Nested Kriging models for large data-sets

François Bachoc, Nicolas Durrande, Didier Rullière

Based on joint work with Clément Chevalier

Mines Saint-Etienne – Chaire OQUAIDO meeting – May 2016

outline :

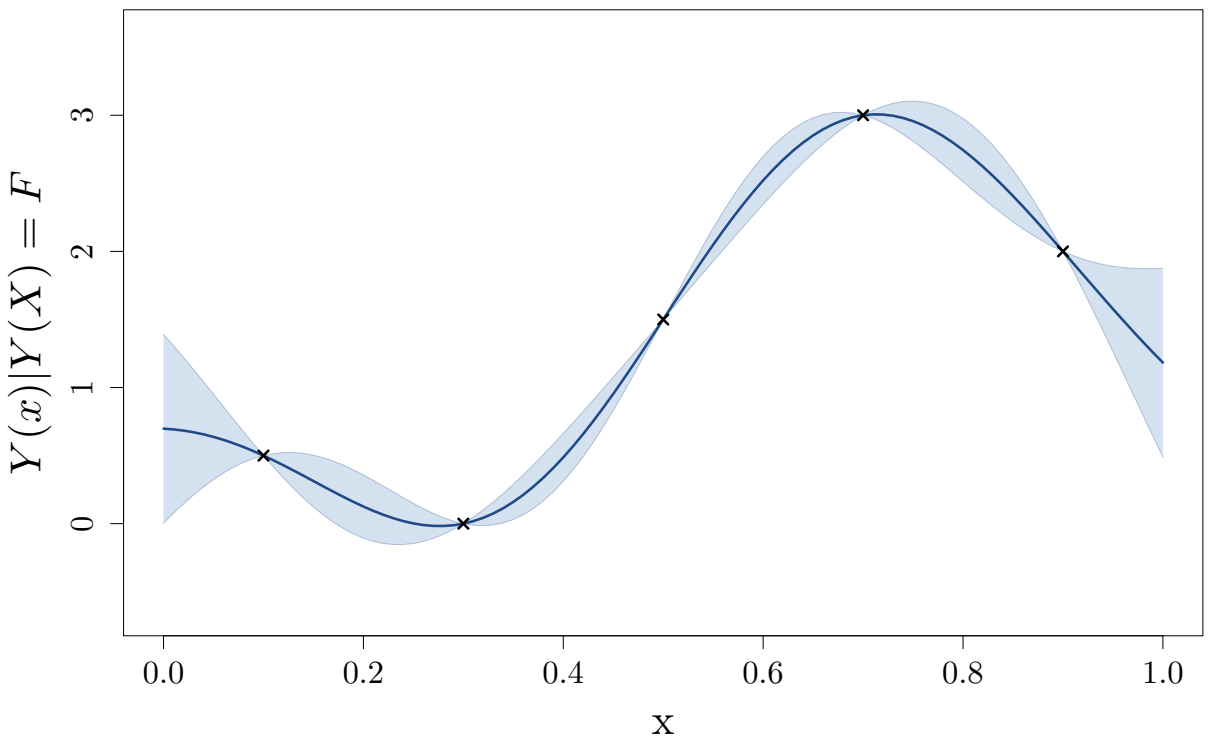
1. Context and notation
2. Litterature review
3. Nested Kriging models
4. Methods consistency
5. Parameter estimation
6. Application to a case study

Context and notation

The conditional distribution of GP (or the BLUE of a second order process) writes

$$m(x) = \mathrm{E}\left[Y(x)|Y(X)=F\right] = k(x,X)k(X,X)^{-1}F$$
$$c(x,x') = \mathrm{Cov}\left[Y(x), Y(x')|Y(X)=F\right] = k(x,x') - k(x,X)k(X,X)^{-1}k(X,x')$$

It can be represented as a mean function with confidence intervals.



If we denote by n the number of observation points, the complexity of building such models is

- $O(n^2)$ in space (storing $k(X, X)$)
- $O(n^3)$ in time (inverting $k(X, X)^{-1}$)

Furthermore, hyperparameter estimation requires to do this many times...

In practice,

- space complexity is often more limiting than time complexity
- the **maximum number of observations** that can be handled **lies in the range [1000, 10000]**.

Various methods have been introduced to deal with a large number of observations :

- methods based on inducing points (sparse GPs)
- methods based on aggregating sub-models
- low rank approximations
- kernels with compact support
- random kitchen sink
- ...

See Rasmussen and Williams, GPML, Chap. 8.

Short literature review

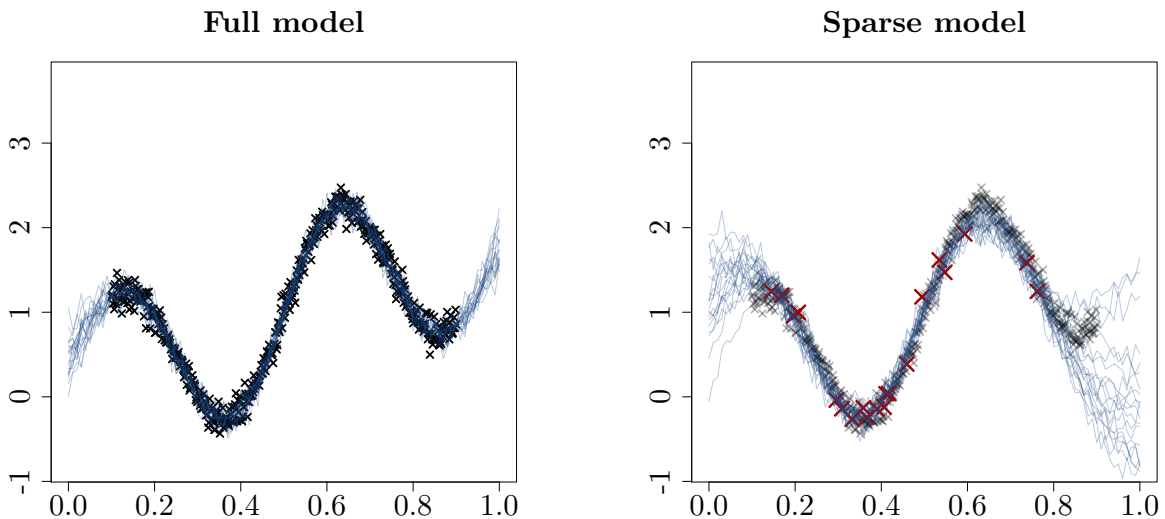
Inducing points (Sparse GPs)

Sparse GPs are based on an approximation of the covariance structure of $(Y(X), Y(x))$ that relies on some inducing variables U.

