Dimensionality reduction and surrogate modelling for high-dimensional UQ problems

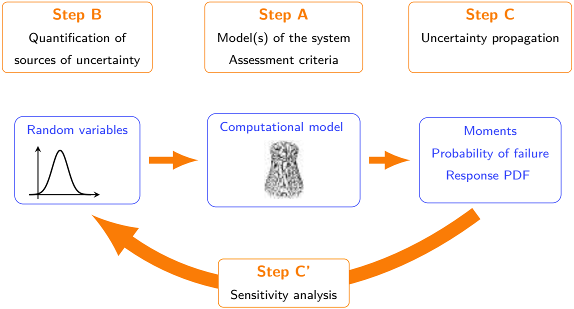
C. Lataniotis, S. Marelli, B. Sudret
Chair of Risk, Safety & Uncertainty Quantification – ETH Zürich

MASCOT NUM 2017, France (Paris), 22.03.2017

Outline

Problem description

UQ framework



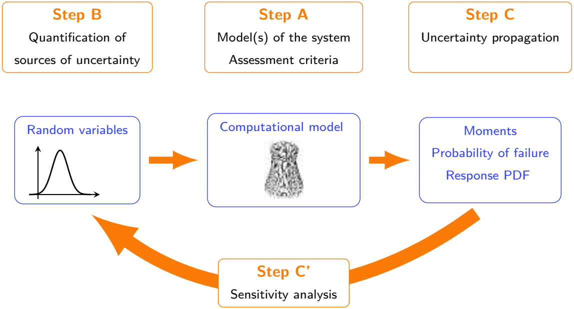
Standard setup

- Prescribed probabilistic model (marginal PDFs and copula)
- Uncertainty propagation by surrogate models (Kriging, PCE)

Sudret, B. (2007). Uncertainty propagation and sensitivity analysis in mechanical models - Contributions to structural reliability and stochastic spectral methods. Habilitation à diriger des recherches, Université Blaise Pascal, Clermont-Ferrand.

Problem description

UQ framework



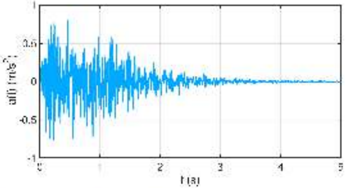
New setup

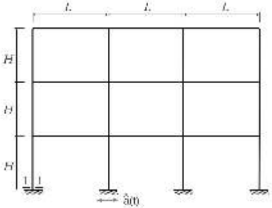
- High input dimensionality, with possible redundancy in the parameters
- Data sets available instead of prescribed PDFs

Goal of the PhD: optimally combine dimensionality reduction with surrogate modelling

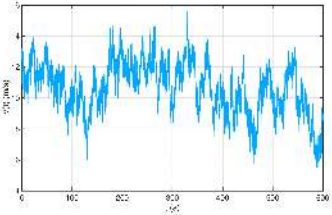
Applications of Interest

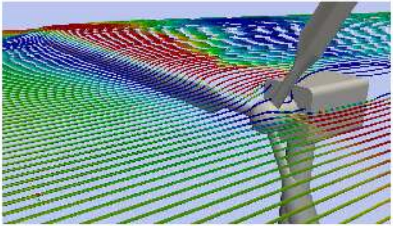
- Earthquake engineering





- Structural health monitoring





Outline

Key ingredients

Surrogate models (SM)

A surrogate model $\tilde{\mathcal{M}}$ is a function s.t.:

$$\mathcal{M}(\mathbf{x}) = \tilde{\mathcal{M}}(\mathbf{x}; \boldsymbol{\Omega}) + \epsilon$$

- Inexpensive to evaluate
- Depends on configuration parameters $\boldsymbol{\Omega}$
- **Non-intrusive**: parameters are inferred from observations $\mathcal{X}, \mathbf{y}$

Examples of surrogate models:

- Kriging
- Polynomial chaos expansions
- Low-rank approximations

Dimensionality Reduction (DR)

A transform $\mathcal{X} \in \mathbb{R}^M \mapsto \mathcal{Z} \in \mathbb{R}^m$ of the form

$$\mathbf{z} = g(\mathbf{x}; \mathbf{w})$$

- Preserves some properties of $\mathbf{x}$ (*e.g.* information content)
- Parameters $\mathbf{w}$ are inferred from observations $\mathcal{X}$

Examples of DR:

- PCA
- Kernel PCA
- Autoencoders

Methodology

- Given observations

$$\mathcal{X} = \left\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}; \mathbf{x}^{(i)} \in \mathbb{R}^M\right\}$$

  and system responses

$$\mathbf{y} = \left\{y_1 = \mathcal{M}(\mathbf{x}^{(1)}), \ldots, y_N = \mathcal{M}(\mathbf{x}^{(N)})\right\}$$

