The What, Why, and How of Multivariate Emulation

Jonathan Rougier

Department of Mathematics

University of Bristol, UK

http://www.maths.bris.ac.uk/~mazjcr/

May 2009, Spring Research Conference, Vancouver

Two types of experiment

System experiment

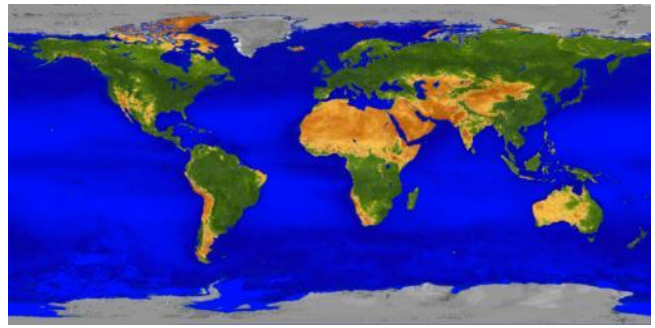
- Multiple experimental units may not be available!
- Many uncontrolled sources of variation
- *Difficulty of doing the experiment you want*

Two types of experiment

System experiment

- Multiple experimental units may not be available!
- Many uncontrolled sources of variation
- *Difficulty of doing the experiment you want*

Computer experiment



- Can do more-or-less any experiments we want
- Have to account for limitations in the model
- *Difficulty of interpreting the experiment you do*

Representing uncertainty

Taking a physical system as an example.

- Denote the system as the (possibly huge) vector Y.
- The model maps some parameters θ into a point or distribution over possible values of Y. Typically θ might comprise
  - Coefficients in the equations;
  - Initial conditions, forcing functions.

Representing uncertainty

Taking a physical system as an example.

- Denote the system as the (possibly huge) vector Y.
- The model maps some parameters θ into a point or distribution over possible values of Y. Typically θ might comprise
  - Coefficients in the equations;
  - Initial conditions, forcing functions.
- Limitations in the model induce uncertainty about the relationship between the system and the model. This takes the form of a joint distribution

$$\pi(Y, \theta^*) = \underbrace{\pi(Y \mid \theta^*)}_{\text{structural}} \times \underbrace{\pi(\theta^*)}_{\text{parametric}}$$

where θ^* is the best/correct/true value of the parameters.

Introducing the Emulator

Often, a large part of the evaluation of $\pi(Y \mid \theta^*)$ comprises the evaluation of a deterministic function $g(\theta)$, termed the simulator.

Emulator, η_θ

An emulator is a framework for making a statistical prediction for $g(\theta)$ at any valid θ, *by conditioning* $g(\theta)$ *on the simulator evaluations*. Denote this prediction as the probability distribution η_θ.

Introducing the Emulator

Often, a large part of the evaluation of $\pi(Y \mid \theta^*)$ comprises the evaluation of a deterministic function $g(\theta)$, termed the simulator.

Emulator, η_θ

An emulator is a framework for making a statistical prediction for $g(\theta)$ at any valid θ, *by conditioning* $g(\theta)$ *on the simulator evaluations*. Denote this prediction as the probability distribution η_θ.

- The emulator can be used to replace evaluations of $g(\theta)$ when these would be too expensive. For example, instead of evaluating $v = g(\theta)$ we could use $\bar{v} = E_{\eta_\theta}(v)$ or $\tilde{v} \sim \eta_\theta$.
- An emulator augments the information in the simulator evaluations with additional judgements about smoothness, and also, if appropriate, about monotinicity, interactions, non-linearity, etc.

Two approaches to computer experiments

Consider an uncertainty analysis experiment, where we want to sample from $\pi(Y)$. Suppose that $\pi(Y \mid \theta^*) = \pi(Y \mid g(\theta^*))$.

Two approaches to computer experiments

Consider an uncertainty analysis experiment, where we want to sample from $\pi(Y)$. Suppose that $\pi(Y \mid \theta^*) = \pi(Y \mid g(\theta^*))$.

In the loop

No time to waste: let's get cracking!

for $i = 1, \ldots, n$ **do**
  $\theta^{(i)} \sim \pi(\theta^*)$
  $v^{(i)} = g(\theta^{(i)})$
  $Y^{(i)} \sim \pi(Y \mid v^{(i)})$
  Save $(\theta^{(i)}, Y^{(i)})$
end for

Result: an *estimate* of $E(Y)$ with an error of $\mathcal{O}(n^{-0.5})$.

