Inverse Modeling via Bayesian Inference

Khachik Sargsyan

Sandia National Laboratories

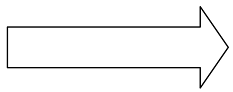
IL-LDRD mtg
April 22, 2025

Uncertainty Quantification in Computational Models

λ

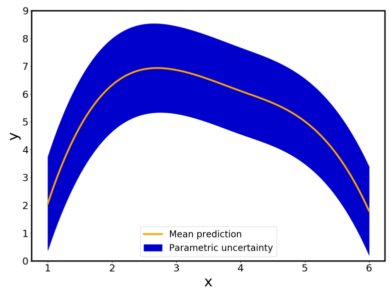
Physical input parameters

Comp. model



$f(x;\lambda)$

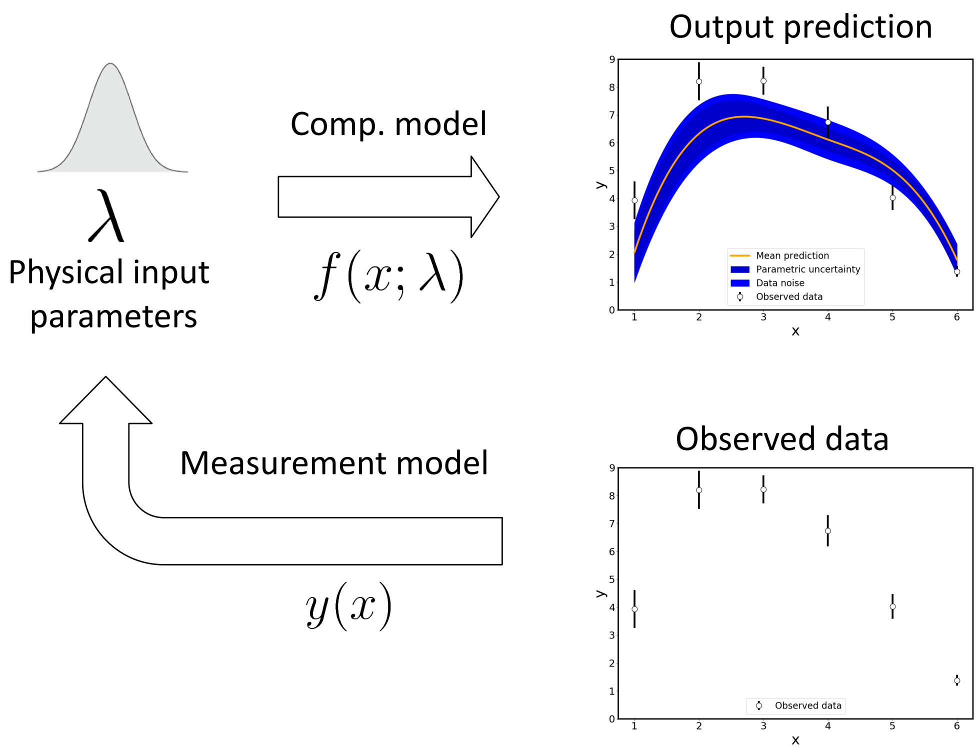
Output prediction



Forward UQ

(Uncertainty propagation, Global sensitivity analysis)

Uncertainty Quantification in Computational Models



Forward UQ

(Uncertainty propagation, Global sensitivity analysis)

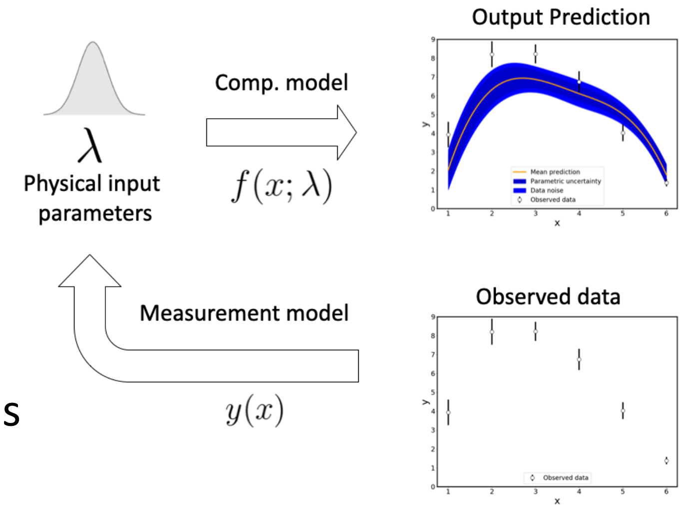
Inverse UQ

(Parameter estimation, Model calibration, validation, Model selection)

TODAY

Inverse UQ

- Compare observational/measurement data with the model
- Tune model parameters (and sometimes model form) to fit the data



- Bayesian methods are best suited for probabilistic inverse problems
  - Meaningful aggregation of various sources of uncertainty
  - Rigorous mathematical footing

Bayes formula

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

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

[Tarantola, 2005]

(Bayesian) Parameter Inference

- Given a model $f(x, \lambda)$ and data $y_i = y(x_i)$, calibrate parameters λ.

