Calibration of computer models

A Closer Look at the Discrepancy Function

Pierre BARBILLON

Fall 2023

Outline

Outline

Validation in [Bayarri et al., 2007] by examinning the Discrepancy.
Estimate tolerance bounds by computing: For a fixed level γ, the tolerance bounds $\tau = \tau(\mathbf{x})$ are then computed such that $\gamma \cdot 100\%$ of the samples satisfy:

- for pure simulator predictions $\left|\hat{f}(\mathbf{x}_{new}, \hat{\boldsymbol{\theta}}) - \zeta^{(i)}(\mathbf{x}_{new})\right| < \tau$
- for bias-corrected predictions $\left|\hat{\zeta}(\mathbf{x}_{new}) - \zeta^{(i)}(\mathbf{x}_{new})\right| < \tau$

where

- $\hat{f}(\mathbf{x}_{new}, \hat{\boldsymbol{\theta}}) = m_D(\mathbf{x}_{new}, \hat{\boldsymbol{\theta}})$ where $\hat{\boldsymbol{\theta}}$ may refer to the posterior mean,
- $\zeta^{(i)}(\mathbf{x}_{new}) = F^{(i)}(\mathbf{x}_{new}, \boldsymbol{\theta}^{(i)}) + \delta^{(i)}(\mathbf{x}_{new})$,
- $F^{(i)}(\mathbf{x}_{new}, \boldsymbol{\theta}^{(i)})$ and $\delta^{(i)}(\mathbf{x}_{new})$ $(1 \leq i \leq N)$ are obtained by an MCMC algorithm sampling from the joint posterior predictive distribution,
- $\hat{\zeta}(\mathbf{x}_{new}) = \frac{1}{N}\sum_{i=1}^{N}\left(F^{(i)}(\mathbf{x}_{new}, \boldsymbol{\theta}^{(i)}) + \delta^{(i)}(\mathbf{x}_{new})\right)$.

Note that the discrepancy can be estimated from the posterior model and discrepancy sampling on a set of new locations: $\mathbf{X}_{new}$:

$$\hat{\delta}_{\hat{\theta}} = \hat{\zeta}_{new} - \hat{f}(\mathbf{x}_{new}, \hat{\boldsymbol{\theta}}).$$

Outline

L_2 Calibration

Defined in [Tuo and Wu, 2016]:

$$\boldsymbol{\theta}_{L_2} = \underset{\Theta}{\operatorname{argmin}} \|\delta_{\boldsymbol{\theta}}(\cdot)\|_{L_2(\mathcal{X})} = \underset{\Theta}{\operatorname{argmin}} \left(\int_{\mathcal{X}} (\zeta(\mathbf{x}) - f(\mathbf{x}, \boldsymbol{\theta}))^2 d\mathbf{x} \right)^{1/2}.$$

[Tuo et al., 2015]proposes to first obtain an estimate $\hat{\zeta}$ of the reality ζ via a Gaussian stochastic stochastic process and then plug it into the minimization problem to get $\hat{\boldsymbol{\theta}}_{L_2}$.
Consistent estimation $\hat{\boldsymbol{\theta}}_{L_2} \to \boldsymbol{\theta}_{L_2}$ provided that $\hat{\zeta}$ is good approximation.
An alternative least square :

$$\hat{\boldsymbol{\theta}}_{LS} = \underset{\Theta}{\operatorname{argmin}} \left(\sum_{i=1}^{n_e} (y_i^e - f(\mathbf{x}^e, \boldsymbol{\theta}))^2 \right)$$

[Tuo et al., 2015] proves the convergence $\hat{\boldsymbol{\theta}}_{LS} \to \boldsymbol{\theta}_{L_2}$
[Wong et al., 2017] uses LS calibration as a plug-in estimator for estimating the discrepancy function via a nonparametric regression

