A Bayesian Framework for Uncertainty Quantification with High-Dimensional Data-Driven Inputs

Nicholas Zabaras

Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853
nzabaras@gmail.com
http://mpdc.mae.cornell.edu/

Dept. of Mathematics & Interdisciplinary Mathematics Inst. joint colloquium, Univ. of South Carolina, Columbia, SC, 7 May 2012
Overview

- Introducing the Problem of Interest: Data-driven Predictive Science
- Stochastic Input Model Reduction
- Multi-output Gaussian Process Regression for Solving SPDEs
- Numerical Examples
- Modeling Correlations
- Conclusions
Problem Definition

Bayesian Approach to UQ: Multi-output GPs
Modeling Correlations with Multi-Output GPs & Examples
Conclusions and Current Research

Deterministic Solver

Bayesian Training
- Tree construction.
- Experimental design.
- Output correlations.
- HDMR terms.

Surrogate Model

Statistics

PDFs

Error bars

Observed input
\[ A = \{ \alpha^{(s)} \}_{s=1}^{S_A} \]
Data-Driven Stochastic Input Modeling

**Input reduction:**
KPCA, Isomap, LLE, LEM, HEM, GTM, etc.

\[ \mathcal{R}(\alpha) = z \]
\[ \alpha = \mathcal{C}(z) \]

**Input reconstruction:**
Optimization problem constrained on physical info.

**Density estimation:**
Polynomial Chaos Representation

\[ z_i = \sum_m \beta_{im} H_m(\xi) \]
Input Reduction: Kernel PCA

Use a non-linear map

$$\Phi : \mathbb{R}^M \rightarrow F \subset \mathbb{R}^M, \quad \alpha \rightarrow \phi(\alpha) = A,$$

to unwrap the observed data

$$\{\alpha^{(s)}\}_{s=1}^{S_A} \rightarrow \{\Phi(\alpha^{(s)})\}_{s=1}^{S_A} = \{A^{(s)}\}_{s=1}^{S_A}$$
on $F$ (feature space). Then, do PCA (K-L) on the feature space $F$.

Reduction achieved by keeping just a few terms of PCA.

References: [Schlkopf et al. (1998)], [Ma & Zabaras (2011)].
Fortunately...

Only dot products are needed:

\[ \Phi(\alpha^{(s)}) \cdot \Phi(\alpha^{(s')}) := k(\alpha^{(s)}, \alpha^{(s')}) \]

Choices of kernels:

- \[ k(\alpha^{(s)}, \alpha^{(s')}) = (\alpha^{(s)} \cdot \alpha^{(s')}) \] (standard PCA).

- Gaussian kernel:

\[ k(\alpha^{(s)}, \alpha^{(s')}) = \exp \left(-\frac{\|\alpha^{(s)} - \alpha^{(s')}\|^2}{2\sigma^2}\right) \] (1)

In this work, we use (c = user defined constant):

\[ \sigma^2 = c \frac{1}{S_A} \sum_{s=1}^{S_A} \min_{s' \neq s} \|\alpha^{(s)} - \alpha^{(s')}\|^2, \quad s' = 1, \ldots, S_A, \] (2)
SGeMS (Stanford Geostatistical Modeling Software).
- 0 – 1 image large scale image of channelized permeability field.
- Cut it in 1000 45 × 45 pieces to generate training set.
- Training set consists of 1000 samples.
- Each sample has 2025 dimensions.
Example: Input Data

- Performing KPCA reduction.
- We keep 30 eigenvalues of feature space.
- 75% of the field energy.
- Residual variance (in feature space) is 0.003.
**Problem Definition**

Given a reduced input $z$, reconstruct a field $\alpha = C(z)$.

Assuming locally linear reduced manifold:

- Find $L$ observed nearest neighbors $z^{(s_l)}$.
- Find the corresponding observed high-dimensional inputs $\alpha^{(s_l)}$.
- Assume the following form for the reconstruction:

\[
C^L(z) = \sum_{l=1}^{L} d^*_l \alpha^{(s_l)}.
\]

where

\[
\{d^*_l\}_{l=1}^L = \text{arg} \min_{\sum_{l=1}^{L} d_l = 1} \| \mathcal{R} \left( \sum_{l=1}^{L} d_l \alpha^{(s_l)} \right) - z \|_2^2.
\]
Example: Reconstruction

Reconstruct a test sample:

- Test sample
- Reconstruction from Kernel PCA with k=30
- Reconstructed with Linear PCA: k = 30

Reconstruction of several test samples:
Problem Definition

Given the reduced observations \( \{ z^{(s)} \}_{s=1}^{S_A} \), deduce their probability density.

- Expand using a GPC representation:
  \[
  z_i(\xi) = \sum_{m} \beta_{im} H_m(\xi), \ i = 1, \ldots, K_r.
  \]

- Coefficients may be found via:
  - **Maximum likelihood**: [Descelie \( \text{et al.} \) (2006)], [Stefanou \( \text{et al.} \) (2009)].
  - **Rosenblatt transformation**: [Rosenblatt (1952)], [Das \( \text{et al.} \) (2009)].
  - **Probabilistic Graphical Model Structure Learning**: [J. Wan & N. Zabaras (2012)].
Problem Definition

Goal

Evaluate the statistics with as few calls to the deterministic solver as possible.
Two-Stage Solution Strategy

\[ \mathcal{X} = \bigcup_{i=1}^{l} \mathcal{X}_i \]

- Deterministic Solver
- MGPR Training
- Mesh Refinement

Iterate until tolerance is reached.

Calculation of the statistics.
General Setting I

- Input space *adaptively* decomposed in *stochastic elements*:

  \[ X = \bigcup_{i=1}^{I} X_i \text{ and } \text{int}(X_i) \cap \text{int}(X_j) = \emptyset, \forall i, j \in I, i \neq j. \]

  We use rectangles:

  \[ X_i = \times_{k=1}^{K} [a_{i}^k, b_{i}^k]. \]

- Assume independent random variables (common case in UQ):

  \[ p(x) = \prod_{k=1}^{K} p_k(x_k). \]
Conditional input probability:

\[ p_i(x) := \frac{p(x)}{P(X_i)} 1_{x_i}(x) = \prod_{k=1}^{K} \frac{p_k(x_k)}{P_k([a_{ik}, b_{ik}])} 1_{[a_{ik}, b_{ik}]}(x_k) \]

\( N \) (fixed) data points drawn from \( p_i(x) \) observed on \( X_i \):

\[ D := \left\{ (x^{(n)}, y^{(n)}) \right\}_{n=1}^{N} \]

A MGP is learnt on each element \( f_i \).

The \textit{global} surrogate:

\[ f(x) := \sum_{i=1}^{I} f_i(x) 1_{x_i}(x). \]
Multi-output Gaussian Process I

- Scale the observed data:

\[
\mu_{\text{obs}, r} = \frac{1}{N} \sum_{n=1}^{N} y_{r}^{(n)}, \quad \sigma_{\text{obs}, r}^2 = \frac{1}{N} \sum_{n=1}^{N} (y_{r} - \mu_{\text{obs}, r})^2.
\]

- Define scaled responses: \( g_r(x) = \frac{f_r(x) - \mu_{\text{obs}, r}}{\sigma_{\text{obs}, r}} \).