  - Linear model $y \approx A\lambda$ with coefficients λ
  - NN model $y \approx NN_\lambda(x)$ with weights/biases λ

- Weighted least-squares fit: $\lambda^* = \text{argmin}_\lambda \sum_{i=1}^{N} w_i^2 \left(f(x_i, \lambda) - y_i\right)^2$

- Bayesian equivalent: $p(\lambda \,|\, y) \propto p(y \,|\, \lambda) p(\lambda) \propto \prod_{i=1}^{N} \exp\left(-\frac{(f(x_i, \lambda) - y_i)^2}{2\sigma_i^2}\right)$

K. Sargsyan (ksargsy@sandia.gov)

Bayes Ingredients

- Prior : knowledge of λ before seeing data (expert opinion, previous analysis, etc…)
- Likelihood : forward model and measurement noise
- Posterior : updated knowledge of λ, combining the prior and the likelihood
- Evidence : normalizing constant, useful for model selection, not for parameter estimation

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

Bayes formula

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

The Prior

Bayes formula

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

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

- The prior comes from physical constraints or prior analysis/data/knowledge
- Prior types: improper prior, objective prior, MaxEnt prior, reference prior, Jeffreys prior
- Prior can be chose 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 (i.e. likelihood) can overrule the prior

The Posterior

Bayes formula

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

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

- Computing full posterior is typically infeasible due to normalizing factor (evidence)
- No closed-form expressions, unless very specialized likelihoods are used
- But, for parameter estimation, it is sufficient to compute the posterior up to a factor
- Explore posterior w/ Markov Chain Monte Carlo (MCMC)
  - Metropolis-Hastings algorithm (a random walk with proposal PDF and accept/reject strategy)
  - Computationally expensive (e.g. $\mathcal{O}(10^5)$ samples)
  - Each sample requires evaluation of the forward model (hence using a pre-trained surrogate)
- Evaluate moments or marginal PDFs from the MCMC statistics

K. Sargsyan (ksargsy@sandia.gov)

The Likelihood

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

Bayes formula

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

- The likelihood requires assumptions regarding model/data relationships

  e.g. Gaussian i.i.d. noise ϵ_i leads to $y_i = f(x_i, \lambda) + \sigma_i \epsilon_i$, where $\epsilon_i \sim \mathcal{N}(0,1)$

  Data Model

- The likelihood becomes essentially a least-squares objective

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

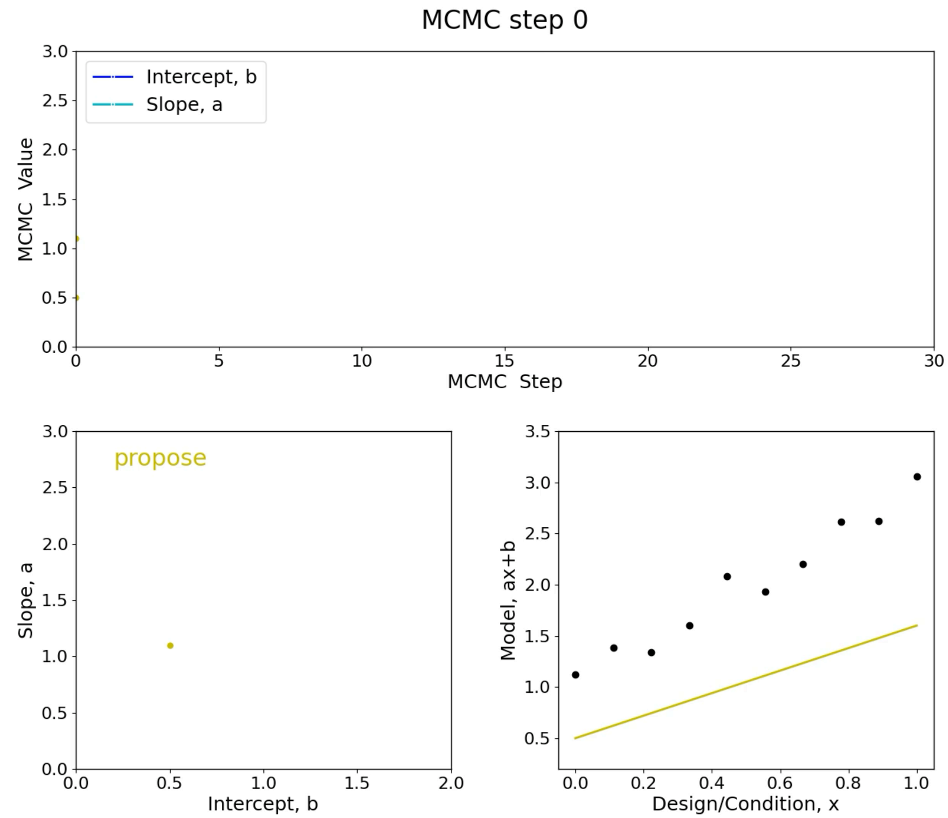
K. Sargsyan (ksargsy@sandia.gov)

Inv UQ: MCMC Example: Line Fit

Linear Model: $f(\overbrace{a,b}^{\lambda};x) = ax + b$

- **Metropolis-Hastings algorithm**
- **Accept/reject mechanism in the parameter space**

- **Generate a random candidate at step t,**
  $\lambda' \sim \pi(\lambda'|\lambda_t)$
- **Calculate the acceptance probability**
  $$\alpha = \min\left(1, \frac{p(\lambda'|y)}{p(\lambda_t|y)} \frac{\pi(\lambda_t|\lambda')}{\pi(\lambda_t|\lambda')}\right)$$
- **Accept with probability α and move to step $t+1$.**



***Note: only posterior ratio matters**

Inv UQ: Bayesian likelihood and MCMC

Bayes formula

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

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

- Denominator $p(y)$ is not important
- Likelihood derived from data model assumptions
- For example, gaussian i.i.d. noise ϵ_i leads to

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

- Markov chain Monte Carlo (MCMC) samples from posterior by marching in the λ-space.
- Likelihood is key:
  - It incorporates statistical assumptions about the discrepancy between model and data.
  - It requires model evaluation at a proposed parameter value λ.

… but it is often infeasible to use model online in an MCMC loop, hence we pre-construct a model surrogate.