Scaled Gaussian Process

[Gu and Wang, 2018]

$$
\begin{aligned}
y_i^e =& f(\mathbf{x}_i^e, \boldsymbol{\theta}) + \mu^\delta(\mathbf{x}_i^e) + \delta_z(\mathbf{x}_i^e) + \epsilon_i \\
\mu^\delta(\mathbf{x}) =& \sum_{i=1}^{q} h(\mathbf{x})\beta_i \\
\delta_z(\cdot) \sim& GP(0, \sigma_\delta^2 c_\delta(\cdot,\cdot)) \text{ s.t. } \int_{\mathcal{X}} \delta_z(\mathbf{x})^2 d\mathbf{x} = Z \\
Z \sim& p_{\delta_z}(\cdot), \quad p_{\delta_z}(z) \propto f_z(Z = z|\lambda) \cdot p_\delta(z|\boldsymbol{\theta}, \Psi)
\end{aligned}
$$

where $p_\delta(z|\boldsymbol{\theta}, \Psi)$ is the implicit prior on Z for a GP on the discrepancy.
Then if f_z constant $\Rightarrow$ Model is equivalent to KOH model.

Comparison GP with SGP

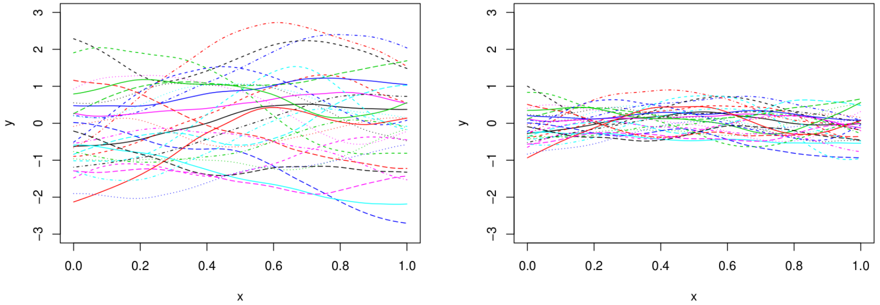


Figure 2. *Fifty samples from the GaSP and discretized S-GaSP are graphed in the left and right panels, respectively, where x_i is equally spaced in $[0, 1]$. For both processes, we let $\mu^\delta = 0$, $\sigma_\delta^2 = 1$ and $\gamma^\delta = 1/2$. In the discretized S-GaSP, $\mathbf{x}_i^C = \mathbf{x}_i$ for $i = 1, ..., N_C$, $N_C = n$ and $\lambda = n/2$ are assumed..*

from [Gu and Wang, 2018].

Outline

Outline

Model Comparison

[Damblin et al., 2016]

- $\mathcal{H}_0 : \zeta(\cdot) = f(\cdot, \boldsymbol{\theta}^*)$ for a "true" $\boldsymbol{\theta}^*$:

$$y_i = f(\mathbf{x}_i, \boldsymbol{\theta}^*) + \epsilon_i^0 \,,$$

where $\epsilon_i^0 \overset{iid}{\sim} \mathcal{N}(0, \lambda_0^2)$.

- $\mathcal{H}_1$: Code discrepancy term $\delta(\mathbf{x})$ s.t. $\zeta(\mathbf{x}) = f(\mathbf{x}, \boldsymbol{\theta}^*) + \delta(\mathbf{x})$:

$$y_i = f(\mathbf{x}_i, \boldsymbol{\theta}^*) + \delta(\mathbf{x}_i) + \epsilon_i^1 \qquad \text{where} \quad \delta(.) \sim \mathcal{GP}(0, \sigma_\delta^2 \Sigma_\psi(.,.)) \quad \text{and} \quad \epsilon_i^1 \overset{iid}{\sim} \mathcal{N}(0, \lambda_1^2)$$

Bayes Factor

$$B_{0,1}(\mathbf{y}^e) := \frac{p(\mathbf{y}^e|\mathcal{H}_0)}{p(\mathbf{y}^e|\mathcal{H}_1)} \quad \text{where} \quad p(\boldsymbol{z}|\mathcal{H}_j) = \int_{\boldsymbol{p}_j} p(\mathbf{y}^e|\boldsymbol{p}_j, \mathcal{H}_j)\pi(\boldsymbol{p}_j) d\boldsymbol{p}_j \,.$$

Intrinsic Bayes Factor

[Berger and Pericchi, 1996]

Main issue: Evidence $p(\mathbf{y}^e|\mathcal{H}_j)$ sensitive to priors $\pi(\boldsymbol{p}_j)$.

