Bayesian model selection for the validation of computer codes

Guillaume Damblin[1,2,3], Merlin Keller[1], Pierre Barbillon[2], Alberto Pasanisi[1], Eric Parent[2]

[1]EDF R&D, [2]AgroParisTech, [3]CEA

ETICS – Ecole Thématique sur les Incertitudes en Calcul Scientifique 2016

Industrial context

- **Numerical simulations** more and more used to complement/replace **physical experiments**, when these are too **costly/dangerous**
- In industrial studies, simulations used for
  - design/optimization tasks
  - reliability assessment and risk analysis
- Can we be sure the computer code used for the simulation **'predicts well enough'** (and **in which sense ?**) the physical phenomena under study ?

$\hookrightarrow$ **Code Validation** answers these questions

What is Code Validation ?

Definition (Advanced Simulation and Computer Program)

'Code Validation provides assurance that the models in the codes produce mathematically correct answers and that the answers reflect physical reality'

Definition (American Institute of Aeronautics and Astronautics [AIAA, 1998])

'The process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model. (AIAA G-077-1998)'

Ok, but how is it done in practice ?

Notations

- Let $r(\mathbf{x}) \in \mathbb{R}$ be the physical **quantity of interest** with respect to a controlable input vector $\mathbf{x} \in \mathbb{R}^d$
- A **computer code** can be seen as a parametric function $y_{\boldsymbol{\theta}}(\mathbf{x})$
  - $\boldsymbol{\theta}$ vector of (unobservable) parameters to be tuned
- $\mathbf{z} = (z_1, \ldots, z_n)$ available field measurements :

$$\begin{aligned} z_i &= r(\mathbf{x}_i) + \varepsilon_i \\ \varepsilon_i &\overset{i.i.d}{\sim} \mathcal{N}(0, \lambda^2) \end{aligned} \quad (1)$$

- Based on $\mathbf{z}$, the following tasks are considered :
  - **Calibration** : Estimate code parameters $\boldsymbol{\theta}$
  - **Validation** : Predict $r(\mathbf{x})$ for $\mathbf{x} \notin \{\mathbf{x}_i\}_{1 \le i \le n}$ using $y_{\boldsymbol{\theta}}(\mathbf{x})$, and assess prediction uncertainty

Current approaches

[Loeppky et al., 2006, Cox et al., 2001]

Test of $\mathcal{H}_0 : r(\cdot) = y_{\boldsymbol{\theta}}(\cdot)$ for a known θ, and computation of confidence intervals for $r(\mathbf{x})$ assuming :

$$z_i = y_{\boldsymbol{\theta}}(\mathbf{x}_i) + \varepsilon_i \tag{2}$$

[Kennedy and O'Hagan, 2001, Bayarri et al., 2007]

Code discrepancy term $b(\mathbf{x})$ s.t. $\mathcal{H}_1 : r(\mathbf{x}) = y_{\boldsymbol{\theta}}(\mathbf{x}) + b(\mathbf{x})$, $(\theta, b(\mathbf{x}))$ estimates and credible intervals for $r(\mathbf{x})$ assuming :

$$\begin{aligned} z_i &= y_{\boldsymbol{\theta}}(\mathbf{x}_i) + b(\mathbf{x}_i) + \varepsilon_i \\ b(.) &\sim \mathcal{GP}(0, \sigma_b^2 \Sigma_{\boldsymbol{\psi}_b}(.,.)) \end{aligned} \tag{3}$$

