Probabilistic Methods for Forward and Inverse Uncertainty Quantification

Khachik Sargsyan

Sandia National Laboratories

Livermore, CA, USA

Zoominar

BYU/UtahState Math

October 12, 2020

Acknowledgements

H. Najm, B. Debusschere, C. Safta, T. Casey, J. Oreluk — SNL, CA
M. Eldred, G. Geraci — SNL, NM

X. Huan — UMich
R. Ghanem — USC
Y. Marzouk, C. Feng — MIT
D. Ricciuto, P. Thornton – ORNL
J. Bender – LLNL

This work was supported by:

- DOE, Advanced Scientific Computing Research (ASCR), SciDAC
- DOE, Basic Energy Sciences (BES)
- DOE, Biological and Environmental Research (BER)
- DOD, DARPA Enabling Quantification of Uncertainty in Physical Systems (EQUiPS) program

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Overview of the talk

- UQ in Computational Science
- Forward UQ with Polynomial Chaos
  - Uncertainty propagation
  - Model surrogate
  - Global sensitivity analysis
  - High-dimensionality challenge
- Inverse UQ with Bayesian inference
  - Markov chain Monte Carlo with model surrogate
  - Embedded model structural error
- Applications
  - Chemistry, Climate, Large Eddy Simulation
- Summary

Outline

The Case for Uncertainty Quantification

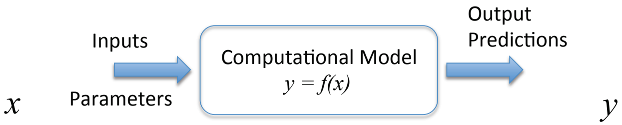
Uncertainty Sources

- Model parameters
- Initial/boundary conditions
- Model geometry/structure
- Lack of knowledge
- Data noise
- Intrinsic stochasticity
- Numerical errors, too

UQ needed for...

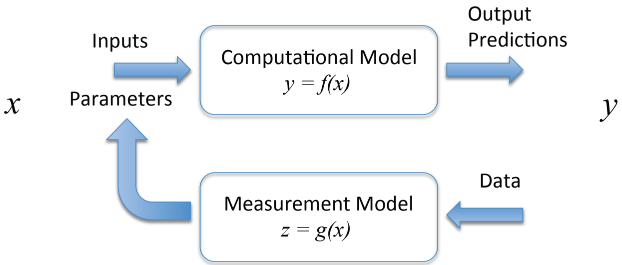
- Model predictions
- Model validation and comparison
- Confidence assessment
- Reliability analysis
- Dimensionality reduction
- Optimal design
- Decision support
- (Noisy) data assimilation

Uncertainty Quantification and Computational Science



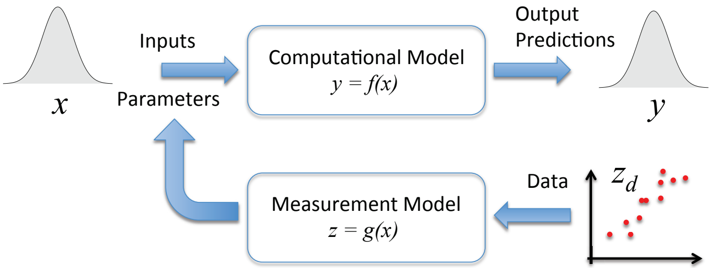
Forward problem

Uncertainty Quantification and Computational Science



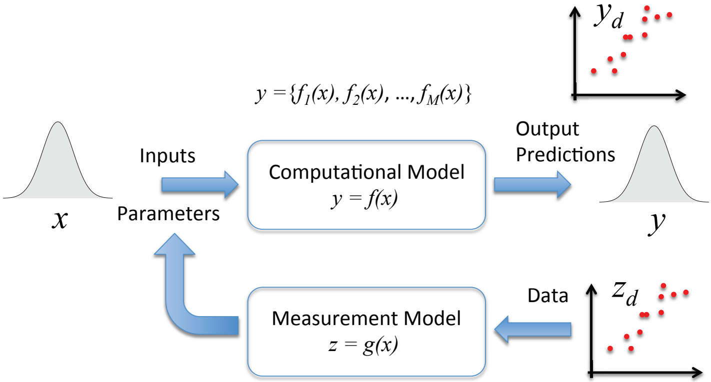
Inverse & Forward problems

Uncertainty Quantification and Computational Science



Inverse & Forward UQ

Uncertainty Quantification and Computational Science



Inverse & Forward UQ

Model validation & comparison, Hypothesis testing

UQ components

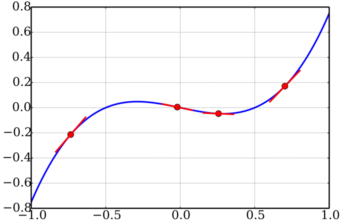
- Locate all sources of (manageable) uncertainties
- Parameter selection/estimation
  - Auxiliary data collection, submodel fitting/regression
  - Expert opinion, physical bounds, maximum entropy
- **Forward** propagation of uncertainties
  - Local SA (deterministic, error propagation)
  - Interval math, evidence theory
  - Global SA (stochastic, variance-based decomposition)
- Calibration/tuning given observations or a higher-fidelity model (**inverse** UQ)
- Model (in)validation
  - Compare model prediction with uncertainties versus data on some QoI
  - Model comparison (Bayes Factors, Model Plausibility)
  - Representation, quantification and propagation of model structural error (no model is perfect)

Outline

Forward UQ

- Local sensitivity analysis and error propagation

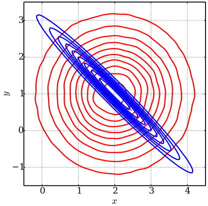
$$\Delta y = \left.\frac{\mathrm{d}f}{\mathrm{d}x}\right|_{x_0} \Delta x$$

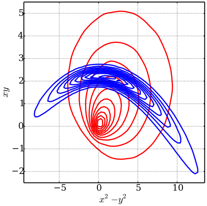


This is ok for:

- small uncertainty
- low degree of non-linearity

- Non-probabilistic methods
  - Evidence theory
  - Fuzzy logic
  - Interval math
  - Misses correlations





- Probabilistic methods – our focus

Polynomial Chaos – functional representation for RVs

- First introduced by Wiener, 1938
- Revitalized by Ghanem and Spanos, 1991
- Convergent series if U has finite variance
- Selection of order p is a modeling choice
- Describes a r.v. U with a vector of *PC modes* $(u_0, u_1, \ldots, u_p)$

$$U \simeq \sum_{k=0}^{p} u_k \psi_k(\xi)$$

- Standard r.v. ξ, standard orthogonal polynomials $\psi_k(\xi)$, *i.e.*

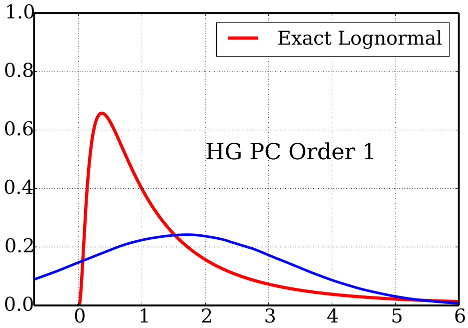
$$\int \psi_i(\xi)\psi_j(\xi)\pi_\xi(\xi)d\xi = \delta_{ij}||\psi_i||^2$$

PC Type	Domain	Density $\pi_\xi(\xi)$	Polynomial	Free parameters
Gauss-Hermite	$(-\infty, +\infty)$	$\frac{1}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}}$	Hermite	none
Legendre-Uniform	$[-1, 1]$	$\frac{1}{2}$	Legendre	none
Gamma-Laguerre	$[0, +\infty)$	$\frac{\xi^\alpha e^{-\xi}}{\Gamma(\alpha+1)}$	Laguerre	$\alpha > -1$
Beta-Jacobi	$[-1, 1]$	$\frac{(1+\xi)^\alpha (1-\xi)^\beta}{2^{\alpha+\beta+1} B(\alpha+1,\beta+1)}$	Jacobi	$\alpha > -1, \beta > -1$

[Wiener, 1938; Ghanem & Spanos, 1991; Xiu & Karniadakis, 2002; Le Maître & Knio, 2010]

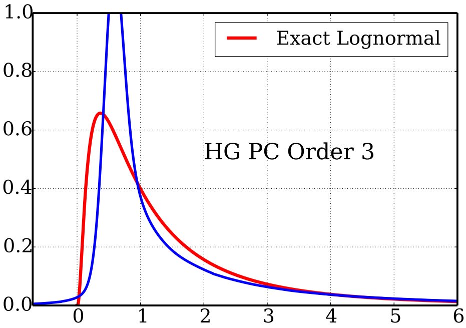
Construction of 1D PC $U \simeq \sum_{k=0}^{p} u_k \psi_k(\xi)$

- Orthogonal projection: $u_k = \frac{1}{||\psi_k||^2} \langle U \psi_k \rangle$
- Need to compute integral $\langle U \psi_k \rangle = \int U(?) \psi_k(\xi) \pi_\xi(\xi) d\xi$
- Need a map $U \leftrightarrow \xi$
- If lucky, there is an explicit formula, e.g. lognormal $U = e^{\xi}$



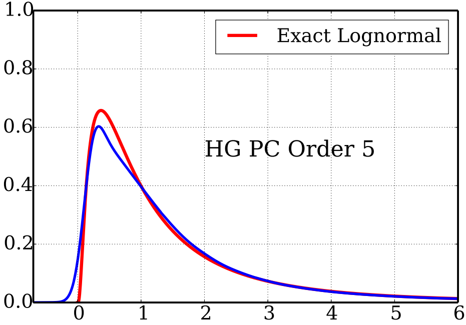
Construction of 1D PC $U \simeq \sum_{k=0}^{p} u_k \psi_k(\xi)$

- Orthogonal projection: $u_k = \frac{1}{||\psi_k||^2} \langle U \psi_k \rangle$
- Need to compute integral $\langle U \psi_k \rangle = \int U(?) \psi_k(\xi) \pi_\xi(\xi) d\xi$
- Need a map $U \leftrightarrow \xi$
- If lucky, there is an explicit formula, e.g. lognormal $U = e^{\xi}$



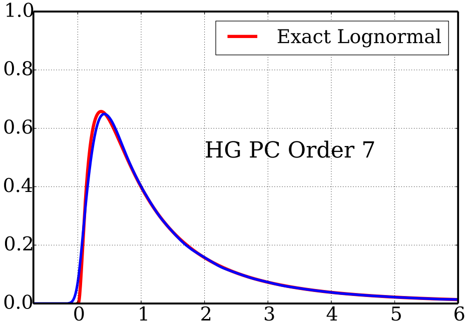
Construction of 1D PC $U \simeq \sum_{k=0}^{p} u_k \psi_k(\xi)$

- Orthogonal projection: $u_k = \frac{1}{||\psi_k||^2}\langle U\psi_k\rangle$
- Need to compute integral $\langle U\psi_k\rangle = \int U(?)\psi_k(\xi)\pi_\xi(\xi)d\xi$
- Need a map $U \leftrightarrow \xi$
- If lucky, there is an explicit formula, e.g. lognormal $U = e^{\xi}$



Construction of 1D PC $U \simeq \sum_{k=0}^{p} u_k \psi_k(\xi)$

- Orthogonal projection: $u_k = \frac{1}{||\psi_k||^2}\langle U\psi_k\rangle$
- Need to compute integral $\langle U\psi_k\rangle = \int U(?)\psi_k(\xi)\pi_\xi(\xi)d\xi$
- Need a map $U \leftrightarrow \xi$
- If lucky, there is an explicit formula, e.g. lognormal $U = e^{\xi}$



Construction of 1D PC $U \simeq \sum_{k=0}^{p} u_k \psi_k(\xi)$

- Orthogonal projection: $u_k = \frac{1}{||\psi_k||^2}\langle U\psi_k\rangle$
- Need to compute integral $\langle U\psi_k\rangle = \int U(?)\psi_k(\xi)\pi_\xi(\xi)d\xi$
- Need a map $U \leftrightarrow \xi$

- CDF transform helps:
  - $U = F_U^{-1}(\frac{\xi+1}{2})$ if ξ is Uniform, Legendre-Uniform PC
  - $U = F_U^{-1}(\Phi(\xi))$ if ξ is Normal, Gauss-Hermite PC

where $F_U(\cdot)$ is the Cumulative Distribution Function (CDF) of U.