Two approaches to computer experiments

Consider an uncertainty analysis experiment, where we want to sample from $\pi(Y)$. Suppose that $\pi(Y \mid \theta^*) = \pi(Y \mid g(\theta^*))$.

In the loop

No time to waste: let's get cracking!

for $i = 1, \ldots, n$ **do**
  $\theta^{(i)} \sim \pi(\theta^*)$
  $v^{(i)} = g(\theta^{(i)})$
  $Y^{(i)} \sim \pi(Y \mid v^{(i)})$
  Save $(\theta^{(i)}, Y^{(i)})$
end for

Result: an *estimate* of $E(Y)$ with an error of $\mathcal{O}(n^{-0.5})$.

Emulator

First, build an emulator η_θ using n carefully chosen evaluations of the simulator, $(G; R)$. Then

for $i = 1, \ldots, N$ **do**
  $\theta^{(i)} \sim \pi(\theta^*)$
  $\tilde{v}^{(i)} \sim \eta_{\theta^{(i)}}$
  $Y^{(i)} \sim \pi(Y \mid \tilde{v}^{(i)})$
  Save $(\theta^{(i)}, Y^{(i)})$
end for

Result ($N \gg n$): an *exact* calculation of $E(Y \mid G; R)$.

Advantages of emulators

1. They allow us to augment the set of n evaluations with additional judgements about the simulator.

Advantages of emulators

1. They allow us to augment the set of n evaluations with additional judgements about the simulator.

2. They provide a framework in which we can explore the behaviour of the simulator (very important for *code verification*).

Advantages of emulators

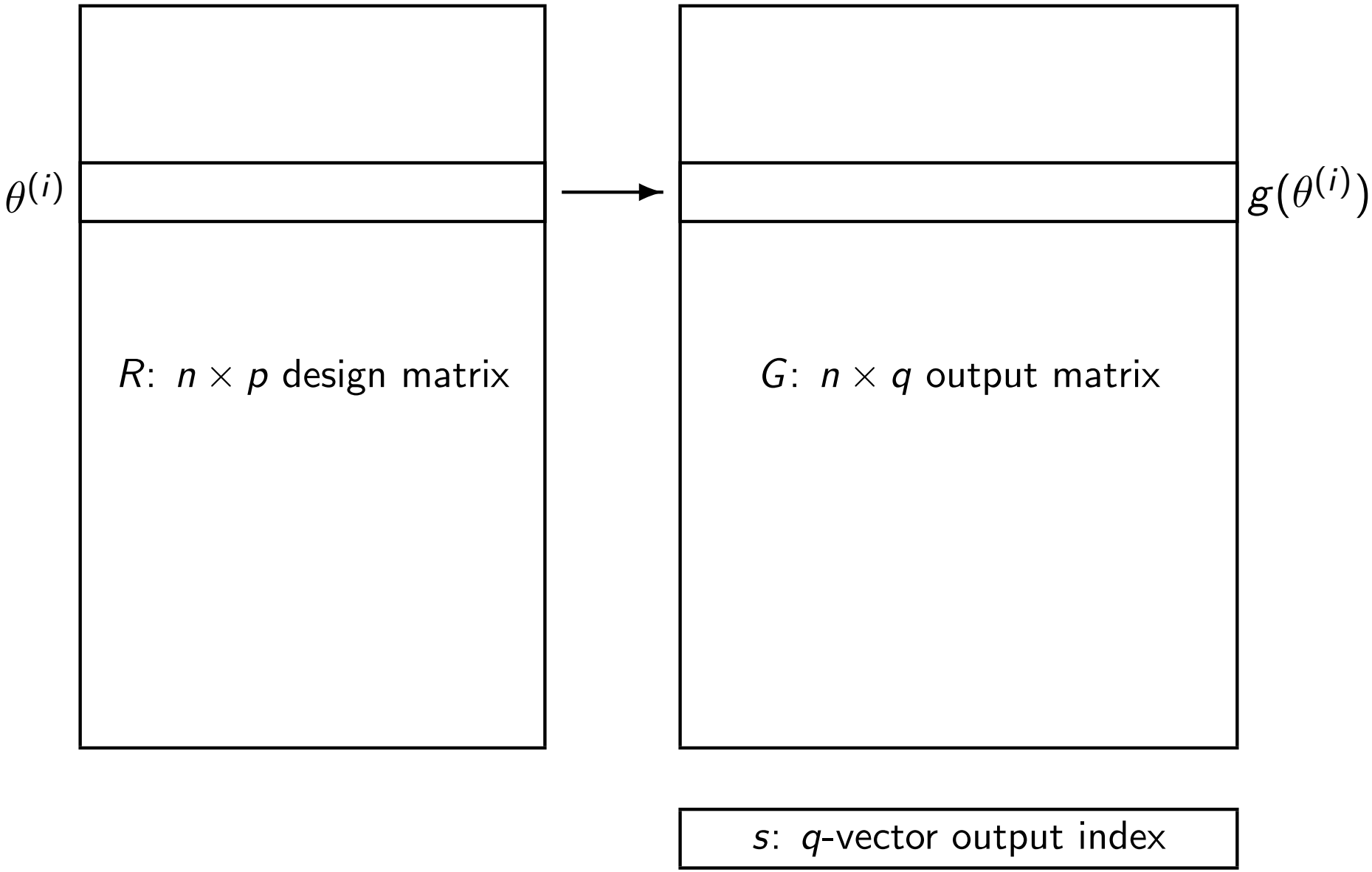
1. They allow us to augment the set of n evaluations with additional judgements about the simulator.