Limits of current approaches

Code discrepancy : pros and cons

- [Bayarri et al., 2007] justify introducing $b(\mathbf{x})$ by arguing that :
  - it **prevents over-fitting** of $\boldsymbol{\theta}$
  - it **ameliorates predictions** of $r(\mathbf{x})$
- However this term remains controversial, due to :
  - **No formal justification** for the presence of $b(\mathbf{x})$ in either [Kennedy and O'Hagan, 2001] or [Bayarri et al., 2007]
  - **Identifiability issues :** $(\boldsymbol{\theta}, b(\mathbf{x}))$ and $(\tilde{\boldsymbol{\theta}}, \tilde{b}(\mathbf{x}))$ equally likely as soon as : $b(\mathbf{x}) + y_{\boldsymbol{\theta}}(\mathbf{x}) = \tilde{b}(\mathbf{x}) + y_{\tilde{\boldsymbol{\theta}}}(\mathbf{x}) := \hat{r}(\mathbf{x})$
  - **Computational/interpretational complexity**

$\hookrightarrow$ Need to **develop a formal test** of the existence of $b(\mathbf{x})$ before dealing with it !

Bayesian Model Selection

In a Bayesian framework, $\mathcal{H}_0 : b(\cdot) = 0$ and $\mathcal{H}_1 : b(\cdot) \neq 0$ are compared through their **posterior odds** :

$$\frac{P(\mathcal{H}_0|\mathbf{z})}{P(\mathcal{H}_1|\mathbf{z})} = \frac{P(\mathcal{H}_0)}{P(\mathcal{H}_1)} \times \frac{p(\mathbf{z}|\mathcal{H}_0)}{p(\mathbf{z}|\mathcal{H}_1)}$$

where

- **Prior odds** chosen equal to one : $\frac{P(\mathcal{H}_0)}{P(\mathcal{H}_1)} = 1$
- $p(\mathbf{z}|\mathcal{H}_j) = \int_{\mathbf{p}_j} p(\mathbf{z}|\mathbf{p}_j, \mathcal{H}_j)\pi(\mathbf{p}_j)d\mathbf{p}_j$ **marginal likelihood** or **evidence** of model $\mathcal{H}_j$, with parameter vector $\mathbf{p}_j$
- $B_{0,1}(\mathbf{z}) := \frac{p(\mathbf{z}|\mathcal{H}_0)}{p(\mathbf{z}|\mathcal{H}_1)}$ is the **Bayes factor** for $\mathcal{H}_0$ over $\mathcal{H}_1$
- $B_{0,1}(\mathbf{z}) > 1$ indicates stronger evidence for $\mathcal{H}_0$ than for $\mathcal{H}_1$
- Bayesian equivalent of likelihood ratio test $\mathrm{lr}(\mathbf{z}) := \frac{\mathrm{p}(\mathbf{z}|\hat{\mathbf{p}}_0, \mathcal{H}_0)}{\mathrm{p}(\mathbf{z}|\hat{\mathbf{p}}_1, \mathcal{H}_1)}$

Intrinsinc Bayes factor [Berger and Pericchi, 1996]

Main issue : Evidence $p(\mathbf{z}|\mathcal{H}_j)$ **sensitive** to priors $\pi(\mathbf{p}_j)$

$\hookrightarrow$ 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 : Replace $\pi(\mathbf{p}_j)$ by **partial posterior** $\pi(\mathbf{p}_j|\mathbf{z}(m))$ given training set $\mathbf{z}(m) \subset \mathbf{z}$, yielding the **partial Bayes factor** :

$$B_{0,1}(\mathbf{z}(-m)|\mathbf{z}(m)) \quad = \quad \frac{B_{0,1}(\mathbf{z})}{B_{0,1}(\mathbf{z}(m))} \tag{4}$$

- $B_{0,1}(\mathbf{z}(-m)|\mathbf{z}(m))$ **well-defined** for $|m| \geq n_0$ large enough :

Intrinsinc Bayes factor obtained by averaging over all $\mathbf{z}(m)$s :

$$B^A_{0,1}(\mathbf{z}) \quad = \quad \frac{B_{0,1}(\mathbf{z})}{C(n, n_0)} \sum_{|m|=n_0} B_{0,1}(\mathbf{z}(m))^{-1} \tag{5}$$