- Need to use compatible priors [Celeux et al., 2006] or objective priors [Casella and Moreno, 2006],
- but marginal likelihood ill-defined (up to arbitrary constant) for improper priors (as objective priors often are).

Idea: using a part of data to obtain a proper prior:

- Partial Bayes Factor:

$$B_{0,1}(\mathbf{y}^e(-m)|\mathbf{y}^e(m)) = \frac{\int l(\boldsymbol{p}_0;\mathbf{y}^e(-m)|\mathbf{y}^e(m))\pi(\boldsymbol{p}_0|\mathbf{y}^e(m))d\boldsymbol{p}_0}{\int l(\boldsymbol{p}_1;\mathbf{y}^e(-m)|\mathbf{y}^e(m))\pi(\boldsymbol{p}_1|\mathbf{y}^e(m))d\boldsymbol{p}_1} = \frac{B_{0,1}(\mathbf{y}^e)}{B_{0,1}(\mathbf{y}^e(m))}\,.$$

- $B_{0,1}(\mathbf{y}^e(-m)|\mathbf{y}^e(m))$ well-defined for $|m| \geq n_0$ large enough:
- Intrinsinc Bayes factor obtained by averaging over all $\mathbf{y}^e(m)$s :

$$B^A_{0,1}(\mathbf{y}^e) = \frac{B_{0,1}(\boldsymbol{z})}{C(n,n_0)} \sum_{|m|=n_0} B_{0,1}(\mathbf{y}^e(m))^{-1}\,.$$

IBF computation under linearization of the code

Linear assumption: $f(\boldsymbol{x}, \boldsymbol{\theta}) = g(\boldsymbol{x})^\top \boldsymbol{\theta}$, with $g(\boldsymbol{x}) \in \mathbb{R}^d$.

Prior choices and consequences:

- Model $\mathcal{H}_0$ boils down to:

$$\mathcal{H}_0 : \mathbf{y}^e \quad \sim \quad \mathcal{N}(G\boldsymbol{\theta}_0; \lambda_0^2 \mathrm{I}_{\mathrm{n_e}}); \quad \boldsymbol{p}_0 = (\boldsymbol{\theta}_0, \lambda_0^2)$$

  where $G = [g(\mathbf{x}_1^e), \cdots, g(\mathbf{x}_{n_e}^e)]^\top$ the $n_e \times p$ design matrix.
  $\hookrightarrow$ Under Jeffreys prior: $\pi(\boldsymbol{p}_0) \propto \lambda_0^{-2}$, $p(\mathbf{y}^e | \mathcal{H}_0)$ explicit.

- Model $\mathcal{H}_1$ boils down to:

$$\mathcal{H}_1 : \mathbf{y}^e \quad \sim \quad \mathcal{N}(G\boldsymbol{\theta}_1; \sigma_\delta^2 V_{k,\psi}); \quad \boldsymbol{p}_1 = (\boldsymbol{\theta}_1, \sigma_\delta^2, \psi, k)$$
$$V_{k,\psi}(i,j) = k\delta_{i,j} + e^{-||\boldsymbol{x}_i - \boldsymbol{x}_j||^2/\psi^2} \quad k = \lambda_1^2 \sigma^{-2}.$$

  - Prior choice: $\pi(\boldsymbol{p}_1) \propto \pi(\psi | k)\pi(k)\sigma^{-2}$ with proper priors for $\pi(\psi|k)\pi(k)$,
  - Integration of $p(\mathbf{y}^e | \boldsymbol{p}_1, \mathcal{H}_1)$: explicit over $(\theta_1, \sigma_\delta^2)$, by Gaussian quadrature over (ψ, k).

