition. The variable $\xi = \Phi(u) \mid \boldsymbol{\Xi(\zeta)}, \boldsymbol{\Upsilon(\kappa)}, \boldsymbol{\theta}_\Xi, \boldsymbol{\theta}_\Upsilon, u$ is called the linked emulator.

Theorem. Linked Gaussian emulator (or Gaussian approximation to a collection of random variables $\Phi_t, t \in T$, where T is an index set) is a stochastic process.

Proof. The linked Gaussian emulator is a Gaussian stochastic process by construction: the mean, variances and correlation structure among *all* variables of this process may be specified. Corresponding formulae rely on the laws of total mean, total variance and total covariance respectively.[6].

Illustrative example. Two functions $f_2(x) = \cos(2x)$ in the domain $x \in [-3, 3]$ and $f_1(z) = \cos(z/2)$ in the domain $z \in [-2, 5]$ are test functions. Model f_2 has been evaluated at 8 training inputs; f_1 — at 5 training data points. The simulators, their designs and emulators are shown in 8. The resulting linked Gaussian approximation to $f_1 \circ f_2$ is shown in 9.

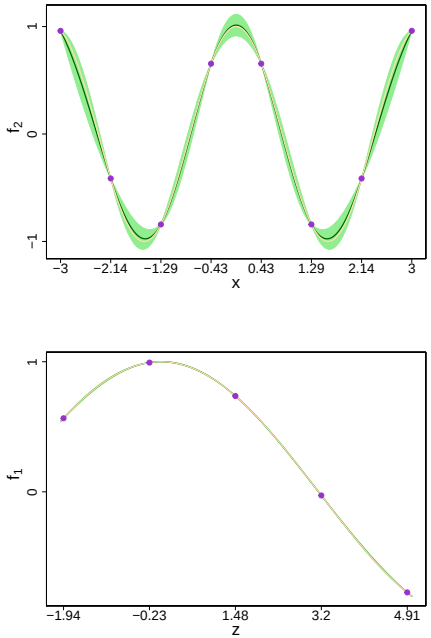

Figure 8: Two test functions: $f_1(x)$ and $f_2(x)$ along with their Gaussian approximations. The dark green lines are the means of emulators, the yellow lines are the true functions. 95% central credible area is shown with the green shaded regions. The purple circles are at the design points.

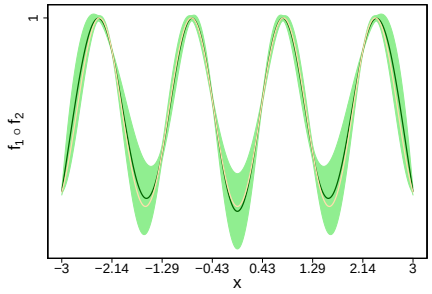

Figure 9: Linked Gaussian approximation to a system of simulators which has never been observed; this is a quite accurate approximation to the system.

In [4] many more mathematical details on how linked approximations of computer models may be constructed are provided. These include linking of *objective* Gaussian process emulators of computer models, analysis of linking of multivariate input-output models, other illustrative examples, and examples of doing so with more realistic computer models of volcano pyroclastic flows and volcano ash plume dispersion.

[6] See details in [3]

Chapter 4. Calibration of a computer model

If observations corresponding to the output of the model are collected, a theoretical model may be assessed on the agreement to the collected data. Moreover, given a computer model and collected data, one might inquire which values of inputs to the model could have generated the collected data; thus, performing calibration *of a model.*

Calibration is, therefore, analogous to finding an inverse image of function $f : Z \to Y$, given a set of n values $\mathbf{y} = (y_1, \ldots, y_n) \in Y$, that is $f^{-1}(\mathbf{y}) = \{\boldsymbol{\zeta} = (\zeta_1, \ldots, \zeta_n) \in Z : f(\zeta_i) = y_i \ \ \forall\, i = 1, \ldots, n\}$. However, this is indeed only an analogy, since in practice calibration involves (a) a computationally intensive computer model or a system of such models, and (b) noisy *collected data which does not arise from the computer model itself but is obtained in either an experiment or observed in nature. Calibration framework which accounts for these two levels of complexity is presented in this study.*

Suppose that n experimental data points $\mathbf{y} = (y_1, \ldots, y_n)^{\mathrm{T}}$ have been collected; each ith point $y_i, i = 1, \ldots, n$ has been obtained under some input conditions $\boldsymbol{\zeta}_i^* \in Z$. These input conditions are assumed to belong to the same input space $\boldsymbol{\zeta}_i^* \in Z$ as the computer model inputs.