Application to Linear Code Validation

Linear assumption : $y_{\boldsymbol{\theta}}(\mathbf{x}) = h(\mathbf{x})^\top \boldsymbol{\theta}$, with $h(\mathbf{x}) \in \mathbb{R}^p$

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

$$\mathcal{H}_0 : \mathbf{z} \sim \mathcal{N}(H\boldsymbol{\theta}_0; \lambda_0^2 \mathrm{I_n}); \quad \mathbf{p}_0 = (\boldsymbol{\theta}_0, \lambda_0^2)$$

$H = [h(\mathbf{x}_1), \cdots, h(\mathbf{x}_n)]^\top$ the $n \times p$ design matrix

$\hookrightarrow$ Under Jeffreys prior : $\pi(\mathbf{p}_0) \propto \lambda_0^{-2}$, $p(\mathbf{z}|\mathcal{H}_0)$ **explicit**

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

$$\mathcal{H}_1 : \mathbf{z} \sim \mathcal{N}(H\boldsymbol{\theta}_1; \sigma^2 V_{k,\Psi}); \quad \mathbf{p}_1 = (\boldsymbol{\theta}_1, \sigma^2, \Psi, k)$$

$V_{k,\Psi}(i,j) = k\delta_{i,j} + e^{-||\mathbf{x}_i - \mathbf{x}_j||^2/\psi^2}$; $k = \lambda_1^2 \sigma^{-2}$

  - Prior choice : $\pi(\mathbf{p}_1) \propto \pi(\Psi|k)\pi(k)\sigma^{-2}$ [Berger et al., 2011]
  - Integration of $p(\mathbf{z}|\mathbf{p}_1, \mathcal{H}_1)$: **explicit** over (θ_1, σ^2), by **gaussian quadrature** over (Ψ, k)

Prior over Gaussian process hyperparameters

Comparison of two proper conditional priors $\pi(\Psi|k)$ combined with $\pi(k) = \mathrm{Beta}(k; 1, 3)$, making $B^A_{0,1}(\mathbf{z})$ easy to compute due to :

Theorem ([Berger et al., 2011])

If $\pi(\Psi, k)$ is proper and $|m| = p + 1$, then $B_{0,1}(\mathbf{z}(m)) \propto 1$

- The uniform prior :

$$\pi(\Psi) \quad \propto \quad \mathbf{1}_{[\Psi_{\min}, \Psi_{\max}]}(\Psi)$$

- The reference prior [Berger et al., 2011] :

$$\pi(\Psi|k) \quad \propto \quad \left\{ \mathrm{tr}[W_\Psi^2] - \frac{1}{n}\mathrm{tr}[W_\Psi]^2 \right\}^{\frac{1}{2}},$$

with $W_\Psi = \frac{\partial V_{k,\Psi}}{\partial \Psi} V_{k,\Psi}^{-1} \left(I - H(HV_{k,\Psi}^{-1}H)^{-1}HV_{k,\Psi}^{-1} \right)$.

Data Simulation

Data simulated according to model $\mathcal{H}_1$, with :

$$\mathbf{x} = \left(\frac{i}{n}\right)_{1\leq i\leq n}, \quad n = 30, \quad \sigma^2 = 0.1, \quad k = 1$$

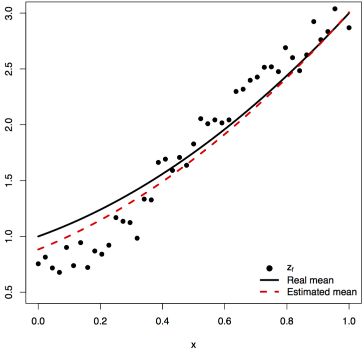
Three different linear models tested :

- constant trend $h(x) = 1$; $\theta_1 = 1$,
- linear trend $h(x) = (1, x)$; $\boldsymbol{\theta}_1 = (1, 1)$,
- quadratic trend $h(x) = (1, x, x^2)$; $\boldsymbol{\theta}_1 = (1, 1, 1)$.