- Instead of computing surrogate $\tilde{\mathcal{M}}(\mathbf{x})$ calculate $\tilde{\mathcal{M}}\left(\mathbf{z}\right)$ were

$$\mathbf{z} = g(\mathbf{x}, \mathbf{w})\,, \, \mathbf{z} \in \mathbb{R}^m\,, \, m < M$$

- Tune DR parameters **w** having the surrogate model performance as objective

Methodology

- Given observations

$$\mathcal{X} = \left\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}; \mathbf{x}^{(i)} \in \mathbb{R}^M\right\}$$

  and system responses

$$\mathbf{y} = \left\{y_1 = \mathcal{M}(\mathbf{x}^{(1)}), \ldots, y_N = \mathcal{M}(\mathbf{x}^{(N)})\right\}$$

- Instead of computing surrogate $\tilde{\mathcal{M}}(\mathbf{x})$ calculate $\tilde{\mathcal{M}}(\mathbf{z})$ were

$$\mathbf{z} = g(\mathbf{x}, \mathbf{w}), \; \mathbf{z} \in \mathbb{R}^m, \; m < M$$

- Tune DR parameters **w** having the surrogate model performance as objective

Treats dim. reduction (DR) and surrogate modelling (SM) as **black boxes**

Methodology

- Given observations

$$\mathcal{X} = \left\{ \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)} : \mathbf{x}^{(i)} \in \mathbb{R}^M \right\}$$

and system responses

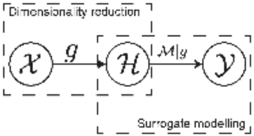
$$\mathbf{y} = \left\{ y_1 = \mathcal{M}(\mathbf{x}^{(1)}), \dots, y_N = \mathcal{M}(\mathbf{x}^{(N)}) \right\}$$

- Instead of computing surrogate $\tilde{\mathcal{M}}(\mathbf{x})$ calculate $\tilde{\mathcal{M}}(\mathbf{z})$ were

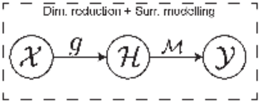
$$\mathbf{z} = g(\mathbf{x}, \mathbf{w}), \; \mathbf{z} \in \mathbb{R}^m, \; m < M$$

- Tune DR parameters $\mathbf{w}$ having the surrogate model performance as objective

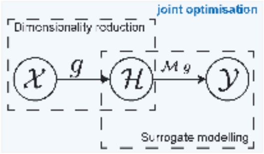
Treats dim. reduction (DR) and surrogate modelling (SM) as **black boxes**



Standard



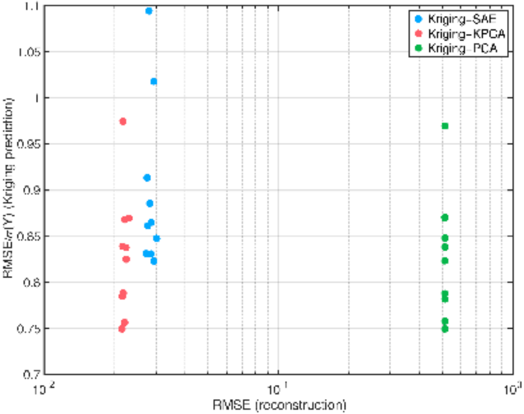
Hinton 2007 (neural network with autoencoder)
Calandra et al. 2016 (GP regression with autoenc.)



DRSM

Hinton, G. E. and Salakhutdinov, R. S. (2006). Reducing the dimensionality of data with neural networks. Science 313.5786, pp. 504–507.
Calandra, R. and Peters J. and Rasmussen C. E. and Deisenroth M.P. (2016). Manifold Gaussian processes for regression. In Neural Networks (IJCNN), 2016 International Joint Conference, pp. 3338-3345. IEEE.

Motivation

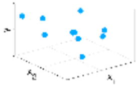


Accuracy of surrogate model not correlated with accuracy of dimensionality reduction

Lataniotis, C. and Marelli, S. and Sudret, B. (2016). Combining feature mapping and Gaussian process modelling in the context of UQ. SIAM Conf. on Uncertainty Quantification (SIAM UQ 2016), Switzerland.

DRSM algorithm

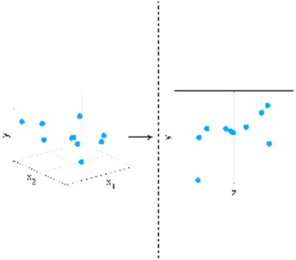
Goal: Find optimal $\mathbf{w}$ *w.r.t.* the surrogate modelling performance of $\tilde{\mathcal{M}}(g(\mathbf{x}, \mathbf{w}))$