The Evidence

Likelihood Prior

$$p(\lambda|y) = \frac{p(y|\lambda)\, p(\lambda)}{p(y)}$$

Posterior

Evidence

- Evidence is not relevant for parameter estimation, but…
- It becomes crucial for model comparison and selection
- Consider a set of models $\{M_1, M_2, \ldots\}$
- Evidence (aka marginal likelihood) is defined as

$$p(y\,|\,M_k) = \int p(y\,|\,\lambda, M_k) p(\lambda\,|\,M_k) d\lambda$$

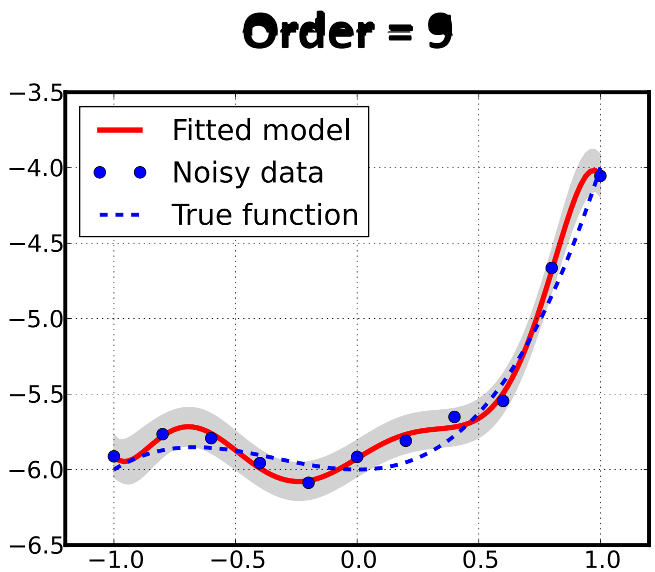
- Compromise between fitting data and model complexity:

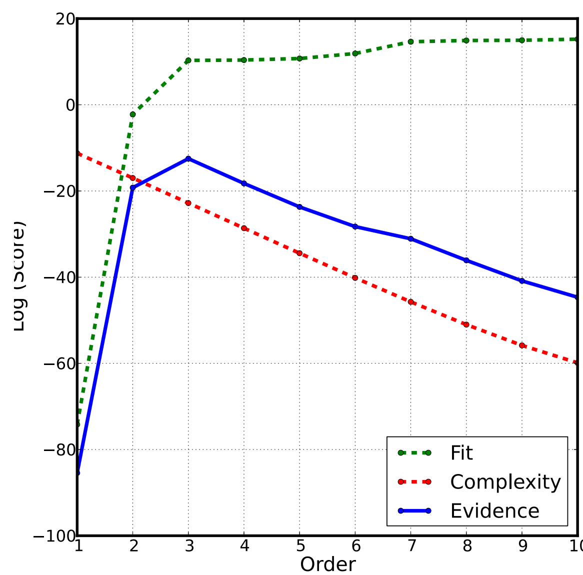
  Occam's razor principle… helps avoid overfitting

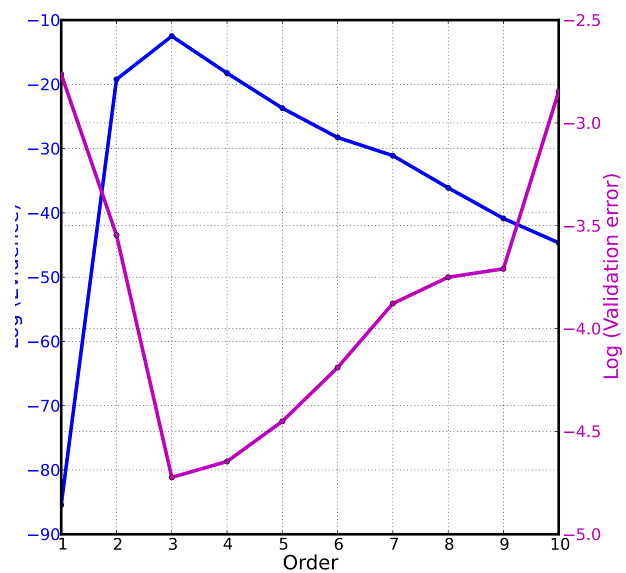
- Model selection: Choose model with maximal evidence
- Model comparison: compute Bayes Factor $BF_{21} = \dfrac{p(y\,|\,M_2)}{p(y\,|\,M_1)}$

Model Evidence = Fit + Complexity

$$\log p(y) = \int \log p(y) p(\lambda|y) d\lambda = \int \log \left[\frac{p(y|\lambda)p(\lambda)}{p(\lambda|y)} \right] p(\lambda|y) d\lambda = \underbrace{\int \log p(y|\lambda) p(\lambda|y) d\lambda}_{Fit} \underbrace{- \int \log \left[\frac{p(\lambda|y)}{p(\lambda)} \right] p(\lambda|y) d\lambda}_{Complexity}$$







K. Sargsyan (ksargsy@sandia.gov)

Linear Models: luxury of analytical solution

$$y \approx A\lambda$$

- Linear least-squares regression

$$\lambda* = \text{argmin}_\lambda ||A\lambda - y||^2$$

- Deterministic solution: $\lambda* = (A^T A)^{-1} A^T y$

- Bayesian posterior PDF… $\lambda_{Post} = \mathcal{N}\left(\lambda*, \sigma^2(\tilde{A}^T A)^{-1}\right)$

Linear Regression: Deterministic-vs-Probabilistic

Given measurements $y \in R^m$, infer signal $\lambda \in R^n$, connected via measurement model $y \approx f(\lambda)$.
E.g. linear model $y = A\lambda + \epsilon$, where $A \in R^{m \times n}$, and $\epsilon \in R^n$ is an iid Gaussian noise.

e.g. CT: x-ray projections from various angular directions; or MRI: spatial Fourier frequencies…..