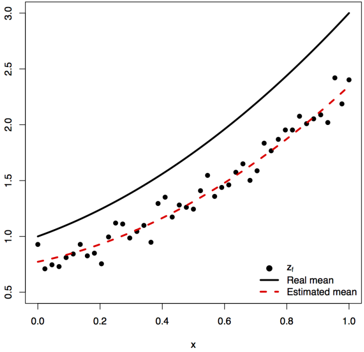
100 datasets simulated for each model & for Ψ varying in $[0, 1]$

- Bayes factor $B^A_{0,1}$ expected to **decrease** with Ψ

Confounding between trend and the discrepancy



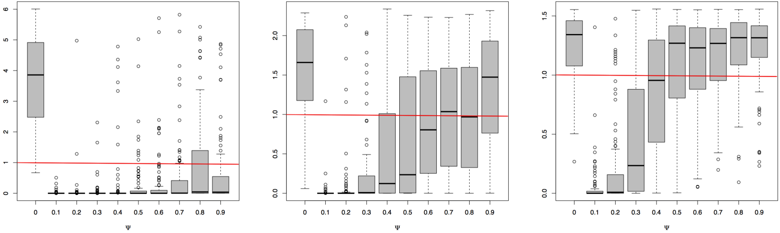
$\Psi = 0.2$



$\Psi = 0.7$

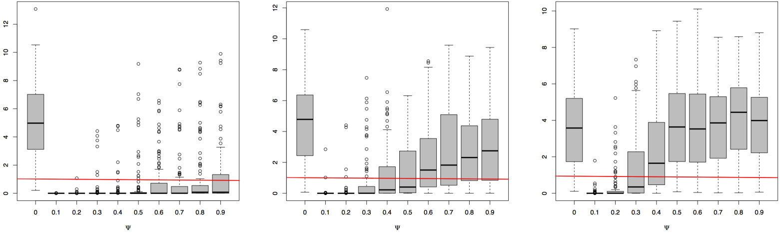
- Data simulated with quadratic trend and $\Psi \in \{0.2, 0.7\}$
- ψ, k, σ^2 estimated by maximum integrated likelihood
- For $\Psi = 0.7$, discrepancy undistinguishable from trend !

Intrinsinc Bayes factor under Uniform-Beta prior



- Boxplots of $B^A_{0,1}(\mathbf{z})$ values over 100 simulations with constant, linear and quadratic trends (left to right)
- $\mathcal{H}_0$ selected over $\mathcal{H}_1$ if $B^A_{0,1}(\mathbf{z}) > 1$
- Due to the confound between nonconstant trend and discrepancy, $\mathcal{H}_0$ and $\mathcal{H}_1$ hard to distinguish for $\psi > 0.3$

Intrinsinc Bayes factor under Reference-Beta prior



- Same remarks as for Uniform-Beta prior
- Higher values of $B^A_{0,1}(\mathbf{z})$ using Reference- rather than Uniform-Beta prior
- Reference prior seems to promote **conservatism** when searching for code discrepancy

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

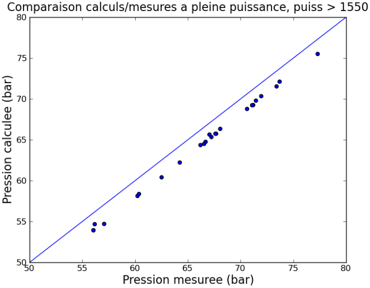
Main code features :

- $d = 20$ input variables ($\mathbf{x} \in \mathbb{R}^{20}$)
- Two outputs of interest (electric power, condenser pressure), seen here as two separate codes
- Code **linearized** in neighbourhood of reference value $\theta^\star$:

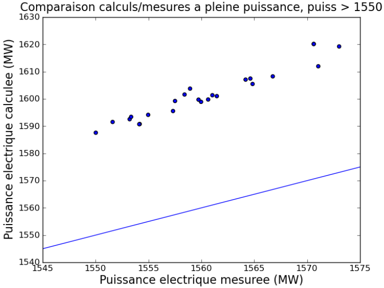
$$y^p_{\boldsymbol{\theta}}(\mathbf{x}_i) \approx y^p_{\boldsymbol{\theta}^\star}(\mathbf{x}_i) + h(\mathbf{x}_i)^\top(\boldsymbol{\theta} - \boldsymbol{\theta}^\star),$$

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

Reference code predictions vs measures



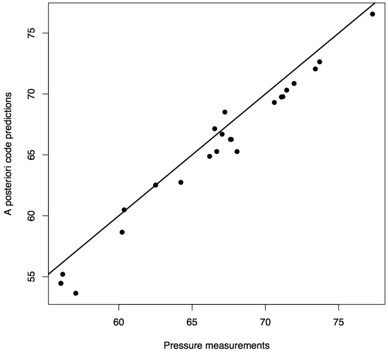
Pressure

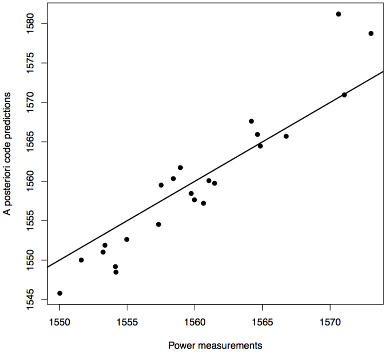


Power

- Systematic bias (constant discrepancy) in code predictions
- May be reduced by calibration

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

Conclusion & Perspectives

Bayesian code validation

- Assesses uncertainties associated to code predictions,
- Corrects code predictions for discrepancy **iff** needed.
- If evidence inconclusive, Bayesian model averaging can be used [Hoeting et al., 1999]

Future work

- Compute reference prior for k
- Relax assumption of code linearity
- Use kriging to emulate costly nonlinear computer codes
- Build physical/numerical DOEs for computer validation
- Efficient likelihood integration in higher dimensions

Short-term perspectives

- Work presented at ENBIS 2015, accepted in QREI

$\hookrightarrow$ EDF/CEA post-doc under project, due to start beg. 2017 (still looking for candidates ! !)

- Envisioned Industrial applications :
  - Power plant production control model (`DYMOLA`)
    - $\rightarrow$ high-dimensional (temporal) output
  - CANOPY joint welding model (`Code_Saturne`)
    - $\rightarrow$ costly ($> 1h$ per evaluation)
  - Hydraulic model of Garrone river (`TELEMAC-2D`)
    - $\rightarrow$ costly and high-dimensional (spatial) output

AIAA (1998).
Guide for the verification and validation of computational fluid dynamics simulations.
American Institute of Aeronautics and Astronautics.

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

Berger, J., De Oliveira, V., and Sanso, B. (2011).
Objective bayesian analysis of spatially correlated data.
Journal of the American Statistical Association, 96(456) :1361–1374.

Berger, J. and Pericchi, L. (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).
Selection bayésienne de variables en régression linéaire.
Journal de la Société Française de Statistique, 147(1) :59–79.

Cox, D., Park, J., and Clifford, E. (2001).
A statistical method for tuning a computer code to a data base.
Computational Statistics and Data Analysis, 37 :77–92.

Hoeting, J., Madigan, D., Raftery, A., and Volinsky, C. (1999).
Bayesian model averaging : A tutorial.
Statistical Science, 14(4) :382–417.

Kennedy, M. and O'Hagan, A. (2001).

Bayesian calibration of computer models.
Journal of the Royal Statistical Society, Series B, Methodological, 63 :425–464.

Loeppky, D., Bingham, D., and Welch, W. (2006).
Computer model calibration or tuning in practice.
Technical Report.