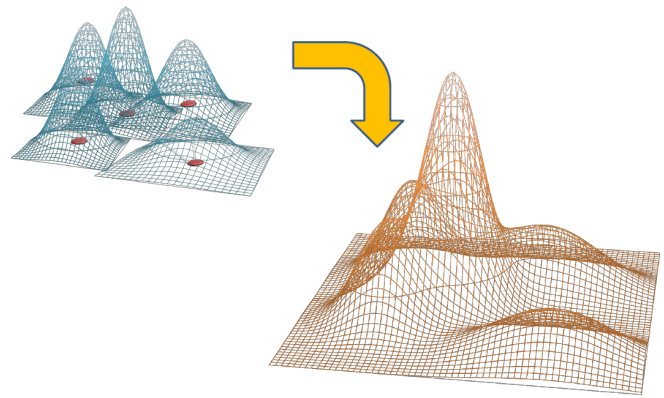
Bayesian model selection for computer model validation via mixture model estimation

Kaniav Kamary
ATER, CNAM

Joint work with É. Parent, P. Barbillon, M. Keller and N. Bousquet

Outline



What is a computer model validation?

Physical phenomena(real system):

- Question: how can we study a real phenomenon?
- What is a computer model?

What is a computer model validation?

$$\text{Physical phenomena(real system)} \xrightarrow[\text{simplify by physical lows}]{\text{Physicien}} \text{Physical model}$$

- Physicien use a part of reality, complicated
- then find a simple representation to which he applies a theory
- the simple representation help us to reproduce the functionality of the physical phenomena
- to explain, analyze and predict some aspects

What is a computer model validation?

$$\text{Physical phenomena(real system)} \xrightarrow[\text{simplify by physical lows}]{\text{Physicien}} \text{Physical model}$$

Example when we want to study movements of a car?

- do not consider the car itself as a representation
- BUT representing it a point in space to which we assign a masse and speed
- describe the movements based on the theory of point mechanics

From this simple representation we can evaluate the movements of the car

What is a computer model validation?

$$\text{Physical model} \xrightarrow[\text{numerical approximation and algorithms}]{\text{Mathematicien}} \text{Numerical code}$$

or computer model
or simulator

- by applying the mathematical approximation techniques and theories
- the physical model is translated to a mathematical model

We obtain therefore what we call numerical code.

What is a computer model validation?

Computer model $\xrightarrow[\text{examine by statistical methods}]{\text{Statisticien}}$ to know if it display a reliable picture of "reality" ?

Statisticien's work in two steps:

- Verification: quantifying the errors produced by the approximate resolution of the mathematical problem
- Validation: try to answer the question on the validity of the mathematical modeling of physical system

What is a computer model validation?

Physical phenomena(real system) $\xrightarrow[\text{simplify by physical lows}]{\text{Physicien}}$ Physical model

$\hookrightarrow$ **Model uncertainty**

Physical model $\xrightarrow[\text{numerical approximation and algorithms}]{\text{Mathematicien}}$ Numerical code

or computer model

or simulator

$\hookrightarrow$ **Numerical uncertainty**

Computer model $\xrightarrow[\text{examine by statistical methods}]{\text{Statisticien}}$ to know if it display a reliable picture of "reality"?

Physical system: Mathematical notation

For unknown real physical system $r(\boldsymbol{x})$

- physical parameters: $\boldsymbol{x} \in \mathbb{X} \subset \mathbb{R}^d$ observable and controllable inputs

the observation y can be obtained as

$$\boldsymbol{x} \in \mathbb{R}^d \rightarrow \boxed{\text{r: physical system}} \rightarrow y \in \mathbb{R}$$

or

$$y = r(\boldsymbol{x}) + \epsilon$$

with $\epsilon \sim \mathcal{N}(0, \lambda)$: measure error that degrades the knowledge of $r(\boldsymbol{x})$.

Computer model: Mathematical notation

Computer model is represented by a function f:

$$f : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}$$
$$(\boldsymbol{x}, \boldsymbol{\theta}) \to f(\boldsymbol{x}, \boldsymbol{\theta})$$

in which

- The inputs $\boldsymbol{\theta}$ are the calibration parameters of the computation code.

The function $f(\boldsymbol{x}, \boldsymbol{\theta})$:

- can have a simple linear structure $f(\boldsymbol{x}, \boldsymbol{\theta}) = g(\boldsymbol{x})\boldsymbol{\theta}$

Or challenges

- can be a complex function with large computation time
  - sparing computation time by meta-modelling

Validation problem: Statistical modelling

Suppose that

$$\delta(\boldsymbol{x}) = r(\boldsymbol{x}) - f(\boldsymbol{x}, \boldsymbol{\theta})$$

$\delta(\boldsymbol{x})$ is a stochastique function and called "code error". Then the observation y can be obtained as

$$y = f(\boldsymbol{x}, \boldsymbol{\theta}) + \delta(\boldsymbol{x}) + \epsilon$$

Remark

- Identifiability issue: How to define $\boldsymbol{\theta}$ and δ w.r.t $r(\boldsymbol{x})$ in a good way? It is not clear.

Does the computer code mimics the reality or

$$y = f(\boldsymbol{x}, \boldsymbol{\theta}) + \epsilon?$$

Validation problem: Statistical modelling

We are interested in testing hypotheses

$$\mathfrak{H}_0 : \delta(\boldsymbol{x}) = 0$$
$$\mathfrak{H}_1 : \delta(\boldsymbol{x}) \neq 0$$

The code validation can be performed using Bayesian model selection methods Damblin et al. (2016).

$$\mathfrak{M}_0 : y = f(\boldsymbol{x}, \boldsymbol{\theta}_0) + \epsilon_0$$
$$\mathfrak{M}_1 : y = f(\boldsymbol{x}, \boldsymbol{\theta}_1) + \delta(\boldsymbol{x}) + \epsilon_1.$$

where $\mathfrak{M}_0$ and $\mathfrak{M}_1$ are the pure code and the discrepancy-corrected code, respectively.