Each ith input $\boldsymbol{\zeta}_i^*$ is comprised of a set of control variables $\boldsymbol{\zeta}_i^{*,c} \in \mathcal{R}^{d_1}$ (say, first d_1 inputs) and a set of calibration variables $\boldsymbol{\zeta}_i^{*,p} \in \mathcal{R}^{d_2}$, the set of the rest $d_2 = d - d_1$ variables; i.e. $\boldsymbol{\zeta}^* = \{\boldsymbol{\zeta}^{*,c}, \boldsymbol{\zeta}^{*,p}\}$. A subset of control variables $\boldsymbol{\zeta}^{*,c}$ is set and known. A subset of $\boldsymbol{\zeta}^{*,p}$ is unknown. However, the unknown conditions are assumed to be the same for all data points, i.e. $\boldsymbol{\zeta}_1^{*,p} = \boldsymbol{\zeta}_2^{*,p} = \ldots = \boldsymbol{\zeta}_n^{*,p} = \boldsymbol{\alpha}$.

Calibration aims at identifying such values of parameters $\boldsymbol{\alpha}$, that computer model output (that is of its approximation) $\boldsymbol{\Xi}(\boldsymbol{\zeta}^*) = (\Xi(\boldsymbol{\zeta}_1^{*,c}, \boldsymbol{\alpha}), \ldots, \Xi(\boldsymbol{\zeta}_n^{*,c}, \boldsymbol{\alpha}))^{\mathrm{T}}$ obtained at inputs $\boldsymbol{\zeta}^*$ with parameters $\boldsymbol{\alpha}$ plugged in matches closely experimental data $\mathbf{y}$.

Conditional on the computer model data $\{\boldsymbol{\zeta}, \boldsymbol{\Xi}(\boldsymbol{\zeta})\}$, within the emulation framework, computer model output $\boldsymbol{\Xi}(\boldsymbol{\zeta}^*) = \boldsymbol{\Xi}(\boldsymbol{\zeta}^{*,c}, \boldsymbol{\alpha})$ is given by the predictive distribution

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}^{*,c}, \boldsymbol{\alpha}) \mid \boldsymbol{\zeta}^{*,c}, \boldsymbol{\alpha}, \boldsymbol{\zeta}, \boldsymbol{\Xi}(\boldsymbol{\zeta}) = (\Xi(\boldsymbol{\zeta}_1^{*,c}, \boldsymbol{\alpha}), \ldots, \Xi(\boldsymbol{\zeta}_n^{*,c}, \boldsymbol{\alpha}))^{\mathrm{T}} \mid \boldsymbol{\zeta}^{*,c}, \boldsymbol{\alpha}, \boldsymbol{\zeta}, \boldsymbol{\Xi}(\boldsymbol{\zeta})\,. \tag{38}$$

This yields the likelihood $L(\boldsymbol{\alpha}; \mathbf{y}) = p(\mathbf{y} \mid \boldsymbol{\zeta}^{*,c}, \boldsymbol{\alpha}, \boldsymbol{\zeta}, \boldsymbol{\Xi}(\boldsymbol{\zeta}))$ for to-be calibrated parameters $\boldsymbol{\alpha}$, which is given by the distribution of experimental data $\mathbf{y} = (y_1, \ldots, y_n)^{\mathrm{T}}$ observed at $\boldsymbol{\zeta}^*$

$$L(\boldsymbol{\alpha}; \mathbf{y}) = p(\Xi(\boldsymbol{\zeta}_1^{*,c}, \boldsymbol{\alpha}) = y_1, \ldots, \Xi(\boldsymbol{\zeta}_n^{*,c}, \boldsymbol{\alpha}) = y_n \mid \boldsymbol{\zeta}^{*,c}, \boldsymbol{\alpha}, \boldsymbol{\zeta}, \boldsymbol{\Xi}(\boldsymbol{\zeta}))\,. \tag{39}$$

To complete specification of the model, one must specify the prior distribution for unknown parameters $\boldsymbol{\alpha}$. In the calibration setting prior distribution $p(\boldsymbol{\alpha})$ on parameters $\boldsymbol{\alpha}$ is, though, possibly vague, but an expert-elicited prior.

