A new exact and scalable model for multitask gaussian process regression : application to the reconstruction of nuclear data in neutronics codes

Olivier TRUFFINET : CEA/DES/ISAS/DM2S/SERMA/LPEC

Supervisors : Karim AMMAR, Bertrand BOURIQUET, Jean-Philippe ARGAUD, Nicolas GERARD CASTAING

A new exact and scalable model for multitask gaussian process regression

1. Multitask gaussian processes

- Reminders on gaussian processes
- The LMC framework
- Common simplifications

2. The projected LMC

- Decoupled expressions
- Resolution : noise model and loss
- Experimental results

3. Application : nuclear data in deterministic neutronics

- Description of the problem
- Results
- Corollary : adaptive sampling

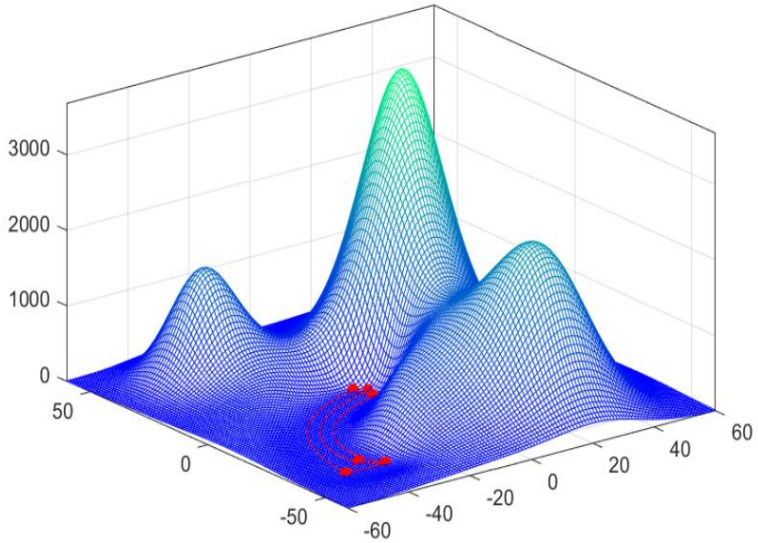
4. Conclusion & Prospects

1. Multitask GP

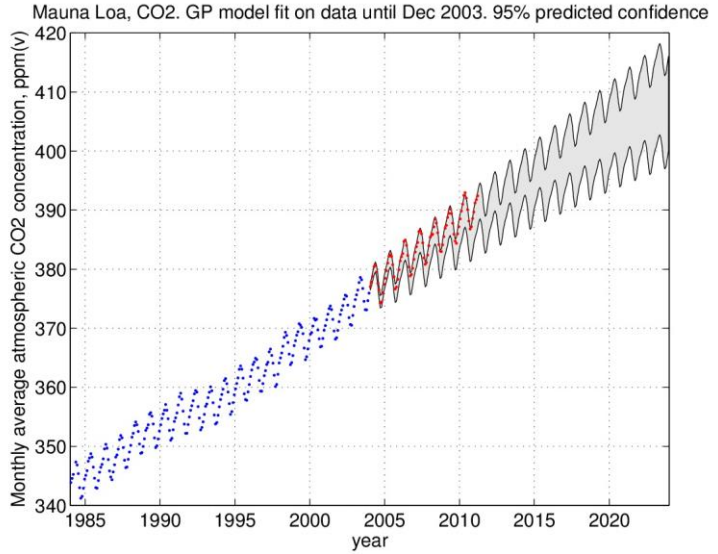
Elements on single-output and multi-ouput gaussian processes

Introduction to gaussian processes

- Very popular methods for regression, classification, optimization…
- Apparented methods : Radial Basis Interpolation, Kernel regression, kriging (+Support Vector Regression, splines)
- Advantages :
  - ❖ Accuracy
  - ❖ No support restriction, trivial handling of multidimensionality
  - ❖ Flexibility (choice of kernels, mean functions…)
  - ❖ Safe optimization with maximum likelihood estimation
  - ❖ Uncertainty metrics thanks to the Bayesian framework
  - ❖ « Non-parametric »
- Flaws : Scalability? **Not so sure…**



Source: T.X.Lin & al, *A Distributed Scalar Field Mapping Strategy for Mobile Robots*



Source: C.E.Rasmussen, personal website

Definition

$\boldsymbol{\theta}$: *parameters of mean, kernel and noise*

- Definition : a gaussian process is a **random variables field** over $\mathbb{R}^d$, such that each of its subsets follows a joint Gaussian distribution. It is entirely defined by its mean function $m(\boldsymbol{x})$ and its covariance function (*kernel*) $k(\boldsymbol{x}, \boldsymbol{x}')$:

$$\begin{cases} \mathbb{E}[f(\boldsymbol{x})] = m(\boldsymbol{x}) \\ Cov[f(\boldsymbol{x}), f(\boldsymbol{x}')] = k(\boldsymbol{x}, \boldsymbol{x}') \end{cases}$$

$f(\boldsymbol{x})$: function to interpolate

- To estimate the process at an unobserved point $\boldsymbol{x}_*$, one expresses the joint probability of known observations $f(\boldsymbol{X})$ and unknown value $f(\boldsymbol{x}_*)$:

$$\begin{bmatrix} \boldsymbol{y} \\ f(\boldsymbol{x}_*) \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} m(\boldsymbol{X}) \\ m(\boldsymbol{x}_*) \end{bmatrix}, \begin{bmatrix} k(\boldsymbol{X}, \boldsymbol{X}) + \sigma^2 I & k(\boldsymbol{x}_*, \boldsymbol{X}) \\ k(\boldsymbol{x}_*, \boldsymbol{X}) & k(\boldsymbol{x}_*, \boldsymbol{x}_*) \end{bmatrix} \right)$$

$\boldsymbol{y} = f(\boldsymbol{X})$ + noise

- One can then condition over $f(\boldsymbol{X})$ to obtain the conditional distribution of $f(\boldsymbol{x}_*)$:

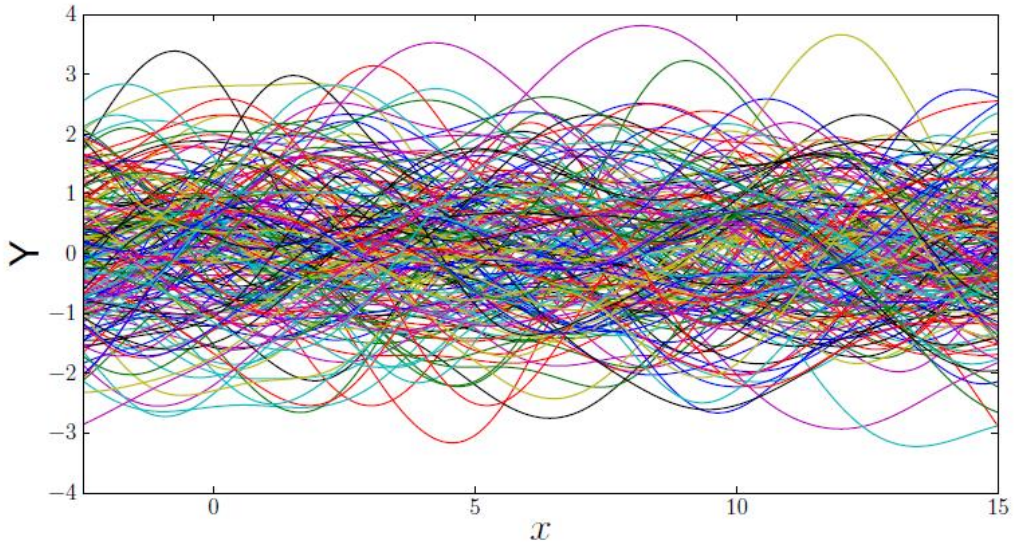
$$f(\boldsymbol{x}_*)|(\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{y} \sim \mathcal{N}(k(\boldsymbol{x}_*, \boldsymbol{X})(k(\boldsymbol{X}, \boldsymbol{X}) + \sigma^2 I)^{-1}\boldsymbol{y}, \; k(\boldsymbol{x}_*, \boldsymbol{x}_*) - k(\boldsymbol{x}_*, \boldsymbol{X})(k(\boldsymbol{X}, \boldsymbol{X}) + \sigma^2 I)^{-1} k(\boldsymbol{x}_*, \boldsymbol{X}))$$

(here in the case $m(\boldsymbol{x}) = 0$)

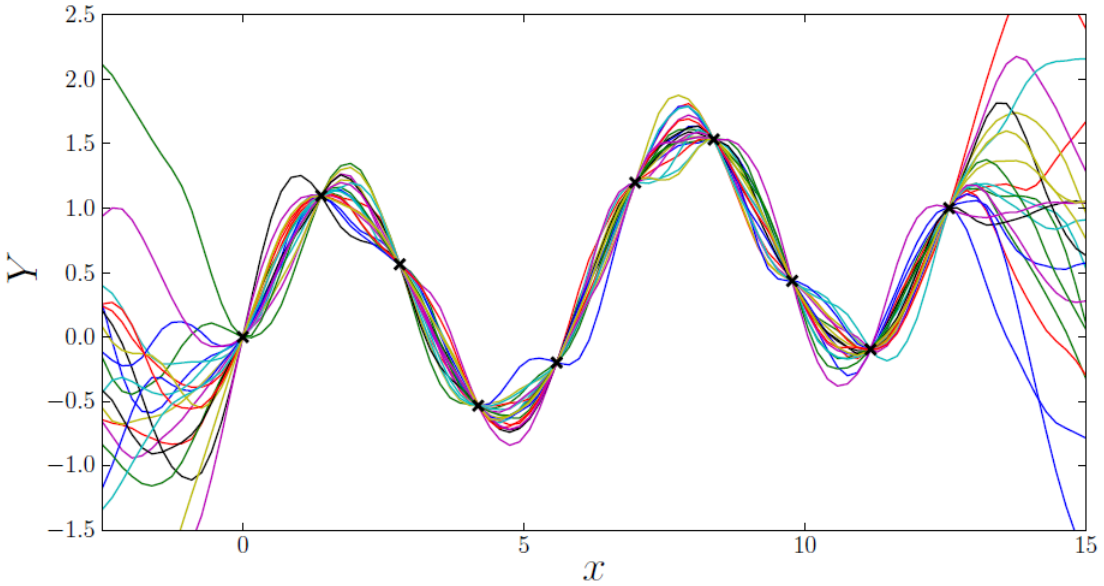
Mean *Variance*

- Marginal log-likelihood (loss function) : $\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) = -\frac{1}{2}\boldsymbol{y}^T k(\boldsymbol{X}, \boldsymbol{X})^{-1}\boldsymbol{y} - \frac{1}{2}\log|k(\boldsymbol{X}, \boldsymbol{X})| + cste$

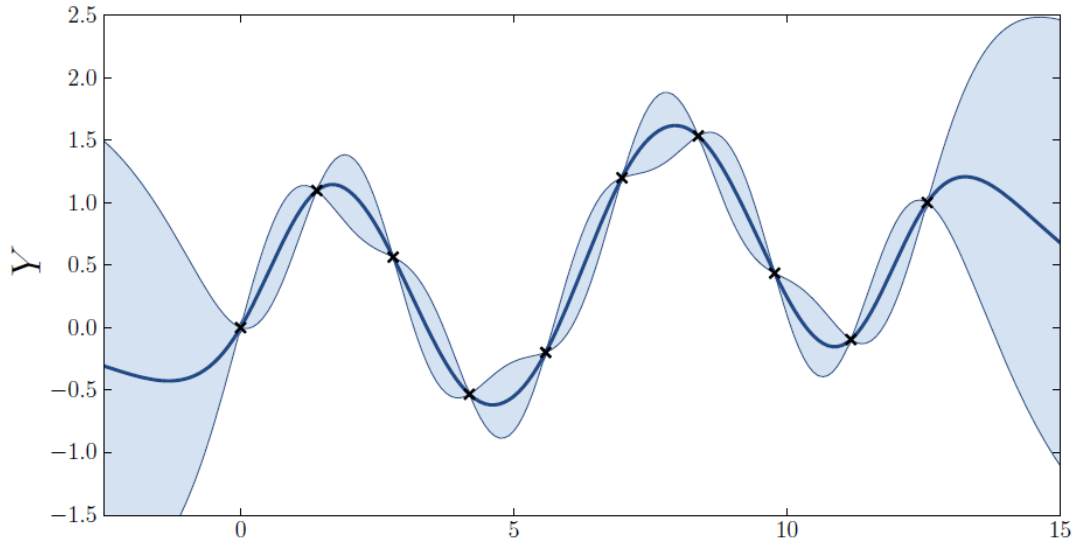
Since $f()$ in unknown, we make the general assumption that it is the sample path of a Gaussian process $Y \sim \mathcal{N}(\mu(), k(,))$:



If we remove all the samples that do not interpolate the observations we obtain:



It can summarized by a mean function and 95% confidence intervals.