[and $\Phi(\cdot)$ is CDF for standard normal]

Essential use of PC in UQ $U \simeq \sum_{k=0}^{K} u_k \Psi_k(\boldsymbol{\xi})$

Strategy:

- Represent model parameters/solution as random variables
- Construct PC for uncertain parameters
- Evaluate PC for model outputs

Advantages:

- Computational efficiency
- Utility
  - Moments: $\mathbb{E}[u] = u_0, \;\; \mathbb{V}[u] = \sum_{k=1}^{K} u_k^2 ||\Psi_k||^2, \;\; \dots$
  - Global Sensitivities – fractional variances, Sobol' indices
  - Uncertainty propagation
  - Surrogate for forward model

Requirements:

- Finite variances (not a handicap in practice)
- Smooth forward functions

PC features: uncertainty propagation

$$U \simeq \sum_{k=0}^{K} u_k \Psi_k(\boldsymbol{\xi})$$

$$Z = f(U) \simeq \sum_{k=0}^{K} c_k \Psi_k(\boldsymbol{\xi})$$

- Basic task: given PC for inputs, find PC for outputs.
- Input-output map can also be defined implicitly, via governing equations $G(Z, U) = 0$.
- Two approaches
  - Intrusive: project governing equations
    - Results in set of equations for the PC modes
    - Requires redesign of computer code
    - PCEs for all uncertain variables in system
  - Non-intrusive: project outputs of interest
    - Sampling to evaluate projection operator
    - Can use existing code as black box
    - Only computes PCEs for quantities of interest

PC surrogate construction

- Build/presume PC for input parameter U

$$U(\boldsymbol{\xi}) = \sum_{k=0}^{K} u_k \Psi_k(\boldsymbol{\xi})$$

with respect to multivariate standard polynomials.

PC surrogate construction

- Build/presume PC for input parameter U

$$U(\boldsymbol{\xi}) = \sum_{k=0}^{K} u_k \Psi_k(\boldsymbol{\xi})$$

with respect to multivariate standard polynomials.

- Input parameters are represented via their cumulative distribution function (CDF) $F(\cdot)$, such that, with $\xi_i \sim$ Uniform$[-1, 1]$

$$U_i = F_{U_i}^{-1}\left(\frac{\xi_i + 1}{2}\right), \qquad \text{for } i = 1, 2, \ldots, d.$$

PC surrogate construction

- Build/presume PC for input parameter U

$$U(\boldsymbol{\xi}) = \sum_{k=0}^{K} u_k \Psi_k(\boldsymbol{\xi})$$

  with respect to multivariate standard polynomials.

- If input parameters are uniform $U_i \sim \text{Uniform}[a_i, b_i]$, then

$$U_i = \frac{a_i + b_i}{2} + \frac{b_i - a_i}{2}\, \xi_i.$$

PC surrogate construction

- Build/presume PC for input parameter U

$$U(\boldsymbol{\xi}) = \sum_{k=0}^{K} u_k \Psi_k(\boldsymbol{\xi})$$

with respect to multivariate standard polynomials.

- Forward function $f(\cdot)$, output Z

$$Z = f(U(\boldsymbol{\xi})) \qquad Z = \sum_{k=0}^{K} c_k \Psi_k(\boldsymbol{\xi})$$

- Global sensitivity information for free
  - Sobol indices, variance-based decomposition.

PC features: moment extraction

$$Z \simeq \sum_{k=0}^{K} z_k \Psi_k(\boldsymbol{\xi})$$

- Expectation: $\langle Z \rangle = z_0$
- Variance σ^2

$$\begin{aligned}
\sigma^2 &= \left\langle (Z - \langle Z \rangle)^2 \right\rangle = \left\langle (\sum_{k=1}^{K} z_k \Psi_k(\boldsymbol{\xi}))^2 \right\rangle \\
&= \left\langle \sum_{k=1}^{K} \sum_{j=1}^{K} z_j z_k \Psi_j(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}) \right\rangle \\
&= \sum_{k=1}^{K} \sum_{j=1}^{K} z_j z_k \left\langle \Psi_j(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}) \right\rangle = \sum_{k=1}^{K} z_k^2 ||\Psi_k||^2
\end{aligned}$$

PC features: Global Sensitivity Analysis $Z(\boldsymbol{\xi}) \simeq \sum_{k=0}^{K} z_k \Psi_k(\boldsymbol{\xi})$

- Main effect sensitivity indices

$$S_i = \frac{Var[\mathbb{E}(Z(\boldsymbol{\xi}|\xi_i)]}{Var[Z(\boldsymbol{\xi})]} = \frac{\sum_{k\in\mathbb{I}_i} z_k^2 ||\Psi_k||^2}{\sum_{k>0} z_k^2 ||\Psi_k||^2}$$

  - $\mathbb{I}_i$ is the set of bases with only ξ_i involved
  - S_i is the uncertainty contribution that is due to i-th parameter only

- Total effect sensitivity indices

$$T_i = 1 - \frac{Var[\mathbb{E}(Z(\boldsymbol{\xi}|\xi_{-i})]}{Var[Z(\boldsymbol{\xi})]} = \frac{\sum_{k\in\mathbb{I}_i^T} z_k^2 ||\Psi_k||^2}{\sum_{k>0} z_k^2 ||\Psi_k||^2}$$

$\mathbb{I}_i^T$ is the set of bases with ξ_i involved, including all its interactions.

PC features: Global Sensitivity Analysis $Z(\boldsymbol{\xi}) \simeq \sum_{k=0}^{K} z_k \Psi_k(\boldsymbol{\xi})$

- Main effect sensitivity indices

$$S_i = \frac{Var[\mathbb{E}(Z(\boldsymbol{\xi}|\xi_i)]}{Var[Z(\boldsymbol{\xi})]} = \frac{\sum_{k\in\mathbb{I}_i} z_k^2||\Psi_k||^2}{\sum_{k>0} z_k^2||\Psi_k||^2}$$

  - $\mathbb{I}_i$ is the set of bases with only ξ_i involved
  - S_i is the uncertainty contribution that is due to i-th parameter only

- Joint sensitivity indices

$$S_{ij} = \frac{Var[\mathbb{E}(Z(\boldsymbol{\xi}|\xi_i, \xi_j)]}{Var[Z(\boldsymbol{\xi})]} - S_i - S_j = \frac{\sum_{k\in\mathbb{I}_{ij}} z_k^2||\Psi_k||^2}{\sum_{k>0} z_k^2||\Psi_k||^2}$$

  - $\mathbb{I}_{ij}$ is the set of bases with only ξ_i and ξ_j involved
  - S_{ij} is the uncertainty contribution that is due to (i, j) parameter pair

Alternative methods to obtain PC coefficients

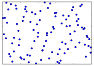
$$Z = f(U(\boldsymbol{\xi})) \simeq \sum_{k=0}^{K} z_k \Psi_k(\boldsymbol{\xi})$$

- Projection $\quad z_k = \frac{\langle f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\rangle}{||\Psi_k||^2}$

  The integral $\langle f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\rangle = \int f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\pi_{\boldsymbol{\xi}}(\boldsymbol{\xi})d\boldsymbol{\xi}$ is estimated by...

  - Monte-Carlo

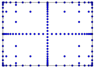
$$\frac{1}{N}\sum_{j=1}^{N} f(\boldsymbol{\xi}_j)\Psi_k(\boldsymbol{\xi}_j)$$



many(!) random samples

  - Quadrature

$$\sum_{j=1}^{Q} f(\boldsymbol{\xi}_j)\Psi_k(\boldsymbol{\xi}_j)w_j$$



samples at quadrature

Alternative methods to obtain PC coefficients

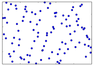
$$Z = f(U(\boldsymbol{\xi})) \simeq \sum_{k=0}^{K} z_k \Psi_k(\boldsymbol{\xi})$$

- Projection $\qquad z_k = \frac{\langle f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\rangle}{||\Psi_k||^2}$

  The integral $\langle f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\rangle = \int f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\pi_{\boldsymbol{\xi}}(\boldsymbol{\xi})d\boldsymbol{\xi}$ is estimated by...

  - Monte-Carlo

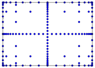
$$\frac{1}{N}\sum_{j=1}^{N} f(\boldsymbol{\xi}_j)\Psi_k(\boldsymbol{\xi}_j)$$



many(!) random samples

  - Quadrature

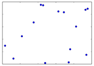
$$\sum_{j=1}^{Q} f(\boldsymbol{\xi}_j)\Psi_k(\boldsymbol{\xi}_j)w_j$$



samples at quadrature

- Bayesian regression

$$P(z_k|f(\boldsymbol{\xi}_j)) \propto P(f(\boldsymbol{\xi}_j)|z_k)P(z_k)$$



any (number of) samples

Alternative methods to obtain PC coefficients

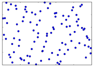
$$Z = f(U(\boldsymbol{\xi})) \simeq \sum_{k=0}^{K} z_k \Psi_k(\boldsymbol{\xi})$$

- Projection $\quad z_k = \frac{\langle f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\rangle}{||\Psi_k||^2}$

  The integral $\langle f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\rangle = \int f(\boldsymbol{\xi})\Psi_k(\boldsymbol{\xi})\pi_{\boldsymbol{\xi}}(\boldsymbol{\xi})d\boldsymbol{\xi}$ is estimated by...

