

# Bayesian Computer Modeling with Emulators

## What are the odds?

Robert L. Wolpert

Department of Statistical Science  
 Nicholas School of the Environment  
 Duke University

Thu, 2011 Jul 07

## Main idea

1. Observe (maybe vector)  $X \in \mathcal{X}$  whose probability density function

$$X \sim p(x | \theta)$$

depends on an uncertain (maybe vector)  $\theta \in \Theta$

2. Want to predict other quantities  $Y \in \mathcal{Y}$  whose distribution

$$Y \sim p(y | \theta)$$

depends on the same  $\theta \in \Theta$

3. While being honest about the uncertainty:

$$Y | X \sim p(y | X = x) = \int_{\Theta} p(y | \theta) p(\theta | x) d\theta$$

## One-Dimensional (Toy) Example

Let  $X$  be the number of successes in  $n = 100$  independent trials, all with the same (unknown) probability  $\theta$  of success.

Suppose we observe  $X = 68$ .

- ▶ Could  $\theta$  be as low as 0.10?
  - ◁ Probably not...  $P[X \geq 68 | \theta = 0.10] = 2.7 \cdot 10^{-45}$ .
  - ▶ The MLE is  $\hat{\theta} = 0.68$ . Does  $\theta$  equal 0.68?
  - ◁ Probably not...  $P[X = 68 | \theta = 0.68] < 0.10$ .
- And  $P[\theta = 0.68 | X = 68] = 0$ .

## The Truth is Revealed...

**Observe:**  $X = 68$   
**Believe:**  $X \sim \text{Bi}(n, p)$   
**Know:**  $n = 100$   
**Unaware:**  $\theta = 0.60$      $\leftarrow \leftarrow \leftarrow \leftarrow$

Likelihood (also  $\chi^2$ ):

$$L(\theta) = \binom{100}{68} \theta^{68} (1 - \theta)^{32} \qquad Q(\theta) = \frac{(100\theta - 68)^2}{100\theta}$$

both optimized at  $\hat{\theta} = 0.68$ . **Is that  $\theta$ ?**

No. Remember? Data were generated with (unknown!)  $\theta = 0.60$ .

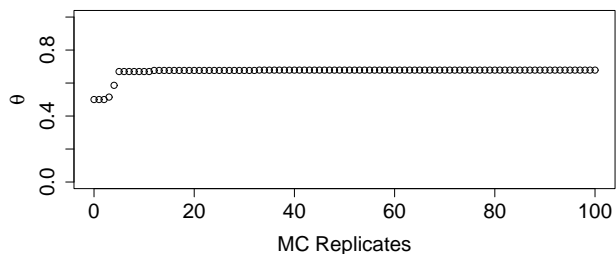
## Stochastic Optimization

Fix  $\theta_0 = 1/2$  (or any other arbitrary point) and, at replicate  $t \in \mathbb{N}$ ,

- ▶ Propose  $\theta^* \sim q(\theta^* | \theta_{t-1})$  (SRW);
- ▶ Accept proposal if  $L(\theta^*) \geq L(\theta_{t-1})$ ; Otherwise stay put.

Results of  $10^2$  replicates with  $q = \text{Un}(-0.10, 0.10)$  RW steps:

**Stochastic Optimization Sequence**



Unchanged after first 33 steps, with  $\hat{\theta} \approx 0.6784$ . Is that  $\theta$ ? No.

## Stochastic Optimization

It's an approximation to the posterior mode,

$$\tilde{\theta} = \operatorname{argmax} L(\theta) = 0.68$$

That's a data-based quantity, which might be far from  $\theta$ .

## Bayesian Posterior

Fix  $\theta_0 = 1/2$  (or any other arbitrary point) and, at replicate  $t \in \mathbb{N}$ ,

- ▶ Propose  $\theta^* \sim q(\theta^* | \theta_{t-1})$  (SRW);
- ▶ Accept proposal with probability  $1 \wedge H$  for Hastings Ratio

$$H = \frac{\pi(\theta^*) L(\theta^*) q(\theta | \theta^*)}{\pi(\theta) L(\theta) q(\theta^* | \theta)}$$

Note the difference to Stochastic Optimization—

$L(\theta^*) \geq L(\theta)$ : SO Accepts  $H \geq 1, \Rightarrow$  MCMC Accepts

$L(\theta^*) < L(\theta)$ : SO Rejects  $H < 1, \Rightarrow$  MCMC MIGHT accept

After  $m = 100$  MCMC steps,  $\bar{\theta}_m = 0.6621$  with sd 0.0443.