Reminder

Bayesian model choice:

- a special case of testing hypotheses theory
- comparison of k potential statistical models towards the selection of model that fits the data "best"
- not to seek to identify which model is "true", but to indicate which fits data better

The most common approaches to Bayesian hypothesis testing in practice

- posterior probabilities of the model given the data
- Bayes factor and its approximations such as the Bayesian information criterion (BIC) and the Deviance information criterion (DIC) and posterior predictive tools and their variants

Standard Bayesian approach to testing

Suppose that two families of models under comparison are given by

$$\mathfrak{M}_0 : \; x \sim \pi_0(x|\theta_0)\,, \; \theta_0 \in \Theta_0 \quad \text{and} \quad \mathfrak{M}_1 : \; x \sim \pi_1(x|\theta_1)\,, \; \theta_1 \in \Theta_1\,,$$

and associate with each model a prior distribution,

$$\begin{aligned} \theta_0 \sim \pi_0(\theta_0) \quad &\text{and} \quad \theta_1 \sim \pi_1(\theta_1)\,, \\ \omega_0 = \pi(\mathfrak{M}_0) \quad &\text{and} \quad \omega_1 = \pi(\mathfrak{M}_1) \end{aligned}$$

Standard Bayesian approach to testing

In order to compare the marginal likelihoods

$$m_0(x) = \int_{\Theta_0} \pi_0(x|\theta_0)\,\pi_0(\theta_0)\,d(\theta_0) \quad \text{and} \quad m_1(x) = \int_{\Theta_1} \pi_1(x|\theta_1)\,\pi_1(\theta_1)\,d(\theta_1)$$

either through Bayes factor or posterior probability, respectively:

$$\mathfrak{B}_{01} = \frac{m_0(x)}{m_1(x)}, \quad \mathbb{P}(\mathfrak{M}_0|x) = \frac{\omega_0 m_0(x)}{\omega_0 m_0(x) + \omega_1 m_1(x)};$$

the latter depends on the prior weights ω_i.

How to make a decision

Bayesian decision step in order to comparing two models

- comparing Bayes factor $\mathfrak{B}_{01}$ with threshold value of one or
- comparing posterior probability $\mathbb{P}(\mathfrak{M}_0|x)$ with bound

When comparing more than two models

- selecting model with highest posterior probability but highly dependent on the prior modeling, even with large databases

How to make a decision

Bayesian decision step in order to comparing two models

- comparing Bayes factor $\mathfrak{B}_{01}$ with threshold value of one or
- comparing posterior probability $\mathbb{P}(\mathfrak{M}_0|x)$ with bound

When comparing more than two models

- selecting model with highest posterior probability but highly dependent on the prior modeling, even with large databases

Interpretation of Bayes Factor

If $\omega_0 = \omega_1 = 0.5$, then

$$\mathfrak{B}_{01} = \frac{m_0(x)}{m_1(x)}, \quad \mathbb{P}(\mathfrak{M}_0|x) = \frac{\mathfrak{B}_{01}}{\mathfrak{B}_{01}+1};$$

Jeffreys gave a scale for interpretation of $\mathfrak{B}_{01}$:

$\mathfrak{B}_{01}$, Strength of evidence supporting $\mathfrak{M}_0$, $\mathbb{P}(\mathfrak{M}_0|x)$

$\mathfrak{B}_{01} < 1$, negative, $\mathbb{P}(\mathfrak{M}_0|x) < 0.5$

$1 < \mathfrak{B}_{01} < 10^{1/2}$, barely worth mentioning, $0.5 < \mathbb{P}(\mathfrak{M}_0|x) < 0.75$

$10^{1/2} < \mathfrak{B}_{01} < 10$, substantial, $0.75 < \mathbb{P}(\mathfrak{M}_0|x) < 0.9$

$10 < \mathfrak{B}_{01} < 10^{3/2}$, strong, $0.9 < \mathbb{P}(\mathfrak{M}_0|x) < 0.96$

$10^{3/2} < \mathfrak{B}_{01} < 100$, very strong, $0.96 < \mathbb{P}(\mathfrak{M}_0|x) < 0.99$

$\mathfrak{B}_{01} > 100$, decisive, $\mathbb{P}(\mathfrak{M}_0|x) > 0.99$

Some difficulties with traditional handling of Bayesian tests

$$\mathfrak{B}_{01} = \frac{m_0(x)}{m_1(x)}, \quad \mathbb{P}(\mathfrak{M}_0|x) = \frac{\omega_0 \mathfrak{B}_{01}}{\omega_0 \mathfrak{B}_{01} + \omega_1};$$

- subsequent and delicate interpretation (or calibration) of the strength towards supporting a given hypothesis or model, because it is not a Bayesian decision rule
- long-lasting impact of the prior modeling, despite overall consistency proof for Bayes factor

Some more difficulties

- discontinuity in use of improper priors since they are not justified in most testing situations, leading to many alternative
- binary (*accept* vs. *reject*) outcome more suited for immediate decision (if any) than for model evaluation, in connection with rudimentary loss function
- lack of assessment of uncertainty associated with decision itself
- difficult computation of marginal likelihoods in most settings with further controversies about which computational solution to adopt

Testing problems via mixture estimation model

A new paradigm for testing: Simple representation of the testing problem as a two-component mixture estimation problem where the weights are formally equal to 0 or 1

- provides a convergent and naturally interpretable solution,
- allowing for a more extended use of improper priors

Inspired from consistency result of Rousseau and Mengersen (2011) on estimated overfitting mixtures

- over-parameterised mixtures can be consistently estimated

[Kamary & Mengersen & Robert & Rousseau, 2014]

Testing problems via mixture estimation model

Given two statistical models,

$$\mathfrak{M}_0 : \; x \sim \pi_0(x|\theta_0)\,, \; \theta_0 \in \Theta_0 \quad \text{and} \quad \mathfrak{M}_1 : \; x \sim \pi_1(x|\theta_1)\,, \; \theta_1 \in \Theta_1\,,$$

embed both within an encompassing mixture

$$\mathfrak{M}_\alpha : \; x \sim \alpha\pi_0(x|\theta_0) + (1-\alpha)\pi_1(x|\theta_1)\,, \; 0 \leq \alpha \leq 1 \qquad (1)$$

- Both models correspond to special cases of (1), one for $\alpha = 1$ and one for $\alpha = 0$
- Draw inference on mixture representation (1), as if each observation was individually and independently produced by the mixture model

[Kamary & Mengersen & Robert & Rousseau, 2014]

Advantages

- relying on a Bayesian estimate of the weight α rather than on posterior probability of model $\mathfrak{M}_1$ does produce an equally convergent indicator of which model is "true"
- interpretation of estimator of α at least as natural as handling the posterior probability, while avoiding zero-one loss setting
- highly problematic computations of the marginal likelihoods is bypassed, since standard algorithms are available for Bayesian mixture estimation
- allows to consider all models at once rather than engaging in pairwise costly comparisons