Computation of the IBF

Proposition

If $\pi(\boldsymbol{p}_1) = \pi(\theta_1, \sigma_\delta^2, \psi, k) = \pi(\psi|k)\pi(k)/\sigma_\delta^2$, $\pi(\psi, k)$ is proper and $m = d + 1$ then

$$B^A_{0,1}(\mathbf{y}^e) = \frac{B_{0,1}(\boldsymbol{z})}{C(n, n_0)} \sum_{|m|=n_0} B_{0,1}(\mathbf{y}^e(m))^{-1} = B_{0,1}(\mathbf{y}^e)$$

In the following,

$$\begin{aligned} \pi(\psi|k) &= \mathcal{U}([0,1])\,, \\ \pi(k) &= Be(1,3)\,. \end{aligned}$$

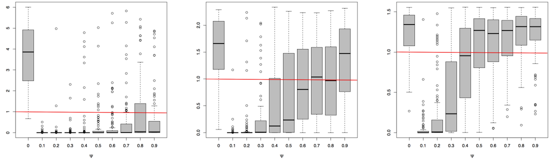
Synthetic data

Data simulated according to model $\mathcal{H}_1$, with $\delta \sim GP(0, \sigma_\delta^2 \Sigma_\psi)$:

$$\mathbf{x}_i^e = \left(\frac{i}{n_e}\right)_{1 \le i \le n}, \quad n_e = 30, \quad \sigma_\delta^2 = 0.1, \quad k = 0.1\,.$$

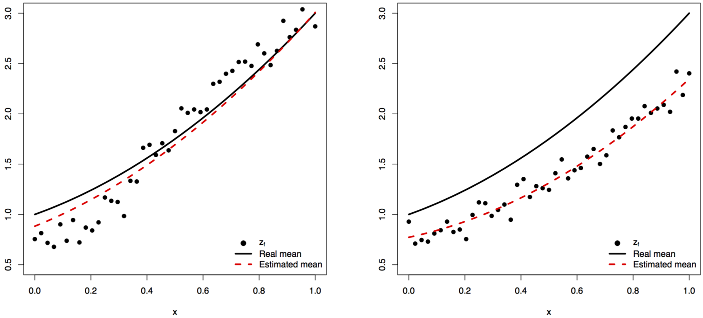
From left to right

- constant trend $g(\mathbf{x}) = 1$; $\theta_1 = 1$,
- linear trend $g(\mathbf{x}) = (1, x)$; $\boldsymbol{\theta}_1 = (1, 1)$,
- quadratic trend $g(\mathbf{x}) = (1, x, x^2)$; $\boldsymbol{\theta}_1 = (1, 1, 1)$.
- Bayes factor $B_{0,1}^A$ expected to decrease with ψ.



Boxplots of $B_{0,1}^A(\mathbf{y}^e)$ values over 100 simulations with constant, linear and quadratic trends (left to right)

Confounding Effect



$\psi = 0.2$ left and $\psi = 0.7$ right

- ψ, k, σ_δ^2 estimated by maximum likelihood.
- For $\psi = 0.7$, discrepancy indistinguishable from quadratic trend!

Case description

- Industrial computer code predicting the **productivity** of an electric power plant, based on measurements (temperature, pressure, discharge, ...) throughout the plant
- $n = 24$ available field measures (results of periodic testing) to validate code

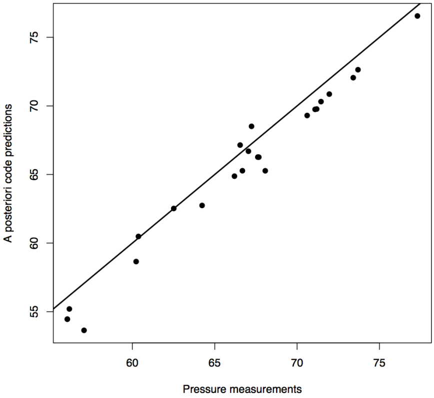
Main code features:

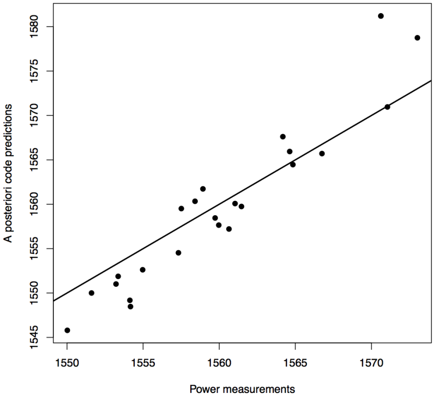
- $p = 20$ input variables ($\boldsymbol{x} \in \mathbb{R}^{20}$)
- $d = 2$ parameters: heat transfer coefficient of the condenser, yield of the main turbine 2 .
- Two outputs of interest (electric power, condenser pressure), seen here as two separate codes
- Code **linearized** in neighbourhood of reference value $\theta^\star$:

$$f(\boldsymbol{x}_i, \boldsymbol{\theta}) \approx f(\boldsymbol{x}_i, \boldsymbol{\theta}^\star) + h(\boldsymbol{x}_i)^\top (\boldsymbol{\theta} - \boldsymbol{\theta}^\star),$$

where $h(\boldsymbol{x}_i) = \nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}_i, \boldsymbol{\theta}^\star)$ evaluated numerically through finite difference

Calibrated code predictions vs measures





Pressure
$B^A_{0,1} = 2 \times 10^{-18}$

Power
$B^A_{0,1} = 3 \times 10^{-3}$

- Bias reduced by calibration, but not supressed
- strong evidence for code discrepancy

Outline

Model selection as a mixture problem

[Kamary et al., 2019] inspired by [Kamary et al., 2014].
Model selection problem:

$$\mathfrak{M}_0 : y_i = f(\mathbf{x}_i, \boldsymbol{\theta}_0) + \epsilon_i^0$$
$$\mathfrak{M}_1 : y_i = f(\mathbf{x}_i, \boldsymbol{\theta}_1) + \delta(\mathbf{x}_i) + \epsilon_i^1 .$$

where $\epsilon_i^0 \overset{iid}{\sim} \mathcal{N}(0, \lambda_0^2)$ and $\epsilon_i^1 \overset{iid}{\sim} \mathcal{N}(0, \lambda_1^2)$

converted into a mixture model:

$$\mathfrak{M}_\alpha : y_i \sim \alpha \left(\ell_{\mathfrak{M}_0}(\boldsymbol{\theta}_0, \lambda_0^2; y_i, \mathbf{x}_i)\right) + (1-\alpha) \left(\ell_{\mathfrak{M}_1}(\boldsymbol{\theta}_1, \lambda_1^2, \delta; y_i, \mathbf{x}_i)\right) .$$

- Model $\mathfrak{M}_\alpha$ is defined under the hypothesis that the likelihood of the model $\mathfrak{M}_1$ is conditioned on δ.
- δ is considered as a parameter of $\mathfrak{M}_1$.
- Conditionnally on δ, the y_i's are considered independent.
- Posterior distribution on α will provide a decision rule for $\mathfrak{M}_0$ against $\mathfrak{M}_1$.