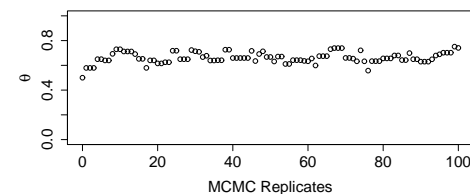
Is that  $\theta$ ?

Yes!

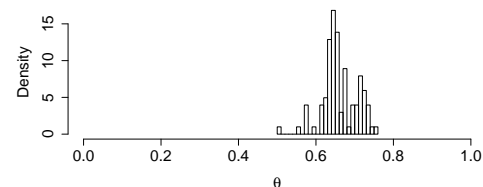
Why?

## Bayesian Posterior, $m = 10^2$ MCMC Steps:

**MCMC Posterior Stream for Bayes Model**

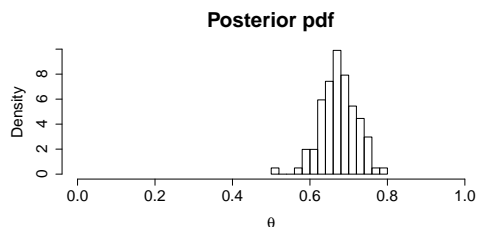
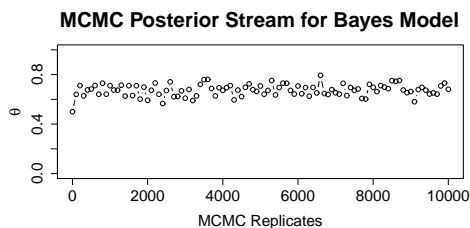


**Posterior pdf**



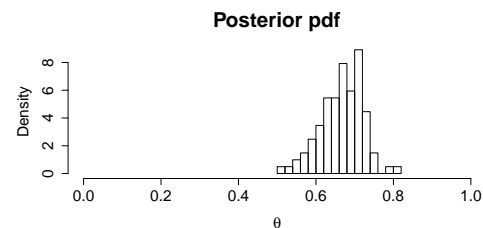
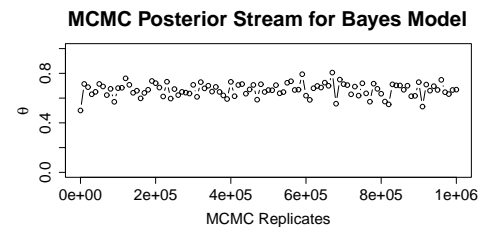
Mean  $\pm 2$  S.D. =  $0.6621 \pm 0.0886$

## Bayesian Posterior, $m = 10^4$ MCMC Steps:



Mean  $\pm 2$  S.D. =  $0.6734 \pm 0.0963$

## Bayesian Posterior, $m = 10^6$ Steps:



Mean  $\pm 2$  S.D. =  $0.6673 \pm 0.1092$

Why aren't the intervals getting narrower???

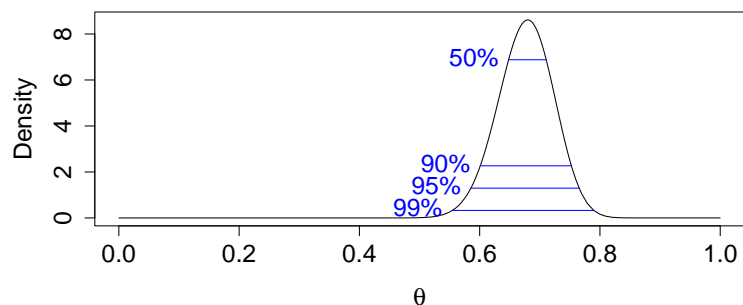
## Bayesian Posterior, $m = \infty$ MCMC Steps:

Q: Why don't the intervals home in on the truth as  $n \rightarrow \infty$ ?

A: Because they *shouldn't*!

We only *observed*  $x = 68$  successes in  $n = 100$  trials.