More advantages

- posterior distribution of α evaluates more thoroughly strength of support for a given model than the single figure outcome of a posterior probability
- additional feature missing from traditional Bayesian answers: a mixture model acknowledges possibility that, for a finite dataset, *both* models or *none* could be acceptable
- non-informative (improper) priors are manageable in this setting, provided both models first reparameterised towards shared parameters, e.g. location and scale parameters
- in special case when all parameters are common

$$\mathfrak{M}_\alpha : \; x \sim \alpha\pi_0(x|\theta) + (1-\alpha)\pi_1(x|\theta)\,, 0 \leq \alpha \leq 1$$

if θ is a location parameter, a flat prior $\pi(\theta) \propto 1$ is available

Mixture estimation using latent variable

Consider sample $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ from (1). Completion by latent component indicators ζ_i leads to completed likelihood

$$\begin{aligned}(\theta, \alpha_0, \alpha_1 \mid \mathbf{x}, \zeta) &= \prod_{i=1}^{n} \alpha_{\zeta_i} \pi_{\zeta_i}(x_i \mid \theta_{\zeta_i}) \\ &= \alpha^{n_1}(1-\alpha)^{n_2} \prod_{i=1}^{n} \pi_{\zeta_i}(x_i \mid \theta_{\zeta_i}),\end{aligned}$$

where

$$(n_1, n_2) = \left(\sum_{i=1}^{n} \mathbb{I}_{\zeta_i=0}, \sum_{i=1}^{n} \mathbb{I}_{\zeta_i=1} \right)$$

under constraint

$$n = \sum_{j=}^{1} \sum_{i=1}^{n} \mathbb{I}_{\zeta_i=j}$$

.

[Diebolt & Robert, 1990]

Mixture estimation using latent variable

Using natural Gibbs implementation

- under a $\mathscr{B}eta(a_1, a_2)$, α is generated from a Beta $\mathscr{B}eta(a_1 + n_1, a_2 + n_2)$
- Gibbs sampling scheme is valid from a theoretical point of view
- convergence difficulties in the current setting, especially with large samples
- due to prior concentration on boundaries of $(0, 1)$ for the mixture weight α

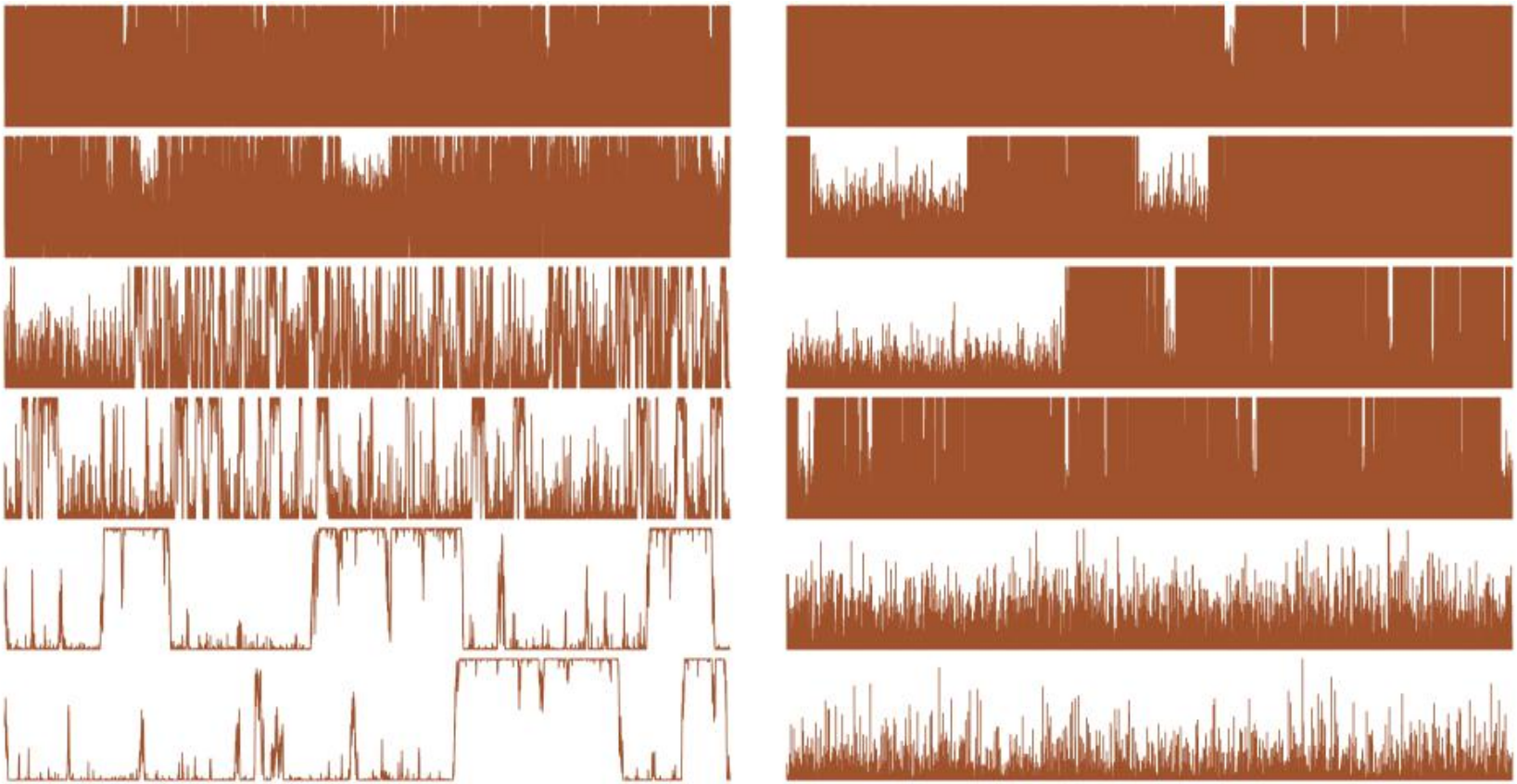
Metropolis-Hastings algorithms

- model parameters θ_i generated from respective full posteriors of both models (i.e., based on entire sample)

$$\pi(\theta_i|\mathbf{x}, \alpha) = \left(\alpha\pi_0(\mathbf{x} \mid \theta_0) + (1-\alpha)\pi_1(\mathbf{x} \mid \theta_1)\right)\pi(\theta_i); \quad i = 0, 1$$

- mixture weight α generated from a random walk proposal on $(0, 1)$

Gibbs versus MH implementation



(Left) Gibbs; (Right) Metropolis–Hastings sequences (α_t) when the mixture model is $\alpha\mathcal{N}(\mu, 1) + (1-\alpha)\mathcal{N}(0, 1)$ for a $\mathcal{N}(0, 1)$ sample of size $N = 5$, 10, 50, 100, 500, 10^3 *(from top to bottom)* based on 10^5 simulations. The y-range range for all series is $(0, 1)$.