2. They provide a framework in which we can explore the behaviour of the simulator (very important for *code verification*).

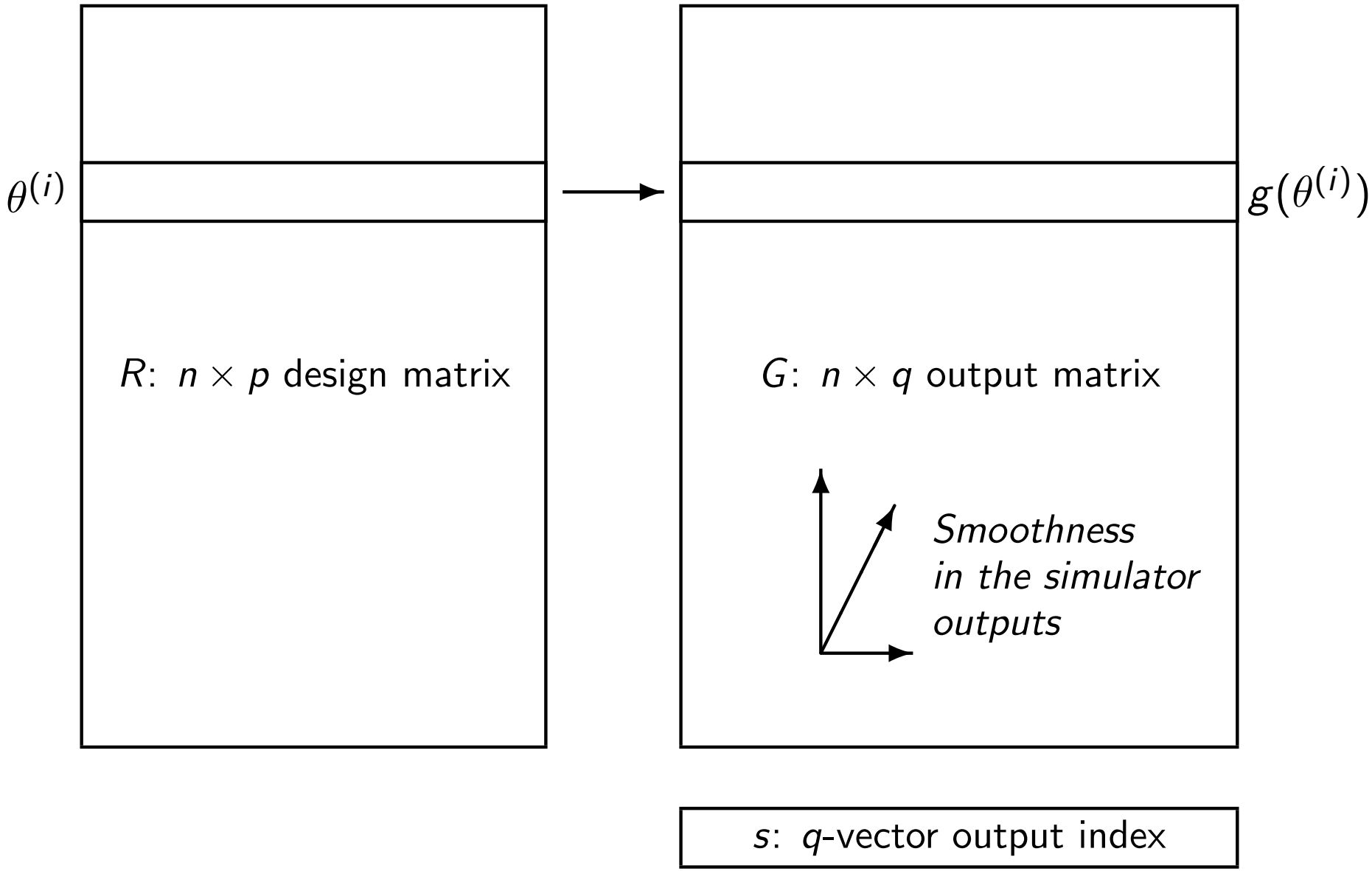
3. They help us to make informative choices for where to evaluate the simulator, and free us from committing to distribution choices for θ^*.