The general principle of is to learn U from $Y(X)$ and then to predict $Y(x)$ from U.

Example :

A naive approach is to consider U as a subset of the observations : $U = Y(X_{SoD})$:



Inducing points (Sparse GPs)

First, let's assume that the location of the inducing points is known. In order to create a Sparse GP model, one need to specify :

- the covariance between U and $Y(X)$
- the covariance between U and $Y(x)$

Various methods have been proposed :

- **Subset of Regressors** minimize the MSE on all dataset $\rightarrow$ Linear regression
- **Deterministic training conditional** Learning the u with linear regression, prediction with Kriging. $\rightarrow$ Same mean as SoR, more realistic variances.
- **Fully Independent Training Conditional (FITC)** same as DTC but with non homogeneous noise variance.

Ref : Quiñonero-Candela and Rasmussen, JLMR 2005

Inducing points (Sparse GPs)

Let n be the number of datapoints and m be the number of inducing points. The complexity is :

Method	Storage	Computation
full GPR	$O(n^2)$	$O(n^3)$
SoD	$O(m^2)$	$O(m^3)$
SoR	$O(mn)$	$O(m^2 n)$
DTC	$O(mn)$	$O(m^2 n)$
FITC	$O(mn)$	$O(m^2 n)$

Ref : GPML, chap. 8

Inducing points (Sparse GPs)

The remaining questions is : **What should be the location of the inducing ?**

In all the methods above, the inducing points can be chosen :

Among the set of observation points

- randomly
- clustering
- **greedy methods**
- as the test points

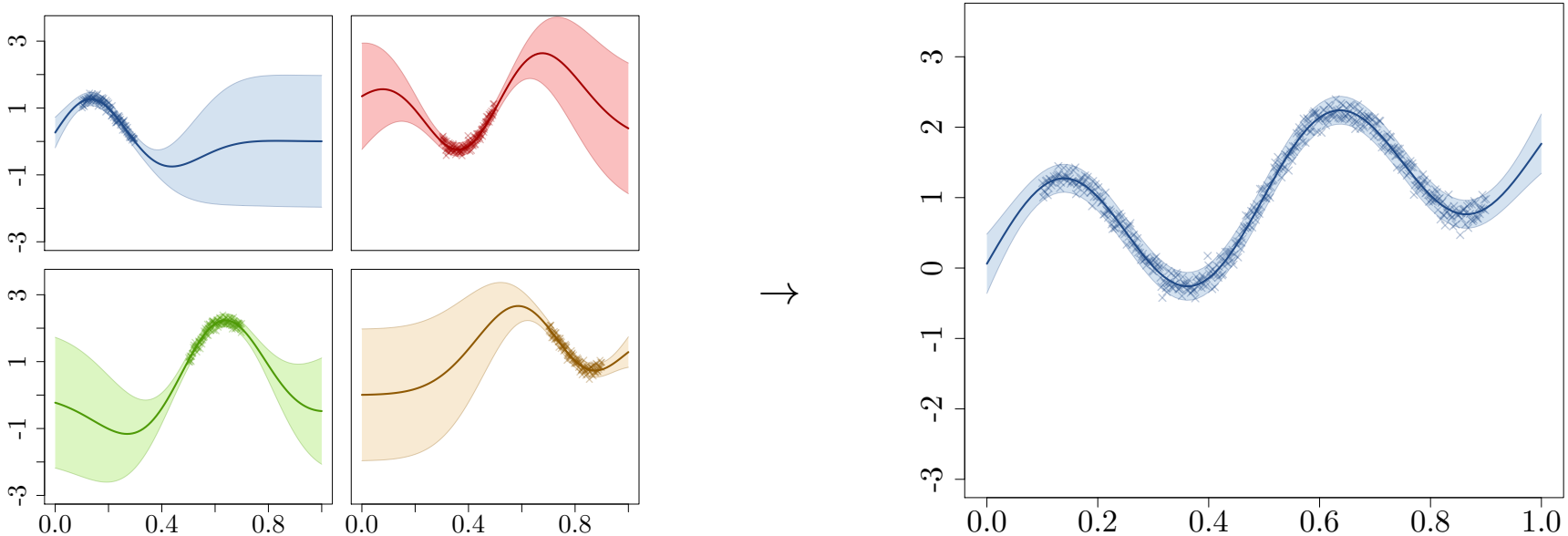
As new points

- maximizing the likelihood
- greedy methods
- variational inference (Titsias AISTATS 2009, Hensman CUAI 2013)

To put it in a nutshell, these methods are based on the fact that lots of data doesn't mean lots of information.

merging sub-models

Another approach is to make many sub-models based on subset of data, and then to find a way to merge these models together



if the number of points per sub-model is fixed to c, there are $p = n/c$ sub-models so the sub-models storage footprint is $O(pc^2) = O(nc)$. Similarly, the complexity is $O(nc^2)$. Note that this is just for the sub-models and not the aggregation procedure.

merging sub-models

Let $f_{M_i}(x)$ be the predictive density of model i at x and $f_M(x)$ denote the aggregated prediction :

Various methods have been proposed in the litterature :

- Product of Experts (PoE)

$$f_M(x) \propto \prod f_{M_I}(x)$$

- Generalised PoE

$$f_M(x) \propto \prod f_{M_I}^{\beta_i}(x)$$

- Bayesian Commitee Machine (BCM)

$$f_M(x) \propto \frac{\prod f_{M_I}(x)}{f_Y^{(N-1)}(x)}$$

- robust BCM

$$f_M(x) \propto \frac{\prod f_{M_I}^{\beta_i}(x)}{f_Y^{(\sum \beta_i - 1)}(x)}$$

Ref : Deisenroth and Wei Ng, ICML proceedings 2015

Nested Kriging Models

Framework - Sub-models

Inputs :
One prediction point : $\mathbf{x} \in \mathbf{D}$.
Initial random field : $\mathbf{Y(x)} \in \mathbf{R}$.
Sub-models vector : $\mathbf{M(x)} = (\mathbf{M_1(x)}, \ldots, \mathbf{M_p(x)}) \in \mathbf{R}^p$.
Sub-models are typically functions of a random vector of observations $Y(X)$.

Known covariances :
we assume that $(Y(x), M(x))$ is centred with $(1+p) \times (1+p)$ covariance matrix :

$$\mathrm{Cov}\left[(Y(x), M(x))\right] = \begin{pmatrix} k(x,x) & k_M(x)^t \\ k_M(x) & K_M(x) \end{pmatrix} \tag{1}$$

$k_M(x)$ is a $p \times 1$ vector with entries $k_M(x)_i = \mathrm{Cov}\left[Y(x), M_i(x)\right]$,
$K_M(x)$ is a $p \times p$ matrix with entries $(K_M(x))_{i,j} = \mathrm{Cov}\left[M_i(x), M_j(x)\right]$.