The solution to the calibration problem is the posterior distribution $p(\boldsymbol{\alpha} \mid \mathbf{y}, \boldsymbol{\zeta}^{*,c}, \boldsymbol{\zeta}, \boldsymbol{\Xi}(\boldsymbol{\zeta}))$. For convenience the short notation is $p(\boldsymbol{\alpha} \mid \mathbf{y})$ which implicitly assumes conditioning on known controlled settings $\boldsymbol{\zeta}^{*,c}$ and computer model data, that is, pairs of observations $\{\boldsymbol{\zeta}, \boldsymbol{\Xi}(\boldsymbol{\zeta})\}$. Posterior distribution is then obtained

$$p(\boldsymbol{\alpha} \mid \mathbf{y}) = \frac{L(\boldsymbol{\alpha}; \mathbf{y}) p(\boldsymbol{\alpha})}{\int L(\boldsymbol{\alpha}; \mathbf{y}) p(\boldsymbol{\alpha})\, \mathrm{d}\boldsymbol{\alpha}}\,. \tag{40}$$

This distribution is well-defined since the prior $p(\boldsymbol{\alpha})$ is defined on closed space of parameters, and, therefore, the posterior is proper. Thus, the numerical solution (40) to the calibration problem always exists.

Qualitative assessment of calibration results. Ideally, posterior $p(\boldsymbol{\alpha} \mid \mathbf{y})$ identifies a small set of most probable parameters $\boldsymbol{\alpha}$ such that computer model data evaluated at these values matches closely collected data $\mathbf{y}$. In the unfortunate scenario, one may find that even being evaluated at the most probable values of $\boldsymbol{\alpha}$, computer model output may not match the collected data, thus, resulting in no agreement between the theory and the data. Such a case would indicate a crucial problem with either a theoretical computer model or a prior on calibration parameters $p(\boldsymbol{\alpha})$, or both; demanding for more inquiry into domain knowledge.

The following posterior corresponding to the calibration of $\boldsymbol{\alpha}$ is

$$p(\boldsymbol{\alpha} \mid \mathbf{y}) \propto \mathcal{N}\left(\mathbf{y} \mid \mu_f^*(\mathbf{t}, \boldsymbol{\alpha}), \sigma_f^{*2}((\mathbf{t}, \boldsymbol{\alpha}), (\mathbf{t}, \boldsymbol{\alpha}))\right) p(\boldsymbol{\alpha})\,. \tag{41}$$

Probabilistic inversion (41) involves numerical computation of the inverse of the covariance matrix $\sigma_f^{*2}((\mathbf{t}, \alpha), (\mathbf{t}, \alpha))$ from the log-likelihood for different values of $\boldsymbol{\alpha}$. This matrix may provide strong correlation between points, such that its correct numerical inversion becomes a challenging problem even if the number of experimental

data points is small. Numerical errors coming from the inversion of the matrix in log-densities may preclude obtaining the correct solution. In order to overcome this computational problem the information about the correlation structure in the likelihood may be omitted, effectively resulting in the following distribution (42) as an approximation to the true posterior (41)

$$p(\boldsymbol{\alpha} \mid \mathbf{y}) \propto \mathcal{N}\left(\mu_f^*(\mathbf{t}, \boldsymbol{\alpha}), \operatorname{diag}\left(\sigma_f^{*2}((\mathbf{t}, \boldsymbol{\alpha}), (\mathbf{t}, \boldsymbol{\alpha}))\right)\right) p(\boldsymbol{\alpha})\,. \tag{42}$$

This distribution does not take information about GASP sample paths into account but only uses the information from the *marginals* of the corresponding likelihood. Computing this approximated solution is trivial.