- Assume conditional independence given the hyper-parameters:

\[
g_r(x)|\theta \sim \mathcal{GP}(0, k(x, x'; \theta)) \).
\]

- Obtain \( \theta \) by maximizing the marginal likelihood of the data.
Scaling back to the original responses, we can easily see that the predictive distribution of \( f_r(\mathbf{x}) \) is Gaussian:

\[
f_r(\mathbf{x}) | \mathcal{D}_r, \theta^* \sim \mathcal{N}(\mu_{f_r}(\mathbf{x}; \theta^*), \sigma^2_{f_r}(\mathbf{x}; \theta^*)) ,
\]

with mean

\[
\mu_{f_r}(\mathbf{x}; \theta^*) = \sigma_{\text{obs}, r} \mathbf{k}^T \mathbf{K}^{-1} \mathbf{z}_r + \mu_{\text{obs}, r},
\]

and variance

\[
\sigma^2_{f_r}(\mathbf{x}; \theta^*) = \sigma^2_{\text{obs}, r} \left( k(\mathbf{x}, \mathbf{x}; \theta^*) - \mathbf{k}^T \mathbf{K}^{-1} \mathbf{k} \right) .
\]
Adaptive Domain Decomposition

Assume given decomposition and current global surrogate learnt from data.

- Refine an element or not?
- What is the most important dimension?
Let $\sigma^2_f_r(x)$ be the predictive variance of the $r = 1, \ldots, M$ output of the local surrogate of $f^i$ at $x \in X_i$.

The predictive variance of the global surrogate is:

$$
\sigma^2_f(x) = \sum_{i=1}^{I} \sigma^2_{f_r}(x) 1_{X_i}(x).
$$

Uncertainty of prediction at test point:

$$
\sigma^2_f(x) := \frac{1}{M} \sum_{r=1}^{M} \sigma^2_{f_r}(x).
$$
Uncertainty of prediction over all the domain:

\[ \sigma^2_{\mathbf{f},p} := \int_{\mathbf{x}} \sigma^2_{\mathbf{f}}(\mathbf{x})p(\mathbf{x}) \, d\mathbf{x}. \]

Element Contribution:

\[ \sigma^2_{\mathbf{f},p} = \sum_{i=1}^{l} \sigma^2_{\mathbf{f},p_i} P(X_i). \]
Refine or Not?

Yes, if

$$\sigma^2_{f,p} P(X_i) > \delta.$$
The Most Important Dimension

- Squared Exponential covariance defines scales (unique to element):
  \[
  k_{SE}(x, x') = s_i^2 \exp \left( -\frac{1}{2} \sum_{d=1}^{D} \frac{(x_k - x'_k)^2}{\ell^2_k} \right),
  \]

- Probability of \( x_k \) falling inside \( X_i \):
  \[
  p^i_k = \int_{a^i_k}^{b^i_k} p_k(x_k) \, dx_k.
  \]
  Notice that if \( p_k \) is uniform, then
  \[
  p^i_k \propto b^i_k - a^i_k,
  \]
  is proportional to the extent of the along dimension \( k \).

- Importance of dimension:
  \[
  I^i_k = \frac{p^i_k}{\ell^i_k}.
  \]
How to Refine an Element?

Split perpendicular to the most important dimension:

\[ k^* = \arg \max_k I_k^i = \arg \max_k \frac{p_k^i}{\ell_k^i}. \]

The element is split at the median of the conditional distribution \( x_{k,\text{med}}^i \), so that both resulting sides are equally probable, i.e.

\[ P_k^i([a_k^i, x_{k,\text{med}}^i]) = P_k^i([x_{k,\text{med}}^i, b_k^i]) = \frac{1}{2}. \]

Notice that if \( p_k \) is uniform, then

\[ x_{k,\text{med}}^i = \frac{b_k^i - a_k^i}{2}. \]
We have to consider two distinct cases:

1. No data have been observed yet and we only have a single element (i.e. \( X \) itself).

2. We have obtained a fit of the response over an element \( X_i \) based on \( N_i \) observations

\[
\mathcal{D}_i = \{ (x_i^{(n)}, y_i^{(n)}) \}_{n=1}^{N_i},
\]

and we have decided to split it in two elements \( X_{i,1} \) and \( X_{i,2} \) so that

\[
X_i = X_{i,1} \cup X_{i,2} \text{ and } X_{i,1} \cap X_{i,2} = \emptyset.
\]
Collection of the Observations I

Given $\mathcal{D}_{i,1}^n$, observe the input point $\mathbf{x}^{\text{new},n+1} \in \mathbf{X}_{i,1}$ that maximizes the joint uncertainty of all outputs biased by the conditional distribution of the element:

$$\sigma_f^2(\mathbf{x}; \theta^*; \mathcal{D}_{i,1}^n) = \frac{1}{M} \sum_{r=1}^{M} \sigma_{f,r}^2(\mathbf{x}; \theta^*, \mathcal{D}_{i,1}^n).$$

That is,

$$\mathbf{x}^{\text{new},n+1} = \arg \max_{\mathbf{x} \in \mathbf{X}_{i,1}} \sigma_f^2(\mathbf{x}; \theta^*; \mathcal{D}_{i,1}^n) p_i(\mathbf{x}).$$

We stop, either if $N$ data points have been collected in $\mathcal{D}_{i,1}^n$, or if

$$\sigma_{f,p_{i,1}}^2(\mathcal{D}_{i,1}^n) P(\mathbf{X}_{i,1}) \leq \delta,$$

where $\sigma_{f,p_{i,1}}^2(\mathcal{D}_{i,1}^n) P(\mathbf{X}_{i,1})$ is the expectation of $\sigma_f^2(\mathbf{x}; \theta^*; \mathcal{D}_{i,1}^n)$ with respect to the conditional probability $p_{i,1}(\mathbf{x})$ of $\mathbf{X}_{i,1}$. 
Local Statistics I

- All quantities in this section are local to the element $X_i$ but the notation is suppressed.

- We are interested in estimating all moments $m^q = (m_1^q, \ldots, m_M^q)$, where

$$m_r^q = \int_{X_i} f_r^q(x)p_i(x)dx,$$  \hspace{1cm} (6)

and $q \geq 1$.

- Let us denote the predictive distribution for the $q$ power of the response at $x \in X_i$ by

$$f_r^q(x)|D_r, \theta^q \sim \mathcal{N}\left(\mu_{f_r^q}(x; \theta^q), \sigma_{f_r^q}^2(x; \theta^q)\right),$$

where $\mu_{f_r^q}(x; \theta^q)$ is the predictive mean and $\sigma_{f_r^q}^2(x; \theta^q)$ the predictive variance for $r = 1, \ldots, M$. 
For convenience, let us write the predictive mean at $\mathbf{x}$ as

$$
\mu_{f_q}(\mathbf{x}; \theta^q) = \sum_{n=1}^{N} \alpha^q_{rn} k(\mathbf{x}^{(n)}, \mathbf{x}; \theta^q) + \mu_{\text{obs},r}^q,
$$

and the predictive variance at $\mathbf{x}$ as

$$
\sigma^2_{f_q}(\mathbf{x}; \theta^q) = (\sigma_{\text{obs},r}^q)^2 \left( k(\mathbf{x}, \mathbf{x}; \theta^q) - k_r^q, (K^q)^{-1} k^q \right) .
$$
To obtain analytic estimates of the predictive distribution of the statistics, let us make the simplifying assumption that the predictions at different input points \( x \) are conditionally independent given the data and the hyper-parameters. Then, by the additivity of independent normal variables, we arrive at the approximation:

\[
m_r^q | \mathcal{D}_r, \mathbf{\theta}^q \sim \mathcal{N} \left( \mu_{m_r^q}, \sigma_{m_r^q}^2 \right),
\]

where the predictive mean of \( m_r^q \) is

\[
\mu_{m_r^q} = \int_{\mathbf{x}_i} \mu_{f_r^q}(\mathbf{x}; \mathbf{\theta}^q) \rho_i(\mathbf{x}) d\mathbf{x},
\]

and its predictive variance

\[
\sigma_{m_r^q}^2 = \int_{\mathbf{x}_i} \sigma_{f_r^q}^2(\mathbf{x}; \mathbf{\theta}^q) \rho_i(\mathbf{x}) d\mathbf{x}.
\]
The integrals can be casted in terms of integrals of the covariance function:

$$\mu_{m^q_r} = \sum_{n=1}^{N} \alpha_{rn}^q \epsilon_{qn} + \mu_r^q,$$

and

$$\sigma^2_{m^q_r} = \left( \sigma_{\text{obs},r}^q \right)^2 \left( c^q - \sum_{n,l=1}^{N} (K^q)^{-1}_{nl} \nu_{nl}^q \right),$$

where

$$\epsilon_{qn}^q = \int_{X_i} k(x^{(n)}, x; \theta^q) p_i(x) dx,$$

$$c^q = \int_{X_i} k(x, x; \theta^q) p_i(x) dx,$$
Local Statistics V

$$\nu_{n_1}^q = \int_{\mathbf{x}_i} k(\mathbf{x}, \mathbf{x}^{(n)}; \theta^q) k(\mathbf{x}, \mathbf{x}^{(l)}; \theta^q) p_i(\mathbf{x}) d\mathbf{x},$$

and $$(K^q)^{-1}_{nl}$$ is the $nl$ element of the inverse $q$ covariance matrix $$(K^q)^{-1}.$$
Notice that $m_r^q$ can be decomposed as

$$m_r^q = \int_x f_r^q(x) p(x) dx$$

$$= \sum_{i=1}^l \int_{x_i} f_r^q(x) \frac{p(x)}{P(X_i)} dx P(X_i)$$

$$= \sum_{i=1}^l \int_{x_i} f_r^q(x) p_i(x) dx P(X_i),$$

or

$$m_r^q = \sum_{i=1}^l m_r^{q,i} P(X_i),$$

where $m_r^{q,i}$ is the local $q$ moment over $X_i$. 
Assuming conditional independence of the predictive distributions given the data and the hyper-parameters, we obtain that

$$m_r^q | D_r, \theta^q \sim \mathcal{N} \left( \mu_{m_r^q}, \sigma_{m_r^q}^2 \right),$$

where the predictive mean is

$$\mu_{m_r^q} = \sum_{i=1}^{l} \mu_{m_r^q,i} P(X_i),$$

and the variance

$$\sigma_{m_r^q}^2 = \sum_{i=1}^{l} \sigma_{m_r^q,i}^2 P(X_i).$$
Finally, we derive a normal approximation to the predictive distribution for the variance of the response $\mathbf{v} = (v_1, \ldots, v_M)$:

$$v_r \sim \mathcal{N}\left(\mu_{v_r}, \sigma^2_{v_r}\right).$$

Under the assumption of conditional independence of $m_r^q$, $q = 1, 2$, the predictive mean of $v_r$ is given by

$$\mu_{v_r} = \mu_m^2 - \mu^2_{m_r} - \sigma^2_{m_r}.$$ 

The predictive variance is

$$\sigma^2_{v_r} = \sigma^2_{m_r} + 4\mu^2_{m_r}\sigma^2_{\mu_r} + 2\sigma^4_{\mu_r}. $$
Consider the real function:

\[ f_1(x) = \begin{cases} 
\sin \left( \frac{\pi x}{5} \right) + \frac{1}{5} \cos \left( \frac{4\pi x}{5} \right), & x \leq 10 \\
\frac{x}{10} - 1, & \text{otherwise}
\end{cases} \]  \tag{7}

on the domain \( X = [0, 20] \). For \( x \leq 10 \), it varies with two different frequencies. For \( x > 10 \) it is linear and finally it has a discontinuity at \( x = 10 \).
Figure: The MSE in the prediction of $f_1(x)$ as a function of the observed samples for MGP and ASGC for various $\epsilon$. 
1D Anisotropic, Discontinuous Function

**Figure:** Left (a): comparison of the predictive mean $\mu_{f_1}(x)$ (dashed blue line) and 95% error bars (shaded grey area) with true response $f_1(x)$ (solid red line), where the symbols mark the observed data. Right (b): predicted length scale across the domain. Tolerances $\delta = 10^{-2}$ with 13 samples gathered.
1D Anisotropic, Discontinuous Function

Figure: Left (a): comparison of the predictive mean $\mu f_1(x)$ (dashed blue line) and 95% error bars (shaded grey area) with true response $f_1(x)$ (solid red line), where the symbols mark the observed data. Right (b): predicted length scale across the domain. Tolerances $\delta = 10^{-4}$ with 25 samples gathered.
Figure: Left (a): comparison of the predictive mean $\mu_{f_1}(x)$ (dashed blue line) and 95% error bars (shaded grey area) with true response $f_1(x)$ (solid red line), where the symbols mark the observed data. Right (b): predicted length scale across the domain. Tolerances $\delta = 10^{-6}$ with 94 samples gathered.
Consider the system of ordinary differential equations [Wan and Karniadakis (2005)]

\[
\begin{align*}
\frac{dy_1}{dt} &= y_1y_3, \\
\frac{dy_2}{dt} &= -y_2y_3, \\
\frac{dy_3}{dt} &= -y_1^2 + y_2^2,
\end{align*}
\]

subject to random initial conditions at \( t = 0 \).

\[y_1(0) = x_1, \quad y_2(0) = x_2, \quad y_3(0) = x_3,\]

where \( x_i, i = 1, 2, 3 \) are random parameters.
Numerical Examples: Krainchnan-Orszag

KO-1:

\[ y_1(0) = 1, \ y_2(0) = 0.1x, \ y_3(0) = 0, \]

KO-2:

\[ y_1(0) = 1, \ y_2(0) = 0.1x_1, \ y_3(0) = x_2, \]

KO-3:

\[ y_1(0) = x_1, \ y_2(0) = x_2, \ y_3(0) = x_3, \]

where

\[ x_i \sim U([-1, 1]), \ i = 1, 2, 3. \]

The error of the statistics will be evaluated using the (normalized) \( L_2 \) norm of the error in variance defined by:

\[
E_{L2} = \frac{1}{M} \sum_{r=1}^{M} (v_{r,MC} - \mu_{v_r})^2 ,
\]  

(8)

where \( \mu_{v_r} \) is the predictive mean of \( v_r \) (Eq. (25)). The results are compared with SGC and ASGC.
Figure: KO-2: the $L_2$ norm of the error in variance as a function of the observed samples for MGP, SGC and ASGC.
Figure: KO-2: The prediction at $y_3$ ($t = 10$) with the stochastic elements (left, $(a)$) and the observed samples (right, $(b)$) for tolerances $\delta = 10^{-3}$.
Numerical Examples: Krainchnan-Orszag