Now it gets messy!

The 'shape' of a MV emulator



The 'shape' of a MV emulator



The NIG implementation

- Emulators typically look quite a lot like regressions:

$$g_j(\theta) = \sum\nolimits_k \beta_k h_k(\theta, s_j) + \varepsilon(\theta, s_j)$$

where $\boldsymbol{\beta}$ comprises uncertain coefficients, $\mathbf{h}$ is specified regressor functions, and ε is a scalar residual process.

The NIG implementation

- Emulators typically look quite a lot like regressions:

$$g_j(\theta) = \sum\nolimits_k \beta_k h_k(\theta, s_j) + \varepsilon(\theta, s_j)$$

  where $\boldsymbol{\beta}$ comprises uncertain coefficients, $\mathbf{h}$ is specified regressor functions, and ε is a scalar residual process.

- The standard conditionally conjugate prior is

$$\begin{aligned} \boldsymbol{\beta} &\perp\!\!\!\perp \varepsilon \mid \tau \\ \boldsymbol{\beta} \mid \tau &\sim \mathsf{N}(\mathbf{m}, \tau\mathbf{V}) \\ \varepsilon \mid \tau &\sim \mathsf{GP}\big(\mathbf{0}, \tau\kappa(\cdot,\cdot;\psi)\big) \\ \tau &\sim \mathsf{IG}(a, d) \end{aligned}$$

  where this is conditional on parameters ψ in the covariance function $\kappa\big((\theta, s), (\theta', s'); \psi\big)$.

- Choosing $\mathbf{h}$, $\mathbf{m}$, $\mathbf{V}$, a, d, κ and ψ is a standard Bayesian statistical challenge, *if we can build emulators quickly.*