Proposition. Let the GASP correlation function be dependent of and decaying with distance along its inputs, then the further apart from each other experimental data inputs are, the more their corresponding cross-correlations are approaching zero. Therefore, the closer approximation (42) is to the true posterior (41).

Proof. Following exposition of the GASP, correlation between two inputs $\boldsymbol{\zeta}_1$ and $\boldsymbol{\zeta}_2$ is $c(\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2) = \prod_{j=1}^{d} c(\zeta_{1j}, \zeta_{2j})$. For the jth coordinate correlations depend only on the distance $c(\zeta_{1j}, \zeta_{2j}) = c(|\zeta_{1j} - \zeta_{2j}|)$, whose properties are $c(\zeta_{1j}, \zeta_{2j}) \longrightarrow 1$ and $c(\zeta_{1j}, \zeta_{2j}) \longrightarrow 0$ as $|\zeta_{1j} - \zeta_{2j}| \longrightarrow \infty$. The last property proves the proposition.

Calibration with respect to noisy collected data. In practice collected data is not perfect and is observed with noise $\boldsymbol{\epsilon}$. That is, true values of $\mathbf{y}$ are not known but may be estimated from the collected data. Collected data is denoted as $\mathbf{y}^E$. In the previous section the ideal scenario of the absence of any noise in the collected data was assumed, resulting in that $\mathbf{y}^E = \mathbf{y}$. In practice, true data $\mathbf{y}$ is not known, but posterior estimate on the true values of $\mathbf{y}$ given the collected data, i.e. the distribution $p(\mathbf{y} \mid \mathbf{y}^E)$ may be obtained.

The more information about the error in the data $\mathbf{y}^E$ is available, the better the estimate of posterior $p(\mathbf{y} \mid \mathbf{y}^E)$ is going to be. The common knowledge is that the larger the noise in the data, the more replicates is required to be taken in order to estimate and reduce the noise. Ultimately, the quality of collected data matters most.

Having obtained posterior on the true values $\mathbf{y} \mid \mathbf{y}^E$, calibration becomes

$$p(\boldsymbol{\alpha} \mid \mathbf{y}^E) = \int_{\mathcal{Y}} p(\boldsymbol{\alpha} \mid \mathbf{y}) p(\mathbf{y} \mid \mathbf{y}^E) d\mathbf{y}\,, \tag{43}$$

where $p(\boldsymbol{\alpha} \mid \mathbf{y})$ is given by equation (40).[7]

This form of the posterior reflects that one may be unsure about the truth, true values $\mathbf{y}$. The more sure one is about the truth, the closer calibration framework (43) is to (40). Calibration (43) also states that one is able to calibrate only to the extent that the observed data allow us to do it. This acknowledges the possibility that calibration task with corrupt data (e.g., data with large noise or biased data or data with limited information on how the data has been obtained) does not allow to perform calibration.

Chapter 5. Censoring for a computer model with zero-inflated output

Computer model of a volcano pyroclastic flow, given a set of initial conditions, produces an output, maximum height of the flow at thousands of geographical locations. The output is non-negative and often results in exact zero, thus, indicating the absence of a flow, and resulting in that the zero-height value of an output has a non-zero *probability to occur, as opposed to all other simulator output values. In order to account for these features of the output, the methodology of a censored GASP approximation to such a computer model is employed. Customarily employed, usual GASP, does not allow to construct an approximation which would incorporate these features of the simulator.*

[7] Computationally approximation to $p(\boldsymbol{\alpha} \mid \mathbf{y}^E)$ may be obtained through sampling from the distribution $p(\mathbf{y} \mid \mathbf{y}^E)$. For each of the r independent samples $\mathbf{y}_{smpl_i} \sim p(\mathbf{y} \mid \mathbf{y}^E)$, $i = 1, \ldots, r$, the distribution $\boldsymbol{\alpha} \mid \mathbf{y}_{smpl_i}$ may be obtained. All combined (to integrate out $\mathbf{y} \mid \mathbf{y}^E$), marginal $p(\boldsymbol{\alpha} \mid \mathbf{y}^E)$ is the posterior estimate of the unknown parameter α. This approach is easily parallelizable, since $p(\boldsymbol{\alpha} \mid \mathbf{y}_{smpl_i})$ is independent from $p(\boldsymbol{\alpha} \mid \mathbf{y}_{smpl_j})$ for all independent samples $\mathbf{y}_{smpl_i}$, $i = 1, \ldots, r$.