Source : Le Riche & Durrande, *An overview of kriging for researchers*

Each curve represents a realization of the GP.

The modeled function is supposed to be such a realization.

Kernel cookbook

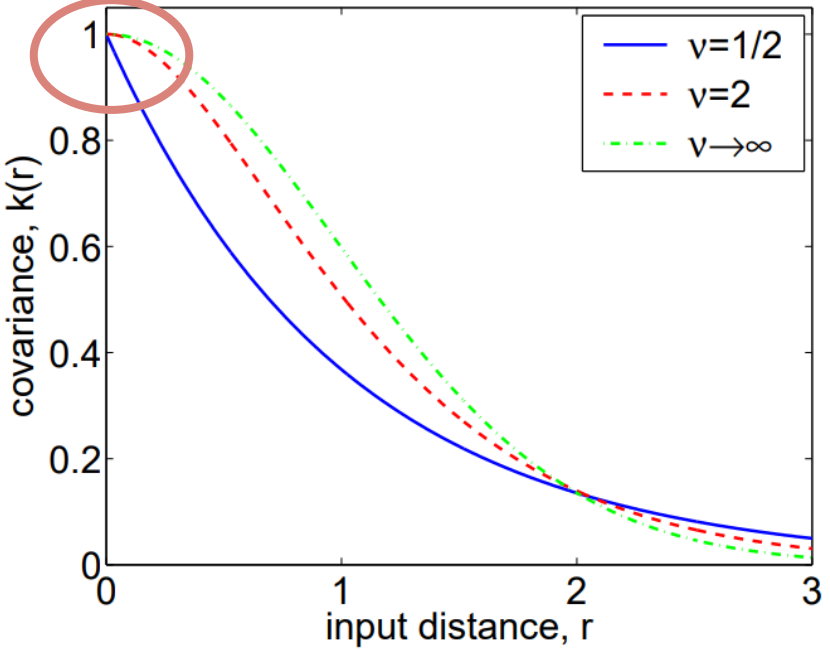
- Stationnary kernels : $k(\boldsymbol{x}, \boldsymbol{x}') = \psi(\|\boldsymbol{x} - \boldsymbol{x}'\|) = \psi(r)$. Highly recommanded unless the data is strongly anisotropic

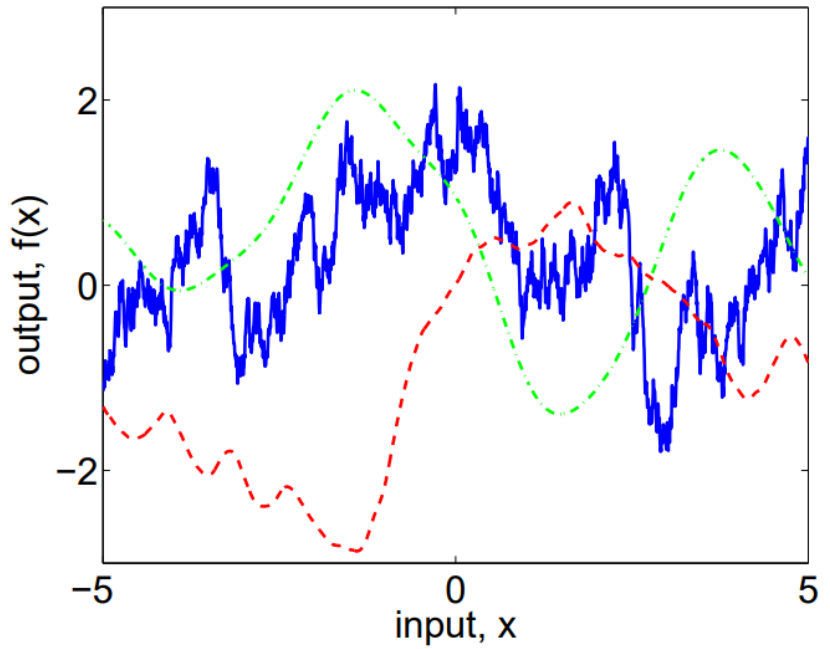
- Standard sationnary kernels : $\psi(r) = e^{-r^2}$ (*squared exponential*), $\psi_\alpha(r) = (1 + \frac{r^2}{2\alpha})^{-1}$ (*rational quadratic* with parameter α), $\psi_\nu(r) = 2^{1-\nu}\left(\sqrt{2\nu}r\right)^\nu K_\nu\left(\sqrt{2\nu}r\right)$ (*Matérn* kernel with parameter ν)…

- The regularity of the estimator is controlled by the derivatives of ψ at $r = 0$ (see figures opposite for processes obtained for Matérn kernels with different values of ν)

- The distance r between two points can be customized, typically by parametrizing a different characteristic length l_i for each variable : $r = \|\boldsymbol{x} - \boldsymbol{x}'\|_{custom} = \sqrt{\sum_{i=1}^{d}\left(\frac{x_i - x'_i}{l_i}\right)^2}$.

- A sum or product of kernels remains a valid kernel: we can for example build a tensorized kernel, $k(\boldsymbol{x}, \boldsymbol{x}') = \prod_{i=1}^{d} k(x_i, x'_i)$





Source : Rassmussen & Williams, *Gaussian Processes for Machine Learning*

Multi-output gaussian processes

- Reminder : equations of a single-output GP:

$$\begin{cases} f(\boldsymbol{x}_*)|(\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{y}) \sim \mathcal{N}\left(\boldsymbol{k}_*^T(\boldsymbol{K} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{y}, \quad k_{**} - \boldsymbol{k}_*^T(\boldsymbol{K} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{k}_* \right) \\ -2\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) = \boldsymbol{y}^T\boldsymbol{K}^{-1}\boldsymbol{y} - \log|\boldsymbol{K}| + n\pi \end{cases}$$

$\boldsymbol{y}$: training output values
$\boldsymbol{X}$: training input values
σ : noise
$\boldsymbol{\theta}$: hyperparameters

- General framework of multi-output GPs:

$$\begin{cases} \boldsymbol{f}(\boldsymbol{x}_*)|(\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{Y}) \sim \mathcal{N}\left(\boldsymbol{k}_*^{M,T}\left(\boldsymbol{K}^M + \boldsymbol{\Sigma}\otimes\boldsymbol{I}\right)^{-1}\boldsymbol{Y}_v, \quad \boldsymbol{k}_{**}^M - \boldsymbol{k}_*^{M,T}\left(\boldsymbol{K}^M + \boldsymbol{\Sigma}\otimes\boldsymbol{I}\right)^{-1}\boldsymbol{k}_*^M \right) \\ -2\log p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta}) = \boldsymbol{Y}_v{}^T\left(\boldsymbol{K}^M + \boldsymbol{\Sigma}\otimes\boldsymbol{I}\right)^{-1}\boldsymbol{Y}_v - \log\left|\boldsymbol{K}^M + \boldsymbol{\Sigma}\otimes\boldsymbol{I}\right| + np\pi \end{cases}$$

With $\boldsymbol{k}(\cdot,\cdot)^M = \begin{pmatrix} k_{1,1}(\cdot,\cdot) & \cdots & k_{1,p}(\cdot,\cdot) \\ \vdots & \ddots & \vdots \\ k_{1,p}(\cdot,\cdot) & \cdots & k_{p,p}(\cdot,\cdot) \end{pmatrix}$ a matricial kernel detailing cross-tasks covariances, $\boldsymbol{Y}_v = \mathrm{vec}(\mathbf{Y})$ the flattened matrix $\mathbf{Y}$, $\boldsymbol{\Sigma} = \left(\sigma_{i,j}\right)_{1\le i\le p}^{1\le j\le p}$ a cross-tasks noise matrix and $\otimes$ the Kronecker product.

Kronecker product : $\boldsymbol{A}\otimes\boldsymbol{B} = \begin{pmatrix} a_{1,1}\boldsymbol{B} & \cdots & a_{1,m}\boldsymbol{B} \\ \vdots & \ddots & \vdots \\ a_{n,1}\boldsymbol{B} & \cdots & a_{n,m}\boldsymbol{B} \end{pmatrix}$. Its shape is the product of shapes of $\boldsymbol{A}$ and $\boldsymbol{B}$.

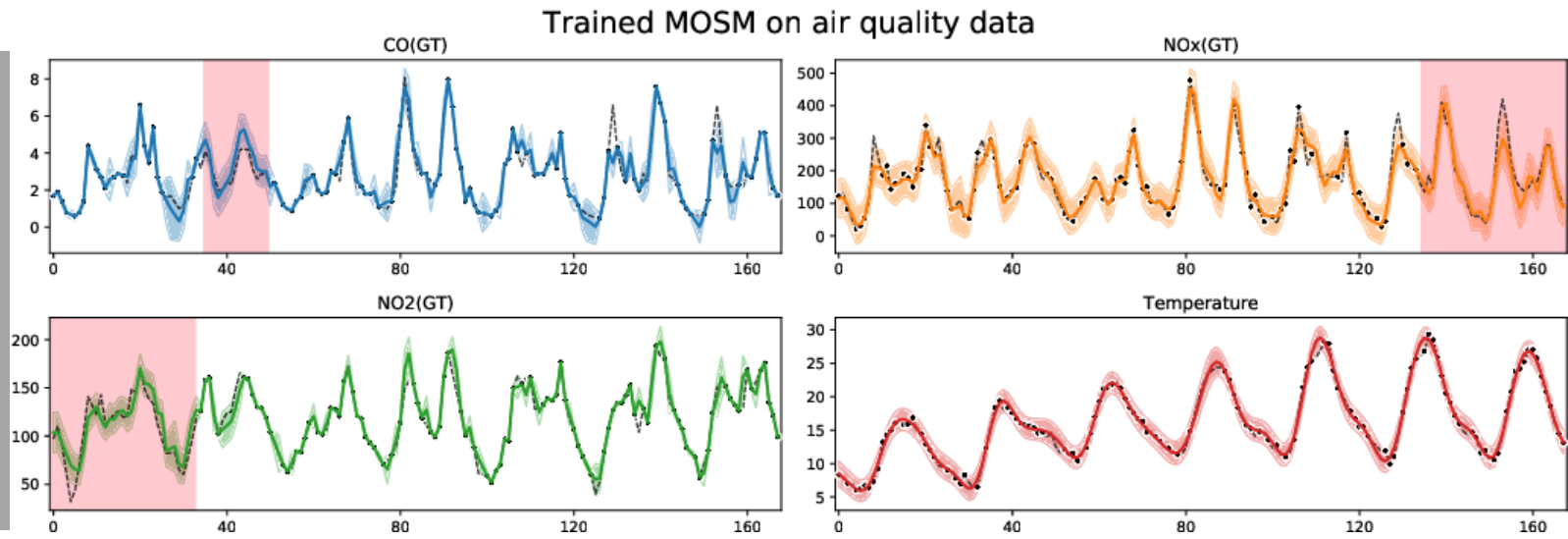
Linear model of co-regionalization

- We model each of the p functions to be reconstructed as a linear combination of q independent latent (unobserved) Gaussian processes u_j :

$$\widehat{y_m}(\boldsymbol{x}) = \sum_{j=1}^{q} H_{m,j} g_j(\boldsymbol{x}), \qquad \forall m \in [\![1; p]\!]$$