Figure: KO-2: The prediction at $y_3 \ (t = 10)$ with the stochastic elements (left, (a)) and the observed samples (right, (b)) for tolerances $\delta = 10^{-5}$. 
Figure: KO-2: The prediction at $y_3$ ($t = 10$) with the stochastic elements (left, (a)) and the observed samples (right, (b)) for tolerances $\delta = 10^{-7}$. 
Numerical Examples: Krainchnan-Orszag

Figure: KO-2: The observed data for Gaussian distribution with zero mean (a) and mean \((-0.5, -0.5)\) (b) (both with unit variance) for tolerance \(\delta = 10^{-2}\).
Figure: KO-2: The observed data for Gaussian distribution with zero mean (a) and mean \((-0.5, -0.5)\) (b) (both with unit variance) for tolerance $\delta = 10^{-3}$. 
Numerical Examples: Krainchnan-Orszag

**Figure:** KO-2: The observed data for Gaussian distribution with zero mean (a) and mean \((-0.5, -0.5\) (b) (both with unit variance) for tolerance \(\delta = 10^{-4}\).
Figure: KO-2: The observed data for Gaussian distribution with zero mean (a) and mean \((-0.5, -0.5\)) (b) (both with unit variance) for tolerance \(\delta = 10^{-5}\).
Figure: KO-2: The variance of $y_3(t)$ for (a) Gaussian distribution with zero mean and unit variance and (b) beta distribution with $a = 2, b = 5$. for tolerance $\delta = 10^{-4}$. 
Figure: KO-3: the $L_2$ norm of the error in variance as a function of the observed samples for MGP, SGC and ASGC.
Figure: KO-3: kernel density estimation of the PDF of $y_2 \ (t = 10)$ (left) and $y_3 \ (t = 10)$ (right) using $10^5$ samples.
We consider a simple stochastic elliptic problem (Nobile et al., 2008).

\[-\nabla \cdot (a_K(\omega, \cdot) \nabla u(\omega, \cdot)) = f(\cdot), \text{ in } D = [0, 1]^2,\]
\[u(\omega, \cdot) = 0, \text{ on } \partial D,\]

Denote the random variables with $\omega$ instead of $x$. Choose a deterministic load

\[f(x, y) = 100 \cos(x) \sin(y),\]

The deterministic problem is solved with FEM using $(20 \times 20$ grid) bilinear quadrilateral elements. The random $a_K(\omega, x)$ is:

\[\log(a_K(\omega, x, y) - 0.5) = 1 + \omega_1 \left(\frac{\sqrt{\pi}L}{2}\right)^{1/2} + \sum_{k=2}^{K} \xi_k \phi_k(x) \omega_k,\]
Numerical Examples: Elliptic Problem II

where

\[ \xi_k := \left( \sqrt{\pi} L \right)^{1/2} \exp \left( \frac{-\left( \left\lfloor \frac{k}{2} \right\rfloor \pi L \right)^2}{8} \right), \quad \text{for } k \geq 2, \]

and

\[ \phi_k(x) := \begin{cases} \sin \left( \frac{\left\lfloor \frac{k}{2} \right\rfloor \pi x}{L_p} \right), & \text{if } k \text{ is even}, \\ \cos \left( \frac{\left\lfloor \frac{k}{2} \right\rfloor \pi x}{L_p} \right), & \text{if } k \text{ is odd}. \end{cases} \]

We choose the \( \omega_k, k = 1, \ldots, K \) to be independent identically distributed random variables

\[ \omega_k \sim U([-\sqrt{3}, \sqrt{3}]). \]

Hence, the stochastic input space is \( \Omega = [-\sqrt{3}, \sqrt{3}]^K \). Finally, we set

\[ L_p = \max\{1, 2L_c\} \quad \text{and} \quad L = \frac{L_c}{L_p}, \]
where $L_c$ is called the *correlation length*. This resembles a Karhunen-Loève expansion of a 2D random field:

$$
\text{Cov}[\log(a_K - 0.5)]((x_1, y_1), (x_2, y_2)) = \exp\left(-\frac{(x_1 - x_2)^2}{L_c^2}\right).
$$

In this study, we set the correlation length to $L_c = 0.6$ and test the convergence of our method for $K = 40$ input dimensions. The results are evaluated by calculating the $L_2$ error in variance using a plain MC estimate with $10^6$ samples.
Figure: Elliptic, \( K = 40 \): The \( L_2 \) norm of the error in variance of the elliptic problem with \( K = 40 \) inputs as a function of the observed samples for MGP and ASGC.
**Numerical Examples: Elliptic Problem**

![Graphs of predicted variance and MC variance](image)

**Figure:** Elliptic, $K = 40$: Predicted variance versus the MC prediction with $10^6$ samples.
Consider the dimensionless form of the Oberbeck-Boussinesq approximation using the vorticity transport equation in streamfunction formulation:

\[
- \frac{\partial}{\partial t} \nabla^2 \psi - \frac{\partial \psi}{\partial y} \frac{\partial}{\partial x} \nabla^2 \psi + \frac{\partial \psi}{\partial x} \frac{\partial}{\partial y} \nabla^2 \psi = - Pr \nabla^4 \psi + Ra Pr \frac{\partial T}{\partial x},
\]

\[
\frac{\partial T}{\partial t} \frac{\partial \psi}{\partial y} \frac{\partial T}{\partial x} - \frac{\partial \psi}{\partial x} \frac{\partial T}{\partial y} = \nabla^2 T,
\]

where Pr and Ra are the Prandtl and Rayleigh numbers respectively. In this formulation, the velocity field is given by

\[
u = \frac{\partial \psi}{\partial x}, \quad v = - \frac{\partial \psi}{\partial x}.
\]
Numerical Examples: Natural Convection

The left vertical wall (cold) is supposed to be a one dimensional Gaussian stochastic process with mean $-0.5$ and exponential covariance

$$\text{Cov}[x_1, x_2] = s^2 \exp \left\{ - \frac{|x_1 - x_2|}{L_C} \right\},$$

where $s^2$ is the variance of the signal and $L_C$ the correlation length.
Numerical Examples: Natural Convection

Computational details:

- We set $L_C = 1$, $s = 1$, $Pr = 1$, $Ra = 5000$.
- We test against 80000 MC samples.
- The problem is solved using spectral elements as implemented in Nektar.
Numerical Examples: Natural Convection

(a) K=8, MGP ($N = 20, \delta = 10^{-3}$): std. of $u$

(b) K=8, MC: std. of $u$

Figure: Natural Convection: prediction for the standard deviation of the velocity $u$ compared to a MC estimate for $K = 8$ input dimensions.
Numerical Examples: Natural Convection

(a) K=8, MGP ($N = 20, \delta = 10^{-3}$): std. of $T$

(b) K=8, MC: std. of $T$

Figure: Natural Convection: prediction for the standard deviation of the temperature $T$ compared to a MC estimate for $K = 8$ input dimensions.
Numerical Examples: Natural Convection

(a) Mean $T$ prediction  (b) Absolute error  (c) 2 std.’s of $T$  (d) Input point

**Figure:** Natural Convection ($K = 4, \delta = 10^{-5}$): Comparing the prediction at a random input point with the true response.
Water and oil, ignore gravity effects and capillary forces, assume porosity is a constant.