Dealing with non-zero probability of a zero-inflated output is inconvenient, and, therefore, it's tempting to ignore this important information this way or another. Two possibilities to doing this exist: first, ignoring zero-output data from the construction of the emulator of the model; second, setting an assumption that the output is a smooth function over the whole range of its values. Both assumptions are unappealing, because either important information from the zero-output data is completely ignored, or the assumption of a smooth function which a priori is known not to hold true is employed. These are the reasons to construct a censored GASP approximation which incorporates appropriate assumptions and its construction involves all possible data.

Let the computer model f be evaluated at n_1 inputs $\boldsymbol{\zeta}^O = (\boldsymbol{\zeta}_1^O, \ldots, \boldsymbol{\zeta}_{n_1}^O)$ resulting in respective outputs $f(\boldsymbol{\zeta}_1^O), \ldots, f(\boldsymbol{\zeta}_{n_1}^O)$. In addition to this simulator data, at n_2 inputs $\boldsymbol{\zeta}^C$ simulator $f(\cdot)$ is known to produce values in the range (a, b) but values of the simulator $f(\cdot)$ at inputs $\boldsymbol{\zeta}^C = \{\boldsymbol{\zeta}_i^C\}_{i=1}^{n_2}$ are not known and are not available. That is, it is known only that $a < f(\boldsymbol{\zeta}_i^C) < b$ for each ith input in a set $\boldsymbol{\zeta}^C$. Originally this work has been motivated by construction of the emulator for the model of the height of a pyroclastic flow, which demands that $a = 0$ and $b = \infty$. Other possibilities for choices of a and b are not considered in this work. Therefore, the output from the computer model itself called $f_a(\cdot) = \max(a, f(\cdot))$, is a censored function. In other words, for each single input $\boldsymbol{\zeta}_i$ the computer model output $f_a(\boldsymbol{\zeta}_i)$

$$f_a(\boldsymbol{\zeta}_i) = \begin{cases} f(\boldsymbol{\zeta}_i), & \text{if } f(\boldsymbol{\zeta}_i) > a \\ a, & \text{otherwise } . \end{cases} \tag{44}$$

Let the latent function $f(\cdot)$ be assigned a Gaussian process prior, i.e. $\Xi(\cdot) \sim \mathcal{GP}(\mu(\cdot), \sigma^2 c(\cdot,\cdot))$. Then, conditional on computer model data $\boldsymbol{\Xi}(\boldsymbol{\zeta}^O)$ and $\boldsymbol{\Xi}(\boldsymbol{\zeta}^C)$ the GASP emulator of f is $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C), \boldsymbol{\theta} \sim \mathcal{GASP}(\mu^*(\cdot), \sigma^{*2}(\cdot,\cdot))$, the predictive distribution $\Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C)$ at a new point $\boldsymbol{\zeta}'$.

In this work outputs $\Xi(\boldsymbol{\zeta}^C)$ are not available. It is only known that $\Xi(\boldsymbol{\zeta}_i^C) < a$ for each ith input $\boldsymbol{\zeta}_i^C$. The following predictive distribution $\Xi_a(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}$ is of interest to us. Here $\mathbf{a}$ is an m-dimensional vector of ones (of length of a vector $\boldsymbol{\zeta}^C$) times constant a.

For a collection of m d-dimensional inputs $\boldsymbol{\zeta} = \{\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_m\}$ (given parameters of the GASP $\boldsymbol{\theta}$) statistical approximation to the outputs at inputs $\boldsymbol{\zeta}$ is $\Xi_a(\boldsymbol{\zeta}) = \{\Xi_a(\boldsymbol{\zeta}_1), \ldots, \Xi_a(\boldsymbol{\zeta}_m)\}$ has a mixture distribution. This distribution consists of three parts: (a) an absolutely continuous distribution with respect to the reference measure that is the sum of a unit point-mass at the m-dimensional vector $(a, \ldots, a)^{\mathrm{T}}$, (b) Lebesgue measure on $\mathbb{R}^m_{>a}$ and (c) mixed type measures on $\mathbb{R}^{|\bar{S}|}_{>a} \times a^{|S|}$, where S is a subset of indices of m variables, i.e. $S \subseteq 1:m = \{1, \ldots, m\}$ and $\bar{S} = \{1, \ldots, m\} \setminus S$ is a complementary subset of S.