Illustrations

Poisson or Geometric

- Models under comparison:

$$\mathfrak{M}_0 : \mathscr{P}(\lambda) \quad \text{and} \quad \mathfrak{M}_1 : \mathscr{G}eo(p)$$

- if both models share a common parameter λ,

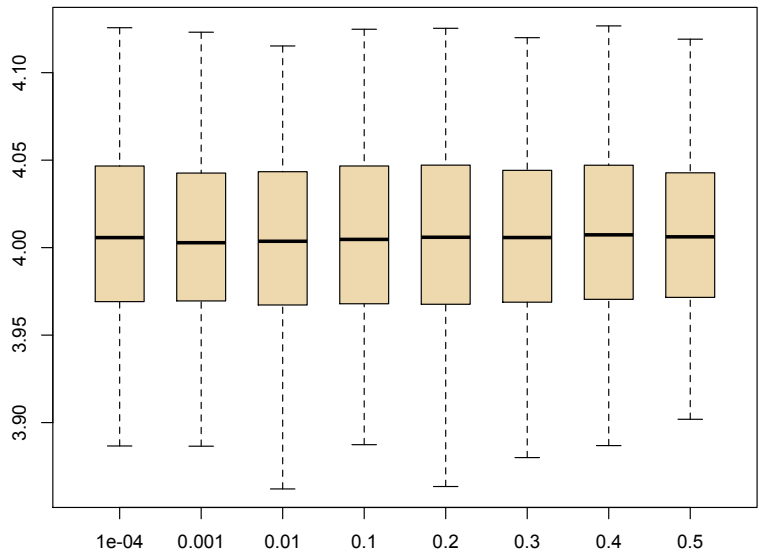
$$\mathfrak{M}_\alpha : \ \alpha\mathscr{P}(\lambda) + (1-\alpha)\mathscr{G}eo(1/1+\lambda)$$

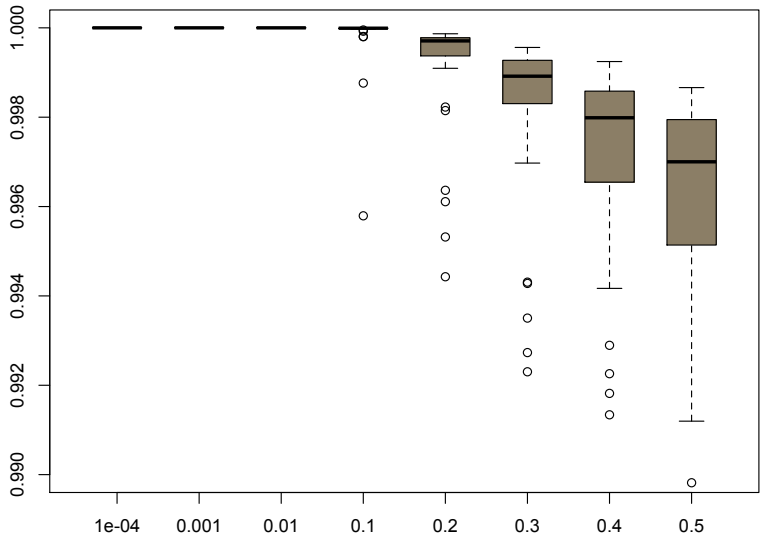
- prior modeling:

$$\pi(\lambda) = 1/\lambda; \quad \alpha \sim \mathscr{B}e(a_0, a_0)$$

- posterior simulation:
  - independent Metropolis–within–Gibbs

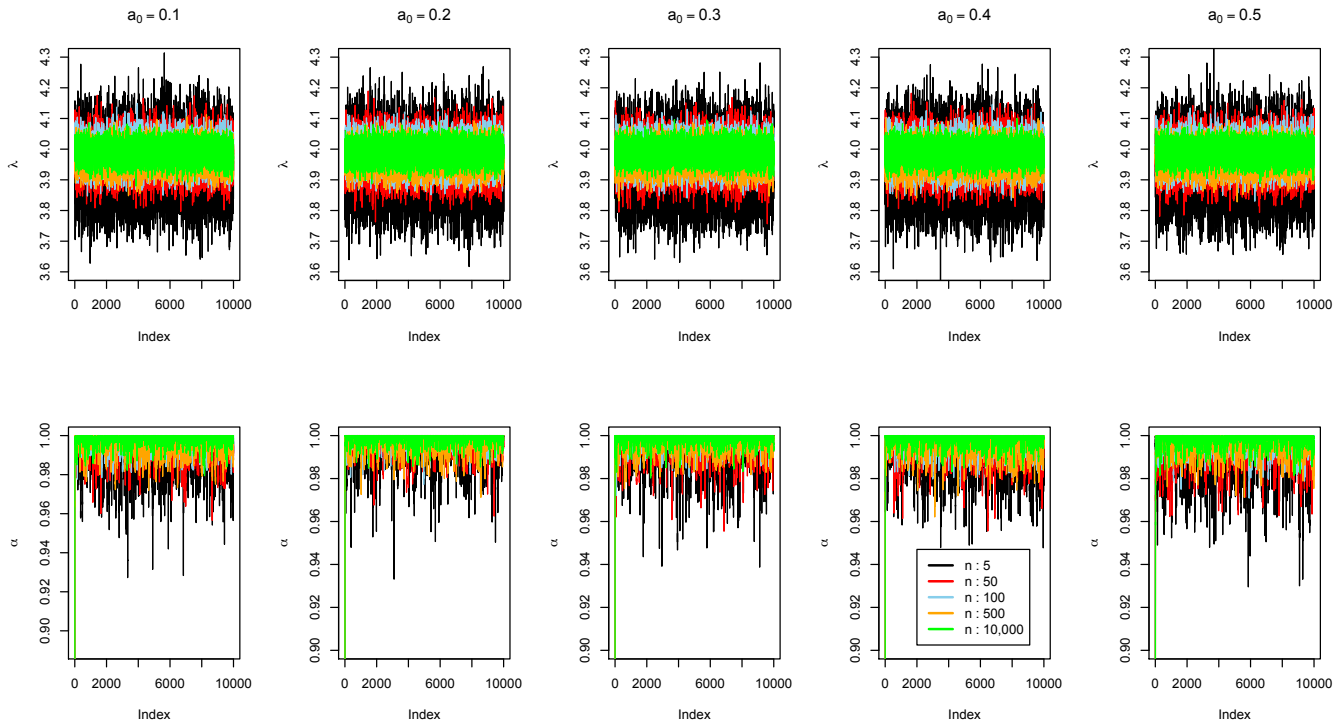
Parameter estimation





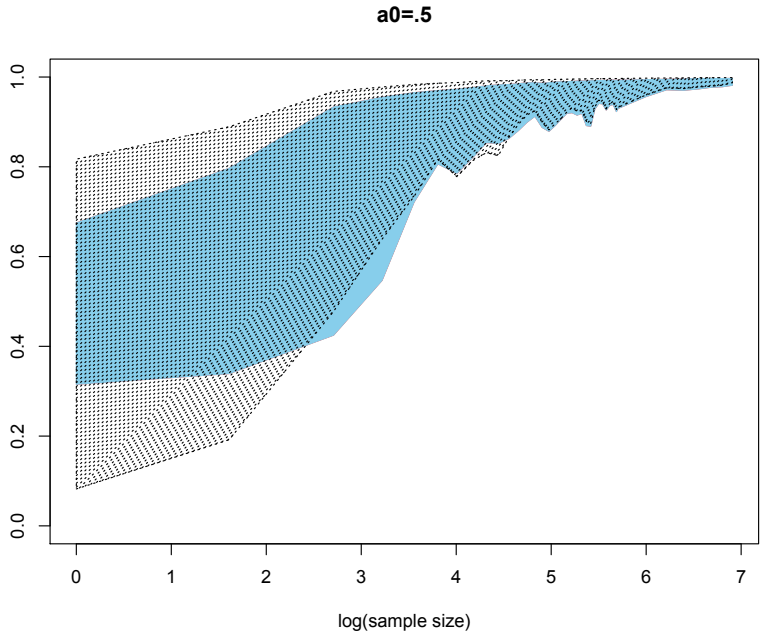
Posterior means of λ and medians of α for 100 Poisson $\mathscr{P}(4)$ datasets of size $n = 1000$, for $a_0 = .0001, .001, .01, .1, .2, .3, .4, .5$. Each posterior approximation is based on 10^4 Metropolis-Hastings iterations.

MCMC convergence



Dataset from a Poisson distribution $\mathscr{P}(4)$: Estimations of *(top)* λ and *(bottom)* α via MH for 5 samples of size $n = 5, 50, 100, 500, 10,000$.

Consistency



Evolution of mixture weight over sample size: Posterior means *(sky-blue)* and medians *(grey-dotted)* of α, over 100 Poisson $\mathscr{P}(4)$ datasets for sample sizes from 1 to 1000.

Code validation: Mixture modelling of competing models

Suppose that

- $X = (x_1 \quad \dots \quad x_n)'$: input physical experiment matrix of size $n \times d$
- $Y = (y_1, \dots, y_n)$: vector of related available measurements of size n

then for $i = 1, \dots, n$, we want to choose a model between

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

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

Code validation via mixture model estimation

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

Remark

- Discrepancy-corrected prediction is traditionally based on the Gaussian processes

[Damblin et al., 2016]

- difficulty item from the dependence of the data which was not the case in the initial implementation of mixture technique.

[Kamary et al., 2014]

We therefore define the mixture of the competing models under the condition of considering $\delta(\boldsymbol{x}_i)$ as a latent variable to be estimated with the other model parameters.

Code validation via mixture model estimation

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

$\hookrightarrow$ after embedding both models within an encompassing mixture model

$$\ell_{\mathfrak{M}_\alpha}(\boldsymbol{\theta}_0, \lambda_0, \boldsymbol{\theta}_1, \lambda_1, \delta; Y, X) = \prod_{i=1}^{n} \left(\alpha \ell_{\mathfrak{M}_0}(\boldsymbol{\theta}_0, \lambda_0; y_i, x_i) + (1-\alpha)\ell_{\mathfrak{M}_1}(\boldsymbol{\theta}_1, \lambda_1, \delta; y_i, x_i)\right). \tag{2}$$

where

$$\ell_{\mathfrak{M}_0}(\boldsymbol{\theta}_0, \lambda_0; y_i, x_i) = \exp\left(-\tfrac{1}{2\lambda_0^2}(y_i - g(\boldsymbol{x}_i)\boldsymbol{\theta}_0)^2\right) \Big/ (2\pi\lambda_0^2)^{n/2}$$
$$\ell_{\mathfrak{M}_1}(\boldsymbol{\theta}_1, \lambda_1, \delta; y_i, x_i) = \exp\left(-\tfrac{1}{2\lambda_1^2}(y_i - g(\boldsymbol{x}_i)\boldsymbol{\theta}_1 - \delta(x_i))^2\right) \Big/ (2\pi\lambda_1^2)^{n/2} \tag{3}$$

Parameter prior choice

In order to able the non-informative priors to be used for mixture parameters, suppose that the parameters θ and λ are shared:

$$\mathfrak{M}_0 : y_i = g(\boldsymbol{x}_i)\boldsymbol{\theta} + \epsilon_i$$
$$\mathfrak{M}_1 : y_i = g(\boldsymbol{x}_i)\boldsymbol{\theta} + \delta(\boldsymbol{x}_i) + \epsilon_i.$$

then $\ell_{\mathfrak{M}_\alpha}$ equals

$$1/(2\pi\lambda^2)^{n/2} \prod_{i=1}^{n} \left(\alpha \exp(-\frac{1}{2\lambda^2}(y_i - g(x_i)\boldsymbol{\theta})^2) + (1-\alpha)\exp(-\frac{1}{2\lambda^2}(y_i - g(x_i)\boldsymbol{\theta} - \delta(x_i))^2) \right) \tag{4}$$

and

$$\pi(\boldsymbol{\theta}, \lambda) = 1/\lambda$$

Parameter prior choice

$\ell_{\mathfrak{M}_\alpha}$ equals

$$1/(2\pi\lambda^2)^{n/2} \prod_{i=1}^{n} \left(\alpha \exp(-\frac{1}{2\lambda^2}(y_i - g(x_i)\boldsymbol{\theta})^2) + (1-\alpha)\exp(-\frac{1}{2\lambda^2}(y_i - g(x_i)\boldsymbol{\theta} - \delta(x_i))^2) \right) \tag{5}$$

The discrepancy δ has a Gaussian process prior as

$$\delta(X) \sim \mathscr{GP}(\mu_\delta, \Sigma_\delta); \quad \Sigma_\delta = \sigma_\delta^2 \mathrm{Corr}_{\gamma_\delta}(x_i, x_{i'})$$

where

$$\mathrm{Corr}_{\gamma_\delta}(x_i, x_{i'}) = \exp\left(-|x_i - x_{i'}|/\gamma_\delta\right).$$

Parameter prior choice

Theorem

Let $g : \mathbb{R}^d \to \mathbb{R}^d; d > 1$ *be a finite-valued function and for any vector* x_i *of size* d, $g(x_{ij}) \neq 0; j = 1, \ldots, d$. *The posterior distribution associated with the prior* $\pi(\boldsymbol{\theta}, \lambda) = 1/\lambda$ *and with the likelihood* $\ell_{\mathfrak{M}_\alpha}$ *is proper when*

- *for any* $0 < k < 1$, *the hyperparameter* σ^2_δ *of the discrepancy prior distribution is reparameterized as* $\sigma^2_\delta = \lambda^2/k$ *and so* $\Sigma_\delta = (\lambda^2/k)Corr_{\gamma_\delta}$ *when* $Corr_{\gamma_\delta}$ *is the correlation function of* δ.
- *the dimensionality,* d, *is less than* n;
- *the mixture weight* α *has a proper beta prior* $\mathscr{B}(a_0, a_0)$;
- γ_δ *has a proper Beta prior* $\mathscr{B}(b_1, b_2)$.
- *proper distribution is used on* k.