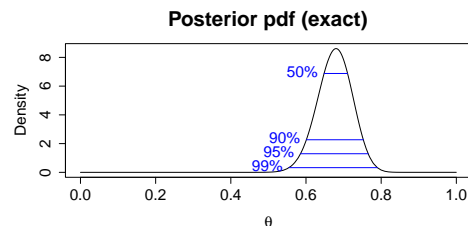
Posterior pdf (exact)



Exact Mean  $\pm 2$  S.D. =  $69/102 = 0.6764706 \pm 0.0922$

Recall— True  $\theta = 0.60$  was used to generate data!

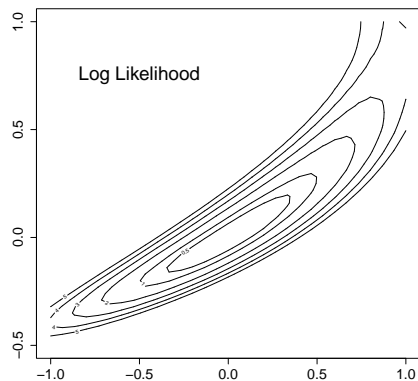
## What is Playful about this Toy Example?



- ▶ In One-Dimension, “Highest Posterior Density” (HPD) regions are **Intervals**;
- ▶ HPD Intervals are more useful than optima—
  - q=50%: [0.6482, 0.7107] Doesn't contain  $\theta = 0.60$
  - q=90%: [0.6011, 0.7525] Doesn't contain  $\theta = 0.60$
  - q=95%: [0.5855, 0.7655] Does contain  $\theta = 0.60$
  - q=99%: [0.5548, 0.7898] Does contain  $\theta = 0.60$
- ▶ In 2+ Dim, HPDs are **High-dimensional bananas**.

## Typical Log LLHs

At best the log likelihood will have banana-shaped contours:



with LOTS of (nearly) equally-likely  $\theta$ 's,  
and "TRUTH" unlikely to be near center!

## Optimize?

One idea... optimize and hope for the best:

Perhaps

$$Y | X = x \sim p(y | \hat{\theta}(x))$$

Understates uncertainty; gives wrong answer.

## Integrate?

Another idea... obey the **Laws of Probability**

$$p(x, \theta) = p(x | \theta)p(\theta)$$

$$p(x) = \int_{\Theta} p(x | \theta)p(\theta) d\theta$$

$$p(\theta | x) = \frac{p(x, \theta)}{p(x)} = \frac{p(x | \theta)p(\theta)}{\int_{\Theta} p(x | \vartheta)p(\vartheta) d\vartheta}$$

$$p(y | x) = \int_{\Theta} p(y | \theta) p(\theta | x) d\theta$$

One problem: Where did the marginal density  $p(\theta)$  come from?

## Prior Distributions:

Where does  $p(\theta)$  come from?

Historical experience, informed opinion, or ignorance:

- ▶ **Historical Experience:** If we've seen 200 successes in 300 previous trials, we might use  $p(\theta) = \text{Be}(200, 100)$  prior;
- ▶ **Informed Opinion:** If a subject-matter expert tells us s/he is certain  $0.5 < \theta < 0.75$  but unsure within that interval, we might use  $p(\theta) = \text{Un}(0.50 - \epsilon, 0.75 + \epsilon)$ ;
- ▶ **Ignorance:** In the absence of other information, we might use a conventional choice of  $p(\theta) \sim \text{Un}(0, 1)$  or  $p(\theta) \sim \text{Be}(\frac{1}{2}, \frac{1}{2})$ .

## What if the Integrals are hard in low dimensions?

### 1. Euler, Simpson, or Runge-Kutta Quadrature:

$$p(y | x) = \frac{\int p(y | \theta)p(x | \theta)p(\theta) d\theta}{\int p(x | \vartheta)p(\vartheta) d\vartheta} \approx \sum p(y | \theta_j)w_j$$

Error  $\epsilon \asymp n^{-4/d}$  for  $\Theta \subseteq \mathbb{R}^d$  so need  $n = O(\epsilon^{-d/4})$  evaluations;  
 OK in dimension  $d \leq 4$  for Simpson,  $d \leq 8$  for RK4.