Deterministic

$$\lambda^* = argmin_\lambda \left[||y - A\lambda||^2 + R(\lambda) \right]$$

Probabilistic/Bayesian

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

$$\lambda^* = argmax_\lambda \log p(\lambda|y) = argmax_\lambda \left[L(\lambda) + \log p(\lambda) \right]$$

- Likelihood is key: it encapsulates the data generation model assumptions (e.g. if correlations)

$$L(\lambda) = \log p(y|\lambda) = \log \pi_N(y - A\lambda) = -\frac{1}{2\sigma^2}||y - A\lambda||^2$$

- Allows exploring the full distribution $p(\lambda|y)$, not just the MAP (max a posteriori) value λ^*
- Allows sequential learning: posterior from one study/probe, feeds to prior in the next study

If $m > n$, this is solved: deterministic: $\lambda^* = (A^TA)^{-1}A^Ty$

probabilistic: $p(\lambda|y) \sim N(\lambda^*, \sigma^2(A^TA)^{-1})$

If $m < n$, there is need for regularization. Or prior, in Bayesian language.
(Bayesian) compressive sensing, LASSO, ….

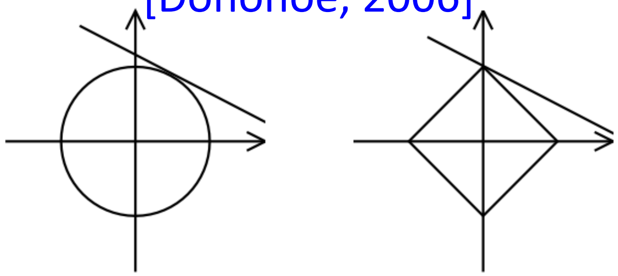
Compressive sensing

Compressive sensing, LASSO, Basis Pursuit:

$$argmin_{\lambda} ||y - A\lambda||_2^2 + ||\lambda||_1$$

Closest Convex apprx.

Roots in sparse signal discovery literature: [Donohoe, 2006]



$$argmin_{\lambda} ||y - A\lambda||_2 \text{ s.t. } ||\lambda||_1 < \epsilon$$

$$argmin_{\lambda} ||\lambda||_1 \text{ s.t. } ||y - A\lambda||_2 < \epsilon$$

Equivalent formulations

Extensions:

- Bayesian Compressive Sensing (BCS): coeffs. with uncertainties, related to relevance vector machine (RVM) [Babacan, 2010], [Sargsyan, 2014],
- Weighted regularization: always better, with judicious choice of weights [Peng, 2013].
- Iterative growth of basis: exploits polynomial structure; increasing the order for the relevant basis terms while maintaining the dimensionality reduction [Jakeman, 2015].

Multiprobe == Multiple Models

$y_1 = A_1\lambda + \epsilon_1$

$\vdots$

$y_K = A_K\lambda + \epsilon_K$

stack them together (fusion)

$$y = \begin{bmatrix} y_1 \\ \hline y_2 \\ \hline \vdots \\ \hline y_K \end{bmatrix} = \begin{bmatrix} A_1 \\ \hline A_2 \\ \hline \vdots \\ \hline A_K \end{bmatrix} \lambda + \begin{bmatrix} \epsilon_1 \\ \hline \epsilon_2 \\ \hline \vdots \\ \hline \epsilon_K \end{bmatrix} = A\lambda + \epsilon$$

- This will alleviate ill-posedness (all the way til we have more data than unknowns?)
- We can incorporate correlations via assumptions on ϵ's

Let's build on this:

- Non-linear models $y_k = f_k(\lambda) + \epsilon_k$: no analytical answer anymore.
  - Resort to Markov chain Monte Carlo (MCMC) sampling? Other tools available… linearization, variational inference, generative models.
- Complex noise structure ϵ_k (IVA merely is independent blocks) is likely non-Gaussian.
  - Is it known? If we can parameterize it, we can infer it!
- Models include further unknown parameters (material properties, geometry): $y_k = f_k(\lambda; \theta_k) + \epsilon_k$
  - Incorporate θ in Bayesian inference: $p(\lambda, \theta \,|\, y) \propto p(y \,|\, \lambda, \theta)\; p(\lambda)p(\theta)$

Inv UQ: Posterior Predictions

- Bayesian inference hinges on likelihood function or data noise (DN) model

  e.g. Gaussian i.i.d. $y_i = f(x_i, \lambda) + \sigma\epsilon_i$, where $\epsilon_i \sim \mathcal{N}(0,1)$

- After we obtain posterior PDF $p(\lambda \,|\, y)$, there are two useful predictive quantities:

  Push-forward (PF): $p(f \,|\, y)$

  Posterior predictive (PP): $p(y^* \,|\, y) = \int p(y^* \,|\, \lambda) p(\lambda \,|\, y) d\lambda$