  - Monte-Carlo

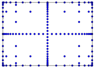
$$\frac{1}{N}\sum_{j=1}^{N} f(\boldsymbol{\xi}_j)\Psi_k(\boldsymbol{\xi}_j)$$



many(!) random samples

  - Quadrature

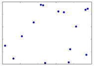
$$\sum_{j=1}^{Q} f(\boldsymbol{\xi}_j)\Psi_k(\boldsymbol{\xi}_j)w_j$$



samples at quadrature

- Bayesian regression

$$\underbrace{P(\boldsymbol{z}|\mathcal{D})}_{\text{Posterior}} \propto \underbrace{P(\mathcal{D}|\boldsymbol{z})}_{\text{Likelihood}}\,\underbrace{P(\boldsymbol{z})}_{\text{Prior}}$$

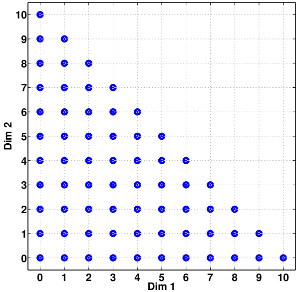


any (number of) samples

Bayesian inference of PC surrogate: high-d, low-data regime

$$Z = f(\boldsymbol{\xi}) \approx \sum_{k=0}^{K} c_k \Psi_k(\boldsymbol{\xi})$$

$$\Psi_k(\xi_1, \xi_2, ..., \xi_d) = \psi_{k_1}(\xi_1)\psi_{k_2}(\xi_2) \cdots \psi_{k_d}(\xi_d)$$

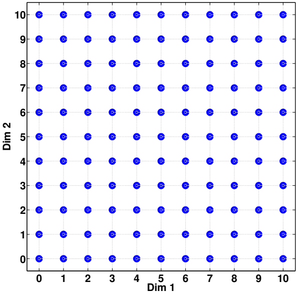


- Issues:
  - how to properly choose the basis set?
  - need to work in underdetermined regime $N < K$: fewer data than bases (d.o.f.)
- Discover the underlying low-d structure in the model
  - sparse regression

Bayesian inference of PC surrogate: high-d, low-data regime

$$Z = f(\boldsymbol{\xi}) \approx \sum_{k=0}^{K} c_k \Psi_k(\boldsymbol{\xi})$$

$$\Psi_k(\xi_1, \xi_2, ..., \xi_d) = \psi_{k_1}(\xi_1)\psi_{k_2}(\xi_2)\cdots\psi_{k_d}(\xi_d)$$



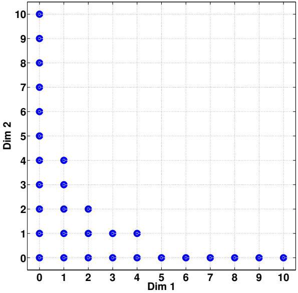
- Issues:
  - how to properly choose the basis set?
  - need to work in underdetermined regime $N < K$: fewer data than bases (d.o.f.)

- Discover the underlying low-d structure in the model
  - sparse regression

Bayesian inference of PC surrogate: high-d, low-data regime

$$Z = f(\boldsymbol{\xi}) \approx \sum_{k=0}^{K} c_k \Psi_k(\boldsymbol{\xi})$$

$$\Psi_k(\xi_1, \xi_2, ..., \xi_d) = \psi_{k_1}(\xi_1)\psi_{k_2}(\xi_2)\cdots\psi_{k_d}(\xi_d)$$

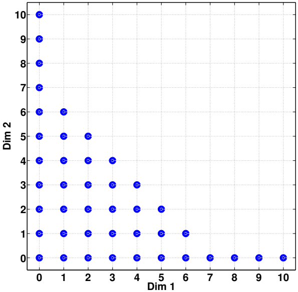


- Issues:
  - how to properly choose the basis set?
  - need to work in underdetermined regime $N < K$: fewer data than bases (d.o.f.)
- Discover the underlying low-d structure in the model
  - sparse regression

Bayesian inference of PC surrogate: high-d, low-data regime

$$Z = f(\boldsymbol{\xi}) \approx \sum_{k=0}^{K} c_k \Psi_k(\boldsymbol{\xi})$$

$$\Psi_k(\xi_1, \xi_2, ..., \xi_d) = \psi_{k_1}(\xi_1)\psi_{k_2}(\xi_2)\cdots\psi_{k_d}(\xi_d)$$

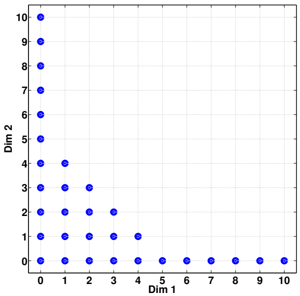


- Issues:
  - how to properly choose the basis set?
  - need to work in underdetermined regime $N < K$: fewer data than bases (d.o.f.)
- Discover the underlying low-d structure in the model
  - sparse regression

Bayesian inference of PC surrogate: high-d, low-data regime

$$Z = f(\boldsymbol{\xi}) \approx \sum_{k=0}^{K} c_k \Psi_k(\boldsymbol{\xi})$$

$$\Psi_k(\xi_1, \xi_2, ..., \xi_d) = \psi_{k_1}(\xi_1)\psi_{k_2}(\xi_2)\cdots\psi_{k_d}(\xi_d)$$



- Issues:
  - how to properly choose the basis set?
  - need to work in underdetermined regime $N < K$: fewer data than bases (d.o.f.)
- Discover the underlying low-d structure in the model
  - sparse regression

Regularization/Sparsity

- N training data points $(\boldsymbol{\xi}_i, Z_i)$ and $K+1$ basis terms $\Psi_k(\cdot)$
- 'Measurement' matrix $\boldsymbol{P}^{N\times(K+1)}$ with $\boldsymbol{P}_{ik} = \Psi_k(\boldsymbol{\xi}_i)$
- Find regression weights $\boldsymbol{c} = (c_0, \dots, c_K)$ so that

$$\boldsymbol{Z} \approx \boldsymbol{P}\boldsymbol{c} \quad \text{or} \quad Z_i \approx \sum_{k=0}^{K} c_k \Psi_k(\boldsymbol{\xi}_i)$$

- The number of polynomial basis terms grows fast; a p-th order, d-dimensional basis has a total of $K+1 = (p+d)!/(p!d!)$ terms.
- For limited data and large basis set ($N \leq K$) this is a sparse signal recovery problem $\Rightarrow$ need some regularization/constraints.
- Least-squares $\quad \text{argmin}_{\boldsymbol{c}} \left\{ ||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2^2 \right\}$
- The 'sparsest' $\quad \text{argmin}_{\boldsymbol{c}} \left\{ ||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2^2 + \alpha ||\boldsymbol{c}||_0 \right\}$
- Compressive sensing $\quad \text{argmin}_{\boldsymbol{c}} \left\{ ||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2^2 + \alpha ||\boldsymbol{c}||_1 \right\}$

Outline

Inverse UQ – Estimation of Uncertain Parameters

- Require joint PDF on input space
- Statistical inference – an inverse problem

- Given Constraints: PDF on uncertain inputs can be estimated using the Maximum Entropy principle
  - MaxEnt Methods

- Given Data: PDF on uncertain inputs can be estimated using Bayes formula
  - **Bayesian Inference**

Bayes formula for Parameter Inference

- Collected data: $\{(x_i, y_i)\}_{i=1}^N$
- Data model: $y_i = f(x_i; \lambda) + \epsilon_i$
- Bayes formula:

$$\underbrace{p(\lambda|y)}_{\text{Posterior}} = \frac{\overbrace{p(y|\lambda)}^{\text{Likelihood}}\ \overbrace{p(\lambda)}^{\text{Prior}}}{\underbrace{p(y)}_{\text{Evidence}}}$$

- Prior: knowledge of λ prior to data
- Likelihood: forward model and measurement noise
- Posterior: combines information from prior and data
- Evidence: normalizing constant for present context

The Prior

- Prior $p(\lambda)$ comes from
  - Physical constraints
  - Prior data/knowledge
- Types of *uninformative* priors
  - Improper prior
  - Objective prior
  - Maxent prior
  - Reference prior
  - Jeffreys prior

$$\underbrace{p(\lambda|y)}_{\text{Posterior}} = \frac{\overbrace{p(y|\lambda)}^{\text{Likelihood}}\ \overbrace{p(\lambda)}^{\text{Prior}}}{\underbrace{p(y)}_{\text{Evidence}}}$$

- It can be chosen to impose *regularization*
- Unknown aspects of the prior can be added to the rest of the parameters as hyperparameters
- The choice of prior can be crucial if data is not informative
- When there is sufficient information in the data, the data can overrule the prior

Construction of the Likelihood $p(y|\lambda)$

$$\underbrace{p(\lambda|y)}_{\text{Posterior}} = \frac{\overbrace{p(y|\lambda)}^{\text{Likelihood}}\ \overbrace{p(\lambda)}^{\text{Prior}}}{\underbrace{p(y)}_{\text{Evidence}}}$$

- Requires a presumed error model
- Data model: $y_i = f(x_i;\lambda) + \epsilon_i$
- Model this error as a random variable, e.g.
  - Error is due to instrument measurement noise
  - Instrument has Gaussian errors, with no bias
  - Measurements are independent

$$\epsilon \sim N(0, \sigma^2)$$

- For any given λ, this implies

$$y_i|\lambda, \sigma \sim N(f(x_i;\lambda), \sigma^2)$$

or

$$p(y|\lambda,\sigma) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(y_i - f(x_i;\lambda))^2}{2\sigma^2}\right)$$

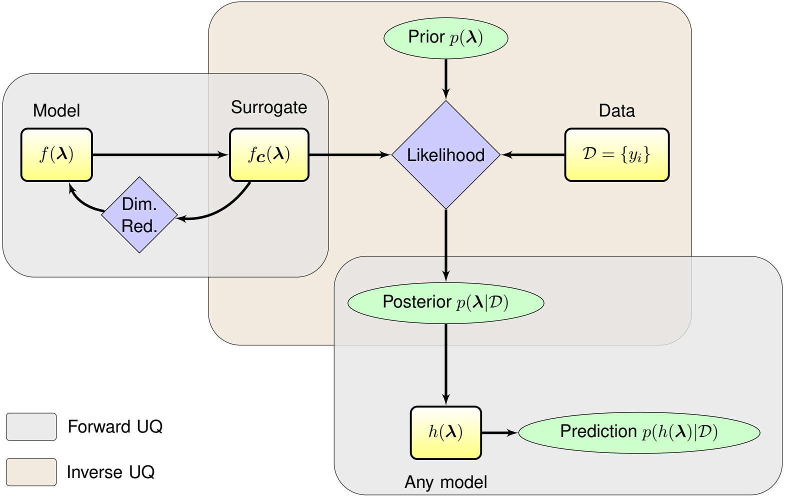
Exploring the Posterior

- Given any sample λ, the un-normalized posterior probability can be easily computed

$$\underbrace{p(\lambda|y)}_{\text{Posterior}} \propto \underbrace{p(y|\lambda)}_{\text{Likelihood}} \underbrace{p(\lambda)}_{\text{Prior}}$$

- Explore posterior w/ Markov Chain Monte Carlo (MCMC)
  - Metropolis-Hastings algorithm:
    - Random walk with proposal PDF & rejection rules
  - Computationally intensive, $\mathcal{O}(10^5)$ samples
  - Each sample: evaluation of the forward model
    - Surrogate models [Marzouk et. al, 2009]
- Evaluate moments/marginals from the MCMC statistics

Forward and Inverse UQ in a workflow



Model Evidence and Complexity

Let $\mathcal{M} = \{M_1, M_2, \ldots\}$ be a set of models of interest

- Parameter estimation from data is conditioned on the model

$$p(\theta|D, M_k) = \frac{p(D|\theta, M_k)\pi(\theta|M_k)}{p(D|M_k)}$$

Evidence (marginal likelihood) for M_k:

$$p(D|M_k) = \int p(D|\theta, M_k)\pi(\theta|M_k)\mathrm{d}\theta$$

Model evidence is useful for model selection

- Choose model with maximum evidence
- Compromise between fitting data and model complexity
  - Optimal complexity – Occam's razor principle
  - Avoid overfitting