Simulation studies

Simulated data: If $x = \{i/n\}_{i=1}^{n}$,

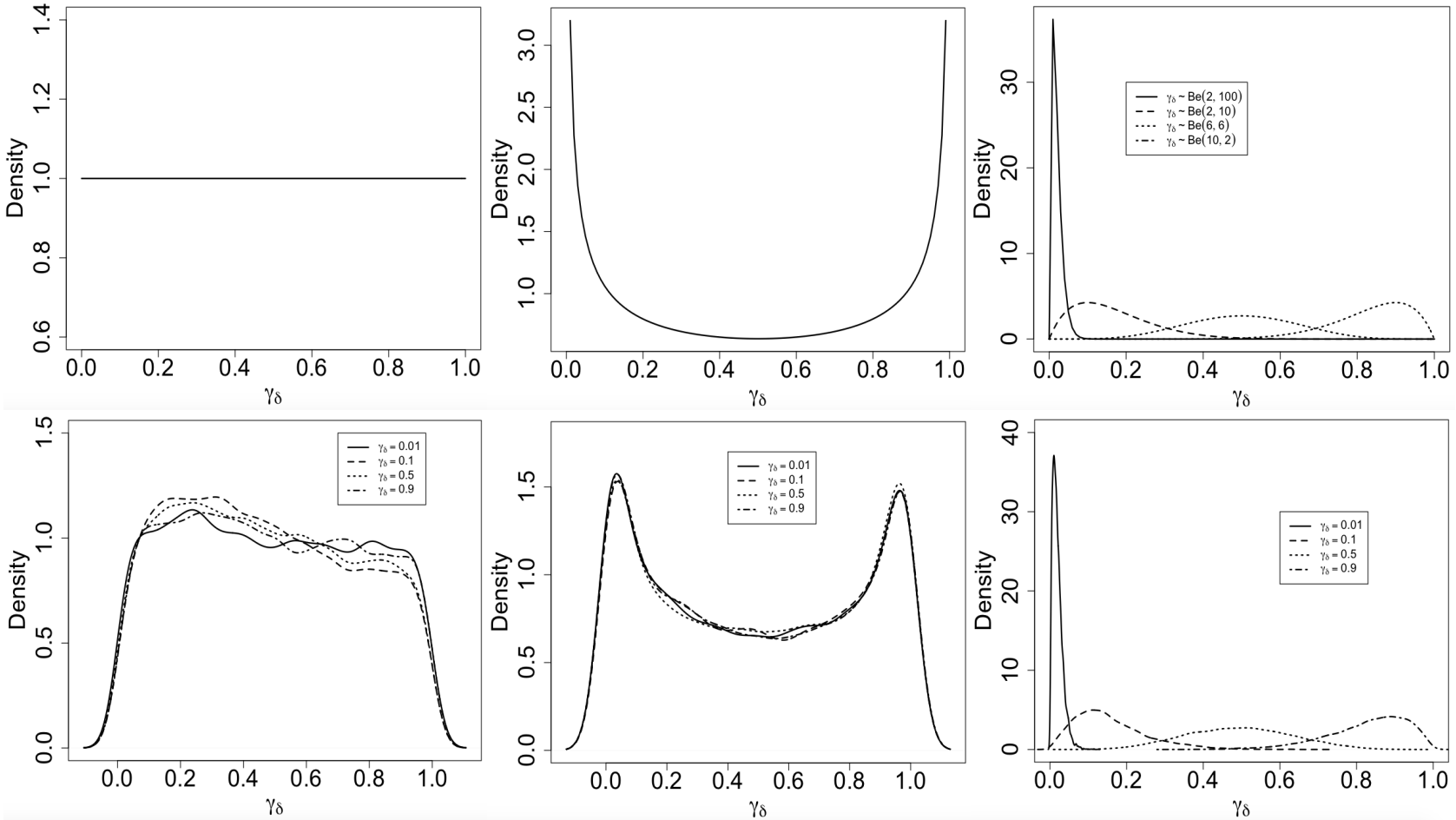
- $g(x)$ with a degree 2 polynomial code in x as $g(x) = (\mathbf{1}, x, x^2)$
- parameter true values, $\boldsymbol{\theta}^* = (4, 1, 2)', \lambda^* = 0.1$ and $k^* = 0.1$

We simulate samples of size n by considering that the true value of the parameter γ_δ varies between $(0, 1)$. Under the priors

- $\pi(\boldsymbol{\theta}, \lambda) = 1/\lambda$
- $\gamma_\delta \sim \mathscr{B}eta(b_1, b_2)$
- $k \sim \mathscr{B}eta(2, 18)$
- $\alpha \sim \mathscr{B}eta(0.5, 0.5)$

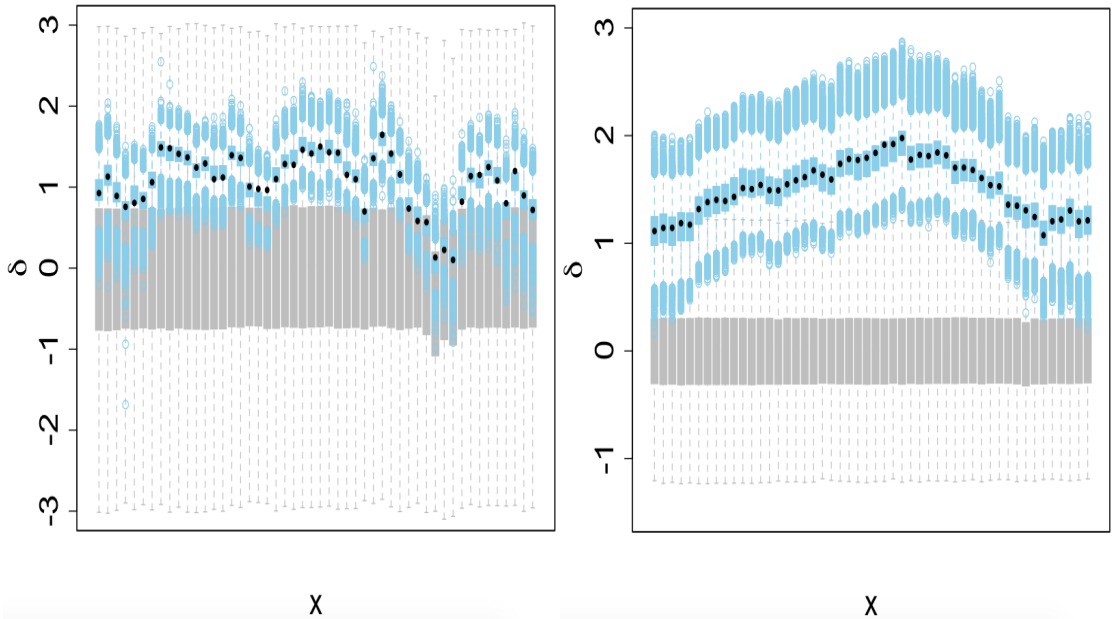
we estimate mixture model $\mathfrak{M}_\alpha$ by implementing Metopolis-within-Gibbs algorithm.