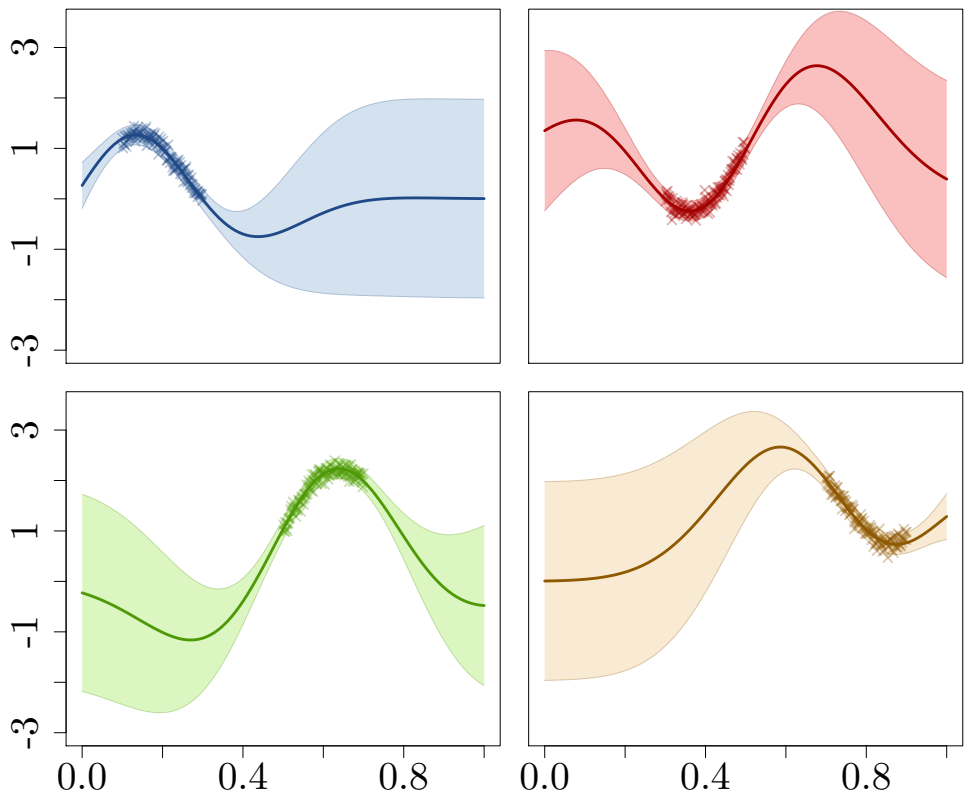
In particular :

- We assume the existence of the first two moments of $(Y(x), M_1(x), \ldots, M_p(x))$
- No other assumption on the joint distribution of $(Y(x), M_1(x), \ldots, M_p(x))$
- $M(x)$ are covariates that are not necessarily a linear combinations of $Y(X)$

Framework - Case of Kriging submodels

Let $X_1, \ldots, X_p$ be matrices corresponding to subsets of points of X.
Define p associated Kriging sub-models (or *experts*) :

$$\begin{cases} \mathbf{M_i(x)} & = & \mathrm{E}\left[Y(x)|Y(X_i)\right] = k(x,X_i)k(X_i,X_i)^{-1}Y(X_i) \\ \left(\mathbf{k_M(x)}\right)_{\mathbf{i}} & = & \mathrm{Cov}\left[Y(x),M_i(x)\right] = k(x,X_i)k(X_i,X_i)^{-1}k(X_i,x) \\ \left(\mathbf{K_M(x)}\right)_{\mathbf{i,j}} & = & \mathrm{Cov}\left[M_i(x),M_j(x)\right] = k(x,X_i)k(X_i,X_i)^{-1}k(X_i,X_j)k(X_j,X_j)^{-1}k(X_j,x). \end{cases}$$



Main questions

Classical kriging outputs (Gaussian case) :

pointwise :

- Kriging mean $\mathrm{E}\left[Y(x)|Y(X)\right]$
- Kriging variance $\mathrm{V}\left[Y(x)|Y(X)\right]$

cross-points :

- Kriging covariances $\mathrm{Cov}\left[Y(x), Y(x')|Y(X)\right]$
- Conditional sample paths

Corresponding questions when aggregating models :

pointwise :

- Aggregation $M_{1\oplus\ldots\oplus p}(x)$ of $M_1(x), \ldots, M_p(x)$, in order to estimate $Y(x)$?
- Variance $v_{1\oplus\ldots\oplus p}(x)$ of the error $M_{1\oplus\ldots\oplus p}(x) - Y(x)$?

cross-points :

- Covariances between $M_{1\oplus\ldots\oplus p}(x)$, $M_{1\oplus\ldots\oplus p}(x')$?
- Conditional sample paths (Gaussian case) ?

Proposed pointwise aggregation

Definition (sub-models aggregation)

For a given point $x \in D$, we define the aggregation of the sub-models (or mixture of experts) by

$$\mathbf{M_{1\oplus\ldots\oplus p}(x) = k_M(x)^t K_M(x)^{-1} M(x).} \tag{2}$$

Some properties for pointwise estimation

- **Optimal** : $M_{1\oplus\ldots\oplus p}(x)$ is the BLUE of $Y(x)$ that writes $\sum_i \alpha_i(x) M_i(x)$.
- **Square error** :
  $v_{1\oplus\ldots\oplus p}(x) = \mathrm{E}\left[(Y(x) - M_{1\oplus\ldots\oplus p}(x))^2\right] = k(x,x) - k_M(x)^t K_M(x)^{-1} k_M(x)$
- **Conditional distribution** : If $(Y(x), M(x))$ is a Gaussian random vector, then the conditional distribution of $Y(x)$ given $M(x)$ is normal with moments

$$\mathrm{E}\left[Y(x) | M_1(x), \ldots, M_p(x)\right] = M_{1\oplus\ldots\oplus p}(x)$$
$$\mathrm{V}\left[Y(x) | M_1(x), \ldots, M_p(x)\right] = v_{1\oplus\ldots\oplus p}(x).$$

- **Full model recovery** : if $M(x)$ writes $M(x) = \Lambda(x) Y(X)$, and if $\Lambda(x)$ is an invertible matrix, then $M_{1\oplus\ldots\oplus p}(x) = k(x,X)k(X,X)^{-1} Y(X)$