- Given quantities: $\mathcal{X}, \mathbf{y}$
- Compute $\mathbf{w}^*$ using objective $J(\mathbf{w})$:
  - Compute $\mathcal{Z} = g(\mathcal{X}, \mathbf{w})$
  - Split $\mathcal{Z}$ into training ($\mathcal{X}_t$) and test set ($\mathcal{X}_{cv}$) with $\mathcal{Z}_t \cup \mathcal{Z}_{cv} = \mathcal{Z}$ (resp. $\mathbf{y}$ to $\mathbf{y}_t, \mathbf{y}_{cv}$)
  - Compute surrogate $\tilde{\mathcal{M}}(\mathbf{z})$ using $\mathcal{Z}_t, \mathbf{y}_t$
  - Evaluate the surrogate response $\tilde{\mathcal{M}}(\mathcal{Z}_{cv})$
  - Compute objective (*e.g.* RMSE):
    $J(\mathbf{w}) = \left(\frac{1}{N_{cv} \operatorname{var}(y_{cv})} \sum_{i=1}^{N_{cv}} \left(y_{cv}^{(i)} - \tilde{\mathcal{M}}\left(\mathbf{z}_{cv}^{(i)}\right) \right)^2 \right)^{1/2}$

DRSM algorithm

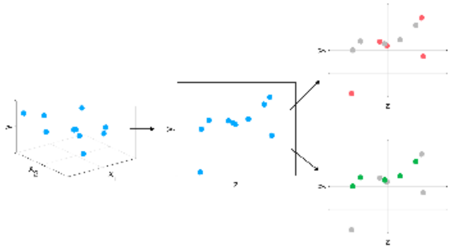
Goal: Find optimal $\mathbf{w}$ *w.r.t.* the surrogate modelling performance of $\tilde{\mathcal{M}}(g(\mathbf{x}, \mathbf{w}))$



- Given quantities: $\mathcal{X}, \mathbf{y}$
- Compute $\mathbf{w}^*$ using objective $J(\mathbf{w})$:
  - Compute $\mathcal{Z} = g(\mathcal{X}, \mathbf{w})$
  - Split $\mathcal{Z}$ into training ($\mathcal{X}_t$) and test set ($\mathcal{X}_{cv}$) with $\mathcal{Z}_t \cup \mathcal{Z}_{cv} = \mathcal{Z}$ (resp. $\mathbf{y}$ to $\mathbf{y}_t, \mathbf{y}_{cv}$)
  - Compute surrogate $\tilde{\mathcal{M}}(\mathbf{z})$ using $\mathcal{Z}_t, \mathbf{y}_t$
  - Evaluate the surrogate response $\tilde{\mathcal{M}}(\mathcal{Z}_{cv})$
  - Compute objective (*e.g.* RMSE):
    $J(\mathbf{w}) = \left(\frac{1}{N_{cv}\,\mathrm{var}(y_{cv})}\sum_{i=1}^{N_{cv}}\left(y_{cv}^{(i)} - \tilde{\mathcal{M}}\left(\mathbf{z}_{cv}^{(i)}\right)\right)^2\right)^{1/2}$

DRSM algorithm

Goal: Find optimal $\mathbf{w}$ *w.r.t.* the surrogate modelling performance of $\tilde{\mathcal{M}}(g(\mathbf{x}, \mathbf{w}))$

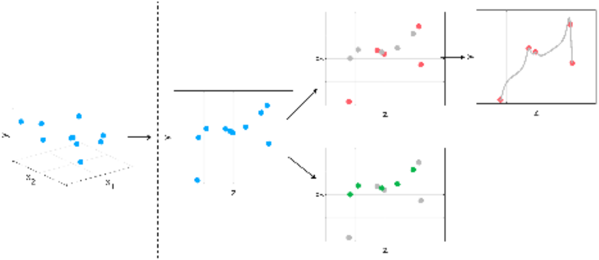


- Given quantities: $\mathcal{X}, \mathbf{y}$
- Compute $\mathbf{w}^*$ using objective $J(\mathbf{w})$:
  - Compute $\mathcal{Z} = g(\mathcal{X}, \mathbf{w})$
  - Split $\mathcal{Z}$ into training ($\mathcal{X}_t$) and test set ($\mathcal{X}_{cv}$) with $\mathcal{Z}_t \cup \mathcal{Z}_{cv} = \mathcal{Z}$ (resp. $\mathbf{y}$ to $\mathbf{y}_t, \mathbf{y}_{cv}$)
  - Compute surrogate $\tilde{\mathcal{M}}(\mathbf{z})$ using $\mathcal{Z}_t, \mathbf{y}_t$
  - Evaluate the surrogate response $\tilde{\mathcal{M}}\left(\mathcal{Z}_{cv}\right)$
  - Compute objective (e.g. RMSE):

$$J(\mathbf{w}) = \left(\frac{1}{N_{cv} \operatorname{var}(y_{cv})} \sum_{i=1}^{N_{cv}} \left(y_{cv}^{(i)} - \tilde{\mathcal{M}}\left(\mathbf{z}_{cv}^{(i)}\right) \right)^2 \right)^{1/2}$$

DRSM algorithm

Goal: Find optimal $\mathbf{w}$ *w.r.t.* the surrogate modelling performance of $\tilde{\mathcal{M}}(g(\mathbf{x}, \mathbf{w}))$