## What if the Integrals are *still* too hard?

### 3. Metropolis/Hastings Markov Chain Monte Carlo:

$$p(y | x) \approx \frac{\sum p(y | \theta_j)}{N}, \quad \{\theta_j\} \sim p(\theta | x)$$

$$\theta_{j+1} = \begin{cases} \theta_j^* & \text{w/prob } [1 \wedge H_j] \\ \theta_j & \text{otherwise} \end{cases}$$

$$\theta_j \rightsquigarrow \theta_j^* \sim g(\theta_j^* | \theta_j), \quad H_j = \frac{p(\theta_j^*) g(\theta_j | \theta_j^*)}{p(\theta_j) g(\theta_j^* | \theta_j)}$$

Works well in high-dimensional problems (despite autocorrelation).  
 Just a technique for approximating integrals— *not* optimization.

## What if the Integrals are harder?

### 2. Monte Carlo Importance Sampling:

$$p(y | x) = \frac{\int p(y | \theta)p(x | \theta)p(\theta) d\theta}{\int p(x | \vartheta)p(\vartheta) d\vartheta} \approx \frac{\sum p(y | \theta_j)w_j}{\sum w_j},$$

$$\theta_j \stackrel{\text{iid}}{\sim} f(\theta) d\theta, \quad w_j = \frac{p(x | \theta_j)p(\theta_j)}{f(\theta_j)}$$

Error  $\epsilon \asymp n^{-1/2}$  for any dimension  
 Hard to find good “importance function”  $f(\theta)$  if  $d$  big.

## Deterministic Computer Models

How can **Probability** help us with  
*Deterministic* computer models?

## What's *random* about a deterministic computer model?

Well... our *uncertainty* about it! Here's an analogy:

- ▶ Find the book closest to you—
- ▶ How long is 2<sup>nd</sup> text line on page 50? Call length “X”
- ▶ Hmmm. 10pt fonts ⇒ ≈ 12 char/in; width ≈ 4–5in...
- ▶ Perhaps... 50–60 chars, w/prob 0.50?
- ▶ Uh oh, paragraph could end... 15–70 chars, w/prob 0.75?
- ▶ Answer (for me):  $X = 61$  (Doob), so *now*  $P[X = 61] = 1$ .

Observation changes probabilities!

## Computer Models and Reality

Denote by  $Y^F(\mathbf{x}_j)$  the value of a **Field** measurement taken at settable and observable input value  $\mathbf{x}_j$ . Typically this will differ by some **measurement error**  $\epsilon_j^F$  from the **Real** value:

$$Y^F(\mathbf{x}_j) = Y^R(\mathbf{x}_j) + \epsilon_j^F, \quad \epsilon_j^F \stackrel{\text{iid}}{\sim} \text{No}(0, 1/\lambda^F).$$

Our computer **Model** will include additional **tuning parameters** and, alas, may also exhibit bias or **discrepancy**:

$$Y^R(\mathbf{x}) = Y^M(\mathbf{x}, u) + \delta_u(\mathbf{x}).$$

We are uncertain about the:

- ▶ Tuning parameter  $u$  (let  $u_*$  be the “right” value);
- ▶ Field measurement-error precision  $\lambda^F$ ;
- ▶ Entire discrepancy function  $\delta_u(\mathbf{x})$ .

## Uncertainty for Complex Models

$$Y^F(\mathbf{x}_j) = Y^R(\mathbf{x}_j) + \epsilon_j^F, \quad Y^R(\mathbf{x}) = Y^M(\mathbf{x}, u) + \delta_u(\mathbf{x}).$$

Complex model uncertainty is much more than

**measurement error** and **systematic errors**.

In studying the “Galform” model for Galaxy Formation, Vernon, Goldstein & Bower (2004) identify **five aspects**:

- |  |  |
|--|--|
| 1. <b>Observational error</b>                                | $\epsilon_j^F = ???$                         |
| 2. <b>Parameter uncertainty</b>                              | $\mathbf{x}, u = ???$                        |
| 3. <b>Simulator uncertainty</b>                              | $Y^M(\mathbf{x}, u) = ???$                   |
| 4. <b>Structural uncertainty</b>                             | $\delta_u(\mathbf{x}) = ???$                 |
| 5. <b>Initial condition and forcing function uncertainty</b> | All the other variables & missing physics... |

Want to see a modest (3-dim)