Example 1 - linear regressions

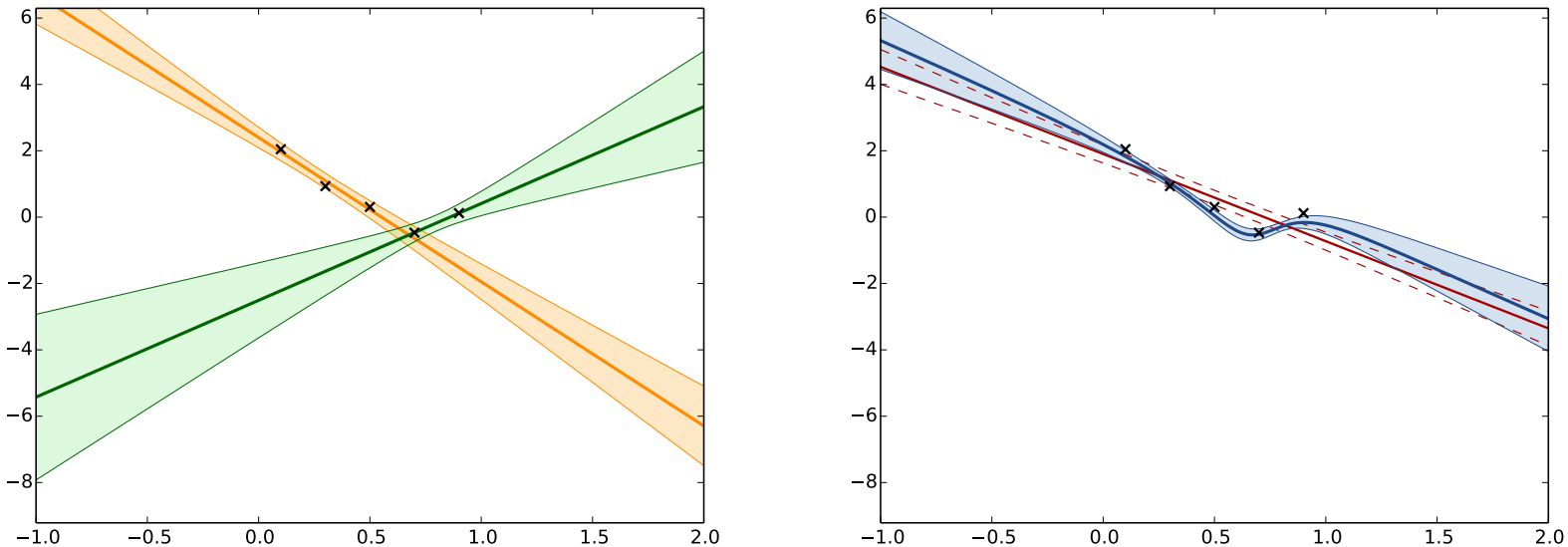


FIGURE: Example 1 : aggregation of two linear regression models. The left panel shows the sub-models and the right one the merged one in blue as well as the full model in red lines. Exhibited confidence bands corresponds to a difference to mean value of two standard deviation.

Example 2 - kriging submodels

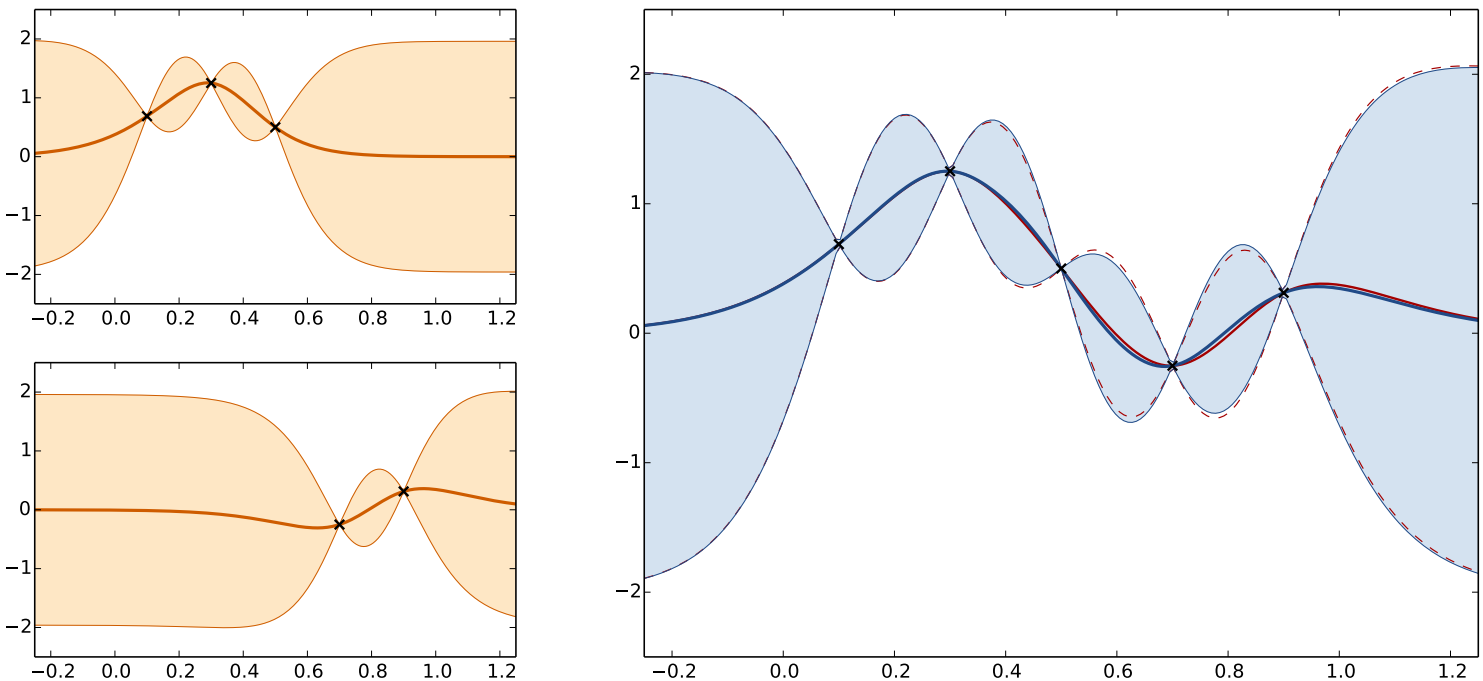


FIGURE: Example 2 : aggregation of two Gaussian process regression models. The left panel shows the sub-models and the right one the merged one in blue as well as the full model in red lines.