\[
\begin{align*}
\nabla \cdot \mathbf{u} &= 0 \\
\mathbf{u} &= -K(x, \omega) \nabla p, \forall x \in D, \\
\frac{\partial S(x, t, \omega)}{\partial t} + \mathbf{u} \cdot \nabla S(x, t, \omega) &= 0, \forall x \in D, t \in [0, T], \\
p &= \bar{p}, \text{ on } \partial D_p, \quad \mathbf{u} \cdot \mathbf{n} = 0, \text{ on } \partial D_u.
\end{align*}
\]

The log-permeability is modeled using data and KPCA as discussed earlier.
Numerical Examples: Flow through porous media with KPCA input

(a) MGP, $\delta = 10^{-3}$  
(b) MGP, $\delta = 10^{-4}$  
(c) MC

Figure: Comparison of standard deviations of the saturation $S$ using MGP with approx. 800 (a) and 6,500 (b) samples with a MC simulation using $10^6$ samples at 0.2 PVI.
Horn Problem I

2D Helmholtz equation in random media:

$$-\nabla^2 p(x, y, \omega) + k^2 (1 + n^2(x, y, \omega)) p(x, y, \omega) = 0,$$

with boundary conditions:

$$\frac{\partial p}{\partial \vec{n}} - ik p = 0, \text{ on } \Gamma_1,$$

$$\frac{\partial p}{\partial \vec{n}} = 0, \text{ on } \Gamma_2,$$

$$p(x, y, \omega) = f(x, y), \text{ on } \Gamma_3.$$

Notation:

- $\vec{n}$: unit normal of the boundary.
- $k$: wave number.
Horn Problem II

- \( n^2(x, y, \omega) \): random reflectivity. For demonstration, we use here \( k = 2 \):

\[
n^2(x, y, \omega) = \sum_{i=1}^{k} \xi_i(\omega)\psi_i(x, y),
\]

with

\[
\psi_1(x, y) = \sin^2(2\pi x) \sin^2(2\pi y),
\]
\[
\psi_2(x, y) = \cos^2(6\pi x) \sin^2(8\pi y).
\]

- \( \Gamma_1 \): outer boundary.
- \( \Gamma_2 \): boundary of the horn.
- \( \Gamma_3 \): source boundary.
Horn Problem III

(a) Illustration

(b) Mesh
Figure: The mean is extremely easy to capture.
Horn Problem

(a) True Standard Deviation (MC)  
(b) MGP $N = 10$, $\delta = 10^{-2}$, 10 observations
Horn Problem

(c) True Standard Deviation (MC)  (d) MGP $N = 10, \delta = 10^{-3}, 20$ observations
(e) True Standard Deviation (MC)  (f) MGP $N = 10$, $\delta = 10^{-4}$, 70 observations
Assume that $f(\cdot)$ is a $q$-dimensional GP:

$$f(\cdot)|B, \Sigma, r \sim \mathcal{N}_q (m(\cdot; B), c(\cdot, \cdot; r)\Sigma),$$

with mean:

$$E [f(x)|B, \Sigma, r] = m(x; B),$$

and covariance:

$$\text{Cov} [f(x_1), f(x_2)|B, \Sigma, r] = c(x_1, x_2; r)\Sigma,$$

over all variables (stochastic input, space and time):

$$x = (\xi, x_s, t).$$

Notation:

- $\xi$: Stochastic inputs.
- $x_s$: Spatial variables.
- $t$: Time.
The Mean Function

• The mean will be a *generalized linear model*:

\[
m(x; B) = B^T h(x),
\]

where

\[
h: \mathbb{R}^k \to \mathbb{R}^m, h(\cdot) = (h_1(\cdot), \ldots, h_m(\cdot)).
\]

• They are formed from a tensor product:

\[
\mathcal{H} = \mathcal{H}_\xi \otimes \mathcal{H}_s \otimes \mathcal{H}_t.
\]

where:

- \( k \): total number of input dimensions \((k = k_\xi + k_s + 1)\).
- \( \mathcal{H}_\xi = \{ h_{\xi,1}(\xi), \ldots, h_{\xi,m_\xi}(\xi) \} \): stochastic basis.
- \( \mathcal{H}_s = \{ h_{s,1}(\mathbf{x}_s), \ldots, h_{s,m_s}(\mathbf{x}_s) \} \): spatial basis.
- \( \mathcal{H}_t = \{ h_{t,1}(t), \ldots, h_{t,m_t}(t) \} \): time basis.
- \( m = m_\xi m_s m_t \): total number of basis functions.

• Parameters to be learnt:

- \( B \in \mathbb{R}^{m \times q} \): the weights.
The Covariance Function

- Remember:
  \[
  \text{Cov} \left[ f(x_1), f(x_2) \middle| \mathcal{B}, \Sigma, r \right] = c(x_1, x_2; r)\Sigma.
  \]

- We assume that:
  \[
  c(x_1, x_2; r) := c_\xi(\xi_1, \xi_2; r_\xi)c_s(x_{s,1}, x_{s,2}; r_s)c_t(t_1, t_2; r_t).
  \]

- We usually choose an Square Exponential for each dimension:
  \[
  c_\xi(\xi_1, \xi_2; r_\xi) = \exp \left\{ -\frac{1}{2} \sum_{s=1}^{k} \frac{(x_{1s} - x_{2s})^2}{r_{\xi,s}^2} \right\},
  \]
  and analogously for \(c_s(\cdot, \cdot; r_s)\) and \(c_t(\cdot, \cdot; r_t)\).

- Notation:
  - \(r \in \mathbb{R}^k\): length scales of input dimension \((r = (r_\xi, r_s, r_t))\)
  - \(\Sigma \in \mathbb{R}^{q \times q}\): correlation matrix (correlates distinct outputs).

- Parameters to learn: \(r\) and \(\Sigma\).
Input Observations

- Given a $\xi \in \mathbb{R}^{k_\xi}$, the computer code reports the response on:
  - a given set of $n_s$ spatial points (e.g. finite element nodes):
    $$X_s = (x_{s,1}, \ldots, x_{s,n_s})^T \in \mathbb{R}^{n_s \times k_s},$$
  - at $n_t$ timesteps:
    $$X_t = (t_1, \ldots, t_{n_t}) \in \mathbb{R}^{n_t \times 1}.$$
  - Sampling a total of $n_\xi$ stochastic input points:
    $$X_\xi = (\xi_1, \ldots, \xi_{n_\xi})^T \in \mathbb{R}^{n_\xi \times k_\xi},$$
  generates a total of
    $$n = n_\xi n_s n_t,$$
  output samples.
Input Observations

Given a $\xi \in \mathbb{R}^{k_\xi}$, the computer code reports the response on:

- a given set of $n_s$ spatial points (e.g. finite element nodes):
  \[ X_s = (x_{s,1}, \ldots, x_{s,n_s})^T \in \mathbb{R}^{n_s \times k_s}, \]

- at $n_t$ timesteps:
  \[ X_t = (t_1, \ldots, t_{n_t}) \in \mathbb{R}^{n_t \times 1}. \]

- Sampling a total of $n_\xi$ stochastic input points:
  \[ X_\xi = (\xi_1, \ldots, \xi_{n_\xi})^T \in \mathbb{R}^{n_\xi \times k_\xi}, \]

which generates a total of
\[ n = n_\xi n_s n_t, \]
output samples

Notice that $n$ becomes extremely large even for a moderate number of observations! We can avoid this problem by arranging the outputs in the right order...
Arranging the Output Observations