- This is equivalent to the previous formulation of a « standard » GP with matrix kernel: $k^M(\boldsymbol{x}, \boldsymbol{x}') = \sum_{j=1}^{q} \boldsymbol{B_j}\, k_j(\boldsymbol{x}, \boldsymbol{x}')$, where k_j is the kernel of process g_j and $\boldsymbol{B_m} = \left(H_{i,m} H_{j,m}\right)_{1 \le i \le p}^{1 \le j \le q}$.
  The cross-task kernels are then : $k_{i,l}(\cdot,\cdot) = \sum_{j=1}^{q} H_{i,j} H_{l,j}\, k_j(\cdot,\cdot)$.

Problem : treating this exactly has time complexity $O\left((np)^3\right)$!



Source : T.Wolff & al, *MOGPTK: The Multi-Output Gaussian Process Toolkit*

Intrinsic model of coregionalization (IMC)

- Simplification of the LMC : all latent processes have the same kernel k_x. $\boldsymbol{K_x} = k_x(\boldsymbol{X}, \boldsymbol{X})$
- $\Rightarrow \boldsymbol{K^M} = \sum_{j=1}^{q}(\boldsymbol{B_j} \otimes \boldsymbol{K_x}) = \left(\sum_{j=1}^{q} \boldsymbol{B_j}\right) \otimes \boldsymbol{K_x} = \boldsymbol{H}\boldsymbol{H^T} \otimes \boldsymbol{K_x} = \boldsymbol{K_t} \otimes \boldsymbol{K_x}$
- Easy computation thanks to the trick :

$\boldsymbol{K_t} \otimes \boldsymbol{K_x} + \boldsymbol{\sigma I} = (\boldsymbol{U_t} \otimes \boldsymbol{U_x})(\boldsymbol{D_t} \otimes \boldsymbol{D_x} + \boldsymbol{\sigma I})\left(\boldsymbol{U_t^T} \otimes \boldsymbol{U_x^T}\right)$, with $\boldsymbol{U_i D_i U_i^T}$ the eigendecomposition of $\boldsymbol{K_i}$

→ Runtime complexity in $\boldsymbol{O}\left(\boldsymbol{n^3} + \boldsymbol{p^3}\right)$ instead of $\boldsymbol{O}\left((\boldsymbol{np})^3\right)$

- Other similar tricks whenever the covariance matrix is a Kronecker product of kernels
- BUT : much lesser expressivity (what if the data has several characteristic lengthscales ?)
- By comparison, for the LMC we have : $\boldsymbol{K^M} = (\boldsymbol{H} \otimes \boldsymbol{I})\, Diag(\boldsymbol{K_i})\left(\boldsymbol{H^T} \otimes \boldsymbol{I}\right)$, with $\boldsymbol{H} \in \mathbb{R}^{p \times q}$ (p number ot tasks, q number of latent processes, n number of points)

Inducing points

- Most common GP approximation
- Principle: few "pseudo-inputs" points with inferred output values replace the much more numerous true datapoints
- Pseudo-points locations are optimized parameters
- Rationale: the set of training points is often redundant. Replacing it with a smaller number of optimized pseudo-inputs greatly speeds up matrix inversions, without degrading performance too much.

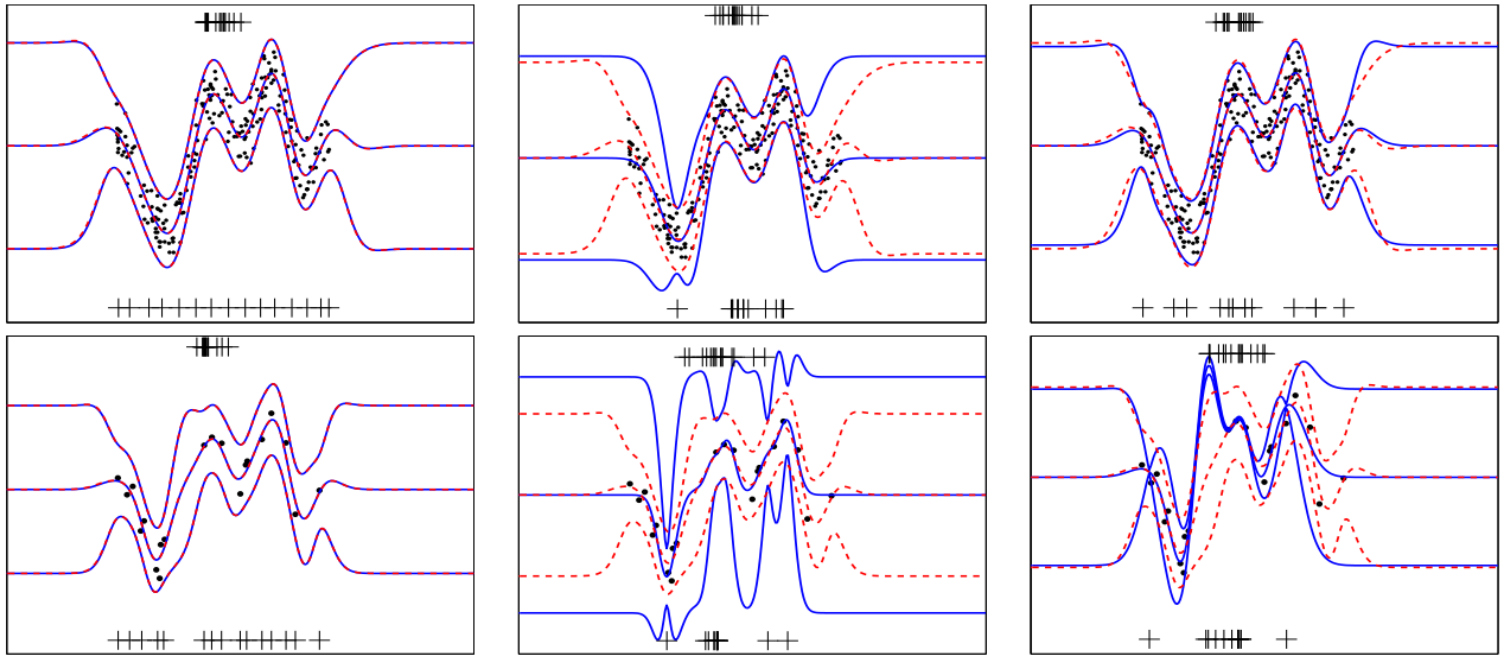


Figure 1: The first row corresponds to 200 training points and the second row to 20 training points. The first column shows the prediction (blue solid lines) obtained by maximizing F_V over the 15 pseudo-inputs and the hyperparameters. The full GP prediction is shown with red dashed lines. Initial locations of the pseudo-inputs are shown on the top as crosses, while final positions are given on the bottom as crosses. The second column shows the predictive distributions found by PP and similarly the third column for SPGP.

Source : Titsias, *Variational Learning of Inducing Variables in Sparse Gaussian Processes*

Variational approach

- Built upon the inducing points approximation

- $\log p(\boldsymbol{y}) = \log\langle p(\boldsymbol{y}|\boldsymbol{f})\rangle_{p(\boldsymbol{f}|\boldsymbol{u})} \geq \langle \log p(\boldsymbol{y}|\boldsymbol{f})\rangle_{p(\boldsymbol{f}|\boldsymbol{u})} \equiv \mathcal{L}$ by convexity, with $\boldsymbol{u}$ the fictive values of f at inducing points locations, and $\boldsymbol{f}$ its possible (non-noisy) values at training locations.

- **The lower bound $\mathcal{L}$ is optimized instead of the true marginal likelihood.** To compute it efficiently, $p(\boldsymbol{f}, \boldsymbol{u} \mid \boldsymbol{y})$ is replaced by a variational distribution $q(\boldsymbol{u})$ which parameters are optimized

- Predictions : $p(f(\boldsymbol{x}_*) \mid \boldsymbol{y}) \approx \int \underbrace{p(f(\boldsymbol{x}_*)|\boldsymbol{u})}_{\text{Standard GP expression}} \underbrace{q(\boldsymbol{u})}_{\text{Gaussian}} d\boldsymbol{u}$

- Numerous variations are possible within this general approach (modifications to $\mathcal{L}$, choice of q), which can in particular enable decoupling of inputs and outputs, or minibatch training (=stochastic optimization with subsets of the dataset).

- **A variational LMC can be built this way** : the variational distribution becomes a product of independent latent GPs, $q(\boldsymbol{u}) = \prod_{j=1}^{q} q_j(\boldsymbol{u})$ → Runtime complexity in $\boldsymbol{O}(\boldsymbol{n^3} + \boldsymbol{p^3})$

2. The projected LMC

A nice finding on the model structure !

Exact posteriors of latent processes

- Question (initially for leave-one-out error computation) : can we « open the box » of the LMC and compute the values taken by latent processes ?

- It turns out that we can : $\begin{cases} \mathbb{E}[\boldsymbol{y}_*|\boldsymbol{Y}] = Diag\left(\boldsymbol{k}_{i*}^T\right)(Diag(\boldsymbol{K_i}) + \boldsymbol{\Sigma_P}\otimes\boldsymbol{I})^{-1}vec(\boldsymbol{TY}) \\ \mathbb{V}[\boldsymbol{y}_*|\boldsymbol{Y}] = Diag(k_{i**}) - Diag\left(\boldsymbol{k}_{i*}^T\right)(Diag(\boldsymbol{K_i}) + \boldsymbol{\Sigma_P}\otimes\boldsymbol{I})^{-1}Diag(\boldsymbol{k_{i*}}) \end{cases}$

With : $\boldsymbol{k_{i*}} = k_i(\boldsymbol{x}_*, \boldsymbol{X}),\ k_{i**} = k_i(\boldsymbol{x}_*, \boldsymbol{x}_*),\ \boldsymbol{K_i} = k_i(\boldsymbol{X}, \boldsymbol{X}),$

And: $\boldsymbol{\Sigma_P} = \left(\boldsymbol{H^T\Sigma\, H}\right)^{-1},\ \boldsymbol{T} = \boldsymbol{\Sigma_P H^T \Sigma^{-1}}$.