Too much model complexity leads to overfitting

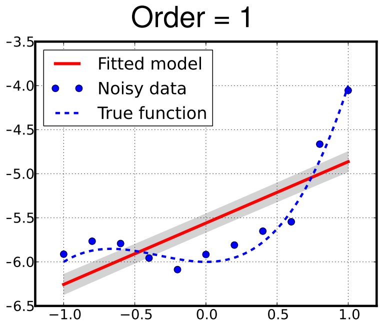
Data model: $i = 1, \ldots, N$

$$\begin{aligned} y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\ \epsilon_i &\sim N(0, s) \end{aligned}$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

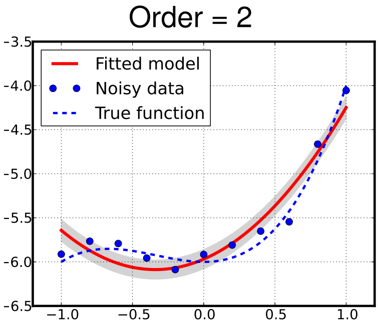
Data model: $i = 1, \ldots, N$

$$\begin{aligned} y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\ \epsilon_i &\sim N(0, s) \end{aligned}$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

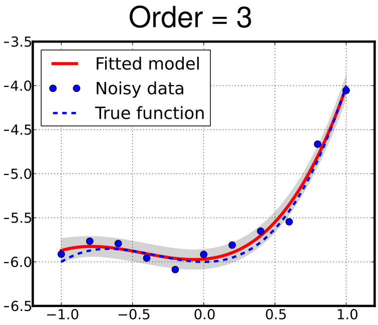
Data model: $i = 1, \dots, N$

$$\begin{aligned} y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\ \epsilon_i &\sim N(0, s) \end{aligned}$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

Uniform priors $\pi(c_k)$, $k = 0, \dots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

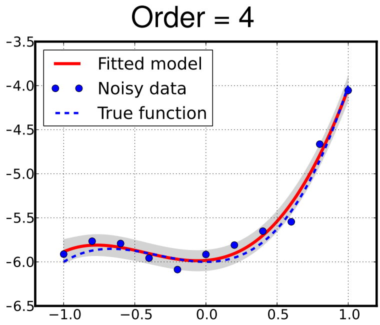
Data model: $i = 1, \ldots, N$

$$
\begin{aligned}
y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\
\epsilon_i &\sim N(0, s)
\end{aligned}
$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$
y_m = \sum_{k=0}^{P} c_k \psi_k(x)
$$

Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

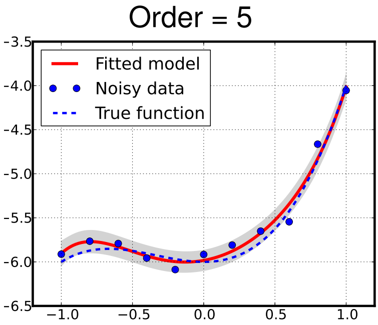
Data model: $i = 1, \ldots, N$

$$\begin{aligned} y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\ \epsilon_i &\sim N(0, s) \end{aligned}$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

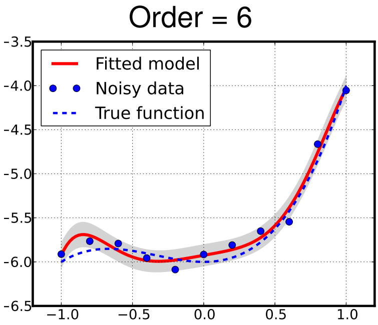
Data model: $i = 1, \ldots, N$

$$\begin{aligned} y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\ \epsilon_i &\sim N(0, s) \end{aligned}$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

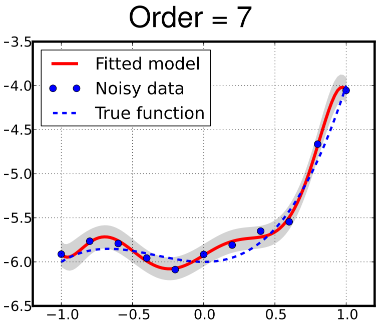
Data model: $i = 1, \ldots, N$

$$y_i = x_i^3 + x_i^2 - 6 + \epsilon_i$$
$$\epsilon_i \sim N(0, s)$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

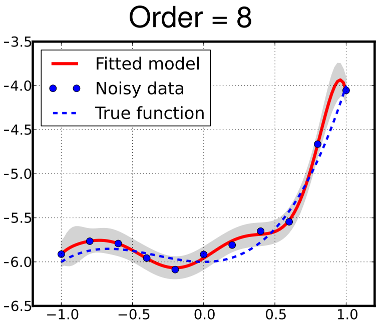
Data model: $i = 1, \ldots, N$

$$\begin{aligned} y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\ \epsilon_i &\sim N(0, s) \end{aligned}$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

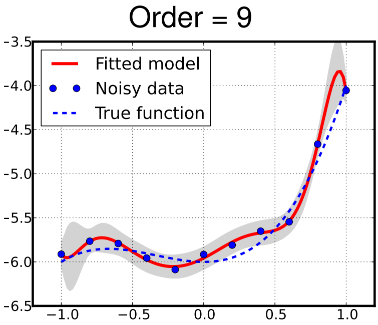
Data model: $i = 1, \ldots, N$

$$\begin{aligned} y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\ \epsilon_i &\sim N(0, s) \end{aligned}$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Too much model complexity leads to overfitting

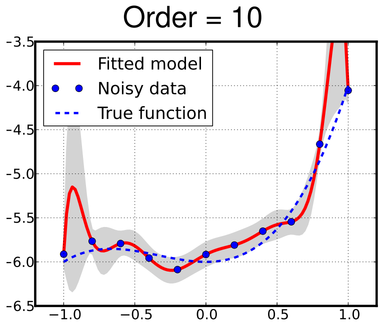
Data model: $i = 1, \ldots, N$

$$\begin{aligned} y_i &= x_i^3 + x_i^2 - 6 + \epsilon_i \\ \epsilon_i &\sim N(0, s) \end{aligned}$$

Bayesian regression with Legendre PCE fit models, order 1-10

$$y_m = \sum_{k=0}^{P} c_k \psi_k(x)$$

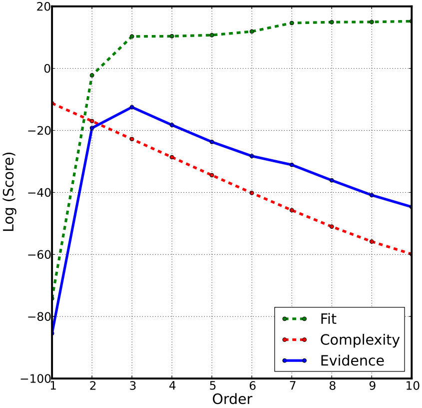
Uniform priors $\pi(c_k)$, $k = 0, \ldots, P$



Fitted model pushed-forward posterior versus the data

Evidence and Cross-Validation Error

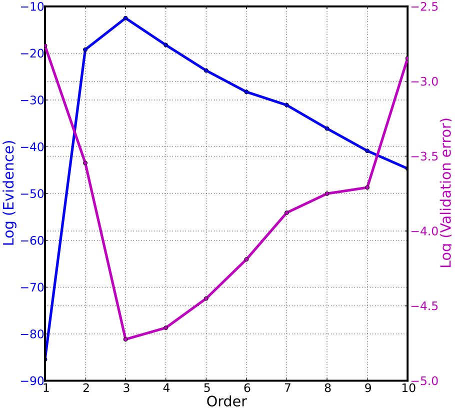
- Model evidence peaks at the true polynomial order of 3
- Cross validation error is equally minimal at order 3
- Models with optimal complexity are robust to cross validation



Log evidence: sum of two scores, balances complexity & fit

Evidence and Cross-Validation Error

- Model evidence peaks at the true polynomial order of 3
- Cross validation error is equally minimal at order 3
- Models with optimal complexity are robust to cross validation



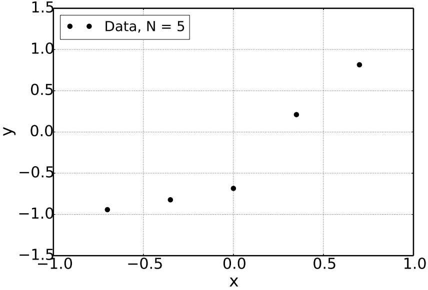
Cross validation error and model evidence versus order

Main target: model error $g(x) \approx f(x; \lambda)$

deviation from 'truth' or from a higher-fidelity model

- ... otherwise called (with slightly altered meanings):
  model discrepancy, model structural error,
  model inadequacy, model misspecification,
  model form error, model uncertainty
- Inverse modeling context
  - Given experimental or higher-fidelity model data, estimate the model error
- Represent and estimate the error associated with
  - Simplifying assumptions, parameterizations
  - Mathematical formulation, theoretical framework
- ...will be useful for
  - Model validation and model comparison
  - Scientific discovery and model improvement
  - Reliable computational predictions

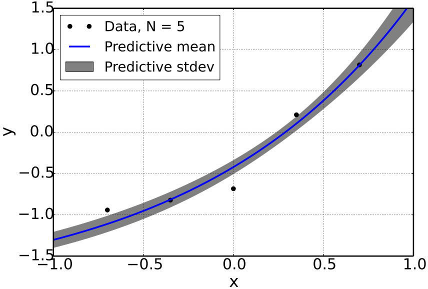
Ignoring model error leads to overconfident and biased predictions



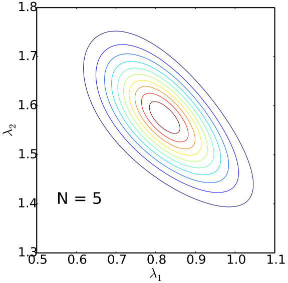
Model-data fit

- Given noisy data, calibrate an exponential model: $g(x) \approx f(x;\lambda)$

Ignoring model error leads to overconfident and biased predictions



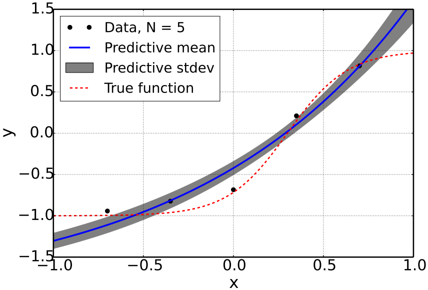
Model-data fit



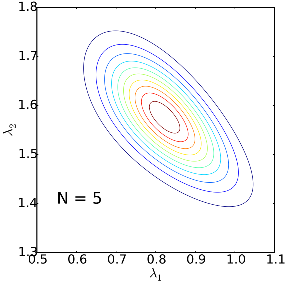
Posterior on parameters

- Given noisy data, calibrate an exponential model: $g(x) \approx f(x;\lambda)$
- Employ Bayesian inference to obtain posterior PDFs on λ

Ignoring model error leads to overconfident and biased predictions



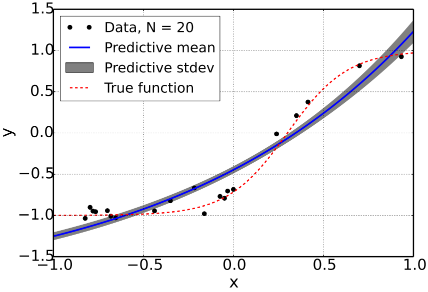
Model-data fit



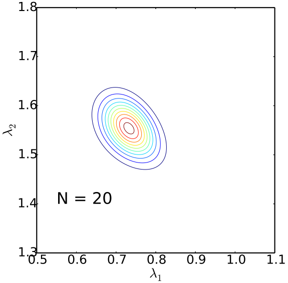
Posterior on parameters

- Given noisy data, calibrate an exponential model: $g(x) \approx f(x; \lambda)$
- Employ Bayesian inference to obtain posterior PDFs on λ
- True model – dashed-red – is *structurally* different from fit model $f(x, \lambda)$