- Each stochastic input $\xi$ corresponds to outputs:
  \[ Y_\xi = (y_{\xi,1}^T \cdots y_{\xi,n_s}^T)^T \in \mathbb{R}^{(n_s n_t) \times q}. \]

- Each $y_{\xi,i} \in \mathbb{R}^{n_t \times q}$ is the response at the spatial point $x_{s,i}$ at each timestep:
  \[ y_{\xi,i} = (y_{\xi,i,1} \cdots y_{\xi,i,n_t})^T \in \mathbb{R}^{n_t \times q}. \]

- Each $y_{\xi,i,j} \in \mathbb{R}^q$ is the response at the spatial point $x_{s,i}$ at time $t_j$:
  \[ y_{\xi,i,j} = (y_{\xi,i,j,1} \cdots y_{\xi,i,j,q})^T \in \mathbb{R}^{q \times 1}. \]

- The observations for different stochastic inputs are arranged as:
  \[ Y = (Y_{\xi_1}^T \cdots Y_{\xi_{n_\xi}}^T)^T \in \mathbb{R}^{n \times q}. \]
Each stochastic input $\xi$ corresponds to outputs:

$$Y_\xi = (y_{\xi,1}^T \ldots y_{\xi,n_s}^T)^T \in \mathbb{R}^{(n_s n_t) \times q}.$$  

Each $y_{\xi,i} \in \mathbb{R}^{n_t \times q}$ is the response at the spatial point $x_{s,i}$ at each timestep:

$$y_{\xi,i} = (y_{\xi,i,1}^T \ldots y_{\xi,i,n_t}^T)^T \in \mathbb{R}^{n_t \times q}.$$  

Each $y_{\xi,i,j} \in \mathbb{R}^q$ is the response at the spatial point $x_{s,i}$ at time $t_j$:

$$y_{\xi,i,j} = (y_{\xi,i,j,1}^T \ldots y_{\xi,i,j,q}^T)^T \in \mathbb{R}^{q \times 1}.$$  

The observations for different stochastic inputs are arranged as:

$$Y = (Y_{\xi_1}^T \ldots Y_{\xi_{n_\xi}}^T)^T \in \mathbb{R}^{n \times q}.$$  

This arrangement is special because it simplifies the form of the covariance and design matrices!
The Covariance and Design Matrices

- The covariance matrix is defined by:

\[ \mathbf{A} \in \mathbb{R}^{n \times n}, \quad A_{ij} = c(x_i, x_j; r). \]

- It can be written as:

\[ \mathbf{A} = \mathbf{A}_\xi \otimes \mathbf{A}_s \otimes \mathbf{A}_t. \]

- The design matrix is defined by:

\[ \mathbf{H} = \left( \mathbf{h}(x_1)^T \ldots \mathbf{h}(x_n)^T \right)^T \in \mathbb{R}^{n \times m}. \]

- It can be written as:

\[ \mathbf{H} = \mathbf{H}_\xi \otimes \mathbf{H}_s \otimes \mathbf{H}_t. \]
The covariance and design matrices

- The covariance matrix is defined by:
  \[ A \in \mathbb{R}^{n \times n}, \quad A_{ij} = c(x_i, x_j; r). \]
- It can be written as:
  \[ A = A_\xi \otimes A_s \otimes A_t. \]
- The design matrix is defined by:
  \[ H = \left( h(x_1)^T \ldots h(x_n)^T \right)^T \in \mathbb{R}^{n \times m}. \]
- It can be written as:
  \[ H = H_\xi \otimes H_s \otimes H_t. \]

In reality, we never have to form the huge matrices in order to carry out inference.
The Likelihood Function

The likelihood function is:

\[ Y | B, \Sigma, r \sim \mathcal{N}_{n \times q} (HB, \Sigma, A), \]

where \( H \in \mathbb{R}^{n \times m} \) is the design matrix:

\[ H = (h(x_1), \ldots, h(x_n))^T, \]

and \( A \in \mathbb{R}^{n \times n} \) is the usual covariance matrix:

\[ A_{ij} = c(x_i, x_j; r). \]

Notation:

- \( \mathcal{N}_{n \times q} \): denotes a matrix-normal distribution.

Using the vectorization operation, it can be written as a simple multi-variate normal distribution.
The Predictive Distribution

Using Bayes theorem, the *predictive distribution* at a new point $\mathbf{x}^*$ is (conditioned on the hyper-parameters):

$$y^* | \mathbf{Y}, \mathbf{B}, \Sigma, \mathbf{r} \sim \mathcal{N}_q \left( \mathbf{m}^* + (\mathbf{Y} - \mathbf{HB})^T \mathbf{A}^{-1} \mathbf{a}^*, \Sigma \left( \mathbf{a}^* - (\mathbf{a}^*)^T \mathbf{A}^{-1} \mathbf{a}^* \right) \right),$$

where

$$\mathbf{m}^* := \mathbf{m}(\mathbf{x}^*; \mathbf{B}) = \mathbf{B}^T \mathbf{h}(\mathbf{x}^*),$$

$$\mathbf{a}^* := (c(\mathbf{x}^*, \mathbf{x}_1; \theta) \ldots c(\mathbf{x}^*, \mathbf{x}_n; \theta))^T,$$

and

$$\mathbf{a}^* = c(\mathbf{x}^*, \mathbf{x}^*; \theta).$$

This results can be generalized to yield joint predictive distributions (e.g. at a single $\xi^*$ and multiple spatial or time inputs).
The Predictive Distribution

Using Bayes theorem, the *predictive distribution* at a new point $x^*$ is (conditioned on the hyper-parameters):

$$y^* | Y, B, \Sigma, r \sim \mathcal{N}_q \left( m^* + (Y - HB)^T A^{-1} a^*, \Sigma (a^* - (a^*)^T A^{-1} a^*) \right) ,$$

where

$$m^* := m(x^*; B) = B^T h(x^*) ,$$

$$a^* := (c(x^*, x_1; \theta) \ldots c(x^*, x_n; \theta))^T ,$$

and

$$a^* = c(x^*, x^*; \theta) .$$

For given $B, \Sigma$ and $r$, the mean of this distribution will be identified as a sample from the space of possible surrogates of the code!
We want to predict the response at a new $\xi^*$, space locations $X_s^* \in \mathbb{R}^{n_s^* \times k_s}$ and $X_t^* \in \mathbb{R}^{n_t^* \times 1}$.

In order to derive the joint predictive distribution (conditional of $B$, $\Sigma$ and $r$ at these points we work as follows:

- Let $Y \in \mathbb{R}^{n \times q}$ be the outputs we have observed at inputs $X_{\xi}, X_s$ and $X_t$.
- Let $Y^* \in \mathbb{R}^{(n_s^* n_t^*) \times q}$ be the outputs we want to predict at inputs $\xi^*, X_s^*$ and $X_t^*$.