Example 3 - fully informative submodels

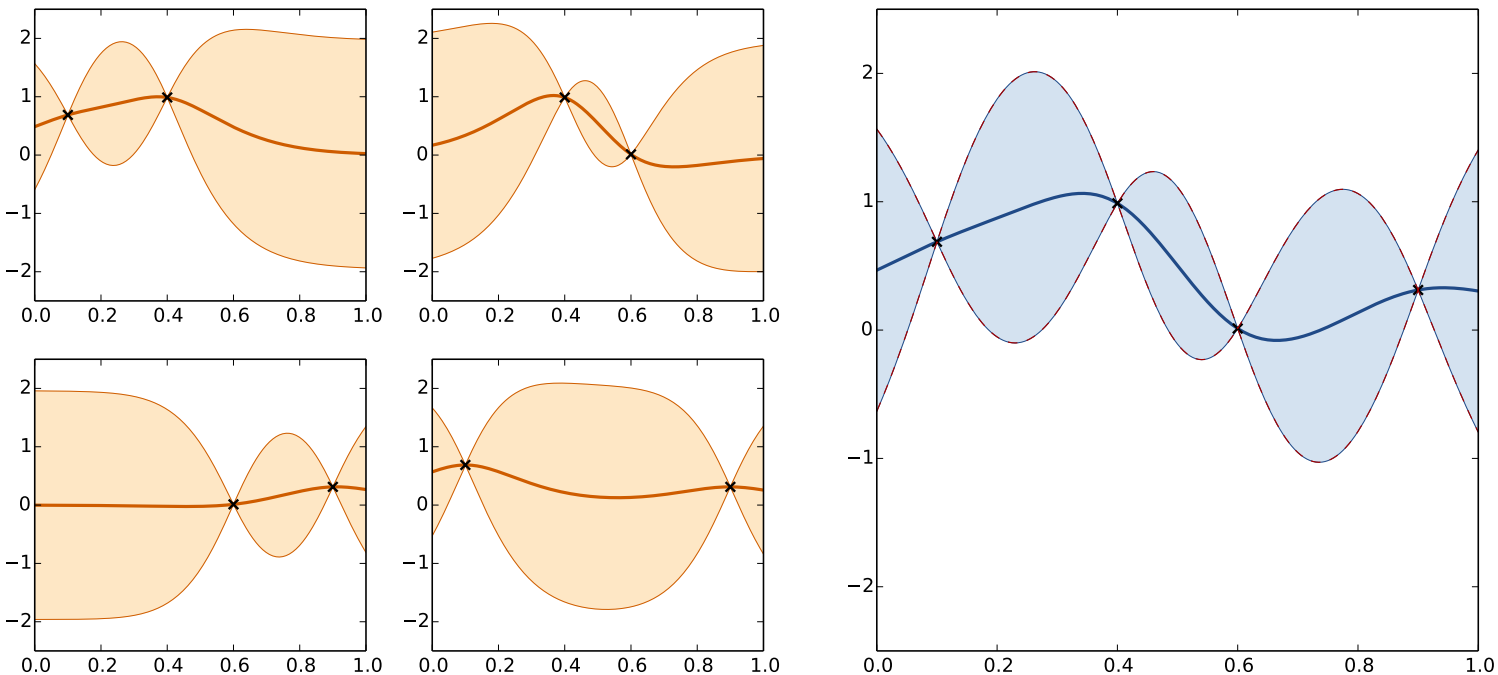


FIGURE: Example of merging sub-models without loss of information. The four submodels are shown on the left panels. As it can be seen on the right panel, the merged model (blue lines and shaded area) as well as the full model (red dashed lines) cannot be distinguished.

Aggregated process

We now focus on the case where (Y, M) is a centred Gaussian process with given covariances

$$\mathrm{Cov}\left[(Y(x), M(x)), (Y(x'), M(x'))\right] = \begin{pmatrix} k(x,x') & k_M(x,x')^t \\ k_M(x',x) & K_M(x,x') \end{pmatrix}. \tag{3}$$

Definition (aggregated process)

We define the process $Y_{1\oplus\ldots\oplus p}$ as

$$\mathbf{Y_{1\oplus\ldots\oplus p}} = \mathbf{M_{1\oplus\ldots\oplus p}} + \varepsilon'_{\mathbf{1\oplus\ldots\oplus p}} \tag{4}$$

where $\varepsilon'_{1\oplus\ldots\oplus p}$ is an independent replicate of $Y - M_{1\oplus\ldots\oplus p}$.

Some properties for aggregated process

- **known distribution** : $Y_{1\oplus\ldots\oplus p}$ is centred with known covariances

$$\begin{aligned} k_{1\oplus\ldots\oplus p}(x,x') = k(x,x') + 2k_M(x)^t k_M^{-1}(x)k_M^{-1}(x,x')k_M^{-1}(x')k_M(x') \\ - k_M(x)^t k_M^{-1}(x)k_M(x',x) - k_M(x')^t k_M^{-1}(x')k_M(x,x'). \end{aligned} \tag{5}$$

- **optimality** : If $M_{1\oplus\ldots\oplus p}(x)$ writes $M_{1\oplus\ldots\oplus p}(x) = \lambda_{1\oplus\ldots\oplus p}(x)^t Y(X)$ and if $M_{1\oplus\ldots\oplus p}(X) = Y(X)$ then

$$\begin{aligned} M_{1\oplus\ldots\oplus p}(x) &= \mathrm{E}\left[Y_{1\oplus\ldots\oplus p}(x) | Y_{1\oplus\ldots\oplus p}(X)\right] \\ v_{1\oplus\ldots\oplus p}(x) &= \mathrm{V}\left[Y_{1\oplus\ldots\oplus p}(x) | Y_{1\oplus\ldots\oplus p}(X)\right]. \end{aligned}$$

- **full model recovery** : If $M(x) = \Lambda(x)Y(X)$ where $\Lambda(x)$ is an invertible matrix, then

$$Y_{1\oplus\ldots\oplus p} \overset{law}{=} Y \text{ and thus } Y_{1\oplus\ldots\oplus p} | Y_{1\oplus\ldots\oplus p}(X) \overset{law}{=} Y | Y(X). \tag{6}$$

