Uncertainty quantification for inverse problems in random neutronics using supervised learning and Monte-Carlo Markov chain methods

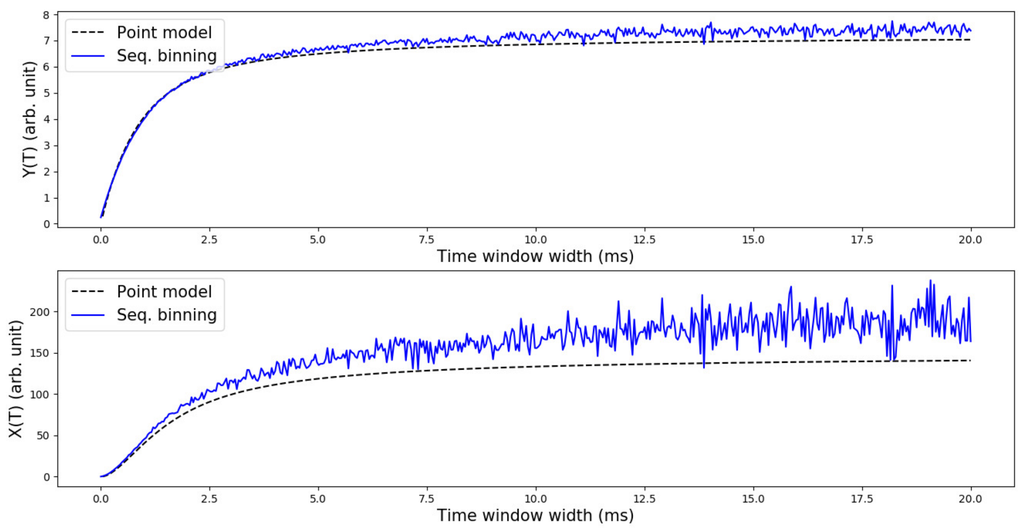
Paul Lartaud[1], Philippe Humbert[1] and Josselin Garnier[2]

Introduction and motivations

- Neutron noise describes a set of non-destructive examinations techniques based on the study of neutron correlations in a multiplying medium. It can be used to identify an unknown fissile material.
- True correlations occur when neutrons arise from the same fission chain. On the other hand, accidental correlations are simultaneous detections of independent neutrons.
- In practical measurements, accidental correlations cause very noisy data acquisition. Inverse problem resolution and uncertainty quantification is made difficult due to the non-linearity of the underlying processes.
- An analytical direct model based on strong physical assumptions can be used in combination to adaptive MCMC methods to solve the inverse problem using a Bayesian approach.
- The inherent model error from the point model assumptions leads to a bias in the posterior distribution. Our goal is to learn a more flexible surrogate model based on multi-output Gaussian process regression and BNN.
- Coupling between the surrogate model and the MCMC sampling can be done to account for measure and model errors. For that purpose, the covariance of the surrogate predictions are injected into the MCMC methods.

1. Neutron correlation measurements

- Numerical measurements of neutron correlations are obtained by Monte-Carlo simulations in the code MCNP6.
- These measurements are used to solve an inverse problem to identify material characteristics θ from neutron correlation measurements y, assuming Gaussian noise $y = f(\theta) + \varepsilon$ with $\varepsilon \sim \mathcal{N}(0, \mathbf{C}_{meas})$
- Each measurement consist of three outputs $y = (R, Y_\infty, X_\infty)$ where Y_∞ and X_∞ are the second and third asymptotic Feynman moments measuring the average number of correlated double and triple detections, and R is the average count rate.
- The second and third asymptotic Feynman moments Y_∞ and X_∞ measure the average number of correlated double and triple detections. The average count rate R is measured as well.



- The systematic bias of the point model due to the strong physical assumptions can be visualised on this figure. Our goal is to reduce the bias by building a surrogate model based on Gaussian process regression.
- MCNP6 is used to create a data set by changing the geometry of the MCNP model (material densities, thicknesses, compositions ...). The data set contains 1125 cases. It is preprocessed with Box-Cox transformations [1].