Sensitivity of the correlation length to the prior choice



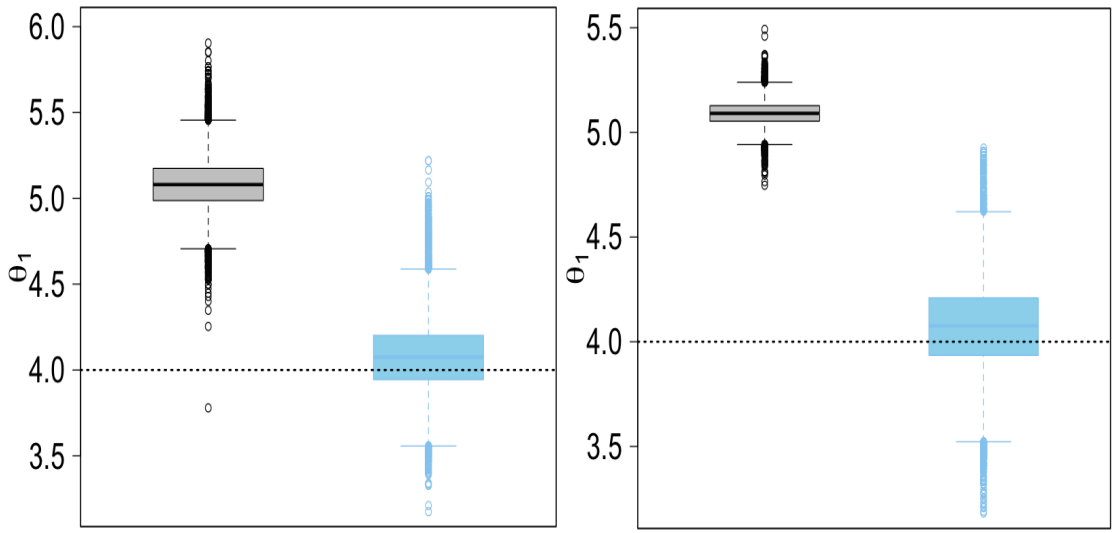
(Top) Beta prior distribution $\mathscr{B}eta(b_1, b_2)$ with (left) $b_1 = b_2 = 1$; (middle) $b_1 = b_2 = 0.5$; (right) different curves are related to the b_1 and b_2 indicated on the plot. (Bottom) Empirical posterior distributions of γ_δ obtained for each prior illustrated on the top when four datasets of size $n = 50$ are simulated from $\mathfrak{M}_1$ for the true value of $\gamma_\delta^* = 0.01, 0.1, 0.5, 0.9$. The number of MCMC iterations is 2×10^4 with a burn-in of 10^3 iterations.

Sensitivity of δ to the choice of δ prior



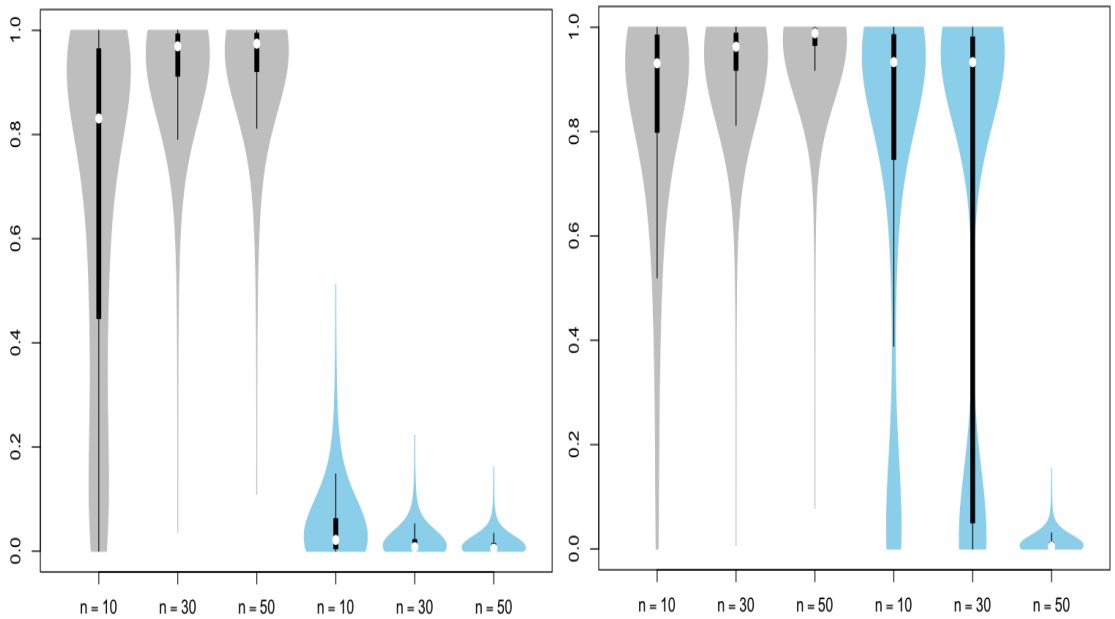
For three datasets of size $n = 50$ simulated from $\mathfrak{M}_1$ when the true value γ_δ^* is (left) 0.1 ; (right) 0.9: Comparison between the posterior distributions of $\delta(x_i); i = 1, \ldots, n$ and the true value (black points). For all plots, the results are shown in (skyblue color) when the prior of $\delta(x)$ is $\mathscr{GP}(1_n, \Sigma_\delta)$ and in (gray color) when $\delta(x) \sim \mathscr{GP}(0_n, \Sigma_\delta)$. The number of MCMC iterations is 2×10^4 with a burn-in of 10^3 iterations.