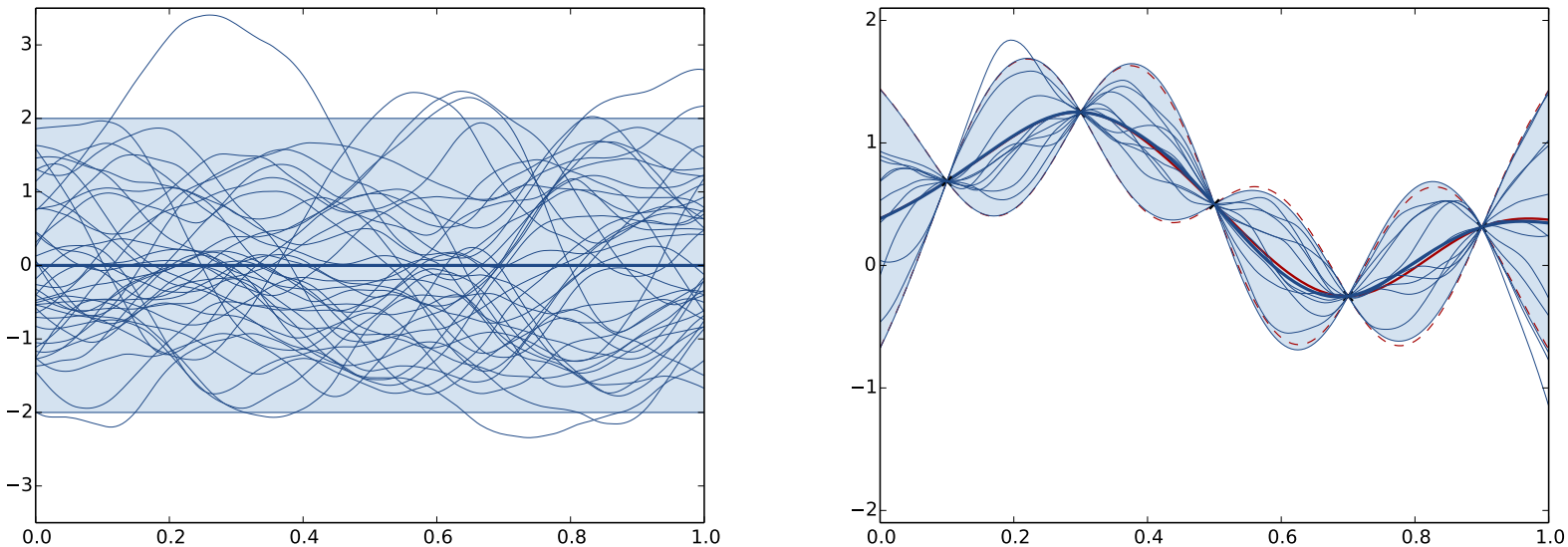
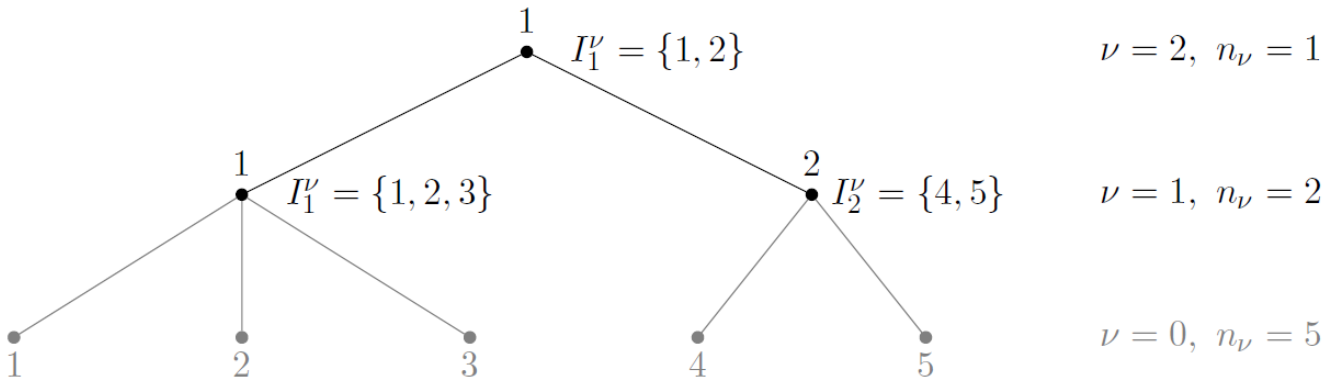


FIGURE: Interpretation of the results from Example 2 as a posterior Gaussian process distribution. The left panel shows the prior $Y_{1\oplus\ldots\oplus p}$ and the right one the conditional distribution given $Y_{1\oplus\ldots\oplus p}(X) = Y(X)$.

Iterative model



From $\begin{cases} (M^\nu(x))_i = M_i^\nu(x) \\ (k^\nu(x))_i = \mathrm{Cov}\left[Y(x), M_i^\nu(x)\right] \\ (K^\nu(x))_{ij} = \mathrm{Cov}\left[M_i^\nu(x), M_j^\nu(x)\right] \end{cases}$ get $\begin{cases} (M^{\nu+1}(x))_i = \alpha_i^{\nu+1}(x)^t \left(M^\nu(x)_{[I_i^{\nu+1}]}\right) \\ (k^{\nu+1}(x))_i = \alpha_i^{\nu+1}(x)^t \left(k^\nu(x)_{[I_i^{\nu+1}]}\right) \\ (K^{\nu+1}(x))_{ij} = \alpha_i^{\nu+1}(x)^t \left(K^\nu_{[I_i^{\nu+1}, I_j^{\nu+1}]}\right) \alpha_j^{\nu+1}(x) \end{cases}$

with vectors of optimal weights $\alpha_i^{\nu+1}(x) = \left(K^\nu_{[I_i^{\nu+1}, I_i^{\nu+1}]}\right)^{-1} \left(k^\nu(x)_{[I_i^{\nu+1}]}\right)$.

Algorithm 1: Iterative kriging algorithm

inputs : M_1, vector of length n_1 (sub-models evaluated at x)
k_1, vector of length n_1 (covariance between $Y(x)$ and sub-models at x)
K_1, matrix of size $n_1 \times n_1$ (covariance between sub-models at x)
I, a list describing the tree structure
outputs: $M_{\nu_{\max}}$, $K_{\nu_{\max}}$

for $\nu = 2, \ldots, \nu_{\max}$ **do**
  for $i = 1, \ldots, n_\nu$ **do**
    $M \leftarrow$ subvector of $M_{\nu-1}$ on I_i^ν
    $K \leftarrow$ submatrix of $K_{\nu-1}$ on I_i^ν
    if $\nu = 2$ **then** $k \leftarrow k_1$ **else** $k \leftarrow \mathrm{Diag}(K)$
    $\alpha_i \leftarrow K^{-1} k$
    $M_\nu[i] \leftarrow (\alpha_i)^t M$
    $K_\nu[i,i] \leftarrow (\alpha_i)^t k$
    for $j = 1, \ldots, i-1$ **do**
      $K \leftarrow$ submatrix of $K_{\nu-1}$ on $I_i^\nu \times I_j^\nu$
      $K_\nu[i,j] \leftarrow (\alpha_i)^t K \alpha_j$
      $K_\nu[j,i] \leftarrow K_\nu[i,j]$

Under some conditions : Algorithm complexity $\mathbf{O(n_p n^2)}$. Storage footprint $\mathbf{O(n_1^2)}$.
n number of observations, n_p number of prediction points, n_1 number of submodels.