2. Multi-Output Gaussian process regression

- A scalar Gaussian process [2] is a collection of random variables such that any finite subset follows a multivariate normal distribution. It is completely defined by its mean function $m(x)$ and covariance function $k(x, x')$.

$$f \sim \mathcal{GP}\left(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\right)$$

- Let us consider a zero-mean Gaussian process $f \sim \mathcal{GP}\left(0, k(\mathbf{x}, \mathbf{x}')\right)$. Let $\mathbf{X} \in \mathbb{R}^{N \times I}$ and $\mathbf{y} \in \mathbb{R}^N$ be respectively N training inputs and outputs, and I is the input dimension. Similarly, let $\mathbf{X}_* \in \mathbb{R}^{N \times I}$ and $\mathbf{y}_* \in \mathbb{R}^N$ be respectively N test inputs and outputs. We assume noisy observations $y = f(x) + \varepsilon$ accounted for by adding a nugget noise variance σ_n^2 in the GP covariance.
- The prediction prediction on the test outputs is given by $\mathbf{f}_* | \mathbf{f}, \mathbf{X}, \mathbf{X}_* \sim \mathcal{N}(\mu_C, K_C)$.

$$\mu_C = K(\mathbf{X}, \mathbf{X}_*)^T K(\mathbf{X}, \mathbf{X})^{-1} \mathbf{f}$$
$$K_C = K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}, \mathbf{X}_*)^T (K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathcal{I})^{-1} K(\mathbf{X}, \mathbf{X}_*)$$

- The Gaussian process is trained by maximizing the marginal likelihood to adjust the covariance kernel hyperparameters.

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^T (K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathcal{I})^{-1} \mathbf{y} - \frac{1}{2} \log |(K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathcal{I})| - \frac{I}{2} \log(2\pi)$$

- Three independent GPs are trained to predict the outputs (R, Y_∞, X_∞) of our model. A Matérn 5/2 kernel is chosen. The optimization of hyperparameters is done with Adam algorithm.
- Now let us consider D-dimesional outputs such that our training outptuts are $\mathbf{y} \in \mathbb{R}^{N \times D}$. To take advantage of intrinsic correlations between outputs, multi-output GP regression is considered. In order to build a multi-dimensional kernel, two methods are studied : linear coregionalization and convolutional GPs [3].
- Linear coregionalization create correlation between Q independent Gaussian processes, defined by their covariance function $k_q(x, x')$ for $q \leq Q$, by introducing a mixing matrix $W \in \mathcal{M}_{D,Q}(\mathbb{R})$ whose coefficients are additional hyperparameters to be optimized. The multioutput GP is given by :

$$f_d(\mathbf{x}) = \sum_{j=1}^{Q} w_{i,j} u_j(\mathbf{x}) \tag{1}$$

In our case, the covariance kernels are optimized separately, and the diagonal coefficients of W are set to 1 since a variance hyperparameter is already included in each k_q for $q \leq Q$. A multioutput covariance kernel can then be built.

$$\mathrm{Cov}(f_d(\mathbf{x}), f_{d'}(\mathbf{x}')) = \sum_{q=1}^{Q} w_{d,q} w_{d',q} k_q(\mathbf{x}, \mathbf{x}') \tag{2}$$

- Similarly, a convolutional GP can be built.

$$f_d(\mathbf{x}) = \sum_{q=1}^{Q} \int G_{d,q}(\mathbf{x} - \mathbf{z}) u_q(\mathbf{x}) d\mathbf{z} \tag{3}$$

where u_q are the independent latent scalar GPs and $G_{d,q}$ are convolutional filters for $d \leq D$ and $q \leq Q$. In order for the integrals to be tractable, the latent covariance kernels are chosen as RBF kernel with lengthscales Λ_q and the convolutional filters are chosen Gaussian with diagonal covariance matrix. Thus, $G_{d,j}(\mathbf{x})$ is the density of a Gaussian random variable $\mathcal{N}(\mathbf{x}, P_d^{-1})$ with a multiplicative constant $S_{d,j}$. The covariance kernel can then be obtained.

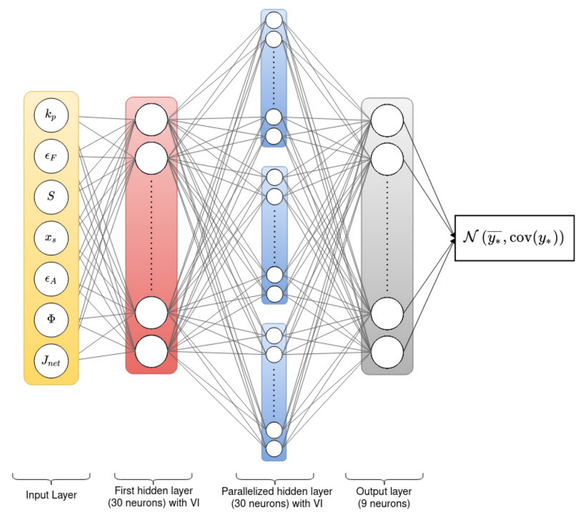
$$\mathrm{Cov}(f_d(\mathbf{x}), f_{d'}(\mathbf{x}')) = \sum_{j=1}^{Q} \frac{S_{d,j} S_{d',j} V_j}{(2\pi)^{I/2} |C_{j,d,d'}|^{1/2}} exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T C_{j,d,d'}^{-1} (\mathbf{x} - \mathbf{x}')\right) \tag{4}$$

with $C_{j,d,d'} = P_d^{-1} + P_{d'}^{-1} + \Lambda_j^{-1}$.