Hypotheses and prior distribution

- Linear code: $f(\mathbf{x}, \boldsymbol{\theta}) = g(\mathbf{x})^\top \boldsymbol{\theta}$.
- GP prior for discrepancy function:

$$\delta(.) \sim \mathcal{GP}(0, \sigma_\delta^2 \Sigma_\psi(.,.)) .$$

- Some parameters are common, $\boldsymbol{\theta}$ and λ^2 so a common prior distribution is chosen for both.

$$\mathfrak{M}_\alpha : y_i \sim \alpha \left(\ell_{\mathfrak{M}_0}(\boldsymbol{\theta}, \lambda^2; y_i, \mathbf{x}_i) \right) + (1-\alpha) \left(\ell_{\mathfrak{M}_1}(\boldsymbol{\theta}, \lambda^2, \delta; y_i, \mathbf{x}_i) \right) .$$

Posterior distribution

Theorem

Let $g : \mathbb{R}^p \to \mathbb{R}^d$ be a finite-valued function and vector $\mathbf{x}_1^e, \ldots, \mathbf{x}_n^e$ such that the rank of $\{g(\mathbf{x}_1^e), \ldots, g(\mathbf{x}_n^e)\}$ is d. The posterior distribution associated with the prior $\pi(\boldsymbol{\theta}, \lambda^2) = {}^1/_{\lambda^2}$ and with the likelihood is proper when

- for any $0 < k < 1$, the hyperparameter σ_δ^2 of the discrepancy prior distribution is reparameterized as $\sigma_\delta^2 = {}^{\lambda^2}/_k$ and so $\Sigma_\psi = ({}^{\lambda^2}/_k)\mathrm{Corr}_{\psi_\delta}$ when $\mathrm{Corr}_{\psi_\delta}$ is the correlation function of δ.
- the mixture weight α has a proper beta prior $\mathcal{B}(a_0, a_0)$;
- ψ_δ has a proper Beta prior $\mathcal{B}(b_1, b_2)$.
- proper distribution is used on k.

Metropolis within Gibbs

Algorithm 1: Metropolis-within-Gibbs algorithm

for t=1,...,T **do**

a) $\delta^{(t)}$ is sampled from $\pi(\delta|\boldsymbol{y}, \mathbf{x}, \boldsymbol{\theta}^{(t-1)}, \lambda^{(t-1)}, k^{(t-1)}, \psi_\delta^{(t-1)}, \alpha^{(t-1)})$ as follows.

a.1) For $i = 1, \ldots, n$; $j = 0, 1$, generate auxiliaire variable $\zeta_i^{(t)}$ from

$$\mathbb{P}(\zeta_i = j|y_i, x_i, \delta^{(t-1)}, \boldsymbol{\theta}^{(t-1)}, \lambda^{(t-1)}, k^{(t-1)}, \psi_\delta^{(t-1)}) .$$

a.2) Generate $\delta^{(t)}$ according to the conditional posterior distribution

$$\delta^{(t)}|\boldsymbol{y}, \mathbf{x}, \zeta^{(t)} = 1, \boldsymbol{\theta}^{(t-1)}, \lambda^{(t-1)}, k^{(t-1)}, \psi_\delta^{(t-1)}, \alpha^{(t-1)} \sim \mathcal{N}_n(\hat{\mu}_\delta, \hat{\Sigma}_\delta) .$$

b) Generate $\boldsymbol{\theta}^{(t)}|\boldsymbol{y}, \mathbf{x}, \boldsymbol{\zeta}^{(t)}, \delta^{(t)}, \lambda^{(t-1)}, k^{(t-1)}, \alpha^{(t-1)} \sim \mathcal{N}_d(\hat{\mu}_{\boldsymbol{\theta}}, \hat{\Sigma}_{\boldsymbol{\theta}})$.

c) Generate $\lambda^{(t)}|\boldsymbol{y}, \mathbf{x}, \boldsymbol{\zeta}^{(t)}, \delta^{(t-1)}, \boldsymbol{\theta}^{(t)}, k^{(t-1)}, \alpha^{(t-1)} \sim \mathcal{IG}(\hat{a}_\lambda, \hat{b}_\lambda)$.

d) Generate $\alpha^{(t)}|\boldsymbol{y}, \mathbf{x}, \boldsymbol{\zeta}^{(t)}, \delta^{(t)}, \boldsymbol{\theta}^{(t)}, \lambda^{(t)}, k^{(t-1)} \sim \mathcal{B}eta(n - m + a_0, m + a_0)$.