Sensitivity of $\boldsymbol{\theta}$ to the choice of δ prior



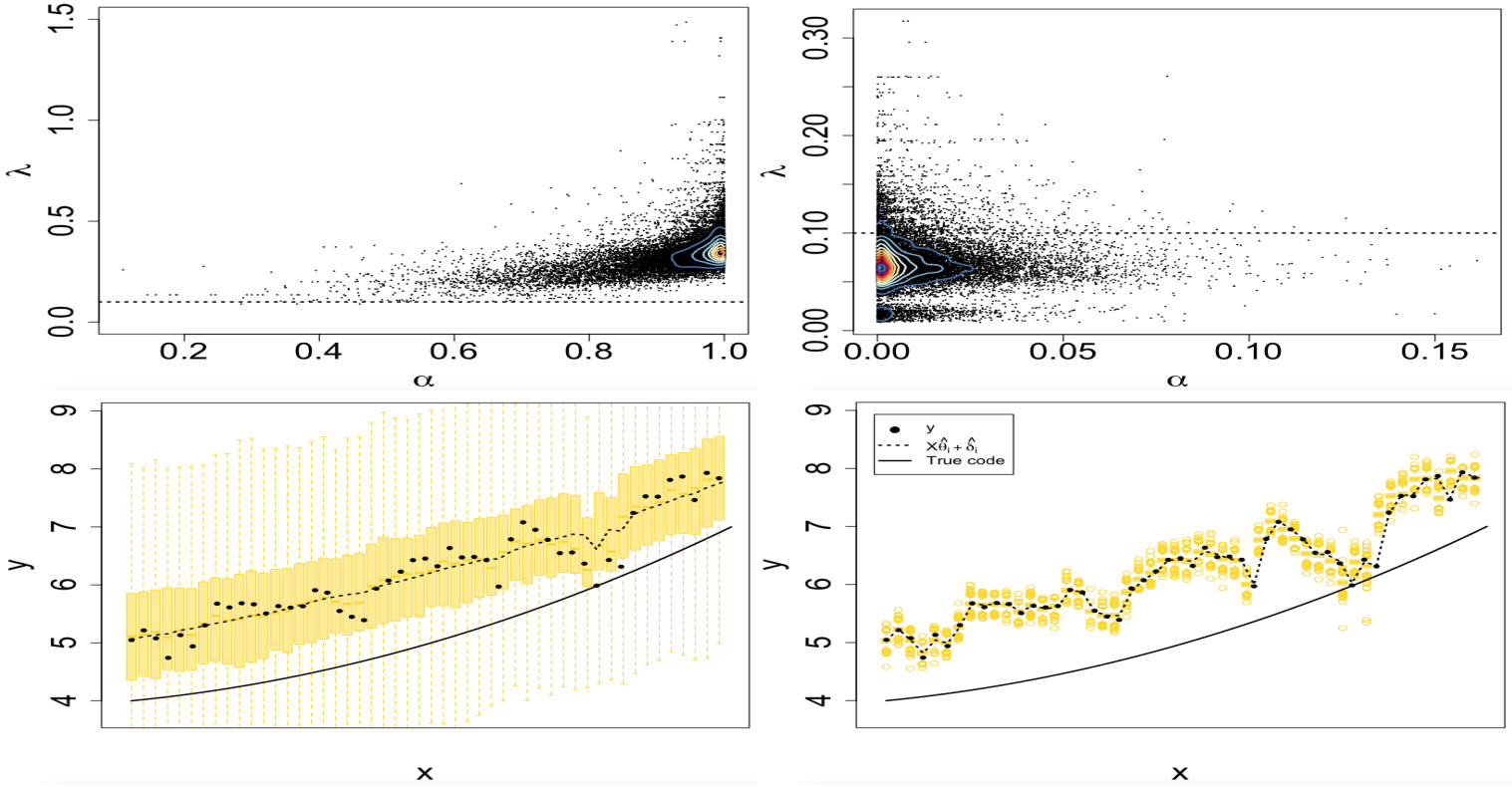
For three datasets of size $n = 50$ simulated from $\mathfrak{M}_1$ when the true value γ_δ^* is (left) 0.1 ; (right) 0.9: Comparison between the posterior distributions of θ_1 and the true value (dotted line). For all plots, the results are shown in (skyblue color) when the prior of $\delta(x)$ is $\mathscr{GP}(1_n, \Sigma_\delta)$ and in (gray color) when $\delta(x) \sim \mathscr{GP}(0_n, \Sigma_\delta)$. The number of MCMC iterations is 2×10^4 with a burn-in of 10^3 iterations.

Sensitivity of α to the choice of δ prior



For three datasets of size $n = 10, 30, 50$ are simulated from $\mathfrak{M}_1$ when the true value γ_δ^* is (left) 0.1; (right) 0.9 : Comparison between the posterior distributions of α, the weight of $\mathfrak{M}_0$ in the mixture model. For different sample sizes, the results are shown in (skyblue color) when the prior of $\delta(x)$ is $\sim \mathscr{GP}(1_n, \Sigma_\delta)$ and in (gray color) when $\delta(x) \sim \mathscr{GP}(0_n, \Sigma_\delta)$. The number of MCMC iterations is 2×10^4 with a burn-in of 10^3 iterations.

Sensitivity of λ to the choice of δ prior



For a sample of size 50 simulated from $\mathfrak{M}_1$ when $\gamma_\delta^* = 0.1$ and $\delta^*(x) \sim \mathcal{GP}(1_n, \Sigma_\delta)$: (Top) Point process representation of the posterior draws of α versus λ, (bottom) Comparison between data points y_i versus x_i (black points), the posterior estimate of $\mathfrak{M}_1$ obtained by averaging over MCMC iterations (dotted line) with overlaying box plots (gold) of the last 1000 posterior draws and the true code (solide line). Mixture model parameters have been estimated when the prior distribution of $\delta(x)$ is (Left) $\mathcal{GP}(0_n, \Sigma_\delta)$; (Right) $\mathcal{GP}(1_n, \Sigma_\delta)$.

Conclusion

The code validation problem

- is considered as a Bayesian model selection
- resort to a technique developed by Kamary et al., 2014 that rely on the encompassing mixture model
- common parameterisation allows for reference priors
- posterior distribution of the parameters is highly subjected to the choice of the discrepancy prior
- posterior distribution of the mixture component weights is used to make a decision about the data
- α can capture the true model only under the condition that the δ prior is informative enough

Perspectives

- what about the real datasets?
- what if the computer model has a complex structure?


Thank
you