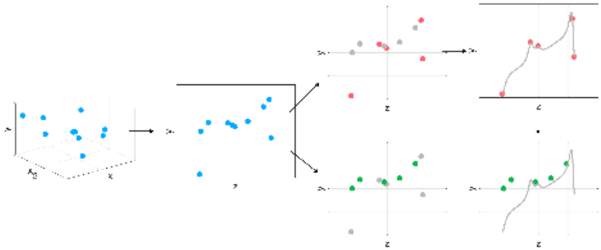


- Given quantities: $\mathcal{X}, \mathbf{y}$
- Compute $\mathbf{w}^*$ using objective $J(\mathbf{w})$:
  - Compute $\mathcal{Z} = g(\mathcal{X}, \mathbf{w})$
  - Split $\mathcal{Z}$ into training ($\mathcal{X}_t$) and test set ($\mathcal{X}_{cv}$) with $\mathcal{Z}_t \cup \mathcal{Z}_{cv} = \mathcal{Z}$ (resp. $\mathbf{y}$ to $\mathbf{y}_t, \mathbf{y}_{cv}$)
  - Compute surrogate $\tilde{\mathcal{M}}(\mathbf{z})$ using $\mathcal{Z}_t, \mathbf{y}_t$
  - Evaluate the surrogate response $\tilde{\mathcal{M}}(\mathcal{Z}_{cv})$
  - Compute objective (e.g. RMSE):

$$J(\mathbf{w}) = \left(\frac{1}{N_{cv}\,\mathrm{var}(y_{cv})}\sum_{i=1}^{N_{cv}}\left(y_{cv}^{(i)} - \tilde{\mathcal{M}}\left(\mathbf{z}_{cv}^{(i)}\right)\right)^2\right)^{1/2}$$

DRSM algorithm

Goal: Find optimal $\mathbf{w}$ *w.r.t.* the surrogate modelling performance of $\tilde{\mathcal{M}}(g(\mathbf{x}, \mathbf{w}))$

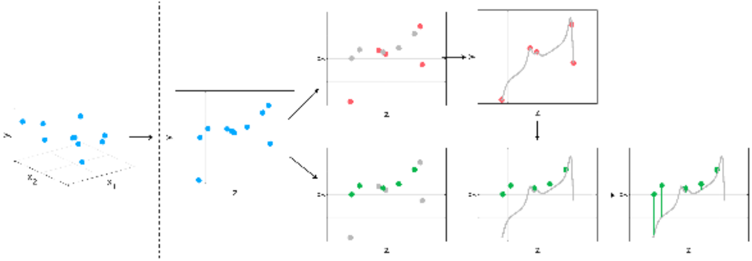


- Given quantities: $\mathcal{X}, \mathbf{y}$
- Compute $\mathbf{w}^*$ using objective $J(\mathbf{w})$:
  - Compute $\mathcal{Z} = g(\mathcal{X}, \mathbf{w})$
  - Split $\mathcal{Z}$ into training ($\mathcal{X}_t$) and test set ($\mathcal{X}_{cv}$) with $\mathcal{Z}_t \cup \mathcal{Z}_{cv} = \mathcal{Z}$ (resp. $\mathbf{y}$ to $\mathbf{y}_t, \mathbf{y}_{cv}$)
  - Compute surrogate $\tilde{\mathcal{M}}(\mathbf{z})$ using $\mathcal{Z}_t, \mathbf{y}_t$
  - Evaluate the surrogate response $\tilde{\mathcal{M}}(\mathcal{Z}_{cv})$
  - Compute objective (e.g. RMSE):

$$J(\mathbf{w}) = \left(\frac{1}{N_{cv} \, \mathrm{var}(y_{cv})} \sum_{i=1}^{N_{cv}} \left(y_{cv}^{(i)} - \tilde{\mathcal{M}}\left(\mathbf{z}_{cv}^{(i)}\right) \right)^2 \right)^{1/2}$$

DRSM algorithm

Goal: Find optimal **w** *w.r.t.* the surrogate modelling performance of $\tilde{\mathcal{M}}(g(\mathbf{x}, \mathbf{w}))$



- Given quantities: $\mathcal{X}, \mathbf{y}$
- Compute $\mathbf{w}^*$ using objective $J(\mathbf{w})$:
  - Compute $\mathcal{Z} = g(\mathcal{X}, \mathbf{w})$
  - Split $\mathcal{Z}$ into training ($\mathcal{X}_t$) and test set ($\mathcal{X}_{cv}$) with $\mathcal{Z}_t \cup \mathcal{Z}_{cv} = \mathcal{Z}$ (resp. $\mathbf{y}$ to $\mathbf{y}_t, \mathbf{y}_{cv}$)
  - Compute surrogate $\tilde{\mathcal{M}}(\mathbf{z})$ using $\mathcal{Z}_t, \mathbf{y}_t$
  - Evaluate the surrogate response $\tilde{\mathcal{M}}\left(\mathcal{Z}_{cv}\right)$
  - Compute objective (*e.g.* RMSE):

$$J(\mathbf{w}) = \left(\frac{1}{N_{cv} \operatorname{var}(y_{cv})} \sum_{i=1}^{N_{cv}} \left(y_{cv}^{(i)} - \tilde{\mathcal{M}}\left(\mathbf{z}_{cv}^{(i)}\right) \right)^2 \right)^{1/2}$$