- Many block-diagonal matrices here; if $\boldsymbol{\Sigma_P}$ is also diagonal, both expression decouple !! Then :

$\begin{cases} \mathbb{E}[y_{j*}|\boldsymbol{Y}] = \boldsymbol{k}_{j*}^T\left(\boldsymbol{K_j} + \sigma_j\boldsymbol{I}\right)^{-1}\boldsymbol{T_i Y} \\ \mathbb{V}[y_{j*}|\boldsymbol{Y}] = k_{j**} - \boldsymbol{k}_{j*}^T\left(\boldsymbol{K_j} + \sigma_j\boldsymbol{I}\right)^{-1}\boldsymbol{k_{j*}} \end{cases}$ → expressions of standard single-output GPs ($\boldsymbol{\Sigma_P} = Diag(\sigma_i)$)

- Same is true for the likelihood : $\prod_{l=1}^{n}\frac{\mathcal{N}(\boldsymbol{Y_j}|\boldsymbol{0},\boldsymbol{\Sigma})}{\mathcal{N}(\boldsymbol{TY_j}|\boldsymbol{0},\boldsymbol{\Sigma_P})} \times \int p(\boldsymbol{G})\prod_{l=1}^{n}\mathcal{N}\left(\boldsymbol{TY_j}|\boldsymbol{G_j},\ \boldsymbol{\Sigma_P}\right)d\boldsymbol{G}$

  ($\boldsymbol{G}$ values of latent processes at observed locations)

Sketch of the proof(s)

- *Block-diagonal matrix algebra* :
  - Start from $\boldsymbol{f}(\boldsymbol{x}_*)|\boldsymbol{x}_*, \boldsymbol{X}, \boldsymbol{Y} \sim \mathcal{N}\left(\boldsymbol{k}_*^{\boldsymbol{M},\boldsymbol{T}}\left(\boldsymbol{K}^{\boldsymbol{M}} + \boldsymbol{\Sigma}\otimes\boldsymbol{I}\right)^{-1}\boldsymbol{Y}_v,\ \boldsymbol{k}_{**}^{\boldsymbol{M}} - \boldsymbol{k}_*^{\boldsymbol{M},\boldsymbol{T}}\left(\boldsymbol{K}^{\boldsymbol{M}} + \boldsymbol{\Sigma}\otimes\boldsymbol{I}\right)^{-1}\boldsymbol{k}_*^{\boldsymbol{M}}\right)$.
  - In each expression, expand inverses with Woodburry's formula :
$$(\boldsymbol{A} + \boldsymbol{UCV})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{U}\left(\boldsymbol{C}^{-1} + \boldsymbol{VA}^{-1}\boldsymbol{U}\right)^{-1}\boldsymbol{VA}^{-1}$$
  - Rearrange using properties of the Kronecker product :
$$(\boldsymbol{A}\otimes\boldsymbol{B})(\boldsymbol{C}\otimes\boldsymbol{D}) = (\boldsymbol{AC}\otimes\boldsymbol{BD}), \qquad \left(\boldsymbol{C}^{\boldsymbol{T}}\otimes\boldsymbol{A}\right)vec(\boldsymbol{B}) = vec(\boldsymbol{ABC}), \qquad (\boldsymbol{A}\otimes\boldsymbol{B})^{-1} = \boldsymbol{A}^{-1}\otimes\boldsymbol{B}^{-1}$$
  - Use backward Woodburry's formula to complete the proof.

- *Ab initio bayesian computation* :
  - Start by computing the posterior of latent processes at training points, $p(\boldsymbol{G}|\boldsymbol{Y})$. By Baye's rule, $p(\boldsymbol{G}|\boldsymbol{Y}) \propto p(\boldsymbol{G}|\boldsymbol{Y})p(\boldsymbol{G})$; by properties of gaussians, $p(\boldsymbol{G}|\boldsymbol{Y})$ is a gaussian in the variables $\boldsymbol{G}$. One can therefore expand $p(\boldsymbol{G}|\boldsymbol{Y})$ and $p(\boldsymbol{G})$ which are given by model definition, and identify terms in $\boldsymbol{G}$.
  - From there, deduce the posterior of latent processes at test point, $p(\boldsymbol{g}_*|\boldsymbol{Y})$. For this, use the laws of total expectation and total variance, conditioning on $\boldsymbol{G}$:

$\mathbb{E}[\boldsymbol{Z}] = \mathbb{E}\left[\mathbb{E}[\boldsymbol{Z}|\boldsymbol{Z}']\right]$; $\mathbb{V}[\boldsymbol{Z}] = \mathbb{E}\left[\mathbb{V}[\boldsymbol{Z}|\boldsymbol{Z}']\right] + \mathbb{V}\left[\mathbb{E}[\boldsymbol{Z}|\boldsymbol{Z}']\right]$, and GP identities yielding $\mathbb{E}[\boldsymbol{g}_*|\boldsymbol{G}]$.

Projected data and projected noise

- What is the meaning of the recurring terms $\boldsymbol{\Sigma_P} = \left(\boldsymbol{H^T \Sigma\, H}\right)^{-1}$ and $\boldsymbol{T} = \boldsymbol{\Sigma_P H^T \Sigma^{-1}}$?
- We have the following set of properties :
  - $\boldsymbol{T}$ is a generalized inverse of $\boldsymbol{H}$: $\boldsymbol{TH} = \boldsymbol{I_q}$. Therefore, $\boldsymbol{HT}$ is a projection.
  - $\boldsymbol{TY}$ is a maximum likelihood estimator of $\boldsymbol{G}$: $\boldsymbol{TY} = \underset{\boldsymbol{G}}{\operatorname{argmax}}\, p(\boldsymbol{Y}|\boldsymbol{G})$.
  - $\boldsymbol{TY}$ is a sufficient statistic of the model for the variable $\boldsymbol{G}$: $p(\boldsymbol{Y}|\boldsymbol{TY}, \boldsymbol{G}) = p(\boldsymbol{Y}|\boldsymbol{TY})$
  - $\boldsymbol{T\Sigma T^T} = \boldsymbol{\Sigma_P}$, so that $\boldsymbol{TY}|\boldsymbol{G} \sim \mathcal{N}(vec(\boldsymbol{G}), \boldsymbol{\Sigma_P})$
  - Latent processes are conditionaly independent $\Leftrightarrow$ $\boldsymbol{\Sigma_P}$ is diagonal

- Interpretation :
  - $\boldsymbol{TY}$ is the data « seen » by the latent processes. In some sense, it is a projection of the full data $\boldsymbol{Y}$.
  - It contains all useful information regarding latent processes, and is the best estimate of their observed values.
  - $\boldsymbol{\Sigma_P}$ is the covariance matrix of this projected data ; if it is diagonal, latent processes are independent.

A compatible parametrization

- **Diagonal Projected Noise (DPN) hypothesis** : enforcing $\boldsymbol{\Sigma_P}$ to be diagonal decouples latent processes and yields runtime complexity $\boldsymbol{O(n^3 + p^3)}$
- **Problem** : this condition couples the mixing matrix $\boldsymbol{H}$ and and noise matrix $\boldsymbol{\Sigma}$ → need for and adapted parametrization.
- Final retained parametrization : set the $\boldsymbol{QR}$ decomposition of $\boldsymbol{H}$, define $\boldsymbol{Q_\perp}$ an orthogonal supplement of $\boldsymbol{Q}$, decompose $\boldsymbol{\Sigma^{-1} = QAQ^T + Q_\perp BQ_\perp^T + QCQ_\perp^T + Q_\perp C^T Q^T}$, set $\boldsymbol{D = R^T AR}$ and $\boldsymbol{M = R^T C}$. Then the DPN condtion is equivalent to ($\boldsymbol{D}$ diagonal), and :

$$\boldsymbol{\Sigma = Q_+ R_+ D_+ R_+^T Q_+^T}, \quad \text{with } \boldsymbol{Q_+} = (\boldsymbol{Q} \,|\boldsymbol{Q_\perp}),\ \boldsymbol{R_+} = \begin{pmatrix} \boldsymbol{R} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I_{p-q}} \end{pmatrix},\ \boldsymbol{D_+^{-1}} = \begin{pmatrix} \boldsymbol{D} & \boldsymbol{M} \\ \boldsymbol{M^T} & \boldsymbol{B} \end{pmatrix};\quad D_+ \equiv \begin{pmatrix} \widetilde{\boldsymbol{D}} & \widetilde{\boldsymbol{M}} \\ \widetilde{\boldsymbol{M}}^T & \widetilde{\boldsymbol{B}} \end{pmatrix}$$

- This yields a convenient form for the marginal log-likelihood :

$$-2\log p(\boldsymbol{Y}) = n(p-q)\pi + \underbrace{2n\log|\boldsymbol{R}| + n\log\widetilde{\boldsymbol{B}}}_{\text{Discarded noise}} + \underbrace{Tr\left(\boldsymbol{Y^T Q_\perp \widetilde{B}^{-1} Q_\perp^T Y}\right)}_{\text{Discarded data}} + \sum_{j=1}^{q} \underbrace{\log \mathcal{N}\left(\boldsymbol{YT_j}|\boldsymbol{0}, \boldsymbol{K_j} + \boldsymbol{\sigma_j I_n}\right)}_{\text{Latent processes}}$$

Results : impact of the noise hypothesis

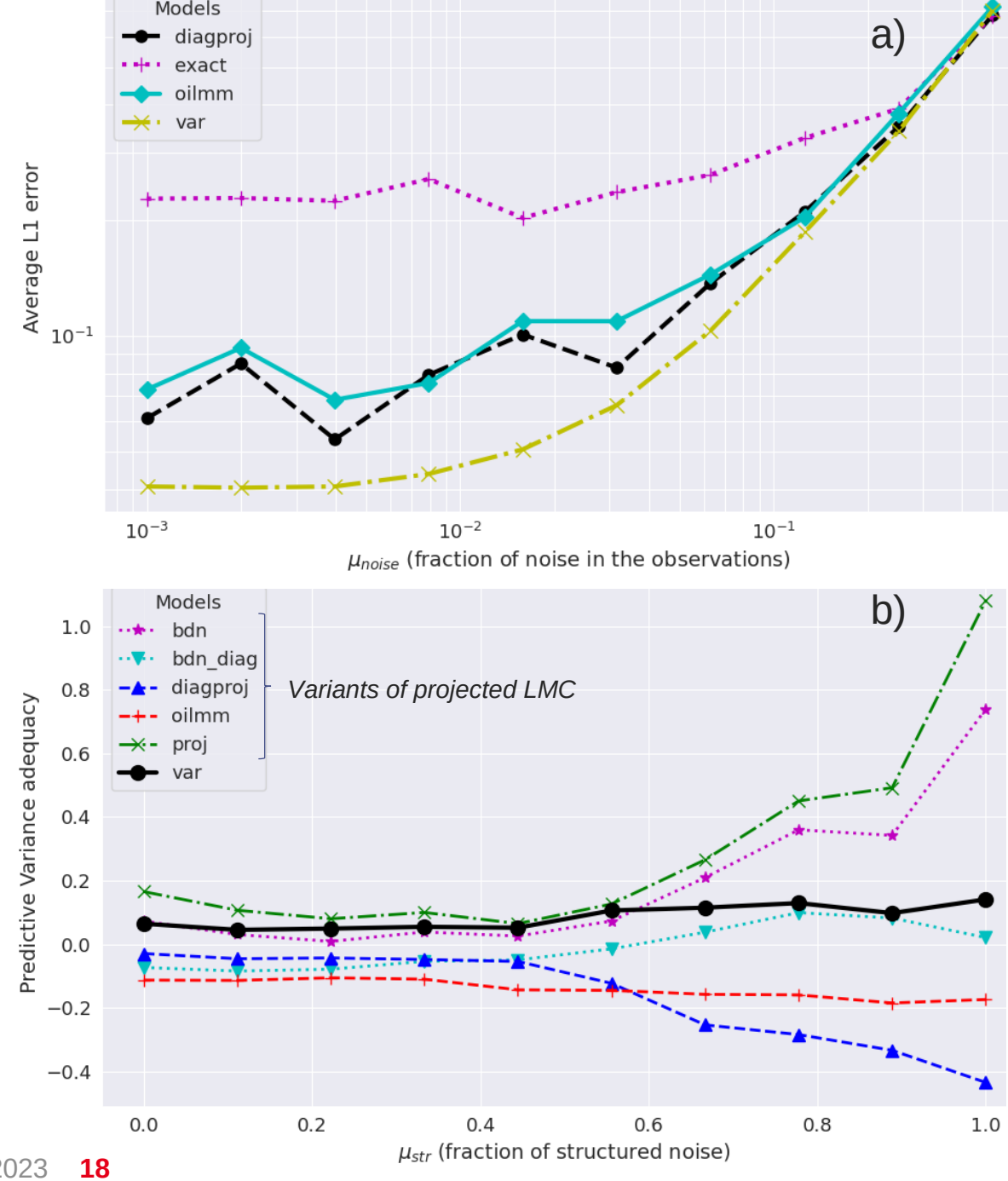
- Tests on synthetic data with controled noise magnitude and share of structured (= mixed cross-tasks) noise // per-task noise
- The model doesn't seem to underperform even for large and heavily-structured noises !