The approximation $\Xi(\cdot) \mid \cdot, \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C), \boldsymbol{\theta}$ induces another stochastic process $\Xi_a(\cdot)$ — approximation to the simulator $f_a(\cdot)$. Stochastic process $\Xi_a(\cdot)$ is specified by a real-valued threshold a and a triple of a mean function $\mu(\cdot)$, variance σ^2 and correlation function $c(\cdot,\cdot)$.

Denote $\mathrm{P}(1:m)$ as a powerset of $1:m$ and cardinality $|S| = \mathbf{card}\, S$, then vector $\boldsymbol{\Xi}_a(\boldsymbol{\zeta})$ has the following joint distribution

$$p(\boldsymbol{\Xi}_a(\boldsymbol{\zeta})) = \begin{cases} \int_{-\infty}^{a} \cdots \int_{-\infty}^{a} p(\boldsymbol{\Xi}(\boldsymbol{\zeta})) \prod_{i=1}^{m} \mathrm{d}\Xi(\boldsymbol{\zeta}_i), & \Xi_a(\boldsymbol{\zeta}_i) = a, i = \overline{1,m}, \\ \\ \forall\, S \in \mathrm{P}(1:m) \setminus \{\emptyset, 1:m\} & \Xi_a(\boldsymbol{\zeta}_i) > a, i \in \bar{S} \\ p(\boldsymbol{\Xi}(\boldsymbol{\zeta})_{\bar{S}}) \int \cdots \int_{\Delta} p(\boldsymbol{\Xi}(\boldsymbol{\zeta})_S \mid \boldsymbol{\Xi}(\boldsymbol{\zeta})_{\bar{S}})\, \mathrm{d}\boldsymbol{\Xi}(\boldsymbol{\zeta})_{\bar{S}}, & \Xi_a(\boldsymbol{\zeta}_i) = a, i \in S \\ \\ p(\boldsymbol{\Xi}(\boldsymbol{\zeta})) & \Xi_a(\boldsymbol{\zeta}_i) > a, i = \overline{1,m}, \end{cases} \tag{45}$$

where $\Delta = (-\infty, a)^{\mathbf{card}\, S}$ and $\overline{1,m} = 1, \ldots, m$. Here $\boldsymbol{\Xi}(\boldsymbol{\zeta})$ has a multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}^*(\boldsymbol{\zeta}), \boldsymbol{\sigma}^{*2}(\boldsymbol{\zeta}, \boldsymbol{\zeta}))$ and $\boldsymbol{\Xi}(\boldsymbol{\zeta})_S = \{\Xi(\boldsymbol{\zeta}_i)\}_{i \in S}$ is a realization with variables S included in vector $\boldsymbol{\Xi}(\boldsymbol{\zeta})$.

Theorem. $\Xi_a(\cdot)$ is a stochastic process.

Joint multivariate distribution of the latent emulator $\boldsymbol{\Xi}(\boldsymbol{\zeta})$ of the simulator f at a set of design points $\boldsymbol{\zeta} = (\boldsymbol{\zeta}^O, \boldsymbol{\zeta}^C)$ (design points $\boldsymbol{\zeta}^O$ with corresponding uncensored computer model output and design points $\boldsymbol{\zeta}^C$ with corresponding censored output) is given as the following set of two distributions

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}^O) \sim \mathcal{N}\left(\boldsymbol{\mu}(\boldsymbol{\zeta}^O), \sigma^2 \mathbf{C}_{z^O}\right),$$
$$\boldsymbol{\Xi}(\boldsymbol{\zeta}^C) \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O) \sim \mathcal{N}\left(\boldsymbol{\mu}^*(\boldsymbol{\zeta}^C), \sigma^{*2}(\boldsymbol{\zeta}^C)\right).$$