Summary

DRSM ingredients

- Dimensionality reduction method
- Surrogate modelling method
- Objective function type
- Optimisation method

Remarks

- Only the forward DR step is required
- Simplified SMs favour computational efficiency
- Multiple CV-folds are preferred. Extreme case: *Leave-One-Out error*
- Global optimisation is expected to perform better

Summary

DRSM ingredients

- Dimensionality reduction method : Kernel PCA
- Surrogate modelling method : Kriging
- Objective function type : Leave-One-Out RMSE
- Optimisation method : Genetic algorithm with BFGS refinement

Remarks

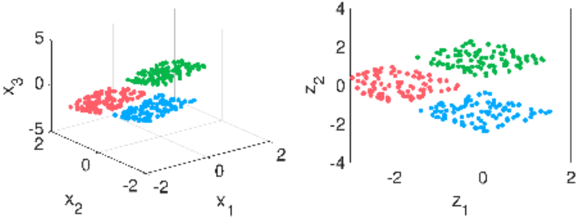
- Only the forward DR step is required
- Simplified SMs favour computational efficiency
- Multiple CV-folds are preferred. Extreme case: *Leave-One-Out error*
- Global optimisation is expected to perform better

One particular DRSM setup is explored next

Outline

Kernel PCA

Recall PCA:



- Calculate the covariance matrix $C = \mathrm{cov}(\mathcal{X})$
- Compute the eigen-decomposition of C:

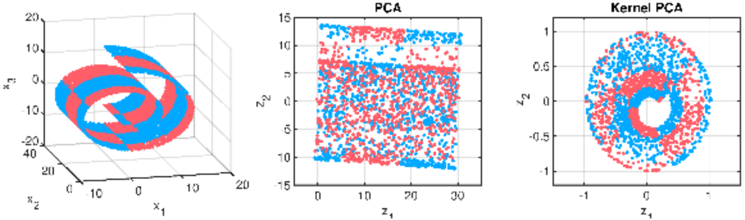
$$Cv = \lambda v$$

- Keep V, an $M \times m$ collection of m eigenvectors v with largest eigenvalues λ
- Obtain $\mathcal{Z}$ by projecting $\mathcal{X}$ onto V:

$$\mathcal{Z} = \mathcal{X} V$$

Kernel PCA

Kernel PCA (KPCA) is a non-linear extension of PCA



- First maps the input to a high dimensional space (*feature space*):

$$\mathbf{x} \in \mathcal{R}^M \rightarrow \Phi(\mathbf{x}) \in \mathcal{H}$$

  then PCA is performed in this space
- Infinite dimensional feature spaces are a common choice
- Calculates $\mathcal{Z}$ as projections onto the first m eigenvectors of $C_{\mathcal{H}} = \text{cov}\left[\Phi(\mathcal{X})\right]$

Problem: In general, $C_{\mathcal{H}}$ and its eigenvectors cannot be computed

Kernel PCA

How to solve the eigen-decomposition problem in $\mathcal{H}$

- Consider Φ that:

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$$

- Although $\boldsymbol{v}_i \rightarrow$ infinite dimensional, $N \rightarrow$ finite, thus:

$$\boldsymbol{v}_i = \sum_{j=1}^{N} \alpha_j^{(i)} \Phi(\mathbf{x}_j)\,,\; i = 1, \ldots, m$$

  where $\boldsymbol{\alpha}^{(i)}$: the **expansion coefficients** of $\boldsymbol{v}_i$

- The eigen-decomposition problem:

$$\lambda \boldsymbol{v} = \boldsymbol{C}_{\mathcal{H}} \boldsymbol{v}$$

  can be cast as:

$$\lambda \boldsymbol{\alpha} = \frac{1}{N} \boldsymbol{K} \boldsymbol{\alpha}$$

  where $\boldsymbol{K}$ is the $N \times N$ kernel matrix with elements $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$

By solving this eigen-decomposition the coefficients $\boldsymbol{\alpha}$ are obtained (not $\boldsymbol{v}$!)