The joint likelihood is:

$$
\begin{pmatrix}
Y \\
Y^*
\end{pmatrix}
\sim \mathcal{N}_{(n+n_s^* n_t^*) \times q}
\begin{pmatrix}
H \\
H^*
\end{pmatrix} B, \Sigma,
\begin{pmatrix}
A \\
(a_{\xi}(\xi^*)^T \otimes A_s^* \otimes A_t^*) \\
A^{**}
\end{pmatrix}
$$

$$
H^* = h_{\xi}(\xi^*)^T \otimes H_s^* \otimes H_t^* \quad \text{(the rest are defined below).}
$$
The Joint Predictive Mean II

- \( \mathbf{A}^{**} \) is the covariance matrix of the new data.

- Use vectorization to write it in terms of a simple multi-variate distribution.

- Find the conditional of \( \mathbf{Y}^* \) on \( \mathbf{Y} \).

- Go back to matrix normal.

- This way you can prove that the mean of the (conditional) predictive distribution in this case is given by:

\[
\mathbf{M}(\xi^*) = (h_\xi(\xi^*)^T \otimes \mathbf{H}_s \otimes \mathbf{H}_t^*) \mathbf{B} + (a_\xi(\xi^*)^T \otimes \mathbf{A}_s \otimes \mathbf{A}_t^*) \mathbf{W} \in \mathbb{R}^{(n_s n_t) \times q},
\]

where:

- \( h_\xi(\xi^*) = (h_{\xi,1}(\xi^*), \ldots, h_{\xi,m_\xi}(\xi^*)) \in \mathbb{R}^{m_\xi \times 1} \).

- \( a_\xi(\xi^*) = (c_\xi(\xi^*, \xi_1; r_\xi), \ldots, c_\xi(\xi^*, \xi_{n_\xi}; r_\xi)) \in \mathbb{R}^{n_\xi \times 1} \).
The Joint Predictive Mean III

- $H^*_s \in \mathbb{R}^{n_s^* \times m_s}$ and $H^*_t \in \mathbb{R}^{n_t^* \times m_t}$ the design matrices of $X^*_s$ and $X^*_t$, resp. For example:
  \[ H^*_s = (h_s(x^*_s,1) \ldots h_s(x^*_s,n^*_s))^T. \]

- $A^*_s \in \mathbb{R}^{n_s^* \times n_s}$ and $A^*_t \in \mathbb{R}^{n_t^* \times n_t}$ the cross covariances of $X^*_s$ and $X^*_t$ with $X_s$ and $X_t$, resp. For example:
  \[ A^*_{s,ij} = c_s(\xi^*_i, \xi_j; r_\xi). \]

- $W = A^{-1}(Y - HB) \in \mathbb{R}^{n \times q}$ another weight matrix.

This can be used to sample the whole response at different stochastic points.
The mean response, can be found by integrating out $\xi$ using the input distribution $p(\xi)$:

$$M = \int M(\xi^*) p(\xi^*) \, d\xi^*.$$  

It is easy to show that:

$$M = (\bar{h}_\xi^T \otimes H_s^* \otimes H_t^*) B + (\bar{a}_\xi^T \otimes A_s^* \otimes A_t^*) W,$$

where

$$\bar{h}_\xi = \int h_\xi(\xi^*) p(\xi^*) \, d\xi^*,$$

and

$$\bar{a}_\xi = \int a_\xi(\xi^*) p(\xi^*) \, d\xi^*.$$  

The second integrals can be evaluated numerically if each dimension of $\xi$ is independent and analytically if it is the uniform distribution.
It is possible to write down a formula for the variance in general (very complicated).

Things are simplified when we choose (constant mean function):

\[ h(x) = 1. \]

Then the covariance matrix between outputs \( i, j = 1, \ldots, q \) is given by:

\[
C^{ij} = \int (M(\xi^*)_i - M_i)(M(\xi^*)_j - M_j)^T p(\xi^*) d\xi^* \in \mathbb{R}((n_s^* n_t^*) \times (n_s^* n_t^*)),
\]

where \((\cdot)_i\) selects the \( i \)-th column of the matrix.
We have shown that:

\[ C_{ij} = (A_s^* \otimes A_t^*) W_i S W_i^T (A_s^* \otimes A_t^*)^T, \]

where

- \( W_i \in \mathbb{R}^{(n_s n_t) \times n_\xi} \) is a reshaping into a matrix of the \( i \)-th column of the weights matrix, \( W_i \in \mathbb{R}^{n_\xi n_s n_t} \).
- The matrix \( S \in \mathbb{R}^{n_\xi \times n_\xi} \) is given by:

\[
S = \int (a_\xi(\xi^*) - \bar{a}_\xi)(a_\xi(\xi^*) - \bar{a}_\xi)^T p(\xi^*) d\xi^*.
\]

Again these integrals can be calculated numerically if \( \xi \) has independent dimensions and analytically if it has the uniform distribution.
It is easily seen that the variance of output \( i = 1, \ldots, q \) at all space and time locations is given by

\[
v^i = \text{diag}(C^{ii}) \in \mathbb{R}^{n_s^* n_t^*}.
\]

Remark:
The covariance matrices \( C^{ij} \) contain much more information than the variance of each output \( v^i \). They basically contain ALL the second order statistics of the response!
The priors as the same as [Conti & O’Hagan (2010)].

We assign independent priors on the hyper-parameters:

\[ \pi(B, \Sigma, r) := \pi(B)\pi(\Sigma)\pi(r). \]

We choose non-informative improper priors for the weights:

\[ \pi(B) \propto 1, \]
\[ \pi(\Sigma) \propto |\Sigma|^{\frac{-q+1}{2}}, \]

and a log-logistic prior for the length scales:

\[ \pi(r) = \prod_{s=1}^{k} \pi(r_s), \quad \pi(r_s) \propto (1 + r_s^2)^{-1}. \]
The posterior of the hyper-parameters is:

$$p(B, \Sigma, r|Y) \propto \pi(B)\pi(\Sigma)\pi(r)p(Y|B, \Sigma, r).$$

It can be sampled via a Gibbs procedure:

- Initialize $B_0, \Sigma_0$ and $r_0$.
- Sample:
  $$B_{i+1} \sim p(B|Y, \Sigma_i, r_i).$$
- Sample:
  $$\Sigma_{i+1} \sim p(\Sigma|Y, B_{i+1}, r_i).$$
- Sample:
  $$r_{i+1} \sim p(r|Y, B_{i+1}, \Sigma_{i+1}).$$
The Conditional Posteriors

- **\( B \)** follows a matrix-normal:
  \[
  p(B|Y, \Sigma, r) = \mathcal{N}_{m \times q}(B; B_0, \Sigma, H^T A^{-1} H),
  \]
  where
  \[
  B_0 = (H^T A^{-1} H)^{-1} H^T A^{-1} Y.
  \]
- **\( \Sigma \)** follows an inverse-Wishart distribution:
  \[
  p(\Sigma|Y, B, r) = W^{-1}(\Sigma; Y^TG Y, n - m),
  \]
  where
  \[
  G = A^{-1} - A^{-1} H(H^T A^{-1} H)^{-1} H^T A^{-1}.
  \]
- The length scales follow:
  \[
  p(r|Y, B, \Sigma) \propto \pi(r)|A|^{-q/2} \exp \left\{ -\frac{1}{2} \text{tr} \left[ \Sigma^{-1}(Y - HB)^T A^{-1}(Y - HB) \right] \right\}.
  \]
  A Metropolis-Hastings procedure must be used for this.
Each sample of the hyper-parameters gives a sample surrogate model for the code, identified as the mean of the predictive distribution.