$$PVA = \log\left(\frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}\frac{(y_{ij}-\widehat{y_{ij}})^2}{\widehat{v_{ij}}}\right)$$

Fig.1:

a) Average test L1 error of the projected model ('diagproj') compared with reference LMC approaches, in function of the proportion of noise

b) Average PVA errors or reference and projected models, in function of the proportion of structured-to-unstructured noise



Results : benefits of a general mixing matrix

- This work is based on a paper on OILMM, a model which derived the same results in the special case of an orthogonal mixing matrix $\boldsymbol{H}$ (see previous graphs)
- This restriction seems to cope very well with the previous case of i.i.d normally-distributed $\boldsymbol{H}$ coefficients
- Can we find realistic cases where this restriction is more of a problem ? → Yes we can :

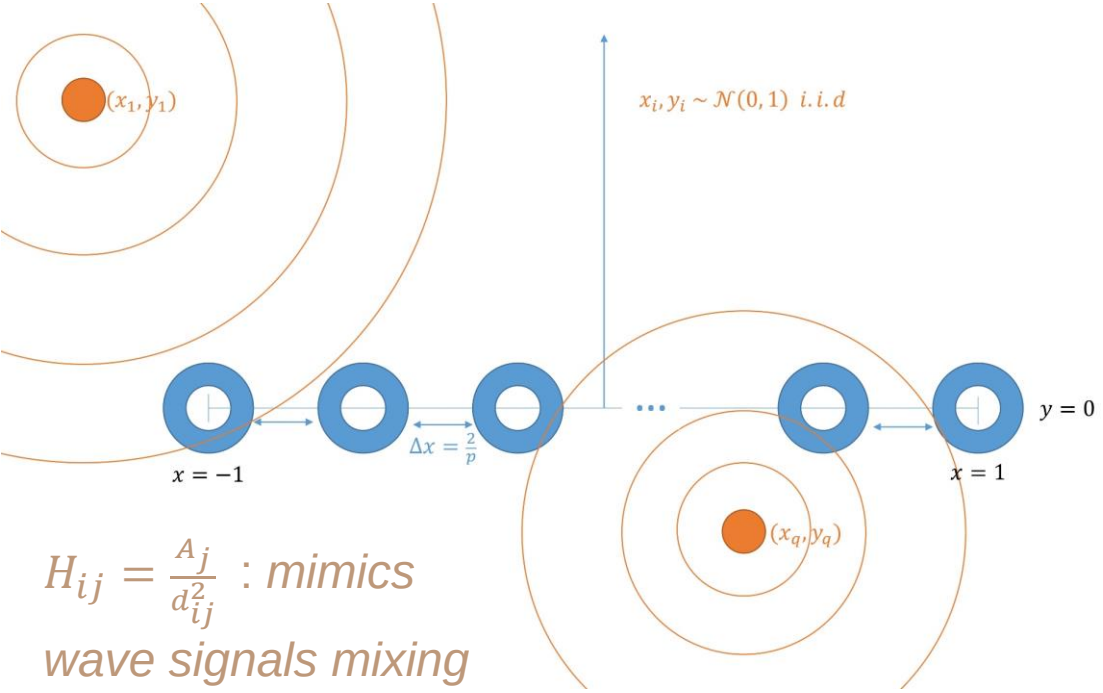


Fig.2: set up of the « physical » toy data leading to non-orthogonal $\boldsymbol{H}$

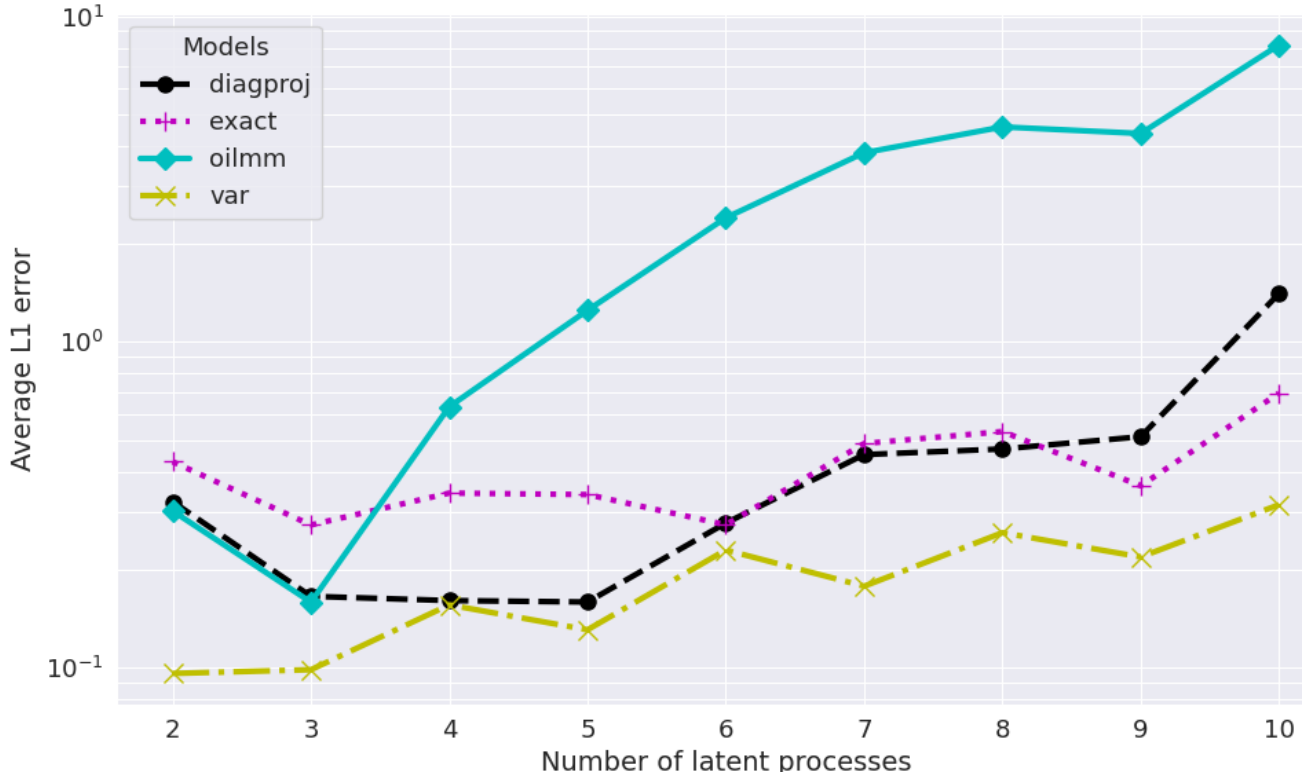


Fig.3: Average test errors of several models in function of $N_{latents}$

3. Application

Homogenized cross-sections in deterministic neutronics codes

Context

- **Neutron modeling:** determination of important core parameters (neutron flux, local power, etc.) by solving the Boltzmann equation: *(+ Bateman equations for evolution of isotopic concentrations)*

$$\frac{1}{v}\underbrace{\frac{\partial\phi(\mathbf{r},v,\mathbf{\Omega},t)}{\partial t}}_{\text{Variation of neutron population}} = +\chi(v)\int_0^{+\infty}dv'\frac{1}{4\pi}\int_{4\pi}d^2\Omega'\,\bar{\nu}\Sigma_f(\mathbf{r},v')\phi(\mathbf{r},v',\mathbf{\Omega}',t) \quad \text{Fissions}$$

$$+\int_{4\pi}d^2\Omega'\int_0^{+\infty}dv'\,[\Sigma_s(\mathbf{r},v'\rightarrow v,\mathbf{\Omega}'\rightarrow\mathbf{\Omega},t)\phi(\mathbf{r},v',\mathbf{\Omega}',t)] \quad \text{Scattering}$$

$$+S_a(\mathbf{r},v,\mathbf{\Omega},t) \quad \text{External source}$$

$$-\Sigma_t(\mathbf{r},v,\mathbf{\Omega},t)\phi(\mathbf{r},v,\mathbf{\Omega},t) \quad \text{Absorption}$$

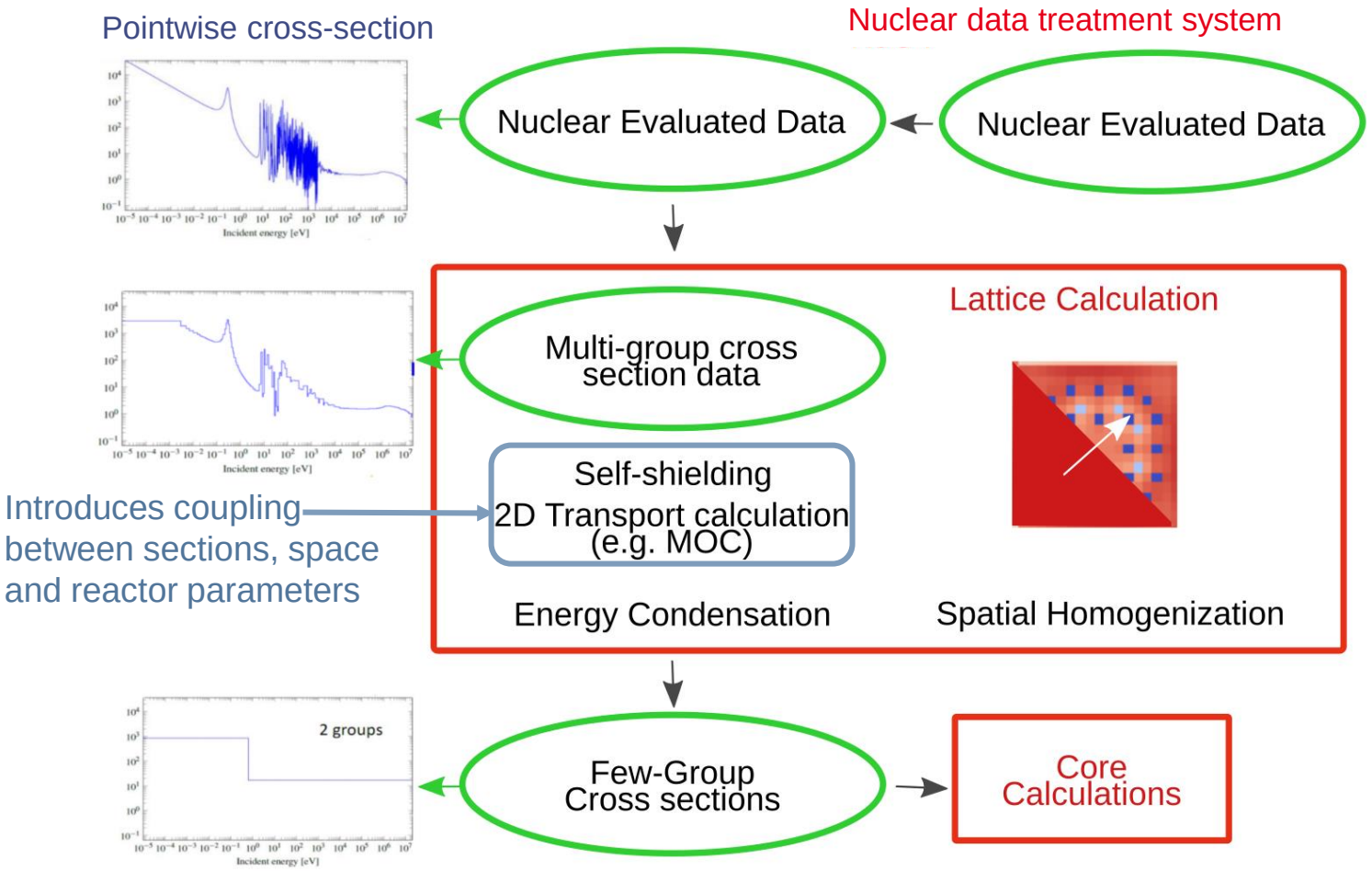
$$-\,\mathrm{div}\,[\mathbf{\Omega}\phi(\mathbf{r},v,\mathbf{\Omega},t)] \quad \text{Leaks}$$