Ignoring model error leads to overconfident and biased predictions



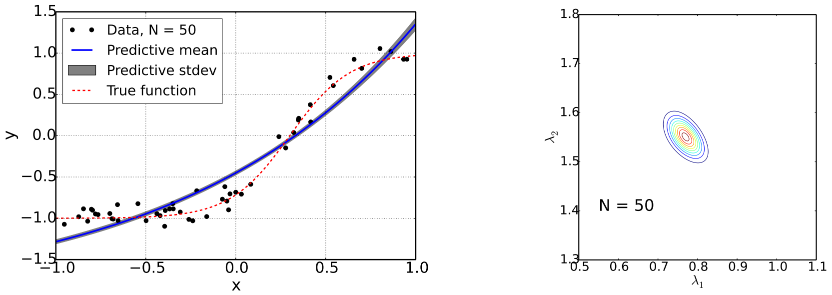
Model-data fit



Posterior on parameters

- Given noisy data, calibrate an exponential model: $g(x) \approx f(x; \lambda)$
- Employ Bayesian inference to obtain posterior PDFs on λ
- True model – dashed-red – is *structurally* different from fit model $f(x, \lambda)$
- Higher data amount reduces posterior and predictive uncertainty
  - Increasingly sure about predictions based on the *wrong* model

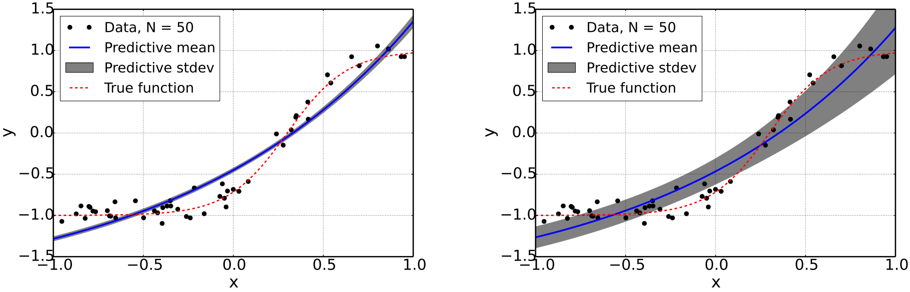
Ignoring model error leads to overconfident and biased predictions



Model-data fit   Posterior on parameters

- Given noisy data, calibrate an exponential model: $g(x) \approx f(x;\lambda)$
- Employ Bayesian inference to obtain posterior PDFs on λ
- True model – dashed-red – is *structurally* different from fit model $f(x, \lambda)$
- Higher data amount reduces posterior and predictive uncertainty
  - Increasingly sure about predictions based on the *wrong* model

Ignoring model error leads to overconfident and biased predictions



No model error treatment

Model error accounted for

- Given noisy data, calibrate an exponential model: $g(x) \approx f(x; \lambda)$
- Employ Bayesian inference to obtain posterior PDFs on λ
- True model – dashed-red – is *structurally* different from fit model $f(x, \lambda)$
- Accounting for model error allows extra uncertainty component to propagate through predictions

Where to put model error?

$$y_i = f(x_i; \lambda) + \delta(x_i) + \epsilon_i$$

- Outside:
  - Explicit GP representation [Kennedy-O'Hagan, 2001]
  - See also [Higdon et. al, 2004], [Bayarri et. al, 2007]
  - Usage: too many to cite
  - Issues: see next slide
  - Variants exist: multiplicative noise, non-linear maps etc.

$$y_i = \tilde{f}(x_i; \lambda, \delta(x_i)) + \epsilon_i$$

- Inside:
  - Increased use, especially in physical models: [Emory et. al, 2011] [Oliver and Moser, 2011], [Morrison et. al, 2016], [Sondak et. al, 2017], [Huan et. al, 2017], [Rizzi et. al, 2018]...
  - Engineering/statistical adjustment [Joseph and Melkote, 2009]
  - Additive corrections to submodels [Strong et. al, 2011]
  - Validation of extrapolative predictions [Oliver et. al, 2014]
  - Field inversion and machine learning [Duraisamy et. al, 2015-]
  - Hybrid correction [He and Xiu, 2016]
  - Random field correction [Brown and Atamturktur, 2016]
  - Hierarchical mixture model [Feng, 2017]
  - Parameter inflation [Pernot et. al, 2017]
  - Hierarchical stochastic model [Wu et. al, 2017]
  - Dynamic discrepancy [Bhat et. al., 2017]

External correction often not satisfactory for physical models

$$y_i = \underbrace{f(x_i;\lambda) + \delta(x_i)}_{\text{truth } g(x_i)} + \epsilon_i$$

- Explicit additive statistical model for model error [KOH, 2001]
- Potential violation of physical constraints
- Disambiguation of model error $\delta(x_i)$ and data error ϵ_i
- Yes, priors help: [Brynjarsdottir and O'Hagan, 2014], [Plumlee, 2017]
- Calibration of model error on measured observable does not impact the quality of model predictions on other QoIs
- Physical scientists are unlikely to augment their model with a statistical model error term on select outputs
  - Calibrated predictive model: $f(x;\lambda) + \delta(x)$ or $f(x;\lambda)$?
- Problem is highlighted in model-to-model calibration ($\epsilon_i = 0$)
  - no a priori knowledge of the statistical structure of $\delta(x)$

Bayesian Framework for Model Error Estimation

$$y_i = f(x_i; \lambda + \delta_\alpha) + \epsilon_i$$

- Given data y_i, perform *simultaneous* estimation of $\tilde{\alpha} = (\lambda, \alpha)$, i.e. model parameters λ and model-error parameters α.
- Bayes' theorem

$$\overbrace{p(\tilde{\alpha}|y)}^{\text{Posterior}} = \frac{\overbrace{p(y|\tilde{\alpha})}^{\text{Likelihood}} \; \overbrace{p(\tilde{\alpha})}^{\text{Prior}}}{\underbrace{p(y)}_{\text{Evidence}}}$$

- In order to estimate the likelihood $L_y(\tilde{\alpha}) = p(y|\tilde{\alpha}) = p(y|\lambda, \alpha)$, one needs uncertainty propagation through $f(x_i; \underbrace{\lambda + \delta_\alpha}_{\text{stochastic}})$,
- ... hence, we employ Polynomial Chaos (PC) representation for δ_α.

Polynomial Chaos Representation of Augmented Input

$$y_i = f(x_i; \lambda + \delta_\alpha) + \epsilon_i$$

- Zero-mean PC form $\delta_\alpha = \sum_{k=1}^{K} \alpha_k \Psi_k(\xi)$
- Functional representation of a large class of random variables
- The PC *germ* ξ is a standard random variable
  - e.g. Uniform$(-1, 1)$ or Normal$(0, 1)$
- The PC bases (e.g. Legendre or Hermite polynomials) are orthogonal w.r.t. PDF of ξ

$$\int \Psi_m(\xi)\Psi_k(\xi)\pi_\xi(\xi)d\xi = 0 \quad \text{for } m \neq k.$$

- PC representation allows efficient
  - Sampling
  - Moment estimation
  - Variance-based decomposition
  - Uncertainty propagation (via NISP)

Model Error – Likelihood construction

$$y_i = f(x_i; \lambda + \delta_\alpha(\zeta)) + \epsilon_i = f_i(\tilde{\alpha}, \zeta) + \epsilon_i$$

- Likelihood $\mathcal{L}_g(\tilde{\alpha}) = p(y|\tilde{\alpha})$ challenging, but can compute moments

$$\mu_i(\tilde{\alpha}) = \mathbb{E}_\zeta[f_i(\tilde{\alpha}, \zeta)] \qquad \text{and} \qquad \sigma_i^2(\tilde{\alpha}) = \mathbb{V}_\zeta[f_i(\tilde{\alpha}, \zeta)] + s_i^2$$

- Gauss-Marginal Approximate Likelihood compares data y_i and model predictions:

$$\mathcal{L}_g(\tilde{\alpha}) \approx \frac{1}{(2\pi)^{N/2}} \prod_{i=1}^{N} \frac{1}{\sigma_i(\tilde{\alpha})} \exp\left(-\frac{1}{2} \left(\frac{y_i - \mu_i(\tilde{\alpha})}{\sigma_i(\tilde{\alpha})} \right)^2 \right)$$

- Non-intrusive spectral projection (NISP) with Polynomial Chaos

$$f_i(\tilde{\alpha}, \zeta) \overset{\text{NISP}}{\simeq} \sum_k f_{ik}(\tilde{\alpha}) \Psi_k(\zeta)$$

- ... provides easy access to mean and variance

$$\mu_i(\tilde{\alpha}) = f_{i0}(\tilde{\alpha}) \qquad \text{and} \qquad \sigma_i^2(\tilde{\alpha}) = \sum_{k\neq 0} f_{ik}^2(\tilde{\alpha}) ||\Psi_k||^2 + s_i^2$$

Model Error – Surrogate and Prediction

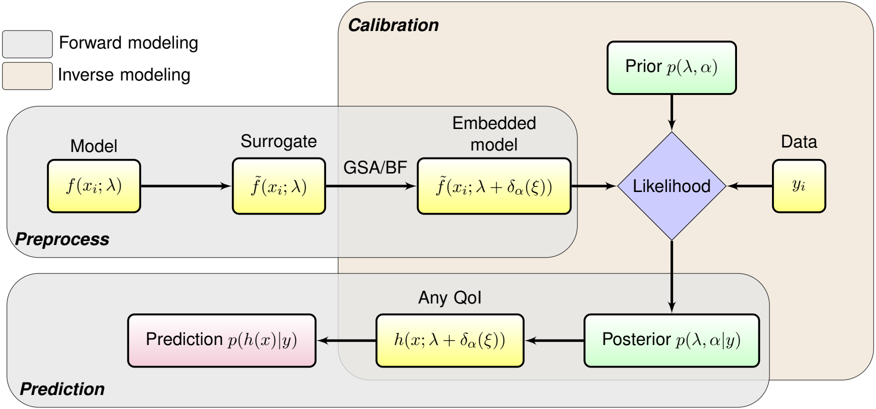
$$f_i(\lambda + \delta_\alpha(\zeta)) = f_i(\tilde{\alpha}, \zeta) \overset{\text{NISP}}{\simeq} \sum_k f_{ik}(\tilde{\alpha}) \Psi_k(\zeta)$$

- NISP is employed both for likelihood computation and for posterior/pushed-forward predictions in general
- In practice, $f_i(\cdot)$ is replaced by a pre-constructed polynomial surrogate
- Note: NISP with finite truncation is exact, if one truncates NISP at the same order as the surrogate of $f_i(\cdot)$
- Posterior predictive moments

$$\mu_i = \mathbb{E}_{\tilde{\alpha}} \left[\mu_i(\tilde{\alpha}) \right]$$

$$\sigma_i^2 = \underbrace{\mathbb{E}_{\tilde{\alpha}} \left[\sigma_i^2(\tilde{\alpha}) \right]}_{\text{Model error}} + \underbrace{\mathbb{V}_{\tilde{\alpha}} \left[\mu_i(\tilde{\alpha}) \right]}_{\text{Posterior uncertainty}} + \underbrace{(\sigma_i^{LOO})^2}_{\text{Surrogate error}} + \underbrace{s_i^2}_{\text{Data noise}}$$