Corresponding joint multivariate distribution of the emulator $\boldsymbol{\Xi}_a(\boldsymbol{\zeta})$ of the simulator output $\mathbf{f}_a(\boldsymbol{\zeta})$ at design input points $\boldsymbol{\zeta}$ is given by the two distributions $\boldsymbol{\Xi}(\boldsymbol{\zeta}^O)$ and $\boldsymbol{\Xi}_a(\boldsymbol{\zeta}^C) \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O) = \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}$. Namely

$$\begin{aligned} \boldsymbol{\Xi}(\boldsymbol{\zeta}^O) &\sim \mathcal{N}\left(\boldsymbol{\mu}(\boldsymbol{\zeta}^O), \boldsymbol{\sigma}^2 \mathbf{C}_{z^O}\right), \\ \boldsymbol{\Xi}_a(\boldsymbol{\zeta}^C) \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O) &\sim \mathcal{TN}_{(-\infty,a)}\left(\boldsymbol{\mu}^*(\boldsymbol{\zeta}^C), \boldsymbol{\sigma}^{*2}(\boldsymbol{\zeta}^C, \boldsymbol{\zeta}^C)\right). \end{aligned}$$

Predictive distribution at a new input to a computer model. In order to obtain predictive distribution of a simulator $\boldsymbol{\Xi}_a(\boldsymbol{\zeta})$ at a new input $\boldsymbol{\zeta}'$, consider the statistical approximation to a latent function $f(\cdot)$. Predictive distribution given by the latent emulator $\Xi(\cdot)$ at a new input $\boldsymbol{\zeta}'$, conditional on evaluations of the computer model $\boldsymbol{\Xi}(\boldsymbol{\zeta})$ at design input points $\boldsymbol{\zeta}$, is

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}) \sim \mathcal{N}(\boldsymbol{\mu}^*(\boldsymbol{\zeta}'), \boldsymbol{\sigma}^{*2}(\boldsymbol{\zeta}', \boldsymbol{\zeta}')). \tag{46}$$

Let $\boldsymbol{\Xi}(\boldsymbol{\zeta}) = (\boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C))$ denote a vector of evaluations of the model at inputs $\boldsymbol{\zeta}^O$ with uncensored outputs and inputs $\boldsymbol{\zeta}^C$ with censored outputs. Computer model outputs $\boldsymbol{\Xi}(\boldsymbol{\zeta}^O)$ is given to us, but $\boldsymbol{\Xi}(\boldsymbol{\zeta}^C)$ are unknown. However, it is known that

$$\boldsymbol{\Xi}(\boldsymbol{\zeta}^C) \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a} \sim \mathcal{TN}_{(-\infty,a)}\left(\boldsymbol{\mu}^*(\boldsymbol{\zeta}^C), \boldsymbol{\sigma}^{*2}(\boldsymbol{\zeta}^C, \boldsymbol{\zeta}^C)\right). \tag{47}$$

Distribution of interest is not closed-form, but a numerical approximation may be obtained. Computationally, in order to account for the censored observations $\boldsymbol{\zeta}^C$, samples from this truncated normal distribution (47) of $\boldsymbol{\Xi}(\boldsymbol{\zeta}^C) \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}$ may be obtained. Say, k samples from the distribution of vector $\{\boldsymbol{\Xi}(\boldsymbol{\zeta}^C)_i\}_{i=1}^k$ are available. For each sample a vector $\boldsymbol{\Xi}(\boldsymbol{\zeta})_i = (\boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C)_i)$ is used to sample from (46), i.e. distribution $\Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta})_i$ which is represented by

$$\Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta})_i \sim \mathcal{N}(\mu^*(\boldsymbol{\zeta}'), \sigma^{*2}(\boldsymbol{\zeta}', \boldsymbol{\zeta}')). \tag{48}$$

Finally, latent marginal distribution $\Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}) = \Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}_a(\boldsymbol{\zeta}^C) = \Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}$ is

$$p(\Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}) = \int \cdots \int p(\Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C)) p(\boldsymbol{\Xi}(\boldsymbol{\zeta}^C) \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}) \, \mathrm{d}\boldsymbol{\Xi}(\boldsymbol{\zeta}^C). \tag{49}$$