Methods consistency

Deisenroth and Ng 2015, Cao and Fleet 2014 and Van Stein et al 2015 propose aggregations of the form

$$M_{1\oplus\ldots\oplus p}(x) = \sum_{k=1}^{p} \alpha_k(x) M_k(x),$$

where $\alpha_k(x)$ increases with $1/v_k(x)$
$\Longrightarrow$ The aggregation cost is negligible !
$\Longrightarrow$ However, we show the following negative result

Proposition

Let the observation domain $\mathcal{X}$ be fixed and bounded, let $x_0 \in \mathcal{X}$ be fixed and let $N, p \to \infty$. For a standard class of covariance functions, with the aggregation methods above, there exists a dense triangular array of observation points so that

$$\liminf_{N,p\to\infty} \mathbb{E}\left(\{Y(x_0) - M_{1\oplus\ldots\oplus p}(x_0)\}^2\right) > 0$$

$\Longrightarrow$ On the contrary, our proposed aggregation method yields a consistent predictor. (Note that many simple predictors are consistent !)

Parameter estimation

Parametric covariance model

Set of covariance functions

$$\{\sigma^2 k_\theta, \sigma^2 \geq 0, \theta \in \Theta\}$$

with $\Theta \subset \mathbb{R}^m$
Yields predictors and predictive variances

$$m_{1\oplus\ldots\oplus p,\theta}(x)$$

and

$$v_{1\oplus\ldots\oplus p,\sigma^2,\theta}(x)$$

Goal : $\hat{\theta}$ and $\hat{\sigma}^2$

Stochastic gradient for $\hat{\theta}$

Let $M_{1\oplus\ldots\oplus p,\theta,-i}(x_i)$ be the Leave One Out prediction of y_i based on the $n-1$ remaining points

We want to use the Leave One Out estimator

$$\hat{\theta} \in \underset{\theta\in\Theta}{\operatorname{argmin}} \frac{1}{n}\sum_{i=1}^{n}\left\{M_{1\oplus\ldots\oplus p,\theta,-i}(x_i) - y_i\right\}^2$$

Computing q Leave One Out errors costs $O(qn^2)$ flops $\Longrightarrow$ stochastic gradient :

$$\theta_{k+1} = \theta_k - a_k h \left\{\frac{1}{\epsilon_k}\left(\frac{1}{q}\sum_{i\in I}\left(M_{1\oplus\ldots\oplus p,\theta+\epsilon_k h,-i}(x_i) - y_i\right)^2 - \frac{1}{q}\sum_{i\in I}\left(M_{1\oplus\ldots\oplus p,\theta-\epsilon_k h,-i}(x_i) - y_i\right)^2\right)\right\}$$

where I is a random sample of size q and h is a random direction

$\Longrightarrow$ Stochastic gradient is not worth it for the exact Gaussian process prediction ($O(n^3)$ cost for q error computations) but is useful with our aggregation method

Comments on stochastic gradient

- A few hours with $n = 10,000$ and $d = 10$ for one descent ($q = 100$, 500 iterations)
- Convergence guaranteed theoretically (and verified numerically) for some step size sequences
- Some other sequences with less theoretical guarantees can work well in practice
- With n large, the impact of the starting point could be limited

Estimation of σ^2

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} \frac{\left(y_i - m_{1\oplus\ldots\oplus p,-i,\hat{\theta}}(x_i)\right)^2}{v_{1\oplus\ldots\oplus p,-i,1,\hat{\theta}}(x_i),}$$

which is equivalent to

$$\frac{1}{n}\sum_{i=1}^{n} \frac{\left(y_i - m_{1\oplus\ldots\oplus p,-i,\hat{\theta}}(x_i)\right)^2}{v_{1\oplus\ldots\oplus p,-i,\hat{\sigma}^2,\hat{\theta}}(x_i)} = 1.$$

Application to a case study

The case study

- Data provided by EDF (Geraud Blatman)
- $10,000$ input-outputs $(x_i, f(x_i))$, with $d = dim(x_i) = 6$ and

$$f(x_i) = \log\left[\sum_{j=1}^{m}(F(x_i, c_j) - m_j)^2\right]$$

with

- c_j : experimental condition
- F : code
- x_i : code parameter
- m_j experimental value

Settings of the case study

- $n = 9000$ data points in the learning base
- $n_t = 1000$ data points in the test base
- One aggregation by our method or the "sum-based" aggregation methods
- $p = 20$ or $p = 90$ aggregated subsamples
- Subsamples chosen with K-means or randomly
- Covariance functions : exponential, Matérn 3/2, Matérn 5/2. Ordinary Kriging
- Covariance parameters chosen by our proposed stochastic gradient method or by minimizing the sum of the likelihoods over the subsamples

Prediction criteria

- MSE (should be minimal)

$$MSE = \frac{1}{n_t} \sum_{i=1}^{n_t} (m_{1\oplus\ldots\oplus p,\hat{\theta}}(x_{t,i}) - f(x_{t,i}))^2,$$

- MNSE (should be close to 1)

$$MSNE = \frac{1}{n_t} \sum_{i=1}^{n_t} \frac{(m_{1\oplus\ldots\oplus p,\hat{\theta}}(x_{t,i}) - f(x_{t,i}))^2}{v_{1\oplus\ldots\oplus p,\hat{\sigma}^2,\hat{\theta}}(x_{t,i})},$$

- CIR (should be close to 0.9)

$$CIR = \frac{1}{n_t} \sum_{i=1}^{n_t} \mathbf{1}\left\{ |m_{1\oplus\ldots\oplus p,\hat{\theta}}(x_{t,i}) - f(x_{t,i})| \leq 1.645 \sqrt{v_{1\oplus\ldots\oplus p,\hat{\sigma}^2,\hat{\theta}}(x_{t,i})} \right\},$$