PP = PF + DN

→ Mean_PP = Mean_PF

→ Var_PP = Var_PF + σ^2

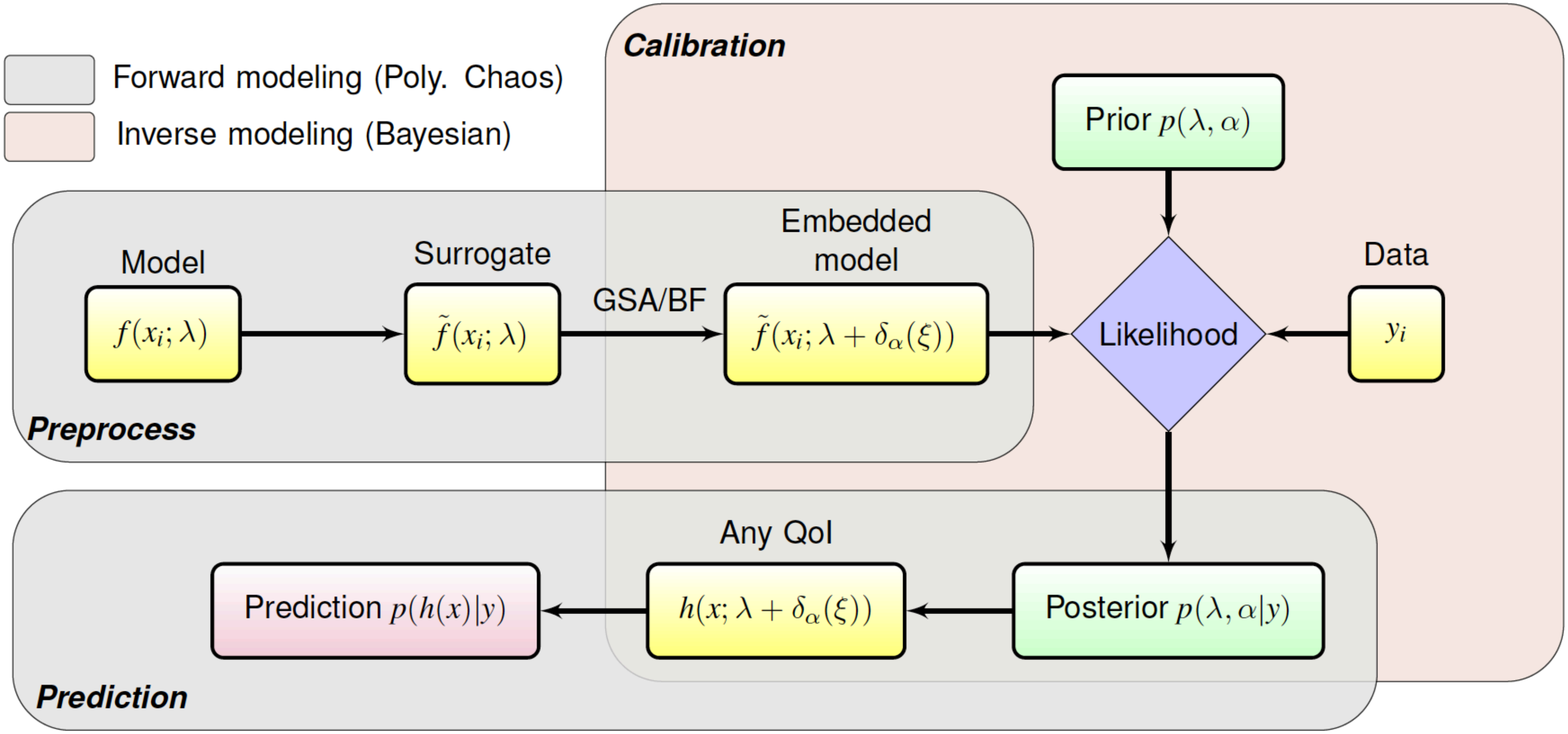
Inv UQ: Challenges

- Model $f(\lambda)$ is expensive
  - Prebuilt and use a surrogate (forward UQ exercise)

- Model $f(\lambda)$ is assumed perfect
  - Can not ignore model structural error
  - Incorporate and correct for model deficiencies

- Model input is high-dimensional: $f(\lambda) = f(\lambda_1, \ldots, \lambda_d)$ for $d \gg 1$
  - MCMC has hard time traveling the high-d posterior surface

Inv UQ: Surrogate-based Bayesian Inference with Embedded Model Error



Inv UQ: High-dimensionality (scalability) challenge

- Starting point of MCMC becomes important. Multiple chains. Tempering to explore/exploit.
- Multimodality, ridges (low-d manifolds with similar posterior values) in posterior shape.
- Important to use good proposal distributions.
- Infer only the most sensitive inputs. Forward UQ / GSA as a preprocessing step.

- MCMC Flavors
  - ✓ Langevin MCMC
  - ✓ Hamiltonian MCMC
  - ✓ Adaptive MCMC
  - ✓ Transitional MCMC
  - ✓ Likelihood-informed subspace / Dimension-Independent MCMC, see [Cui, 2016].

- Variational approximations, see [Blei, 2017].
- Transport maps to directly map prior to posterior, see [Parnot, 2018].
- Amortized inference (pre-build a map from data to posterior), see [Ganguly, 2023].
- Approximate Bayesian computation, see [Sunnåker, 2013].

UQ Software: incomplete list

- DAKOTA: UQ, Optimization and more.
  Targeted for High Performance Systems
  https://dakota.sandia.gov
- UQTk: A C/C++ library for a range of UQ workflows
  https://www.sandia.gov/uqtoolkit
- PyTUQ: Python library for a range of UQ workflows
  https://github.com/sandialabs/pytuq

SciDAC Scientific Discovery through Advanced Computing

FASTMATH

- PyApprox https://github.com/sandialabs/pyapprox
- MUQ https://mituq.bitbucket.io/
- OpenTURNS https://openturns.github.io/www/
- UQPy https://github.com/SURGroup/UQpy
- UQLab https://www.uqlab.com/
- ChaosPy https://github.com/jonathf/chaospy
- PSUADE https://github.com/LLNL/psuade
- CUQIpy https://cuqi-dtu.github.io/CUQIpy/
- ….

PyTUQ

- PyTUQ: Python library for a range of UQ workflows
  https://github.com/sandialabs/pytuq

- Released in March 2025
- Python successor of UQTk
- Member of software stack of (re-)newly proposed SciDAC FASTMath institute

Range of basic and advanced UQ tools and workflows, such as

- Surrogate construction
- Global sensitivity analysis
- Karhunen-Loeve decomposition
- Uncertainty propagation
- Polynomial chaos
- Rosenblatt transformation
- Bayesian linear regression
- MCMC flavors (adaptive MCMC, Hamiltonian MCMC)
- Gaussian processes
- Bayesian compressive sensing
- Evidence computation
- Embedded model error

Inverse UQ: Summary

- Parameter estimation / model calibration / inverse modeling
- Bayesian inference is a major tool
- Do not confuse with term 'inference' in ML!