e) Generate $k^{(t)}$ from a random walk Metropolis-Hastings algorithm conditionally to $(\boldsymbol{y}, \mathbf{x}, \boldsymbol{\zeta}^{(t)}, \delta^{(t)}, \boldsymbol{\theta}^{(t)}, \lambda^{(t)}, \alpha^{(t)}, \psi_\delta^{(t-1)})$.

f) Generate $\psi_\delta^{(t)}$ from a random walk Metropolis-Hastings algorithm conditionally to $(\boldsymbol{y}, \mathbf{x}, \boldsymbol{\zeta}^{(t)}, \delta^{(t)}, \boldsymbol{\theta}^{(t)}, \lambda^{(t)}, \alpha^{(t)}, k^{(t)})$.

Synthetic example $\mathfrak{M}_0$

Code is a quadratic function.
50 datasets of size $n = 30$ from $\mathfrak{M}_0 : y_i = g(x)^\top \boldsymbol{\theta}^* + \epsilon_i$.
Priors as in the theorem, $\alpha \sim \mathcal{B}eta(1, 1)$, $\delta \sim \mathcal{GP}(0_n, \Sigma_\psi)$, $\psi_\delta \sim \mathcal{B}eta(1, 1)$ and $k \sim \mathcal{B}eta(1, 1)$.
Number of MCMC iterations is 10^4 with a burn-in of 10^3 iterations

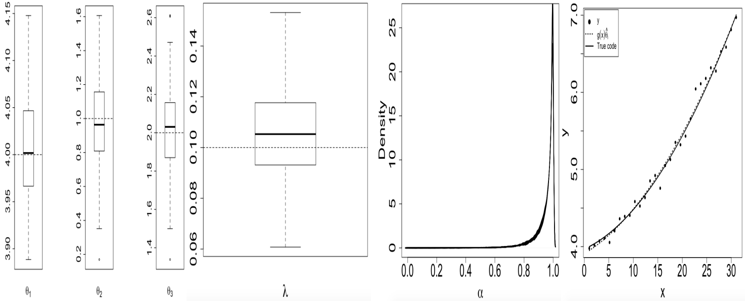


Figure: Posterior mean estimates of $\boldsymbol{\theta}, \lambda^2$, Posterior densities of α, Posterior prediction of the code.

Synthetic example $\mathfrak{M}_1$

Code is a quadratic function.
50 samples of size 50 simulated from $\mathfrak{M}_1$ when ψ_δ^* varies between 0.01 and 0.9, $\delta^*(x) \sim \mathcal{GP}(0_n, \Sigma_\psi)$, $\lambda^2* = 0.1$ and $k^* = 0.1$.
Priors as in the theorem, $\alpha \sim \mathcal{B}eta(1,1)$, $\delta \sim \mathcal{GP}(0_n, \Sigma_\psi)$, $\psi_\delta \sim \mathcal{B}eta(1,1)$ and $k \sim \mathcal{B}eta(1,1)$.
Number of MCMC iterations is 10^4 with a burn-in of 10^3 iterations.

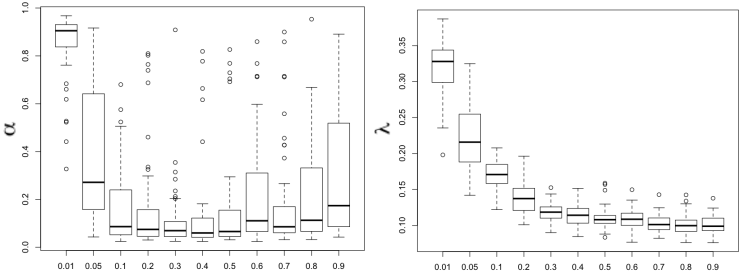


Figure: Posterior mean estimates for α and λ^2.