Flux = neutron population

Cross-sections

- 6+1 phase space, resonant phenomena → complex discretization

Deterministic neutronics code

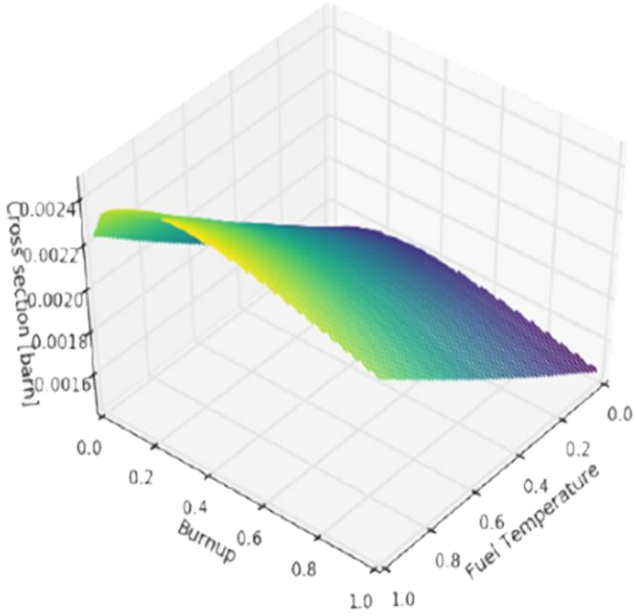


Source: E.Szames, Few group cross section modeling by machine learning for nuclear reactor

Homogenized cross-sections (HXS)

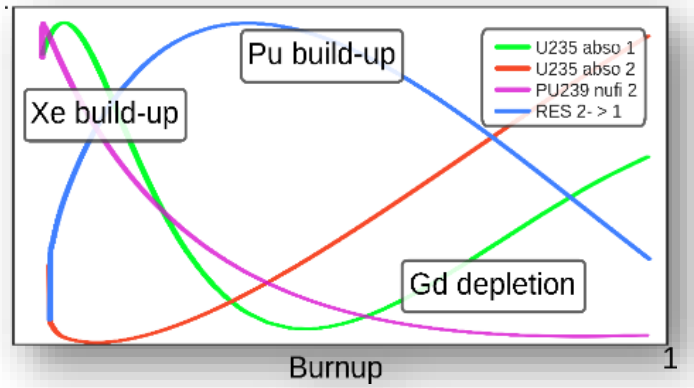
- After a lattice calculation (assembly-wise), point cross sections are: homogenized in a region v, condensed over an energy range g, for a reaction type r, and a particularized isotope i.

$$\sigma_{v,i,r,g} = \frac{\int_v \int_g (\sigma_{i,r}\phi)(\vec{r}, E) dE d\vec{r}}{\int_v \int_g \phi(\vec{r}, E) dE d\vec{r}}$$



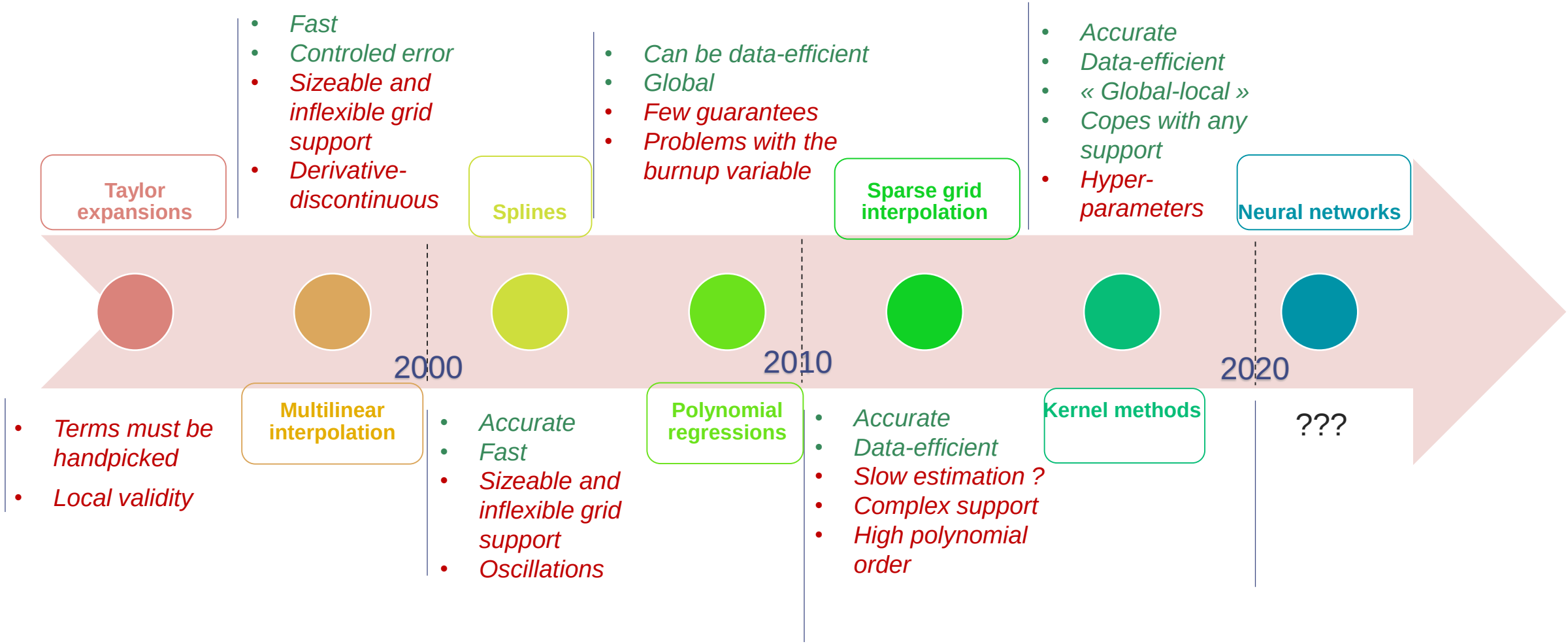
- Problem to be solved: create and store a large number (up to millions) of highly correlated multivariate approximation models, one for each homogenized cross-section.

- Constraints: high accuracy requirement (relative error$\sim 10^{-3}$), low memory footprint, smallest possible support, fast estimation



Source: E.Szames

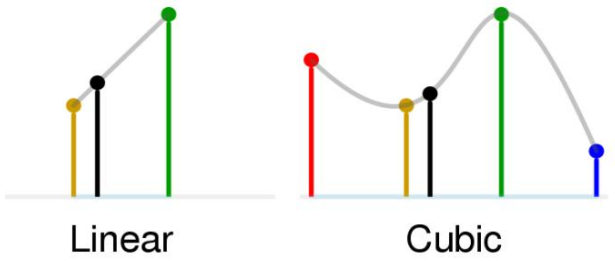
State of the art

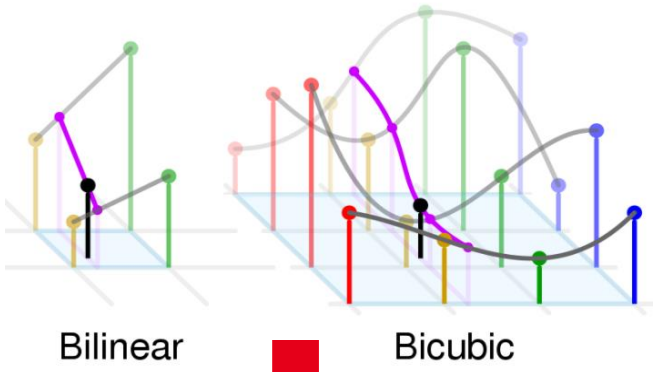


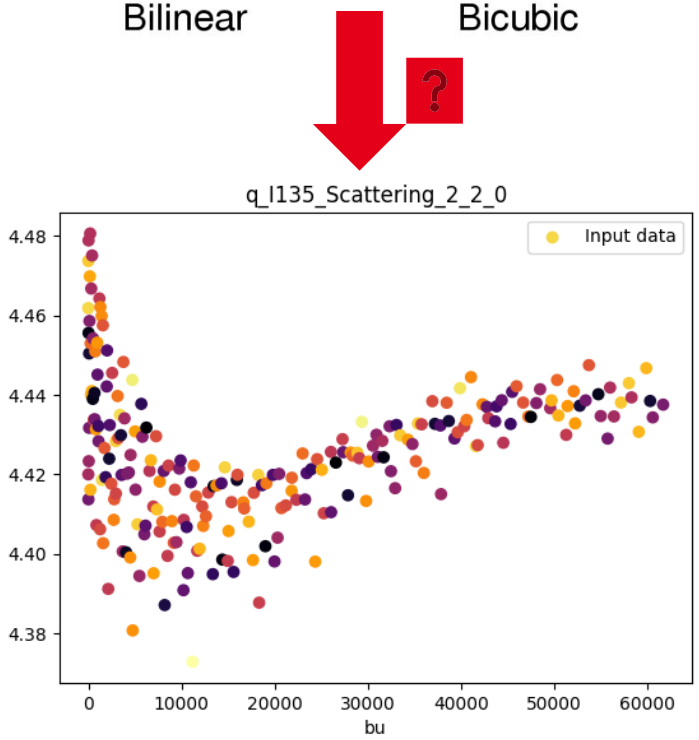
Current standard

- Most common method today: multilinear interpolation
- Strengths: simplicity, estimation speed, error control
- Inconvenients: improvable accuracy, number of coefficients, curse of dimensionality
- Consequence: cross-sections libraries explode in size (up to several hundred GB)
- Possible solutions:
  - Get rid of grids
  - Exploit HXS redundancy