Questions to consider:

- How expensive is the model (can we afford MCMC or pre-build surrogate)?
- How trustworthy is the model (model structural error)?
- What is the data measurement noise model (build the likelihood)?
- Any prior knowledge or regularization?
- How many parameters to tune (advanced MCMC methods to operate in low-d)?

Literature: Inverse UQ

- A. Tarantola, "Inverse Problem Theory and Methods for Model Parameter Estimation", SIAM, (2005).
- D.S. Sivia and J. Skilling "Data Analysis: A Bayesian Tutorial 2nd Edition”, Oxford U Press, (2008).

- Y. Marzouk, H. Najm, “Dimensionality Reduction and Polynomial Chaos Acceleration of Bayesian Inference in Inverse Problems”, J. Comp. Phys., 228:6, 1862-1902, (2009).
- R. Baptista, Y. Marzouk, O. Zahm, "On the Representation and Learning of Monotone Triangular Transport Maps", Foundations of Computational Mathematics, 24, 2063-2108, (2024)
- M. Parno, Y. Marzouk, “Transport Map Accelerated Markov Chain Monte Carlo”, SIAM/ASA Journal on Uncertainty Quantification, 6:2, (2018).
- M. Sunnåker, A. Busetto, E. Numminen, J. Corander, M. Foll, C. Dessimoz, “Approximate Bayesian Computation”. PLoS Comput Biol 9(1): e1002803 (2013).
- A. Ganguly, S. Jain, U. Watchareeruetai, "Amortized Variational Inference: A Systematic Review", Journal of Artificial Intelligence Research 78, 167-215 (2023).
- D. Blei, A. Kucukelbir, J. McAuliffe, "Variational Inference: A Review for Statisticians", Journal of the American Statistical Association, 112:518, 859-877 (2017).
- T. Cui, K. Law, Y. Marzouk, "Dimension-independent likelihood-informed MCMC", Journal of Computational Physics, 304, 109-137 (2016).

- 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, 246-276, (2015).
- K. Sargsyan, X. Huan, H. Najm. “Embedded Model Error Representation for Bayesian Model Calibration”, Int. J. Uncertainty Quantification, 9:4, (2019).

Trunk

Linear Models: fixed σ

- Gaussian likelihood with fixed σ

$$p(y\,|\,c) \propto \frac{1}{\sigma^N}\prod_{i=1}^{N}\exp\left(-\frac{(\tilde{A}c)_i - \tilde{y}_i)^2}{2\sigma^2}\right)$$

- Leads to Gaussian posterior PDF

$$p(c\,|\,y) \propto p(y\,|\,c)p(c) \sim \mathcal{N}\left((\tilde{A}^T\tilde{A})^{-1}\tilde{A}^T y, \sigma^2(\tilde{A}^T\tilde{A})^{-1}\right)$$

- As well as Gaussian push-forward and posterior predictive

Linear Models: inferred σ

- Gaussian likelihood with inferred σ $$p(y\,|\,c,\sigma^2) \propto \frac{1}{\sigma^N}\prod_{i=1}^{N}\exp\left(-\frac{(\tilde{A}c)_i - \tilde{y}_i)^2}{2\sigma^2}\right)$$

- Leads to normal-inverse-gamma posterior PDF

$$p(c,\sigma^2\,|\,y) \propto p(y\,|\,c,\sigma^2)p(c,\sigma^2) \sim NIG$$

- … but only its marginals are interesting/useful

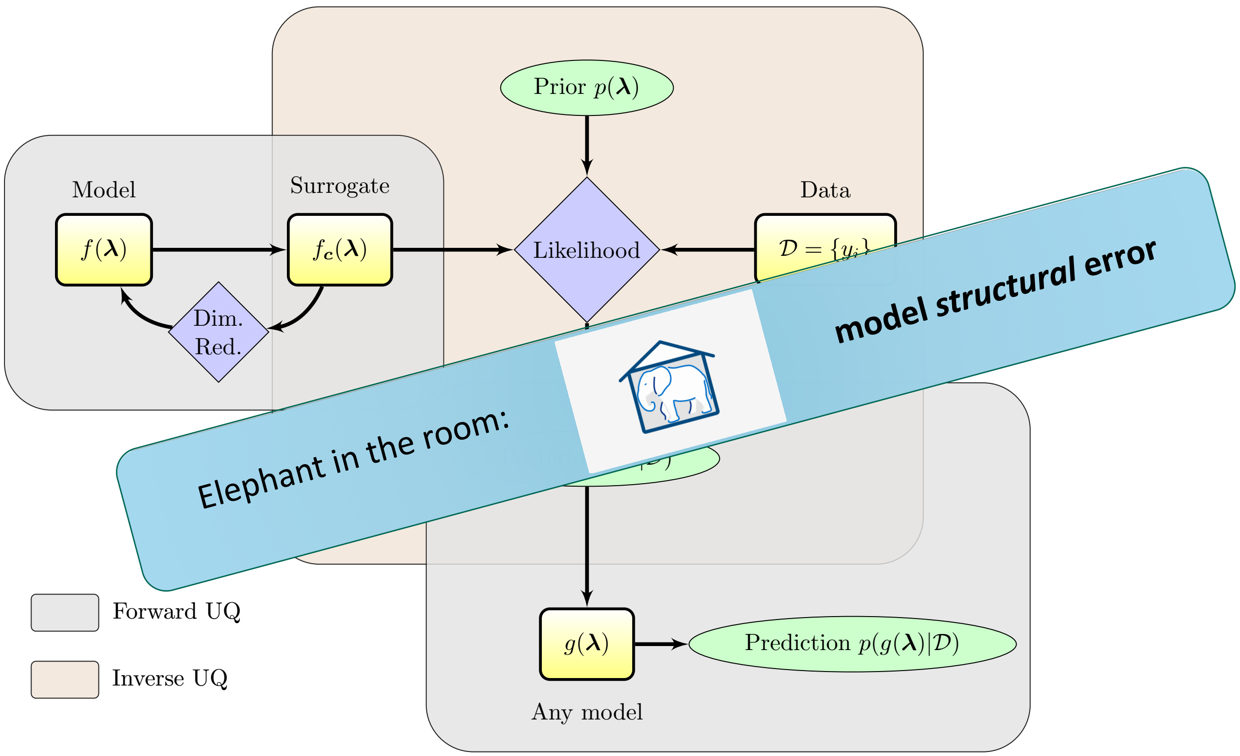
$$p(c\,|\,y,\sigma^2) \sim \mathcal{N}\left((\tilde{A}^T\tilde{A})^{-1}\tilde{A}^Ty, \sigma^2(\tilde{A}^T\tilde{A})^{-1}\right)$$