- These different GPs are trained on a set of data created from MCNP6 simulations, with 450 training points. The test set contains 675 points.

3. Bayesian neural network

- Bayesian neural networks are also tested to serve as a surrogate model. The BNN is trained on the same dataset for 3000 epochs. The architecture of the BNN is described on the right.
- The BNN is considered to provide additional flexibility. The output of the BNN is a multivariate distribution, such that it can be used to provide a mean prediction, and its covariance.
- The priors on the weights are independent normal distributions $\mathcal{N}(0, 1)$. Each hidden layer neuron uses a sigmoid activation function.
- The posterior distribution of the weights is approximated with Variational Inference.



4. Surrogate models performance

- Predictions are largely improved with surrogate models. Independent GPs provide good predictions, but are unable to predict covariance of the outputs.
- The Linear Model of Coregionalization (LMC) provides better performance than Convolutional GPs. This may be caused by the use of Gaussian filters and RBF kernels in Convolutional GPs.

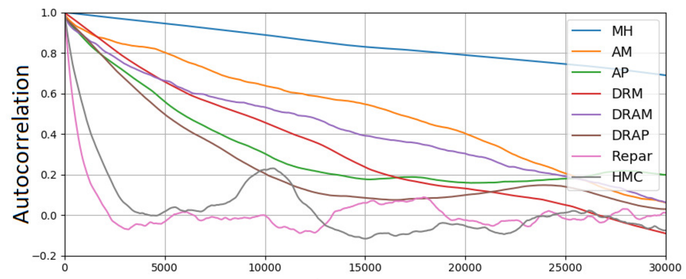
Point Model	MAE	MSE	MAPE	Prob. coverage
Count Rate	5.61×10^3	4.56×10^7	22.3%	–
Second Feynman	2.28×10^{-1}	2.03×10^0	8.89%	–
Third Feynman	3.70×10^1	2.06×10^5	15.5%	–
Independent GPs	**MAE**	**MSE**	**MAPE**	**Prob. coverage**
Count Rate	2.02×10^2	1.31×10^5	0.83%	95.7%
Second Feynman	6.02×10^{-2}	1.06×10^{-1}	2.77%	93.1%
Third Feynman	8.89×10^0	1.46×10^4	9.50%	92.2%
LMC - 2 latent GPs	**MAE**	**MSE**	**MAPE**	**Prob. coverage**
Count Rate	2.11×10^2	1.06×10^5	0.92%	94.8%
Second Feynman	6.67×10^{-2}	5.65×10^{-2}	3.41%	87.9%
Third Feynman	5.93×10^0	2.60×10^3	9.53%	93.2%

LMC - 3 latent GPs	**MAE**	**MSE**	**MAPE**	**Prob. coverage**
Count Rate	1.65×10^2	8.91×10^4	0.91%	93.8%
Second Feynman	6.36×10^{-2}	1.22×10^{-1}	2.61%	93.4%
Third Feynman	8.91×10^0	1.35×10^4	8.83%	94.1%
Convolutional GPs	**MAE**	**MSE**	**MAPE**	**Prob. coverage**
Count Rate	8.05×10^2	3.31×10^6	4.94%	96.6%
Second Feynman	1.32×10^{-1}	2.74×10^{-1}	6.19%	97.5%
Third Feynman	5.04×10^0	7.35×10^5	22.5%	97.5%
Bayesian NN	**MAE**	**MSE**	**MAPE**	**Prob. coverage**
Count Rate	5.55×10^2	3.39×10^6	2.21%	92.4%
Second Feynman	1.55×10^{-1}	1.47×10^0	3.65%	93.3%
Third Feynman	1.87×10^1	3.86×10^4	9.01%	95.5%

- Covariance estimates are robust for all methods, although LMC with 2 latent processes provide less robust estimates. On the other hand, LMC with 3 latent processes displays good performance.