## Pedagogic Example

The mass  $y(t)$  of Silane gas  $\text{SiH}_4$  decreases in the manufacture of silicon solar cells through the reaction  $\text{SiH}_4 \rightarrow \text{Si} + 2\text{H}_2$ ; a plausible model is linear decay:

$$\frac{d}{dt}y(t) = -u y(t), \quad y(0) = y_0$$

for (unknown) reaction rate  $u$  and (known) initial concentration  $y_0$ . The obvious solution to this ODE is our “computer model”

$$Y^M(t, u) = y_0 \exp(-u t)$$

Imagine however that a portion  $c$  remains unreacted, however, so the model is (a little) wrong— the *real* mass is

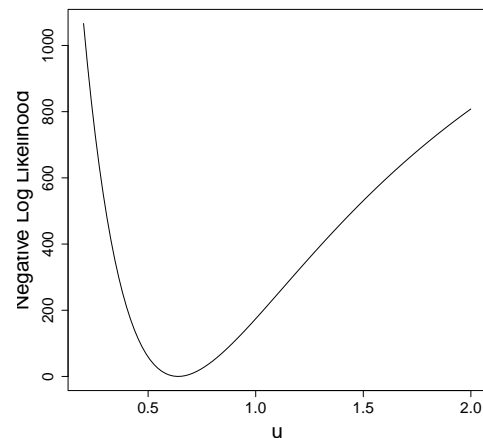
$$Y^R(t) = (y_0 - c) \exp(-u_* t) + c = Y^M(t, u_*) + \delta_u(\mathbf{x}).$$

## Field Data

$t$	$Y^F(t)$		
0.110	4.730	4.720	4.234
0.432	3.177	2.966	3.653
0.754	1.970	2.267	2.084
1.077	2.079	2.409	2.371
1.399	1.908	1.665	1.685
1.721	1.773	1.603	1.922
2.043	1.370	1.661	1.757
2.366	1.868	1.505	1.638
2.688	1.390	1.275	1.679
3.010	1.461	1.157	1.530

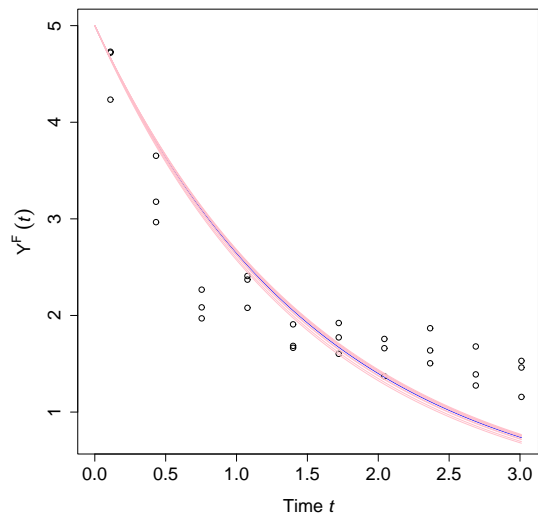
Three noisy replicates at each of 10 field points

## Negative Log Likelihood



Minimum occurs at  $\hat{u} \approx 0.6364$ — how good is model at  $\hat{u}$ ?

## Model Fit and Field Data



Not very!

Maybe there's discrepancy...

## Discrepancy

$$Y^F(\mathbf{x}_j) = Y^M(\mathbf{x}, u) + \epsilon_j^F + \delta_u(\mathbf{x})$$

We were uncertain about the:

- ▶ Tuning parameter  $u$  ( $\hat{u} \approx 0.6364$ );
- ▶ Field m.e. precision  $\lambda^F$  ( $\hat{\lambda}^F \approx 21.74$ );
- ▶ Discrepancy function  $\delta_u(\mathbf{x})$  (probably not zero!).

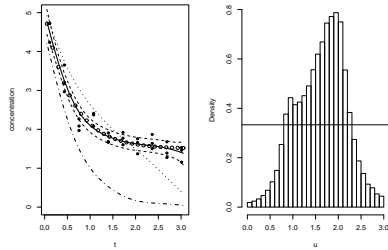
Let's pick independent prior distributions:

- ◁  $u \sim \text{Un}(0.2, 2.0)$
- ◁  $\lambda^F \sim \text{Ex}(\frac{1}{5 \times 21.74})$
- ◁  $\delta_u(\cdot) \sim \text{GP}(0, c^B(\cdot, \cdot | \beta_b^y))$

and then find the posterior!