$$p(\sigma^2\,|\,y) \sim IG\left(\frac{N-K}{2}, \frac{N-K}{2}\hat{\sigma}^2\right)$$

$$p(c\,|\,y) \sim St\left((\tilde{A}^T\tilde{A})^{-1}\tilde{A}^Ty, \sigma^2(\tilde{A}^T\tilde{A})^{-1}, N-K\right)$$

Effective stdv. = residual RMSE

Inv UQ: Surrogate-enabled Bayesian inference



+ **Model Structural Error**

+ **Surrogate Error**

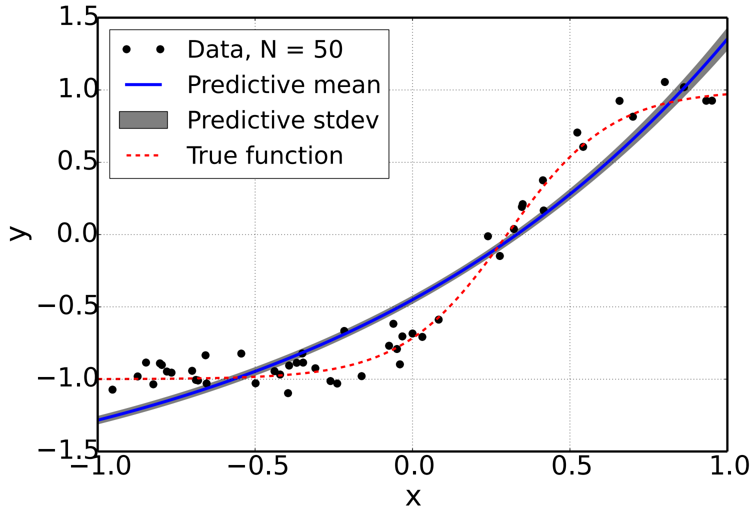
Prediction Variance = **Parametric Uncertainty** + **Data Noise**

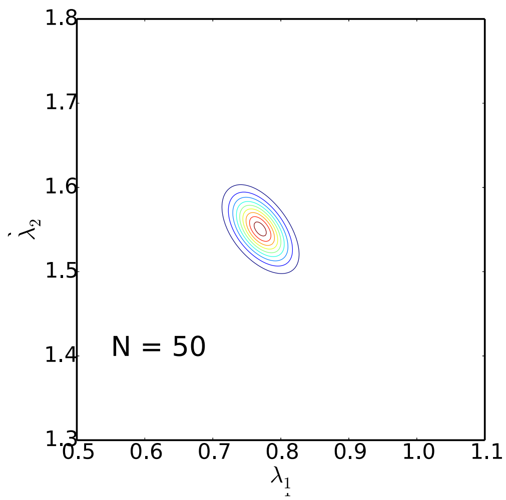
Inv UQ: Model Error

Model error: otherwise called (with slightly altered meanings):

model discrepancy, model structural error, model inadequacy, model misspecification, model form error, model uncertainty.

Ignoring model error leads to overconfident and biased predictions





Inv UQ: How to Correct for Model Error

External: Conventional statistical Gaussian Process correction [Kennedy, O'Hagan, 2001]

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

Challenges when it comes to physical models.

Embedded Intrusive: Model-specific corrections: changing the model/code

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

Embedded Non-Intrusive: Model-agnostic: cast deterministic model parameter as a random variable [Sargsyan, 2019]

$$g(x_i) = f(\lambda + \delta_\alpha(x_i), x_i) + \epsilon_i$$

Inv UQ: Case for Embedded Model Error

- Allows meaningful extrapolation
- Respects physics
- Disambiguates model and data errors
- Predictive uncertainty attribution