5. Posterior distribution sampling

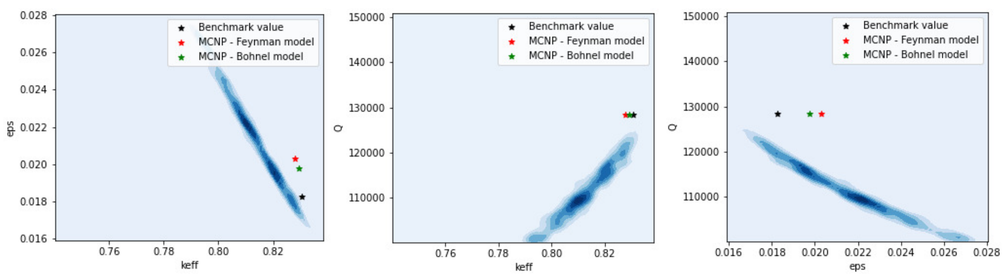
- Forward model is strongly non-linear and leads to degenerate target distributions. The decorrelation times for different MCMC methods are shown below. The lower the decorrelation time, the faster the convergence towards the invariant distribution.



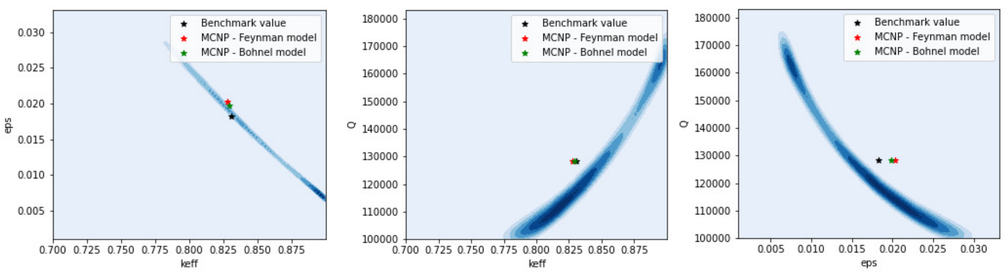
- When using surrogate models, the systematic model bias can be accounted for by adding the predicted covariance to the likelihood.

$$L(\mathbf{y}|\theta) \propto \prod_{i=1}^{N} \exp\left(-\frac{1}{2}(y_i - f(\theta))^T (\mathbf{C_{meas}} + \mathrm{Cov}(\theta))^{-1} (y_i - f(\theta))\right)$$

- The test case considered is taken from the ICSBEP Handbook [4]. The posterior distribution is sampled with Adaptive Metropolis [5] with 2×10^6 iterations, a burn-in phase of 5000 steps.
- The 2D marginal densities are presented using the point model only, and using the LMC with 3 latent processes.
- With the point model, the posterior distribution is thin and the real values are outside the sampled distribution due to the inherent model bias.



- In the second case, the posterior distribution is broader and the theoretical values of the parameters are closer to the distribution, which accounts for the systematic model error. This surrogate model improves the estimation of the posterior $p(\theta|\mathbf{y})$, although some residual bias appear due to the difference between the test example and our training data.



Conclusion and prospects

- Surrogate models based on multioutput Gaussian process regression with Linear Coregionalization allow for improved neutron correlation predictions, and provide a covariance estimation for the predictions.
- MCMC sampling coupled with the surrogate model predictions covariance makes it possible to better account for systematic model bias and aleatoric error of the neutron correlation measurements.
- The coupling between MCMC and BNN is yet to be implemented.
- Some residual model biases are still present. Surrogate model predictions and robustness need to be improved.

References

[1] R. M. Sakia, "The box-cox transformation technique: a review," *Journal of the Royal Statistical Society: Series D (The Statistician)*, vol. 41, no. 2, pp. 169–178, 1992.
[2] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer school on machine learning*, pp. 63–71, Springer, 2003.
[3] E. V. Bonilla, K. Chai, and C. Williams, "Multi-task gaussian process prediction," *Advances in neural information processing systems*, vol. 20, 2007.
[4] J. B. Briggs, L. Scott, and A. Nouri, "The international criticality safety benchmark evaluation project," *Nuclear science and engineering*, vol. 145, no. 1, pp. 1–10, 2003.
[5] H. Haario, E. Saksman, and J. Tamminen, "An adaptive metropolis algorithm," *Bernoulli*, pp. 223–242, 2001.

[1] CEA, DAM, DIF, F-91 297 Arpajon, France [2] Centre de Mathématiques Appliquées (CMAP), École polytechnique, 91128 Palaiseau Cedex, France paul.lartaud@polytechnique.edu