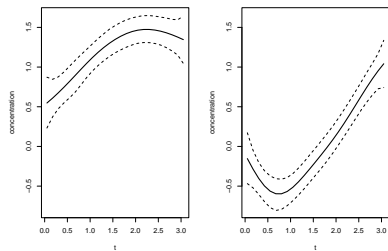
## Correcting for Discrepancy

Discrepancy-corrected predictions of  $Y^F$  with 90% intervals



Post'r Dist'n of  $u$

Post'r dist'n of  $\delta_u(\cdot)$  for  $u = E[u|Dat]$



Post'r dist'n of  $\delta_u(\cdot)$  for  $u = \hat{u}$

## What did we do?

We identified the model discrepancy  $\delta_u(t)$  and learned  $u$ .

For this simulation the Truth was:

$$Y^F(t_i) = 1.5 + (3.5) e^{-1.7 t_i} \pm \epsilon_i$$

$$\delta_{u^*}(t) = Y^R(t) - Y^M(t, 1.7)$$

$$= 1.5(1 - e^{-1.7 t})$$

$$\lambda = 1/(0.3^2) \approx 11.11$$

With these, we were able to correct for the discrepancy and make reliable out-of-sample predictions of  $Y^F(t')$  at unobserved times  $t'$ , with honest reflection of uncertainty.

## How did we do it?

To simulate the posterior distribution of  $u$ ,  $\lambda^F$ , and  $\delta_u(\cdot)$ , at each Monte Carlo step  $m$  we had to:

- ▶ Draw a random  $u^*$ , and consider replacing  $u^{m-1}$  with  $u^*$ ; the "consideration" entails evaluating  $Y^M(t_i, u^*)$  at each field location  $t_i \in \mathcal{D}$ .
- ▶ Draw a new random  $\lambda^{F^m}$  from its complete conditional distribution given  $u^{m-1}$  and  $\delta_u^{m-1}(t)$  (easy).
- ▶ Draw a new random  $\delta_u^m(t)$  from its complete conditional distribution (high-dimensional  $\Rightarrow$  hard if  $|\mathcal{D}|$  is big).

All inside a loop, for  $m = 1$  to  $m = 1,000,000!$

## A way out...

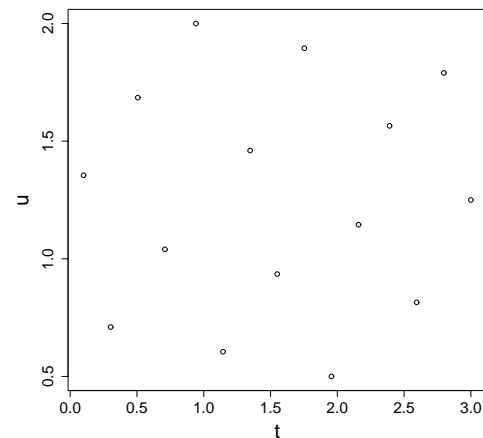
- ▶ For RHIC (for example) it may take 1 hr to evaluate  $Y^M(t_i, u)$
- ▶ We don't have 1,000,000 hours.
- ▶ If evaluating the model  $Y^M(t_i, u)$  at all field locations  $t_i \in \mathcal{D}$  and tuning values  $u$  is too slow, consider an **Emulator**
- ▶ Fit a new GaSP  $Y^E(\mathbf{x}, u) \sim \text{GP}(\text{Mean}, \text{Cov})$  as a **fast surrogate** for  $Y^M(t_i, u)$ , with honest reflection of uncertainty.
- ▶ The "fitting" is done at a carefully chosen set of data values:

## Model Data

$t_i$	$u_i$	$Y^M(t_i, u_i)$	$t_i$	$u_i$	$Y^M(t_i, u_i)$
2.159	1.145	0.422	0.941	2.000	0.761
0.303	0.710	4.032	0.709	1.040	2.392
1.753	1.895	0.180	1.144	0.605	2.502
0.506	1.685	2.132	2.391	1.565	0.118
1.956	0.500	1.880	1.550	0.935	1.174
2.594	0.815	0.604	2.797	1.790	0.033
1.347	1.460	0.700	0.100	1.355	4.366
3.000	1.250	0.118			

Fifteen points  $(t_i, u_i) \in [0, 3] \times [0.5, 2]$ , space-filling LHC design

## Space-filling LHC design



Unlike a lattice, the 1-dim projections are spread widely.