Kernel PCA

How to calculate $\mathcal{Z}$

The k-th component of the i-th sample of $\mathcal{Z}$, denoted by $z_i^{(k)}$, is the projection of $\Phi(\mathbf{x}_i)$ onto $\mathbf{v}_k$:

$$
\begin{aligned}
z_i^{(k)} &= \Phi(\mathbf{x}_i)^T \mathbf{v}_k \\
&= \sum_{j=1}^{N} \alpha_j^{(k)} \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x}_j) \\
&= \sum_{j=1}^{N} \alpha_j^{(k)} \kappa(\mathbf{x}_i, \mathbf{x}_j)
\end{aligned}
$$

Remarks

- Some details are omitted here (*e.g.* centering the kernel)
- In contrast to PCA going from $\mathcal{Z}$ back to $\mathcal{X}$ is non-trivial (**pre-image problem**)
- The **ARD** kernel will be used:

$$
\kappa(\mathbf{x}_i, \mathbf{x}_j; \mathbf{w}) = \exp\left(-\frac{1}{2} \sum_{k=1}^{M} \frac{1}{w_k^2} \left(x_i^{(k)} - x_j^{(k)}\right)^2\right)
$$

Schölkopf, B., Smola, A. J., and Müller, K.-R., (1999). Kernel Principal Component Analysis, Advances in Kernel Methods–Support Vector Learning, MIT Press, Cambridge, MA pp. 327-352.
Ham, J., Lee, D. D., Mika, S., and Schölkopf, B. (2004). A kernel view of the dimensionality reduction of manifolds. In Proceedings of the twenty-first international conference on Machine learning (p. 47), ACM.

Kriging reminder

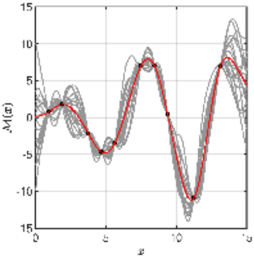
Kriging metamodelling in a nutshell

Universal Kriging:

$$\tilde{\mathcal{M}}(\boldsymbol{x}) = \boldsymbol{\beta}^T \boldsymbol{f}(\boldsymbol{x}) + \sigma^2 Z(\boldsymbol{x}, R(\boldsymbol{x}, \boldsymbol{x}_{\text{ED}}; \hat{\boldsymbol{\theta}}))$$

$$\hat{\boldsymbol{\theta}} = \arg\min J(\hat{\boldsymbol{\theta}})$$

Objective function varies depending on the estimation method (*e.g.* **maximum likelihood**, **cross-validation**, *etc.*)



Using a Kriging metamodel as a surrogate

A Kriging metamodel forms a group of curves characterized by their:

- Mean (model output):

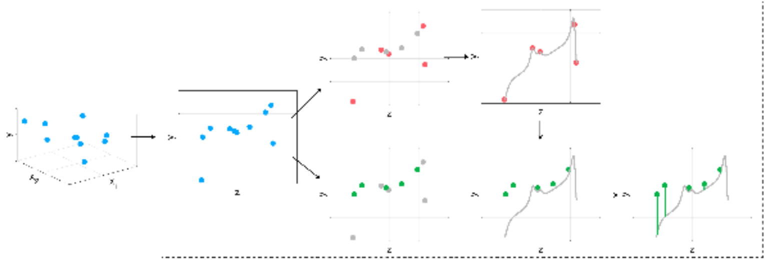
$$\mu_{\hat{Y}}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x})^T \boldsymbol{\beta} + \boldsymbol{r}(\boldsymbol{x})^T \boldsymbol{R}^{-1} (\boldsymbol{y} - \boldsymbol{F}\boldsymbol{\beta})$$

- Variance (local error estimate):

$$\sigma^2_{\hat{Y}}(\boldsymbol{x}) = \sigma^2 \left(1 - \boldsymbol{r}^T(\boldsymbol{x})\boldsymbol{R}^{-1}\boldsymbol{r}(\boldsymbol{x}) + \boldsymbol{u}^T(\boldsymbol{x})(\boldsymbol{F}^T\boldsymbol{R}^{-1}\boldsymbol{F})^{-1}\boldsymbol{u}(\boldsymbol{x})\right)$$

DRSM objective: Kriging Leave-One-Out error

Recall DRSM algorithm

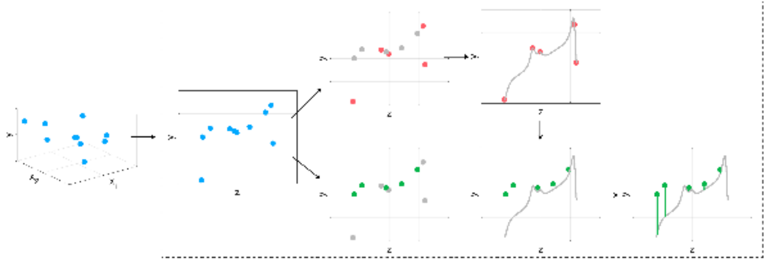


The Leave-One-Out RMSE error reads:

$$J(\mathbf{w}) = \left(\frac{1}{N \mathrm{Var}\left[\mathbf{y}\right]} \sum_{i=1}^{N} \left(\mathcal{M}(z_i) - \mu_{\hat{Y},(-i)}(z_i) \right)^2 \right)^{1/2}$$