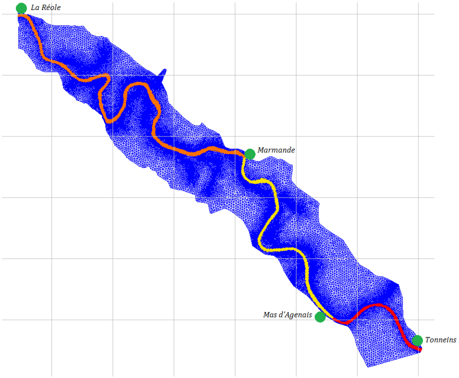
Hydraulic application: Garonne river

- TELEMAC 2D models the flow of the Garonne between Tonneins and la Réole:

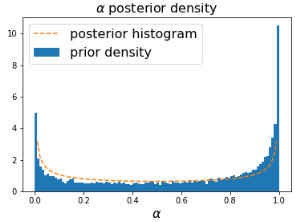
$$h_i = f(q_i, \mathbf{K_s}),$$

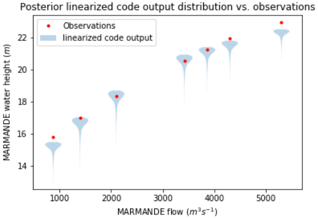
with:

- $\mathbf{h}_i$ water heights,
- $\mathbf{K_s} = (K_{s1}, \ldots, K_{s5})$ Strickler coefficients (5 friction coefficients)
- q_i river flow at Tonneins
- Linearization of the model around a reference value for the Strickler coefficient (limited to the most inflential ones).
- Only 7 data points available.



Results





Observation nb.	1	2	3	4	5	6	7
Bias probability	0.513	0.473	0.452	0.448	0.451	0.472	0.514

Table: Probability of a code bias for each observation in Marmande

Outline

Variable selection in the discrepancy function

[Barbillon et al., 2021]

Barbillon, P., Forte, A., and Paulo, R. (2021).
Screening the discrepancy function of a computer model.
arXiv preprint arXiv:2109.02726.

Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C.-H., and Tu, J. (2007).
A framework for validation of computer models.
Technometrics, 49(2):138–154.

Berger, J. O. and Pericchi, L. R. (1996).
The intrinsic bayes factor for model selection and prediction.
Journal of the American Statistical Association, 91(433):109–122.

Casella, G. and Moreno, E. (2006).
Objective Bayesian variable selection.
Journal of the American Statistical Association, 101(473):157–167.

Celeux, G., Marin, J.-M., and Robert, C. (2006).
Sélection bayésienne de variables en régression linéaire.
Journal de la société française de statistique, 147(1):59–79.

Damblin, G., Keller, M., Barbillon, P., Pasanisi, A., and Parent, É. (2016).
Bayesian model selection for the validation of computer codes.
Quality and Reliability Engineering International, 32(6):2043–2054.

Gu, M. and Wang, L. (2018).
Scaled gaussian stochastic process for computer model calibration and prediction.
SIAM/ASA Journal on Uncertainty Quantification, 6(4):1555–1583.

Kamary, K., Keller, M., Barbillon, P., Goeury, C., and Éric Parent (2019).
Computer code validation via mixture model estimation.

Kamary, K., Mengersen, K., Robert, C. P., and Rousseau, J. (2014).
Testing hypotheses via a mixture estimation model.
arXiv preprint arXiv:1412.2044.

Tuo, R., Wu, C. J., et al. (2015).
Efficient calibration for imperfect computer models.
The Annals of Statistics, 43(6):2331–2352.

Tuo, R. and Wu, J. (2016).
A theoretical framework for calibration in computer models: parametrization, estimation and convergence properties.
SIAM/ASA Journal on Uncertainty Quantification, 4(1):767–795.

Wong, R. K., Storlie, C. B., and Lee, T. (2017).
A frequentist approach to computer model calibration.

Journal of the Royal Statistical Society: Series B (Statistical Methodology), 79(2):635–648.