## Model our Uncertainty about $Y^M(\mathbf{x}, u)$ with a GaSP

$Y^M(\mathbf{x}_i, u_i)$  Model output at input (vector)  $\mathbf{x}_i \in \mathcal{X}$ ,  
tuning parameter (vector)  $u_i \in \mathcal{U}$   
Only observed at a few points  $\mathcal{D}^M = \{\mathbf{x}_i, u_i\}_{i \in I}$

$Y^E(\mathbf{x}, u)$  Emulator:  $Y^E(\mathbf{x}, u) \sim \text{GP}(\text{Mean}, \text{Cov})$   
Prior: Mean =  $\Psi'(\mathbf{x}, u)\beta^m$ , Cov =  $c^M(\cdot, \cdot; \beta^v)$   
Post'r: Usual Gaussian updating formulas

Emulator is just an interpolator, *with uncertainty measure*

## A few details...

Mean, Covariance:

Set  $\mathbf{z} = (\mathbf{x}, u)$ ,  $\beta = (\beta^m, \beta^v)$ :

$$E[Y^E(\mathbf{z}) | \mathbf{y}^M, \beta] = \Psi'(\mathbf{z})\beta^m + r_z'[\Gamma^M]^{-1}(\mathbf{y}^M - \Psi'(\mathbf{z}^M)\beta^m)$$

$$V[Y^E(\mathbf{z}), Y^E(\mathbf{z}^*) | \mathbf{y}^M, \beta] = c^M(\mathbf{z}, \mathbf{z}^*; \beta^v) - r_z'[\Gamma^M]^{-1}r_{z^*}$$

where  $\Gamma^M = c^M(\mathbf{z}^M, \mathbf{z}^M; \beta^v)$ ,  $r_z = c^M(\mathbf{z}, \mathbf{z}^M; \beta^v)$ .

Given  $\beta^m, \beta^v$  these are usual Kalman filter formulae.

## A few more details...

Mean & Variance Hyperparameters  $\beta^m, \beta^v$ :

- ▶ **MLE**: Weighted linear regression using Model Data:  
Find  $\hat{\beta}^m, \hat{\beta}^v$  by maximizing Gaussian LLH for:

$$Y^M(\mathbf{z}_i) \sim \text{No}\left(\Psi'(\mathbf{z}_i)\beta^m, c^M(\mathbf{z}_i, \mathbf{z}_j; \beta^v)\right)$$

- ▶ **Bayes**: Specify joint prior dist'n for  $\beta^m, \beta^v$  and integrate.

General consensus is that MLE is good enough approximation at this level of hierarchy.

## Yet more details...

Most common covariance function is isotropic **Power Exponential**:

$$c(\mathbf{z}, \mathbf{z}^*; \beta^v) = \frac{1}{\lambda} \exp\left\{-\sum_i \left|\frac{z_i - z_i^*}{\rho_i}\right|^{\alpha_i}\right\}$$

where  $\beta^v = (\lambda, \{\alpha_i, \rho_i\})$

- ▶  $\lambda > 0$  is (constant) 1/variance of  $Y^E(\mathbf{z})$ ;
- ▶  $\rho_i > 0$  is length scale for  $i$ th component of  $\mathbf{z} = (\mathbf{x}, u)$ ;
- ▶  $\alpha_i \in [1, 2]$  is length scale for  $i$ th component (commonly  $\alpha_i \equiv 1.9$  is posited, not estimated).

**Isotropy** only sensible after suitable transformations.

## Wrap-Up

- ▶ Learning physics from complex models is hard, but
- ▶ **Bayesian Statistics** and meso-scale **Model Emulators** can help.
- ▶ We can use them to help identify **Regions of Parameter Space** that lead to model predictions consistent with observations—
- ▶ Or, if **no part** of parameter space works, that can shine light on model discrepancies and perhaps lead to **new insights**.

The approach has been applied successfully in up to 50 or so dimensions (but try to stay below 20 or so if you can).

**Thanks for your attention and interest!**

wolpert@stat.duke.edu      <http://www.stat.duke.edu/~rlw/>

(or just Google me: **Robert Wolpert**)