DRSM objective: Kriging Leave-One-Out error

Recall DRSM algorithm



The Leave-One-Out RMSE error reads:

$$J(\mathbf{w}) = \left(\frac{1}{N \mathrm{Var}\left[\mathbf{y}\right]} \sum_{i=1}^{N} \left(\mathcal{M}(z_i) - \mu_{\hat{Y},(-i)}(z_i) \right)^2 \right)^{1/2}$$

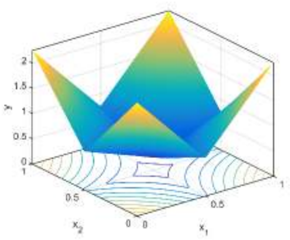
For Kriging surrogates $\mu_{\hat{Y},(-i)}(z_i)$ can be computed efficiently

Dubrule, O. (1983). Cross validation of kriging in a unique neighborhood. Mathematical Geology 15.6, pp. 687-699.

Outline

Sobol function

Model description



$$\mathcal{M}(\mathbf{x}) = \prod_{i=1}^{20} \frac{|4x_i - 2| + \alpha_i}{1 + a_i}, \; x_i \in [0,1]$$

where

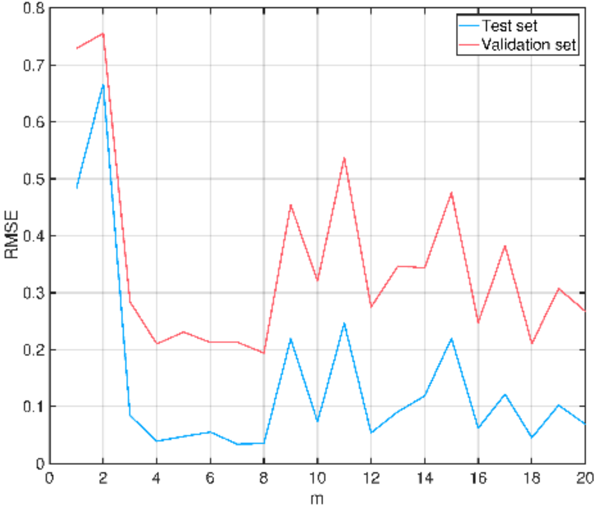
$$\boldsymbol{\alpha} = [1, 2, 5, 10, 20, 50, 100, 500, , \ldots, 500]^T$$

DRSM parameters

KPCA	
Kernel:	ARD
Kriging	
Correlation function:	Matérn 5/2 (isotropic)
Trend type:	linear
Optimisation:	GA + BFGS
Optimisation	
Method:	GA+ BFGS
Population size:	60
Max. Iterations:	100
Conv. tolerance:	10^{-3}

Training: 400 random samples
Validation: 1000 random samples

Sobol function: m vs surrogate model performance



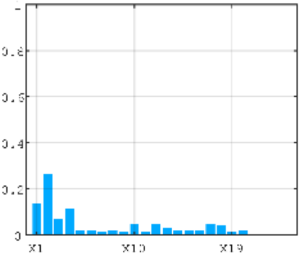
Selected $m = 1$

Sobol function: Optimal KPCA parameters ($m = 4$)

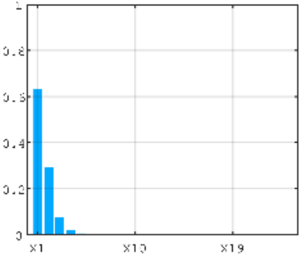
ARD Kernel reminder:

$$\kappa(\mathbf{x}_i, \mathbf{x}_j; \mathbf{w}) = \exp\left(-\frac{1}{2}\sum_{k=1}^{M}\frac{1}{w_k^2}\left(x_i^{(k)} - x_j^{(k)}\right)^2\right)$$

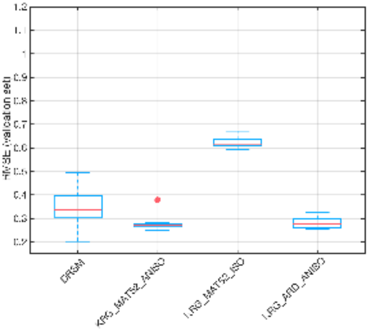
$1/w_i$ (ARD kernel)

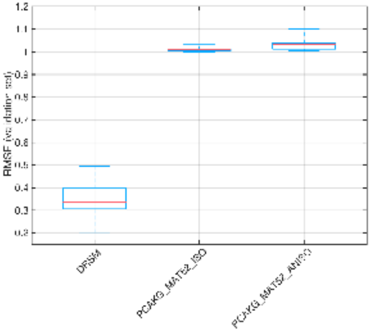


Global sensitivity analysis



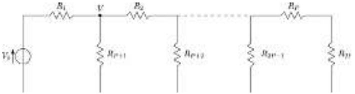
DRSM vs plain Kriging





Electrical resistor networks

Dataset description



- Courtesy of **SANDIA National Labs**
- Network comprised of 80 uncertain resistors
- Output of interest is voltage at V
- Effect of resistors to V decays with distance