How to obtain a sample of the statistics on a stochastic element:

- Sample $\mathbf{B}, \Sigma, \mathbf{r}$ from the posterior distribution.
- Calculate the mean $\mathbf{M}$ and all the covariance matrices $\mathbf{C}^{ij}, i, j = 1, \ldots, q$.
- Taking many samples, one can construct error bars for the statistics.
Changes in the Tree Construction: Refine or Not?

Refine the element $\Xi_i$, if:

$$\int_0^T \int_{\Omega} \int_{\Xi_i} \text{tr} \left[ \Sigma (a^* - a(\xi^*, x_s^*, t^*))^T A^{-1} a(\xi^*, x_s^*, t^*)) \right] p(\xi^*) dx_s^* dt^* \frac{1}{T|\Omega|q} P(\Xi_i) > \delta,$$

- The part in the trace is the covariance of the (single output) predictive distribution at a new point $x^* = (\xi^*, x_s^*, t^*)$.
- We average its diagonal (predictive variances) over the number of outputs (trace of the covariance divided by $q$).
- Average over the space domain $\Omega$ ($|\Omega|$ = volume)
- We average over time ($T$ is the final observation time).
- $P(\Xi_i)$ is the probability of the stochastic element.

There is a (not so simple) formula for that! Numerically (wrt $\xi^*$) if the $\xi$’s are independent, analytically for uniform input. The integrals wrt $x_s$ and $t$ are always analytical.
Changes in the Tree Construction: Splitting Dimension

Split perpendicular to the most *important dimension*:

\[ s^* = \arg \max_s I_s^i = \arg \max_s \frac{p_s^i}{r_{\xi, k}^i}. \]

The element is split at the median of the conditional distribution \( x_{k, \text{med}}^i \), so that both resulting sides are equally probable, i.e.

\[ P_s^i([a_s^i, \xi_{s, \text{med}}^i]) = P_s^i([\xi_{s, \text{med}}^i, b_s^i]) = \frac{1}{2}. \]

Notice that if \( p_s \) is uniform, then

\[ \xi_{s, \text{med}}^i = \frac{b_s^i - a_s^i}{2}. \]

The only difference is that we use the posterior mean of the length scales \( \hat{r}_\xi \).
KO Problem - Data Collection

- This is a 3D dynamical system \((q = 3)\). No space variables \((k_s = 0)\).

- For each stochastic input point \(\xi\), we observe the response at \(n_t = 20\) time instants:

\[
Y_\xi = \begin{pmatrix}
y_1(\xi, t_1) & y_2(\xi, t_1) & y_3(\xi, t_1) \\
y_1(\xi, t_2) & y_2(\xi, t_2) & y_3(\xi, t_2) \\
\vdots & \vdots & \vdots \\
y_1(\xi, t_{20}) & y_2(\xi, t_{20}) & y_3(\xi, t_{20})
\end{pmatrix}
\]

- Gathering everything together, we build the observed-data matrix:

\[
Y = \begin{pmatrix}
Y_{\xi_1} \\
\vdots \\
Y_{\xi_{n_\xi}}
\end{pmatrix}
\]
KO Problem - Predictions

- Predictions are made on a denser set of time instants $n_t^* = 100$.
- For each sample of the posterior of $\mathbf{B}$, $\mathbf{\Sigma}$ and $\mathbf{r}$ we can get a sample of the following:
  - The mean response of the surrogate is identified by the integrating the mean of the joint predictive $\mathbf{M}(\xi^*) \in \mathbb{R}^{n_t^* \times q}$.
  - The variance of the response for output $i = 1, 2, 3$ is the diagonal part of $\mathbf{C}^{ii} \in \mathbb{R}^{n_t^* \times n_t^*}$.
  - The pdf of a given output $i$ at a given time $t$ is found by sampling the joint distribution at different $\xi$'s and using a kernel density estimator on the obtained data.
- In the plots that follow, each line corresponds to different samples of $\mathbf{B}$, $\mathbf{\Sigma}$ and $\mathbf{r}$. 
KO-1 Mean

\[ \delta = 10^{-1}, \text{ 10 observations} \]

\[ \delta = 10^{-3}, \text{ 80 observations} \]

Figure: This is the third column of the matrix \( \mathbf{M} \) (expectation of \( \mathbf{M}(\xi^*) \) with respect to \( \xi^* \)). Different lines correspond to different samples of \( \mathbf{B}, \mathbf{\Sigma} \) and \( \mathbf{r} \). Blue dashed line is the true mean (calculated via MC).
KO-1 Variance

Figure: These are the diagonal elements of $C^{11}$ which correspond to the variance we usually plot. Different lines correspond to different samples of $B, \Sigma$ and $r$. Blue dashed line is the true variance (calculated via MC).
KO-1 Variance

(a) $\delta = 10^{-3}$, 80 observations

(b) $\delta = 10^{-4}$, 80 observations
KO-2 Mean

(c) $\delta = 10^{-1}$, 20 observations

(d) $\delta = 10^{-2}$, 40 observations
KO-2 Variance

(e) $\delta = 10^{-1}$, 20 observations

(f) $\delta = 10^{-2}$, 40 observations
KO-2 Variance

(g) $\delta = 10^{-3}$, 80 observations

(h) $\delta = 10^{-4}$, 260 observations
(i) True and MGP predictions

To find the pdf: 1) Sample $\mathbf{B}$, $\Sigma$ and $\mathbf{r}$. 2) Sample 10,000 $\xi^*$’s. 3) For each $\xi^*$, calculate the mean of the (single input) predictive distribution at $\mathbf{x} = (\xi^*, 10)$. 4) The third element of the mean is an estimate of $y_3(t=10)$. 5) Gather all 10,000 samples and plot a kernel density estimator. 6) Go to 1.
Problem Definition and Input Reduction
Bayesian Approach to UQ: Multi-output GPs
Modeling Correlations with Multi-Output GPs & Examples
Conclusions and Current Research

KO-2 PDF

(k) True and MGP predictions

(l) $\delta = 10^{-2}$, 40 observations
Problem Definition and Input Reduction
Bayesian Approach to UQ: Multi-output GPs
Modeling Correlations with Multi-Output GPs & Examples
Conclusions and Current Research

KO-2 PDF

(m) True and MGP predictions

(n) $\delta = 10^{-3}$, 80 observations

Materials Process Design and Control Laboratory
Cornell University
(o) True and MGP predictions

(p) $\delta = 10^{-4}$, 260 observations
Conclusions

- The Bayesian approach offers
  - Probability measure over possible surrogates (finite number of samples)
  - Active Learning (Experimental Design)

- Non-stationary responses can be modeled by trees of surrogates

- Samples of the surrogate distribution can be used for:
  - Uncertainty Quantification tasks (semi-analytically for uniform inputs)
  - Sensitivity analysis
  - Model calibration

- Easy coupling with HDMR (ANOVA) as a subroutine (high-dimensions)
On-going research

- Capturing non-linear correlations between outputs (corregionalization model)
- Coupling with data-driven high-dimensional stochastic input models
- Investigating feasibility of fully Bayesian trees
- Designing *Best-C++: Bayesian Exploration Statistical Toolbox*: Statistical Toolbox for Exascale Computing (Boost, ScaLAPACK, BLAS, MPI2, LGPL) - to be widely released to DoE and DoD laboratories this year