→ Multitask GPs are natural candidates



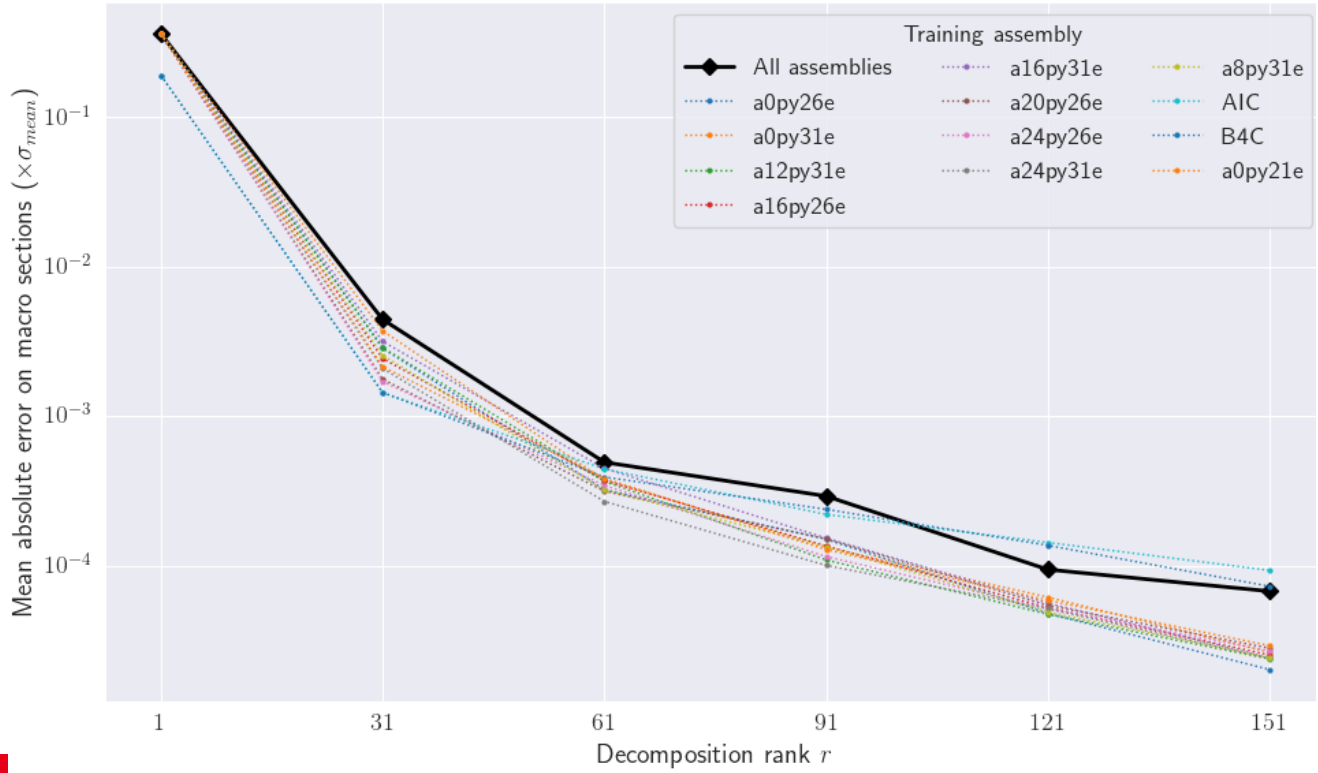




Source : Cmglee

Nuclear data redundancy

- Parallel work: exploiting the redundancy of nuclear data to reduce the size of calculations (typically linear dimension reduction). Reduced Order Models, compression...
- Examples: linear compression of local burnups [Tom1] and HXS [Tom2], PCA of concentration data [Hua], autoencoding of scattering matrices [Whe], multi-output neural network [Sza2]…
- Early thesis work: linear dimension reduction and extrapolation using Empirical Interpolation Method [Tru].

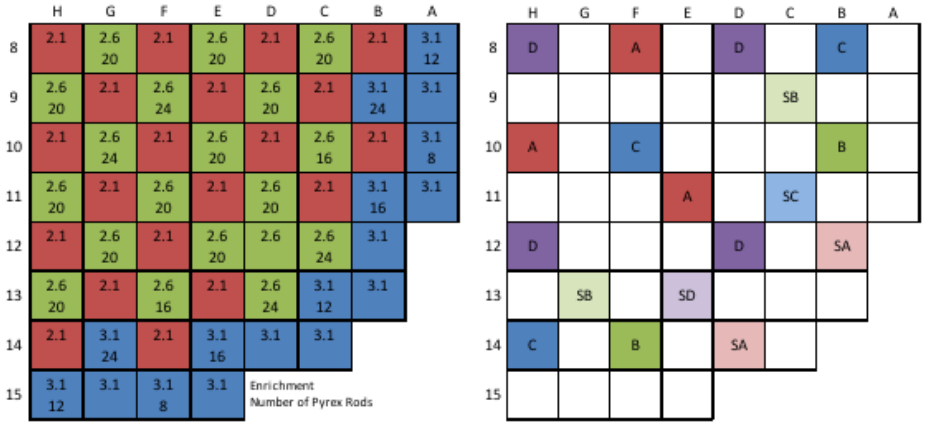


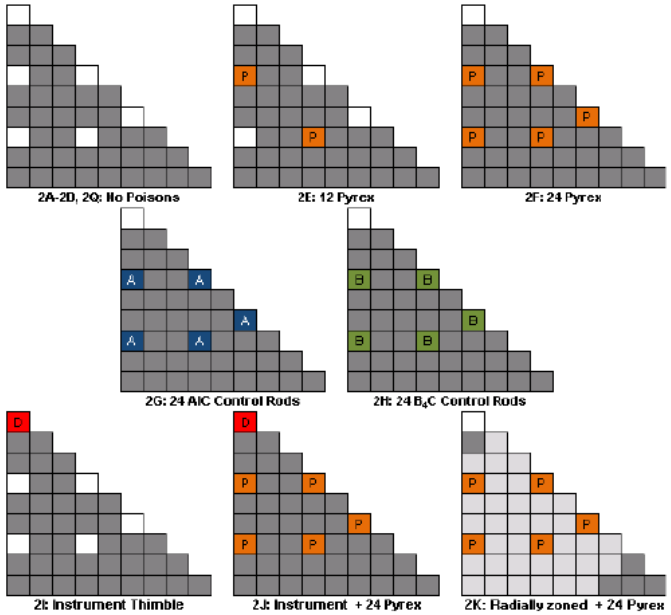
Source : *An EIM-based compression-extrapolation tool for efficient treatment of homogenized cross-section* data, Truffinet, Ammar, Gérard Castaing, Argaud, Bouriquet, Annals of Nuclear Energy, 2023

Each plot represents the error made when extrapolating - via the EIM algorithm - data from one assembly to all others. We can see that these errors are minor, and that each assembly contains as much useful information as the sum of them (black curve).

Experimental test-case

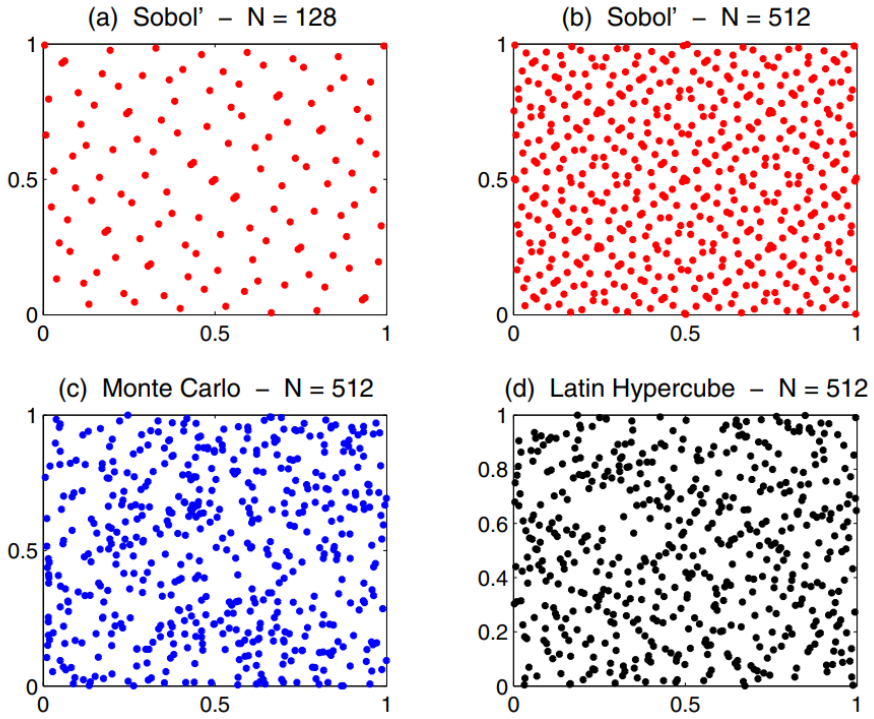
- Data generated with Apollo3® + SIGAR launcher
- 2 assemblies used in this presentation : a «standard» one (2.6% enrichment, no control rods or burnable poisons) and a 3.1% enriched assembly with gadolinium rods
- Variables : burnup Bu, fuel temperature T_f, moderator temperature T_m, boron concentration C_b
- Full grids for comparison with multilinear interpolation
- Several sets generated according to discretization finesse: 2 or 20 energy groups, average per cell or per assembly. From 70 to 100,000 HXS / assembly according to these choices



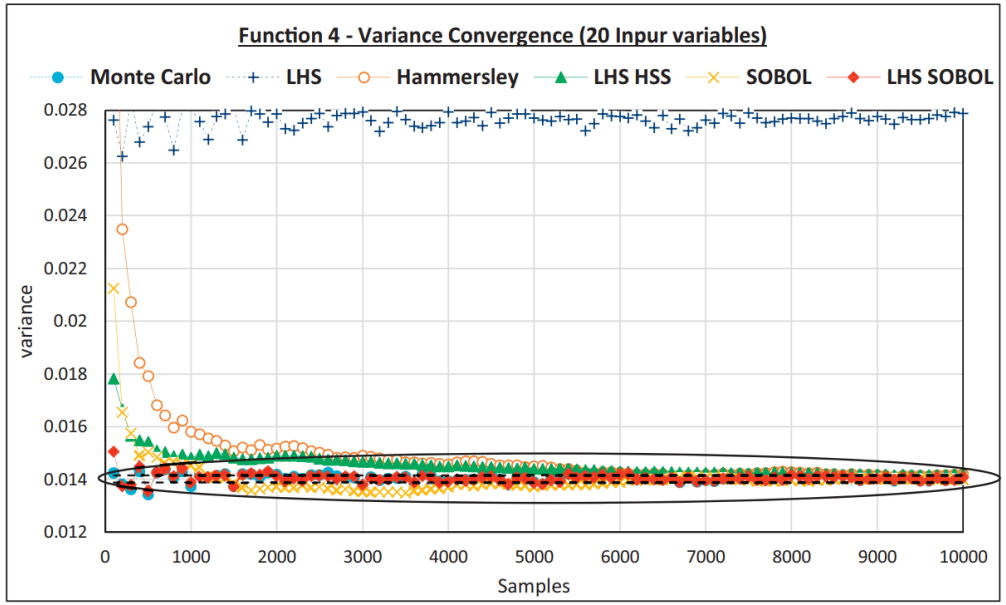


Low-discrepancy sequences

- Sets of points (deterministic or semi-random) in $\mathbb{R}^d$ filling space as homogeneously as possible
- Most common methods: Sobol' sequences and Latin Hypercube Sampling (LHS)
- Sobol' sequences generally superior for numerical quadrature and Uncertainty Quantification, tasks close to ours
- Generated point sets:
  - For training, Sobol' sets with 128, 256, 512 and 1024 points (optimal properties when the number of points is a power of 2);
  - For testing, 512-point LHS.