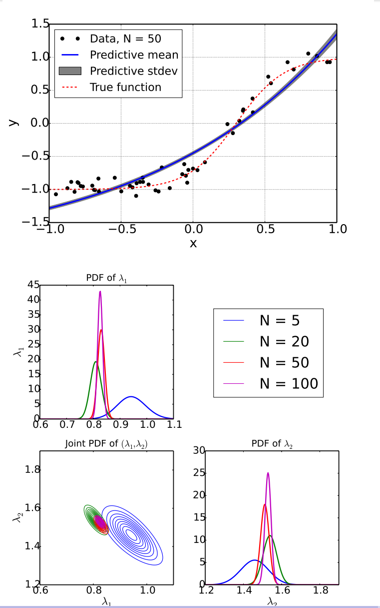
Model error embedding – workflow

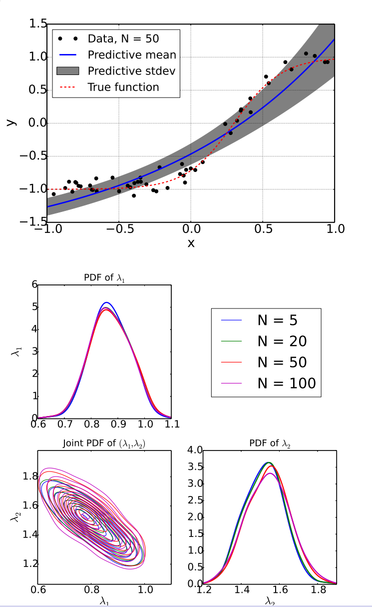


- Predictive uncertainty decomposition: Total Variance =

  Posterior uncertainty + Data noise + Model error + Surrogate error

.. back to toy example



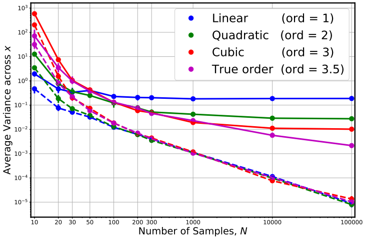


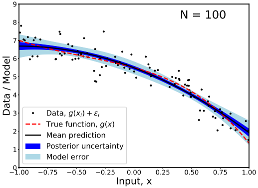
More data leads to 'leftover' model error

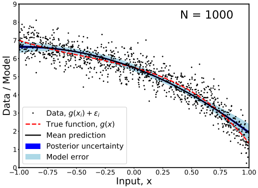
Calibrating a quadratic $f(x) = \lambda_0 + \lambda_1 x + \lambda_2 x^2$
w.r.t. 'truth' $g(x) = 6 + x^2 - 0.5(x+1)^{3.5}$ measured with noise $\sigma = 0.1$.

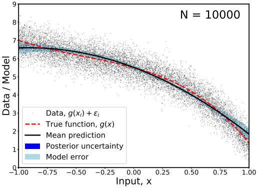
Summary of features:

- Well-defined model-to-model calibration
- Model-driven discrepancy correlations
- Respects physical constraints
- Disambiguates model and data errors
- Calibrated predictions of multiple QoIs









Outline

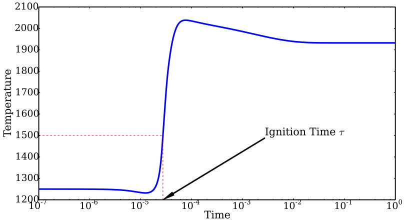
Ignition time in chemical kinetics

- Two-step global reaction model calibrated against shock tube experimental data
- Operating conditions: pressure P, initial temperature T_0 & equivalence ratio ϕ

$$C_{12}H_{26} + \frac{25}{2}O_2 \xrightarrow{k_1} 12CO + 13H_2O$$

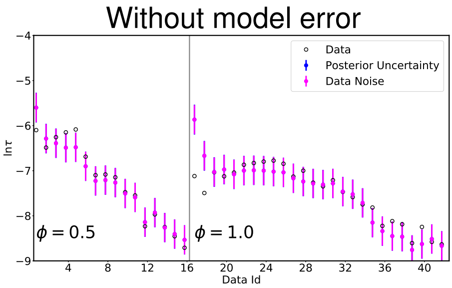
$$CO + \frac{1}{2}O_2 \underset{k_{2b}}{\overset{k_{2f}}{\rightleftharpoons}} CO_2.$$

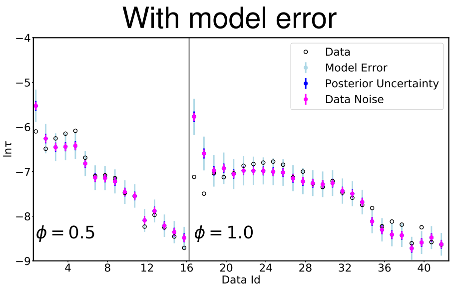
$$k_1 = Ae^{\left(-\frac{E}{RT}\right)}[C_{12}H_{26}]^{0.25}[O_2]^{1.25}$$



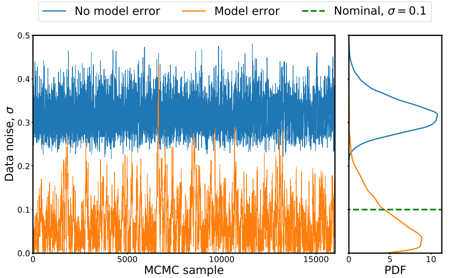
- Data: log(ignition time)
- Embedding
  $(\ln A, E) = \sum_k \boldsymbol{\alpha}_k \Psi_k(\boldsymbol{\xi})$

Ignition time in chemical kinetics



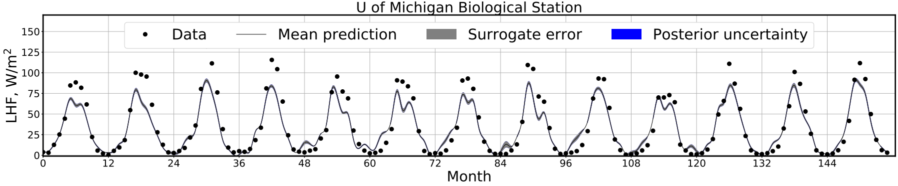


- Model error disambiguated from data error
- Data error correctly captured
- Meaningful extrapolative predictions



E3SM Land Model (ELM)

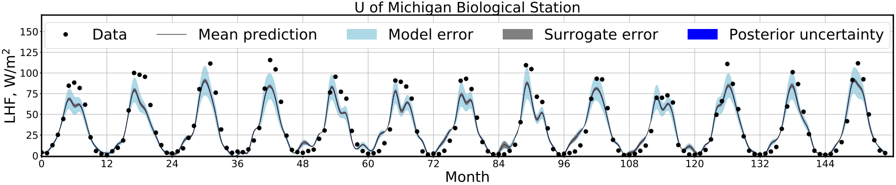
- US Department of Energy (DOE) sponsored Earth system model
- Land, atmosphere, ocean, ice, human system components
- High-resolution, employ DOE leadership-class computing facilities



- Conventional calibration without model error

E3SM Land Model (ELM)

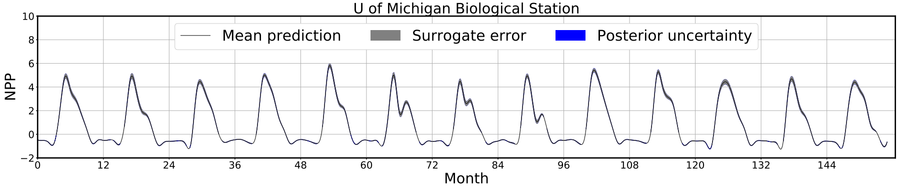
- US Department of Energy (DOE) sponsored Earth system model
- Land, atmosphere, ocean, ice, human system components
- High-resolution, employ DOE leadership-class computing facilities



- Predictive variance decomposition with model-error component
- ... with predictive uncertainty that captures model error

E3SM Land Model (ELM)

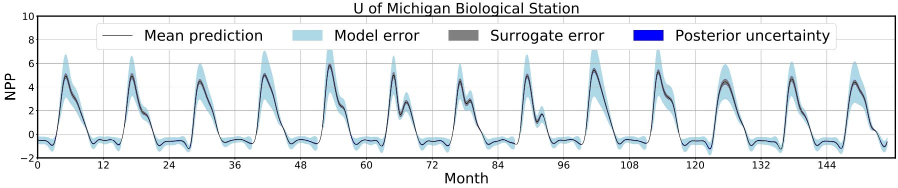
- US Department of Energy (DOE) sponsored Earth system model
- Land, atmosphere, ocean, ice, human system components
- High-resolution, employ DOE leadership-class computing facilities



- Predictive variance decomposition with model-error component
- Allows meaningful prediction of other QoIs
  (e.g. no data/observable)

E3SM Land Model (ELM)

- US Department of Energy (DOE) sponsored Earth system model
- Land, atmosphere, ocean, ice, human system components
- High-resolution, employ DOE leadership-class computing facilities

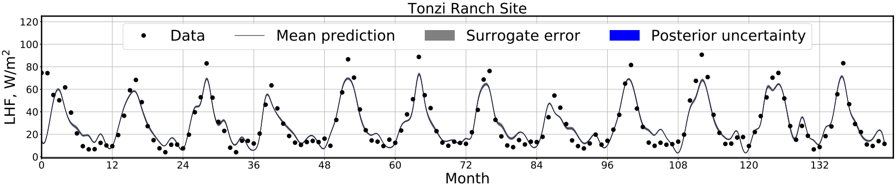


- Predictive variance decomposition with model-error component
- Allows meaningful prediction of other QoIs (e.g. no data/observable)
- ... with predictive uncertainty that captures model error

E3SM Land Model (ELM)

E3SM Energy Exascale Earth System Model

- US Department of Energy (DOE) sponsored Earth system model
- Land, atmosphere, ocean, ice, human system components
- High-resolution, employ DOE leadership-class computing facilities

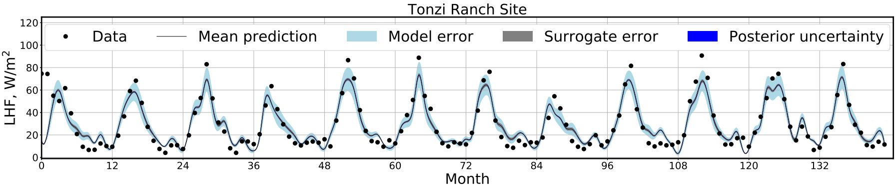


- Predictive variance decomposition with model-error component
- Allows (a more dangerous) extrapolation to other sites

E3SM Land Model (ELM)

E3SM Energy Exascale Earth System Model

- US Department of Energy (DOE) sponsored Earth system model
- Land, atmosphere, ocean, ice, human system components
- High-resolution, employ DOE leadership-class computing facilities