- MNLP (should be minimal)

$$= \frac{1}{n_t} \sum_{i=1}^{n_t} \left(\frac{1}{2} \log(2\pi v_{1\oplus\ldots\oplus p,\hat{\sigma}^2,\hat{\theta}}(x_{t,i})) + \frac{(m_{1\oplus\ldots\oplus p,\hat{\theta}}(x_{t,i}) - f(x_{t,i}))^2}{2 v_{1\oplus\ldots\oplus p,\hat{\sigma}^2,\hat{\theta}}(x_{t,i})} \right),$$

Prediction results (1/3)

	SPV	PoE	gPoE1	gPoE2	BCM	rBCM	Opt
MSE (log lik)	0.0068	**0.00455**	0.00456	**0.00455**	0.00467	0.00456	0.00459
MNSE (log lik)	343	382	1780	**19.1**	384	1780	372
CIR (log lik)	0.482	0.224	0.177	**0.636**	0.193	0.197	0.36
MNLP (log lik)	167	186	884	**6.22**	187	884	181
MSE (loo)	0.00394	0.00741	**0.00276**	0.00741	0.0471	0.00832	0.00283
MNSE (loo)	0.602	**1.26**	2.38	0.0632	7.09	4.88	0.687
CIR (loo)	0.939	**0.894**	0.851	1.00	0.224	0.546	0.936
MNLP (loo)	-1.38	-1.31	-1.31	-0.409	1.61	-0.0511	$\mathbf{-1.53}$

TABLE: $p = 20$ aggregated subsamples selected with the K means algorithm. Matérn 5/2 covariance function.

Prediction results (2/3)

	SPV	PoE	gPoE1	gPoE2	BCM	rBCM	Opt
MSE (log lik)	0.00815	0.0105	0.00561	0.0105	0.00811	**0.00487**	0.00567
MNSE (log lik)	139	339	802	**3.77**	259	723	206
CIR (log lik)	0.509	0.031	0.037	**0.74**	0.047	0.153	0.259
MNLP (log lik)	65.3	165	396	**−0.242**	125	357	98.9
MSE (loo)	0.00695	0.0448	0.00681	0.0448	0.0648	0.0083	**0.00452**
MNSE (loo)	0.614	6.92	4.14	0.0769	9.78	4.57	**0.937**
CIR (loo)	0.951	0.262	0.703	1.00	0.106	0.623	**0.922**
MNLP (loo)	−1.05	1.63	−0.299	0.461	3.07	−0.0764	**−1.3**

TABLE: $p = 90$ aggregated subsamples selected with the K means algorithm. Matérn 5/2 covariance function.

Prediction results (3/3)

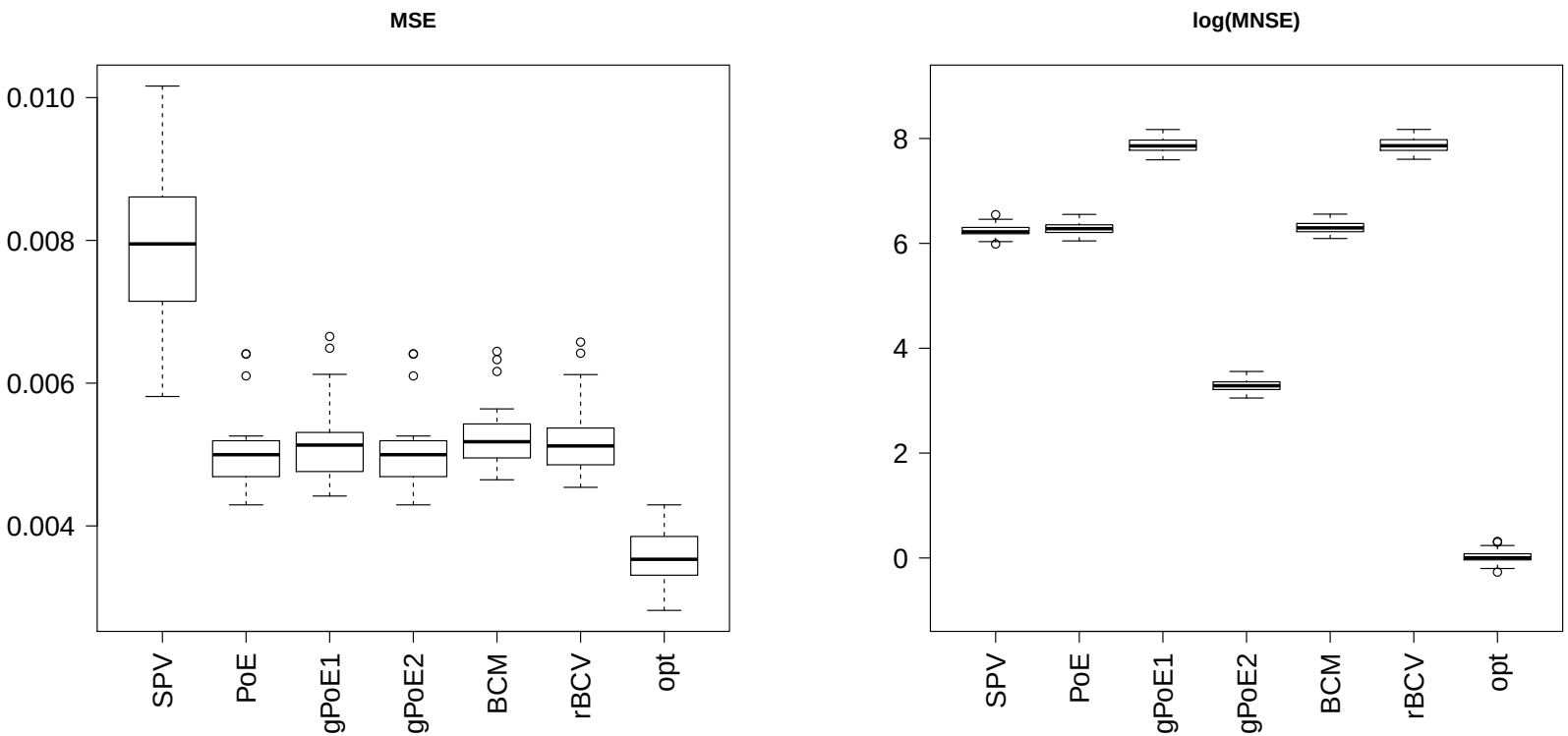


FIGURE: Box plot of 20 values of MSE and log(MNSE) where 20 learning and test sets are randomly generated. $p = 20$ subsamples obtained from the K means algorithm ; Matérn 5/2 covariance function. Covariance parameters estimated by log lik for SPV, PoE, gPoE1, gPoE2, BCM and rBCM and by LOO for our aggregation procedure.

Thank you for your attention !