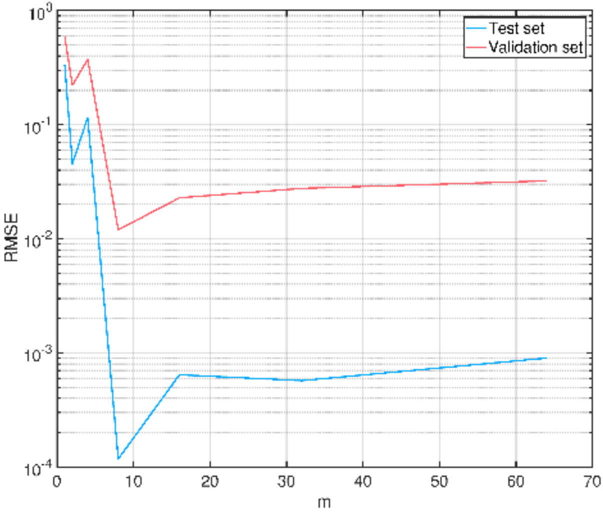
DRSM parameters

KPCA	
Kernel:	ARD
Kriging	
Correlation function:	Matérn 5/2 (isotropic)
Trend type:	linear
Optimisation:	GA + BFGS
Optimisation	
Method:	GA + BFGS
Population size:	150
Max. Iterations:	100
Conv. tolerance:	10^{-3}

Training: 500 samples
Validation: 500 samples

Jakeman, D., Michael S. E., and Khachik S. (2015). Enhancing l1-minimization estimates of polynomial chaos expansions using basis selection. Journal of Computational Physics 289, pp. 18-34.

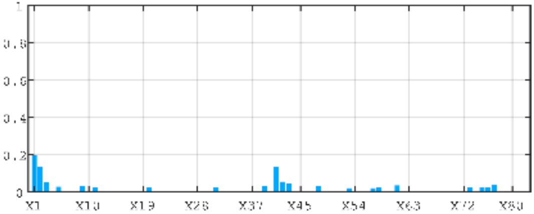
Resistor networks: m vs surrogate model performance



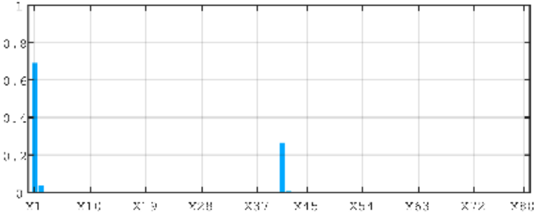
Selected $m = 8$

Resistor networks: Optimal KPCA parameters ($m = 8$)

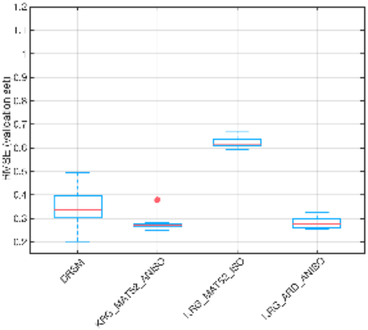
$1/w_i$
(ARD kernel)

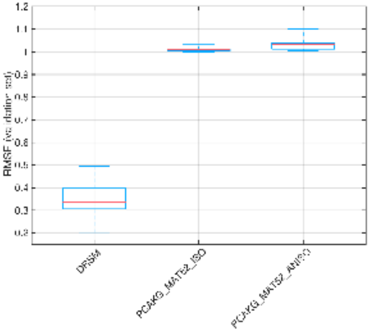


Global sensitivity analysis



Resistor networks: DRSM vs plain Kriging





Outline

Summary & Outlook

Summary

- A new framework for SM with high-dimensional inputs is proposed
- It is currently at validation stage
- It shows superior performance compared to traditional approaches (disjoint DR and SM)
- An interesting by-product of DRSM with KPCA using ARD kernel: pseudo-sensitivity indices

Outlook

- Explore different setups, *e.g.* PCE instead of Kriging and Autoencoders instead of KPCA
- Test performance of various DRSM set-ups on problems of interest (*e.g.* time-series inputs)

Summary & Outlook

Summary

- A new framework for SM with high-dimensional inputs is proposed
- It is currently at validation stage
- It shows superior performance compared to traditional approaches (disjoint DR and SM)
- An interesting by-product of DRSM with KPCA using ARD kernel: pseudo-sensitivity indices

Outlook

- Explore different setups, *e.g.* PCE instead of Kriging and Autoencoders instead of KPCA
- Test performance of various DRSM set-ups on problems of interest (*e.g.* time-series inputs)

Thank you very much for your attention!

UQLab

The Framework for Uncertainty Quantification

www.uqlab.com

Chair of Risk, Safety & Uncertainty Quantification

http://www.rsuq.ethz.ch