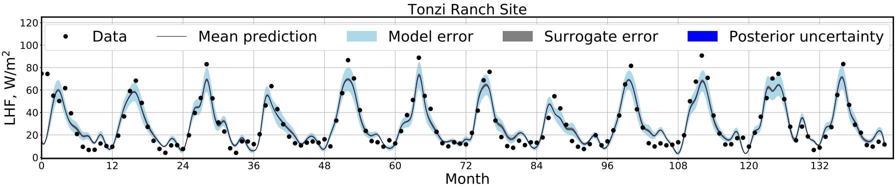


- Predictive variance decomposition with model-error component
- Allows (a more dangerous) extrapolation to other sites
- ... with predictive uncertainty that captures model error

E3SM Land Model (ELM)

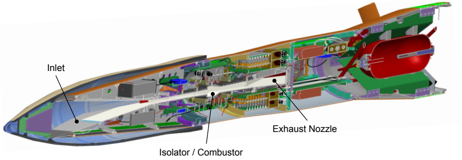
E3SM Energy Exascale Earth System Model

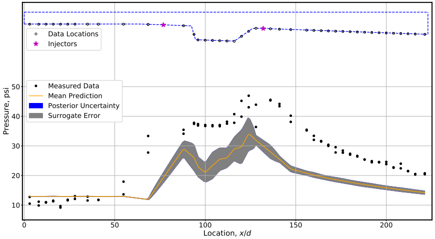
- US Department of Energy (DOE) sponsored Earth system model
- Land, atmosphere, ocean, ice, human system components
- High-resolution, employ DOE leadership-class computing facilities

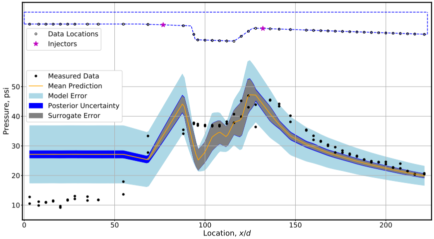


LES: Turbulent Combustion in Scramjet Engine

- HIFiRE (Hypersonic International Flight Research and Experimentation) scramjet
- Pressure data from NASA Langley Research Center
- Highly complex LES model







- Augmenting model error leads to more ‘physical’ likelihood

Outline

Summary

- Forward UQ: Polynomial Chaos representation of RVs
  - Non-intrusive spectral projection
  - Surrogate construction, Bayesian regression
  - **High-D challenge: sparse PC via compressive sensing**
- Inverse UQ: Bayesian inference for parameter estimation
  - Bayesian parameter estimation
  - **Model error quantification: embedded model error approach**
- All developments done within UQTk, lightweight C++/Python library out of SNL-CA (*github.com/sandialabs/UQTk*)

Literature : General UQ

Ghanem, R., Spanos, P., “Stochastic Finite Elements: A Spectral Approach”, Springer Verlag, (1991).

Xiu, D., Karniadakis, G., “The Wiener-Askey Polynomial Chaos for Stocahstic Differential Equations”, *SIAM J. Sci. Comp.*, 24(2), 619-644, (2002).

Le Maître, O., Knio, O., “Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics”, Springer-Verlag, (2010).

Najm, H., “Uncertainty Quantification and Polynomial Chaos Techniques in Computational Fluid Dynamics”, *Ann. Rev. Fluid Mech.*, 41(1):35-52, (2009).

Xiu, D., “Numerical Methods for Stochastic Computations: A Spectral Method Approach”, Princeton U. Press (2010).

Marzouk, Y., Najm, H., “Dimensionality Reduction and Polynomial Chaos Acceleration of Bayesian Inference in Inverse Problems”, *J. Comp. Phys.*, 228(6):1862-1902, (2009).

Literature, continued

Thank you!

Bayesian compressive sensing

S. Ji, Y. Xue and L. Carin, “Bayesian Compressive Sensing”, *IEEE Trans. Signal Proc.*, 56(6), (2008).

K. Sargsyan, C. Safta, H. Najm, B. Debusschere, D. Ricciuto, P. Thornton, “Dimensionality reduction for complex models via Bayesian compressive sensing”, *Int. J. Uncertainty Quantification*, 4(1), 63-93, (2014).

D. Ricciuto, K. Sargsyan, P. Thornton, “The Impact of Parametric Uncertainties on Biogeochemistry in the E3SM Land Model”, *J of Advances in Modeling Earth Systems*, 10(2), 297-319, (2018).

Model structural error

M. Kennedy and A. O’Hagan, “Bayesian Calibration of Computer Models”, *Journal of the Royal Statistical Society*, Series B. 63, 425-464, 2001.

K. Sargsyan, H. Najm, R. Ghanem, “On the Statistical Calibration of Physical Models”, *Int. J. Chem. Kinetics*, 47(4), 2015.

X. Huan et. al., “Global Sensitivity Analysis and Estimation of Model Error, Toward Uncertainty Quantification in Scramjet Computations”, *AIAA Journal*, 56 (3), 2018.

K. Sargsyan, X. Huan, H. Najm. “Embedded Model Error Representation for Bayesian Model Calibration”, arXiv:1801.06768, *Int. J. Uncert. Quant.*, 9(4), 2019.

Postdoc positions available, Livermore, CA

A few postdoctoral open positions at Sandia National Labs in Livermore, broadly at the intersection of **machine learning** and **uncertainty quantification**, including both fundamental work and applications to a range of domain problems.

http://www.sandia.gov/careers/students_postdocs/postdocs.html, click 'View All Jobs' and search for Job IDs:

- 672950 - Advanced ML methods, probabilistic neural networks, neural ODEs
- 673228 - Bayesian inference and ML of interatomic potentials, UQ in molecular dynamics
- 672487 - Develop and deploy advanced UQ methods for multi-physics computational codes

Additional Material

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + \ c_1\psi_1(\xi_1) \ + \ c_2\psi_1(\xi_2) \ + \ c_3\psi_1(\xi_3) \ +$$

$$+ \ c_4\psi_2(\xi_1) \ + \ c_5\psi_1(\xi_1)\psi_1(\xi_2) \ + \ c_6\psi_1(\xi_1)\psi_1(\xi_3) \ + \ c_7\psi_2(\xi_2) \ + \ c_8\psi_1(\xi_2)\psi_1(\xi_3) \ + \ c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + \ c_1^2\langle\psi_1^2\rangle \ + \ c_2^2\langle\psi_1^2\rangle \ + \ c_3^2\langle\psi_1^2\rangle \ +$$

$$+ \ c_4^2\langle\psi_2^2\rangle \ + \ c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_7^2\langle\psi_2^2\rangle \ + \ c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_9^2\langle\psi_2^2\rangle$$

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$, number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + c_1\psi_1(\xi_1) + c_2\psi_1(\xi_2) + c_3\psi_1(\xi_3) +$$

$$+ c_4\psi_2(\xi_1) + c_5\psi_1(\xi_1)\psi_1(\xi_2) + c_6\psi_1(\xi_1)\psi_1(\xi_3) + c_7\psi_2(\xi_2) + c_8\psi_1(\xi_2)\psi_1(\xi_3) + c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + c_1^2\langle\psi_1^2\rangle + c_2^2\langle\psi_1^2\rangle + c_3^2\langle\psi_1^2\rangle +$$

$$+ c_4^2\langle\psi_2^2\rangle + c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_7^2\langle\psi_2^2\rangle + c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_9^2\langle\psi_2^2\rangle$$

Main effect sensitivities ξ_1 ξ_2 ξ_3

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + \ c_1\psi_1(\xi_1) \ + \ c_2\psi_1(\xi_2) \ + \ c_3\psi_1(\xi_3) \ +$$

$$+ \ c_4\psi_2(\xi_1) \ + \ c_5\psi_1(\xi_1)\psi_1(\xi_2) \ + \ c_6\psi_1(\xi_1)\psi_1(\xi_3) \ + \ c_7\psi_2(\xi_2) \ + \ c_8\psi_1(\xi_2)\psi_1(\xi_3) \ + \ c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + \ c_1^2\langle\psi_1^2\rangle \ + \ c_2^2\langle\psi_1^2\rangle \ + \ c_3^2\langle\psi_1^2\rangle \ +$$

$$+ \ c_4^2\langle\psi_2^2\rangle \ + \ c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_7^2\langle\psi_2^2\rangle \ + \ c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_9^2\langle\psi_2^2\rangle$$

Main effect sensitivities ξ_1 ξ_2 ξ_3

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + \; c_1\psi_1(\xi_1) \; + \; c_2\psi_1(\xi_2) \; + \; c_3\psi_1(\xi_3) \; +$$

$$+ \; c_4\psi_2(\xi_1) \; + \; c_5\psi_1(\xi_1)\psi_1(\xi_2) \; + \; c_6\psi_1(\xi_1)\psi_1(\xi_3) \; + \; c_7\psi_2(\xi_2) \; + \; c_8\psi_1(\xi_2)\psi_1(\xi_3) \; + \; c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + \; c_1^2\langle\psi_1^2\rangle \; + \; c_2^2\langle\psi_1^2\rangle \; + \; c_3^2\langle\psi_1^2\rangle \; +$$

$$+ \; c_4^2\langle\psi_2^2\rangle \; + \; c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \; + \; c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \; + \; c_7^2\langle\psi_2^2\rangle \; + \; c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \; + \; c_9^2\langle\psi_2^2\rangle$$

Main effect sensitivities ξ_1 ξ_2 ξ_3

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + c_1\psi_1(\xi_1) + c_2\psi_1(\xi_2) + c_3\psi_1(\xi_3) +$$

$$+ c_4\psi_2(\xi_1) + c_5\psi_1(\xi_1)\psi_1(\xi_2) + c_6\psi_1(\xi_1)\psi_1(\xi_3) + c_7\psi_2(\xi_2) + c_8\psi_1(\xi_2)\psi_1(\xi_3) + c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + c_1^2\langle\psi_1^2\rangle + c_2^2\langle\psi_1^2\rangle + c_3^2\langle\psi_1^2\rangle +$$

$$+ c_4^2\langle\psi_2^2\rangle + c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_7^2\langle\psi_2^2\rangle + c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_9^2\langle\psi_2^2\rangle$$

Total sensitivities ξ_1 ξ_2 ξ_3

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + c_1\psi_1(\xi_1) + c_2\psi_1(\xi_2) + c_3\psi_1(\xi_3) +$$
$$+ c_4\psi_2(\xi_1) + c_5\psi_1(\xi_1)\psi_1(\xi_2) + c_6\psi_1(\xi_1)\psi_1(\xi_3) + c_7\psi_2(\xi_2) + c_8\psi_1(\xi_2)\psi_1(\xi_3) + c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + c_1^2\langle\psi_1^2\rangle + c_2^2\langle\psi_1^2\rangle + c_3^2\langle\psi_1^2\rangle +$$
$$+ c_4^2\langle\psi_2^2\rangle + c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_7^2\langle\psi_2^2\rangle + c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_9^2\langle\psi_2^2\rangle$$

Total sensitivities ξ_1 ξ_2 ξ_3

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + c_1\psi_1(\xi_1) + c_2\psi_1(\xi_2) + c_3\psi_1(\xi_3) +$$
$$+ c_4\psi_2(\xi_1) + c_5\psi_1(\xi_1)\psi_1(\xi_2) + c_6\psi_1(\xi_1)\psi_1(\xi_3) + c_7\psi_2(\xi_2) + c_8\psi_1(\xi_2)\psi_1(\xi_3) + c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + c_1^2\langle\psi_1^2\rangle + c_2^2\langle\psi_1^2\rangle + c_3^2\langle\psi_1^2\rangle +$$
$$+ c_4^2\langle\psi_2^2\rangle + c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_7^2\langle\psi_2^2\rangle + c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + c_9^2\langle\psi_2^2\rangle$$