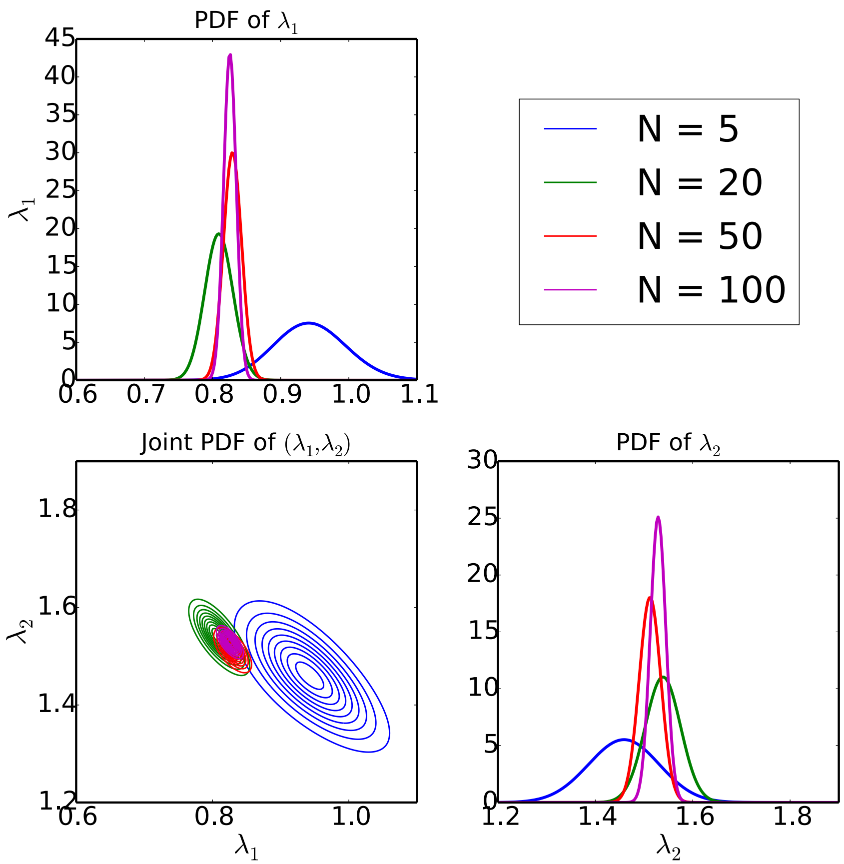
$$g(x_i) = f(\lambda + \delta_\alpha(x_i), x_i) + \epsilon_i$$

- Infer physical parameters λ and model-error parameters α together
- In practice, cast the parameters as a PC expansion $\lambda = \sum_k \alpha_k \Psi_k(\xi)$
- Propagate (forward UQ) PC $f\left(\sum_k \alpha_k \Psi_k(\xi), x_i\right) \approx \sum_k f_k(\alpha)\Psi_k(x_i)$
- Build approximate likelihood functions based on data model $g(x_i) = \sum_k f_k(\alpha)\Psi_k(x_i) + \epsilon_i$
  (e.g. match moments, or Gaussian iid approximation)

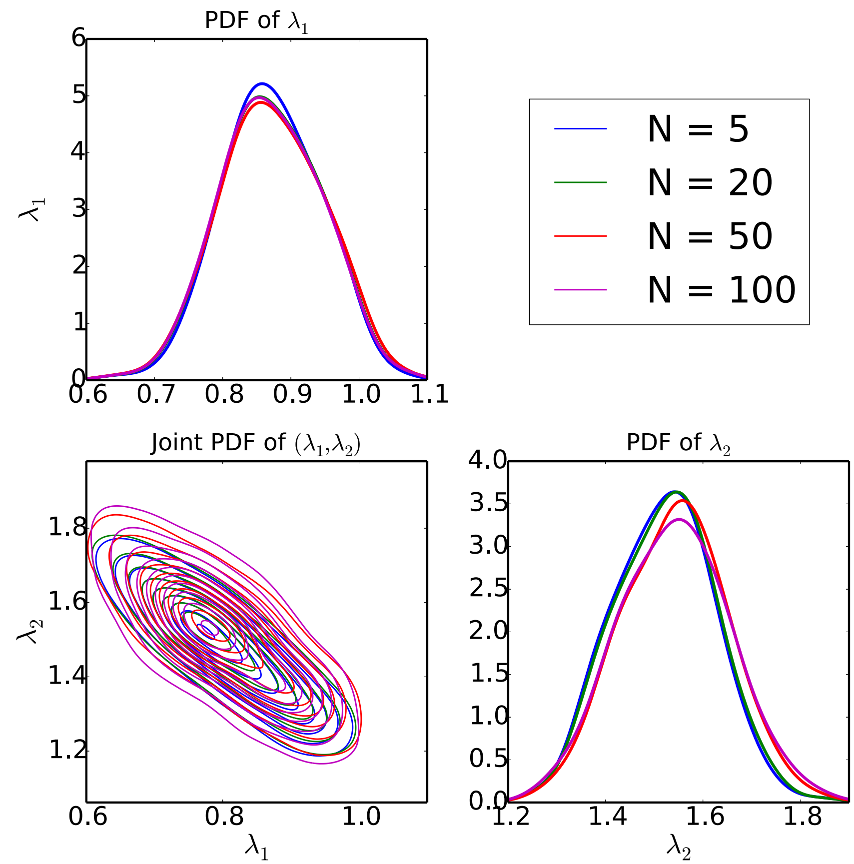
Inv UQ: Embedded Model Error

Predictive uncertainty of “physical” parameters

Without model error



With model error



K. Sargsyan (ksargsy@sandia.gov)

Further notes/questions/challenges:

- BIE: uses Bayesian paradigm, but largely interested in MAP value x^* only (via deterministic optimization).
  - Q: are we interested in the full posterior distribution?
  - What do we do with it? Propagate thru other models? Make better decisions?

- Adjoints useful for optimization, but similarly useful for MCMC if we are after full uncertainties.
  - Q: do we have adjoints? I.e. gradient of the model $f(x; \theta)$ with respect to x and θ.

- Inputs (x, θ) and output y are high-dimensional:
  - use SVD (linear) compression and work in the latent space.
  - nonlinear compression (manifold learning)

- Models are expensive:
  - Prebuilt surrogates (mathematical form, poly or NN…), ROMs (projection, Q: is it always available?), multifidelity.
  - Big Q: what do these models look like? Shared inputs? Shared outputs? Nestedness?

- Models themselves are wrong:
  - Incorporate model deficiency: external (KOH/Higdon) or embedded (S.).
  - Basically hierarchical Bayes.