Source : G.Blatman & al, *Quasi random numbers in stochastic finite element analysis*



Source : N.Dige & al, *Efficient sampling algorithm for large-scale optimization under uncertainty problem*

Performance metrics

- HXS are spread over 10 orders of magnitude (!!), and for most of them vary little around their mean value → important to **center and reduce data to obtain errors compatible with ML standards**

- **Retained metrics** :

  - « Absolute» error (*ML*) measured on standardized data, in % : $Err_{abs} = \left|\tilde{y}_{true} - \tilde{y}_{pred}\right| * 100$

  - Relative error (*neutronics*) measured on raw data, in pcm : $Err_{rel} = \frac{|y_{true} - y_{pred}|}{y_{true}} * 10^5 = \frac{|\tilde{y}_{true} - \tilde{y}_{pred}|}{\tilde{y}_{true} + mean(y)/std(y)} * 10^5$

  - "Compression rate": number of coefficients in the multilinear model / number of coefficients in the model

  - T_{train} : duration of model training ; N_{points} : number of support points

  - α-CI : proportion of predictions within the predicted 5% confidence interval. **Target value : 95%**

  - Predictive Variance Adequacy : $PVA = \log\left(\frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}\frac{\left(y_{ij} - \widehat{y_{ij}}\right)^2}{\widehat{v_{ij}}}\right)$. **Target value : 0**

Modeling choices : grid search

- For all discrete or non-optimizable choices, we rely on a grid search (testing all combinations)

- **Kernel type**: *squared exponential* ($\psi(r) = e^{-r^2}$), *rational quadratic* of learned parameter α ($\psi_\alpha(r) = (1 + \frac{r^2}{2\alpha})^{-1}$), *Matérn* of parameter $\nu = 2.5$ ($\psi_\nu(r) = 2^{1-\nu}\left(\sqrt{2\nu}r\right)^{\nu} K_\nu\left(\sqrt{2\nu}r\right)$) → Little impact on accuracy

- **Variables combination** : → Huge impact on accuracy (factor 3 on errors)

  - $k_1 = \alpha\, k_{base}(Bu, C_b) + \beta\, k_{fuel}(Bu, T_f) + \gamma k_{mod}(Bu, T_m)$ (α, β, γ learned)
  - $k_2 = \alpha\, k_{base}(Bu, T_m, C_b) + \beta\, k_{fuel}(Bu, T_f)$ (α, β learned)
  - $k_3 = \alpha\, k_{base}(Bu, T_f, T_m, C_b)$ (α learned)
  - $k_4 = \alpha\, k_{base}(Bu, T_f, T_m, C_b) + \beta\, k_{bu}(Bu)$ (α, β learned)

- **Mean function** : fixed to a constant mean (variable means not yet implemented for projected model)

- Training dynamics : test of several optimization algorithms. Selected one : ADAM with exponential learning rate decay between 2 values to be adjusted. Automatic stop in case of stagnation.

  → Convergent and monotonic learning, little sensitive to the chosen bounds.

Results on the « light » test case : 70 outputs

	Mean absolute error (%)	Max absolute error (%)	Mean relative error (pcm)	Max relative error (pcm)	alpha-CI (%)	PVA	Compression rate	T_{train} (s)	N_{points}
Projected LMC	1,09	28	108	1948	98	-0.86	93	41	128
Variational LMC	0,91	19	92	1635	99	-1,04	114	44	128
IMC	1,09	28	118	2280	99	-2,14	37	19	128
Single-output GPs	1,16	19	129	2040	99	-0,94	35	178	128
Multilinear interpolation	2,81	21	197	1817	-	-	1	-	4704

- Performance of the variational model depends on the number of induction points, set here at 1.5x this of input points
- Accuracy and error estimation can be further improved by adding task-level polynomial mean functions (not available yet for projected LMC…)
- Models have been tested on a heavier dataset with ~100 000 outputs, making for 380 Mo of data ; training is not fully optimized yet, but runs in < 500s and mostly beats multilinear interpolation !

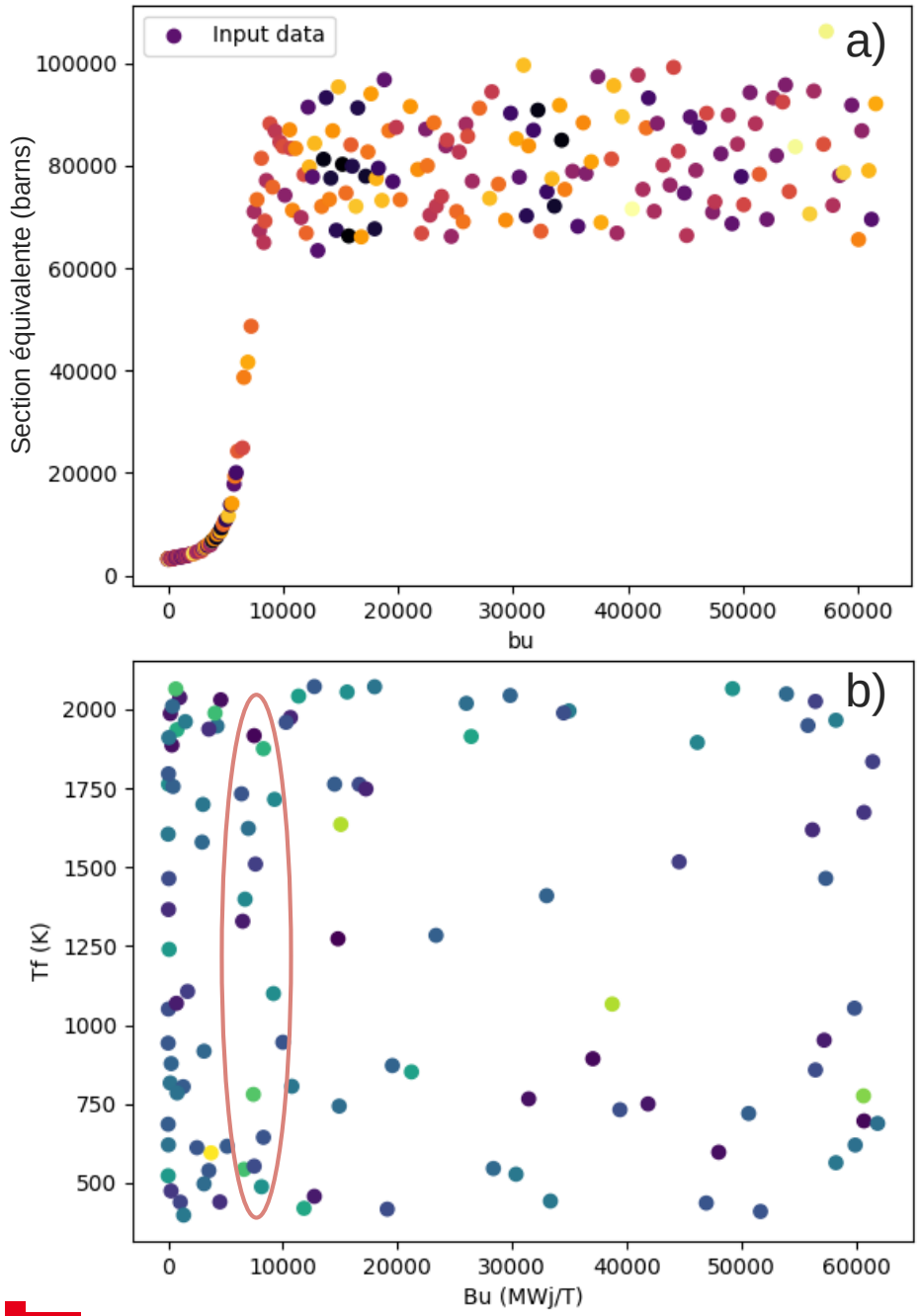
4. Conclusion

Conclusion

- **Theoretical aspects :**
  - Better understanding of the structure of the Linear Model of Coregionalization
  - Elaboration of a new multitask GP model, more expressive than the commonly-used IMC, more straightforward and less parametrized than variational ones
  - Authorizes or facilitates some operations on multitask GP, such as fast support updating (online training) or leave-one-out error computation

- **Applicative results :**
  - Improved accuracy with 40x less support → fewer calls to network code
  - Reduction of HXS library size by a factor of $\approx \mathbf{100}$
  - Reliable estimation of reconstruction uncertainty, easy propagation of uncertainties
  - Accepts any support (useful in case of missing or corrupted data, for flexible support refinement or reduction, etc.)
  - Possibility of adaptive support construction

Adaptive sampling



Section équivalente (barns)

- The GP variance estimator can be used to find areas in space where the model is unsure of its predictions → good candidates for placing new points there.
- Numerous refinements in the literature, e.g. based on LOO-CV error (Le Gratiet & Cannamela, 2012)
- Significant impact on maximum errors (÷4 for small number of points)Much smaller impact on mean errors
- So far, this has been done with single-outputs GP, summing scores of all outputs with equal weights. **Projected LMC will allow to work at the level of latent functions, with natural weighting → better results ?**

Fig.4:

a) Thermal absorption section of Gd157

b) Aspec of supports obtained by LOO-CV adaptive sampling on the gadolinium dataset (113 points)

Thanks for your attention !

Olivier TRUFFINET

olivier.truffinet@cea.fr

06 78 07 69 06

Références

[Bon] *Generic Inference in Latent Gaussian Process Models*, Bonilla, Krauth & Dezfouli, Journal of Machine Learning Research, 2019

[Bot] *Polynomial interpolation of few-group neutron cross sections on sparse grids*, Botes & Bokov, Annals of Nuclear Energy, 2014

[Bru] *Scalable Exact Inference in Multi-Output Gaussian Processes,* Bruinsma & al*,* Proceedings of Machine Learning Research*,* 2020

[Hua] *PWR pin-homogenized cross-sections analysis using big-data technology,* Hua, Li & Wang, Progress in Nuclear Energy, 2020

[Ras] *Gaussian Processes for Machine Learning*, Rassmussen & Williams, The MIT Press, 2006

[Sza1] *Reconstruction of few-group homogenized cross section by kernel method and active learning*, Szames, Tomatis, Ammar & Martinez, M&C 2019

[Sza2] *Few group cross section modeling by machine learning for nuclear reactor*, Neural and Evolutionary Computing. Université Paris-Saclay, 2020

[Tom1] *Compression of 3D pin-by-pin burnup data*, Tomatis & Dall'Osso, Annals of Nuclear Energy, 2020

[Tom2] *A multivariate representation of compressed pin-by-pin cross-sections*, Tomatis, EPJ Nuclear Sci. Technol, 2021

[Tru] *An EIM-based compression-extrapolation tool for efficient treatment of homogenized cross-section* data, Truffinet, Ammar, Gérard Castaing, Argaud & Bouriquet, Annals of Nuclear Energy, 2023

[Wat] *Improved cross-section modeling methodology for coupled three-dimensional transient simulations*, Watson, Ivanov, Annals of Nuclear Energy, 2002

[Whe] *Data Reduction in Deterministic Neutron Transport Calculations Using Machine Learning*, Whewell & McClarell, Annals of Nuclear Energy, 2022

[Zim] *Building neutron cross-section dependencies for few-group reactor calculations using stepwise regression,* Zimin & Semenov, Annals of Nuclear Energy, 2005