Total sensitivities ξ_1 ξ_2 ξ_3

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + \; c_1\psi_1(\xi_1) \; + \; c_2\psi_1(\xi_2) \; + \; c_3\psi_1(\xi_3) \; +$$

$$+ \; c_4\psi_2(\xi_1) \; + \; c_5\psi_1(\xi_1)\psi_1(\xi_2) + \; c_6\psi_1(\xi_1)\psi_1(\xi_3) \; + \; c_7\psi_2(\xi_2) \; + \; c_8\psi_1(\xi_2)\psi_1(\xi_3) \; + \; c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + \; c_1^2\langle\psi_1^2\rangle \; + \; c_2^2\langle\psi_1^2\rangle \; + \; c_3^2\langle\psi_1^2\rangle \; +$$

$$+ \; c_4^2\langle\psi_2^2\rangle \; + \; c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle + \; c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \; + \; c_7^2\langle\psi_2^2\rangle \; + \; c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \; + \; c_9^2\langle\psi_2^2\rangle$$

Joint sensitivities (ξ_1, ξ_2) (ξ_1, ξ_3) (ξ_2, ξ_3)

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + \ c_1\psi_1(\xi_1) \ + \ c_2\psi_1(\xi_2) \ + \ c_3\psi_1(\xi_3) \ +$$

$$+ \ c_4\psi_2(\xi_1) \ + \ c_5\psi_1(\xi_1)\psi_1(\xi_2) \ + \ c_6\psi_1(\xi_1)\psi_1(\xi_3) \ + \ c_7\psi_2(\xi_2) \ + \ c_8\psi_1(\xi_2)\psi_1(\xi_3) \ + \ c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + \ c_1^2\langle\psi_1^2\rangle \ + \ c_2^2\langle\psi_1^2\rangle \ + \ c_3^2\langle\psi_1^2\rangle \ +$$

$$+ \ c_4^2\langle\psi_2^2\rangle \ + \ c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_7^2\langle\psi_2^2\rangle \ + \ c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_9^2\langle\psi_2^2\rangle$$

Joint sensitivities (ξ_1, ξ_2) (ξ_1, ξ_3) (ξ_2, ξ_3)

Sensitivity indices are directly computable from PC

$$g(\boldsymbol{\xi}) = \sum_{k=0}^{P} c_k \Psi_k(\boldsymbol{\xi})$$

Consider dimensionality $d = 3$, total order $p = 2$,
number of PC terms $P + 1 = (d + p)!/(d!p!) = 10$.

$$g(\xi_1, \xi_2, \xi_3) = c_0 + \ c_1\psi_1(\xi_1) \ + \ c_2\psi_1(\xi_2) \ + \ c_3\psi_1(\xi_3) \ +$$

$$+ \ c_4\psi_2(\xi_1) \ + \ c_5\psi_1(\xi_1)\psi_1(\xi_2) \ + \ c_6\psi_1(\xi_1)\psi_1(\xi_3) \ + \ c_7\psi_2(\xi_2) \ + \ c_8\psi_1(\xi_2)\psi_1(\xi_3) \ + \ c_9\psi_2(\xi_3)$$

Variance contributions

$$Var(g) = 0 + \ c_1^2\langle\psi_1^2\rangle \ + \ c_2^2\langle\psi_1^2\rangle \ + \ c_3^2\langle\psi_1^2\rangle \ +$$

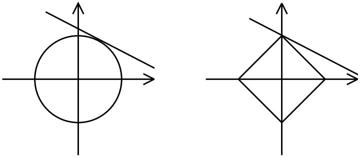
$$+ \ c_4^2\langle\psi_2^2\rangle \ + \ c_5^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_6^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_7^2\langle\psi_2^2\rangle \ + \ c_8^2\langle\psi_1^2\rangle\langle\psi_1^2\rangle \ + \ c_9^2\langle\psi_2^2\rangle$$

Joint sensitivities (ξ_1, ξ_2) (ξ_1, ξ_3) (ξ_2, ξ_3)

Compressive sensing and regularization

- Least-squares $\text{argmin}_{\boldsymbol{c}} ||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2^2$
- Tikhonov regularization; Ridge regression
$$\underset{\boldsymbol{c}}{\text{argmin}} ||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2^2 + ||\boldsymbol{c}||_2^2$$
- The 'sparsest' $\text{argmin}_{\boldsymbol{c}} \left\{||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2^2 + \alpha||\boldsymbol{c}||_0\right\}$
- Compressive sensing, LASSO, basis pursuit
$$\underset{\boldsymbol{c}}{\text{argmin}} \left\{||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2^2 + \alpha||\boldsymbol{c}||_1\right\}$$
  - ... or $\text{argmin}_{\boldsymbol{c}} ||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2$ s.t. $||\boldsymbol{c}||_1 < \epsilon$
  - ... or $\text{argmin}_{\boldsymbol{c}} ||\boldsymbol{c}||_1$ s.t. $||\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{c}||_2 < \epsilon$

$\Rightarrow$ discovery of sparse signals



Compressive sensing: enhancements

- Bayesian extension: $\mathrm{argmin}_{\boldsymbol{c}}\{\overbrace{||\boldsymbol{Z}-\boldsymbol{P}\boldsymbol{c}||_2}^{\text{Likelihood}}+\overbrace{\alpha||\boldsymbol{c}||_1}^{\text{Prior}}\}$
  - Get coefficients with uncertainties
  - Fights overfitting better
  - Connections with relevance vector machine (RVM)
- Weighted regularization
  - Always better, if you know how to weigh
- Iterative growth of polynomial basis
  - Exploit the structure of polynomial bases for smarter search
  - An iterative procedure that allows increasing the order for the relevant basis terms while maintaining the dimensionality reduction [Sargsyan *et al.* 2014], [Jakeman *et al.* 2015].
  - Iterations inform the weighting procedure

Case for Model Error Embedding

Ideally, modelers want predictive *errorbars*:
inserting randomness on the outputs has issues, so...

$$y_i = \tilde{f}(x_i; \lambda, \delta_\alpha) + \epsilon_i$$

- Embed model error in specific submodel phenomenology
  - a modified transport or constitutive law
  - a modified formulation for a material property
  - turbulent model constants
- Allows placement of model error term in locations where key modeling assumptions and approximations are made
  - as a correction or high-order term
  - as a possible alternate phenomenology
- Naturally preserves model structure and physical constraints
- Disambiguates model/data errors

Embedded Model Error Options

- Explore different model forms,

Intrusive $$y_i = \tilde{f}(x_i; \lambda, \delta_\alpha(x_i)) + \epsilon_i$$

- Additive stochastic corrections to existing inputs

Non-intrusive $$y_i = f(x_i; \lambda + \delta_\alpha(x_i)) + \epsilon_i$$

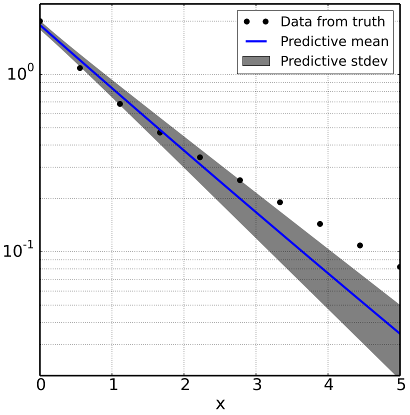
- ... even simpler, x-independent

$$y_i = f(x_i; \lambda + \delta_\alpha) + \epsilon_i$$

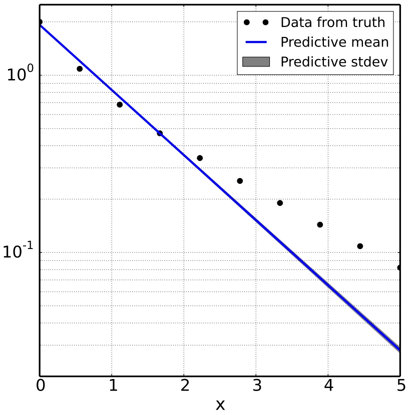
Predictions account for model error

Calibrating single-exponential models
with data from a double exponential model $g(x) = e^{-0.5x} + e^{-2x}$

Linear-exponential $f(x, \lambda) = e^{\lambda_1 + \lambda_2 x}$



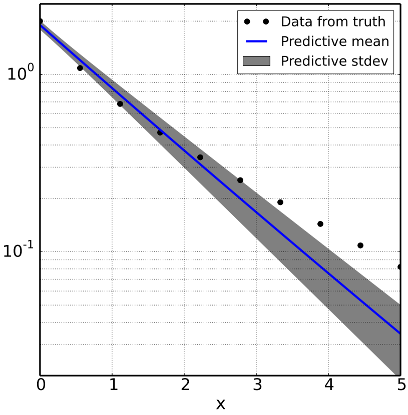
Additive Gaussian error



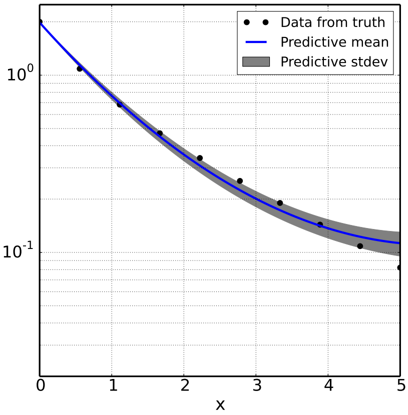
Predictions account for model error

Calibrating single-exponential models
with data from a double exponential model $g(x) = e^{-0.5x} + e^{-2x}$

Linear-exponential $f(x, \lambda) = e^{\lambda_1 + \lambda_2 x}$



Quadratic-exponential $f_2(x, \lambda) = e^{\lambda_1 + \lambda_2 x + \lambda_3 x^2}$



Chemistry problem – ABC

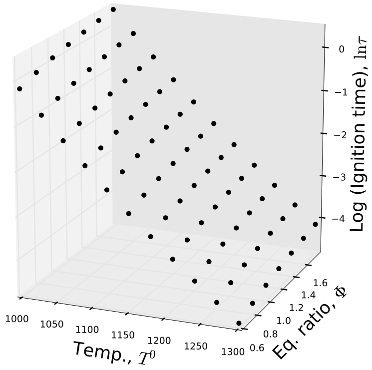
- Homogeneous ignition, methane-air mixture
- Single-step global reaction model calibrated against a detailed chemical kinetic model

- Data: ignition time; range of initial T & equivalence ratio
- Single-step model:

$$CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O$$

$$\begin{aligned} \mathfrak{R} &= [CH_4][O_2]k_f \\ k_f &= A\exp(-E/R^oT) \end{aligned}$$

- $(\ln A, E) = \sum_k \boldsymbol{\alpha}_k \Psi_k(\boldsymbol{\xi})$



Quality of Uncertain Calibrated Model Predictions

Calibrated uncertain fit model is consistent with the detailed-model data.

Over the range of (T^0, Φ):

- MAP predictive mean ignition-time is centered on the data
- MAP predictive stdv is consistent with the scatter of the data