Predictive distribution of the emulator $\Xi_a(\cdot)$ at a new input to the computer model $\boldsymbol{\zeta}'$ consists of two parts: a point mass at a and a Lebesgue measure on $\mathbb{R}_{>a}$. Namely, the distribution is

$$\Xi_a(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a} = \begin{cases} \Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}, & \Xi_a(\boldsymbol{\zeta}') > a \\ \int_{-\infty}^{a} p(\Xi(\boldsymbol{\zeta}') \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}) \, \mathrm{d}\Xi(\boldsymbol{\zeta}'), & \Xi_a(\boldsymbol{\zeta}') = a \,. \end{cases} \tag{50}$$

Theorem. $\Xi_a(\cdot) \mid \boldsymbol{\Xi}(\boldsymbol{\zeta}^O), \boldsymbol{\Xi}(\boldsymbol{\zeta}^C) < \mathbf{a}$ is a stochastic process.

Construction of an adequate emulator is essential for subsequent proper use of such an emulator for decision and policy making. If the purpose is not only to construct an emulator as an approximation to the simulator, but to use this emulator for providing adequate uncertainty estimates and estimation of a probability of an event, e.g., a natural hazard from a volcano pyroclastic flow, then it is preferred to use a valid approximation, which in this case is given by a censored GASP.

One illusive limitation of the proposed framework is that a constructed latent process may have to perform extrapolation in a region with no positive output values. However, the "censoring" operation performs conditioning of a latent process on the zero-output. This is because of this conditioning that censored emulator still performs an interpolation rather than extrapolation.

Chapter 6. Design of experiments for a large-scale computer model

A simulator is defined as LARGE-SCALE *if the number of inputs is such that construction of its emulator (which involves optimization over its parameters) is prohibitively time-consuming. In order to facilitate the exploration of such a simulator, useful is to divide inputs* $\mathbf{x}$ *into two groups, that is* $\mathbf{x} = (\boldsymbol{\kappa}, \boldsymbol{\lambda})$*. First, design over the range of* $\boldsymbol{\kappa}$*, choosing, say,* m *points* $\{\boldsymbol{\kappa}_i\}_{i=1}^m$*. For each* $\boldsymbol{\kappa}_i$ *develop a Gaussian process emulator over the rest of inputs* $\boldsymbol{\lambda}$*. Second, for each set of* fixed *inputs* $\boldsymbol{\lambda}$*, an emulator over* $\boldsymbol{\kappa}$ *conditional on fixed* $\boldsymbol{\lambda}$ *is constructed.*

This methodology may be used for facilitating parameter estimation and fast emulation of a model with many inputs. Depending on the purpose and implementation of this methodology in practice, but the Gaussian process over the entire input space may be lost, although useful approximations are constructed.

My biography

This monograph is a follow-up after my dissertation "On Uncertainty Quantification for Systems of Computer Models" with which I completed philosophy doctorate *(PhD) in Statistical Science at Duke University, USA in 2017. In my dissertation I have developed and analyzed a fully probabilistic Bayesian framework for testing theoretical scientific models, often realized as scientific computer models, with respect to experimental data.*

Bibliography

[1] James Berger, 2001, Objective Bayesian analysis for spatially correlated data, *Journal of the American Statistical Association* **96**(456), 1361-1374.

[2] Mengyang Gu, 2018, Robust Gaussian stochastic process emulation, *The Annals of Statistics* **46**, 3038-3066.

[3] Ksenia Kyzyurova, 2017, On Uncertainty Quantification for Systems of Computer Models, *Philosophy doctorate thesis*, Duke University.

[4] Ksenia Kyzyurova, 2018, Coupling computer models through linking their statistical emulators, *SIAM/ASA Journal on Uncertainty Quantification* **6**(3), 1151-1171.

[5] Dennis Lindley and Adrian Smith, 1972, Bayes estimates for the linear model, *Journal of the Royal Statistical Society: Series B (Methodological)* **34**(1),1-18.