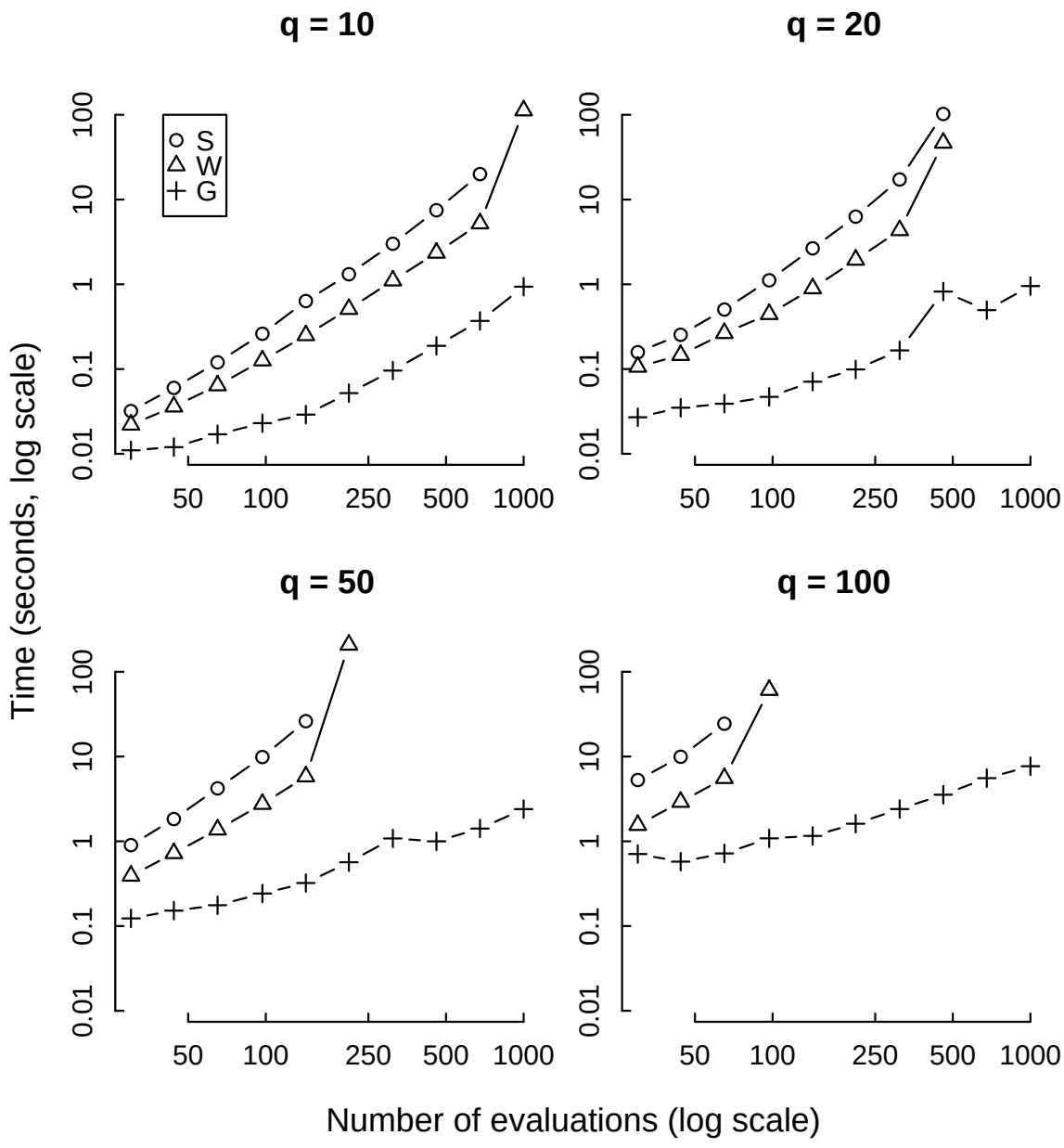
Numerical cost of building an emulator

- Until recently, it was thought that the cost of building an emulator (once all the bits were specified) was $\mathcal{O}(n^3 q^3)$ flops. On a desktop computer, this is about $n = 200$ runs of a simulator with $q = 50$ outputs.

Numerical cost of building an emulator

- Until recently, it was thought that the cost of building an emulator (once all the bits were specified) was $\mathcal{O}(n^3 q^3)$ flops. On a desktop computer, this is about $n = 200$ runs of a simulator with $q = 50$ outputs.
- But now we know that the Outer Product Emulator (Rougier, 2008) can do this calculation in $\mathcal{O}(n^3) + \mathcal{O}(q^3)$ flops. This allows us to go at least an order of magnitude bigger in both n and q (or go a *lot* quicker). Thus it becomes possible to emulate a spatial field of, e.g., temperatures, for the whole of the globe.
- We can go bigger again if the output index has separable structure, e.g. into space and time.

Numerical cost of building an emulator (in pictures)



The OPE in one slide

The OPE requires three conditions:

1. Rectangular outputs; i.e. the same output indices $s_1, \ldots, s_q$ regardless of the value of θ.
2. A separable residual covariance function,

$$\kappa\big((\theta, s_j), (\theta', s_{j'}); \psi\big) = \kappa^{\theta}\big(\theta, \theta'; \psi_\theta\big) \times \Sigma^{s}_{jj'}.$$

3. A set of regressors that is the pairwise product of regressors in θ and regressors in s:

$$\mathbf{h}(\theta, s) = \mathbf{h}^{\theta}(\theta) \otimes \mathbf{h}^{s}(s).$$

The OPE in one slide

The OPE requires three conditions:

1. Rectangular outputs; i.e. the same output indices $s_1, \ldots, s_q$ regardless of the value of θ.
2. A separable residual covariance function,

$$\kappa\big((\theta, s_j), (\theta', s_{j'}); \psi\big) = \kappa^{\theta}\big(\theta, \theta'; \psi_\theta\big) \times \Sigma^{s}_{jj'}.$$

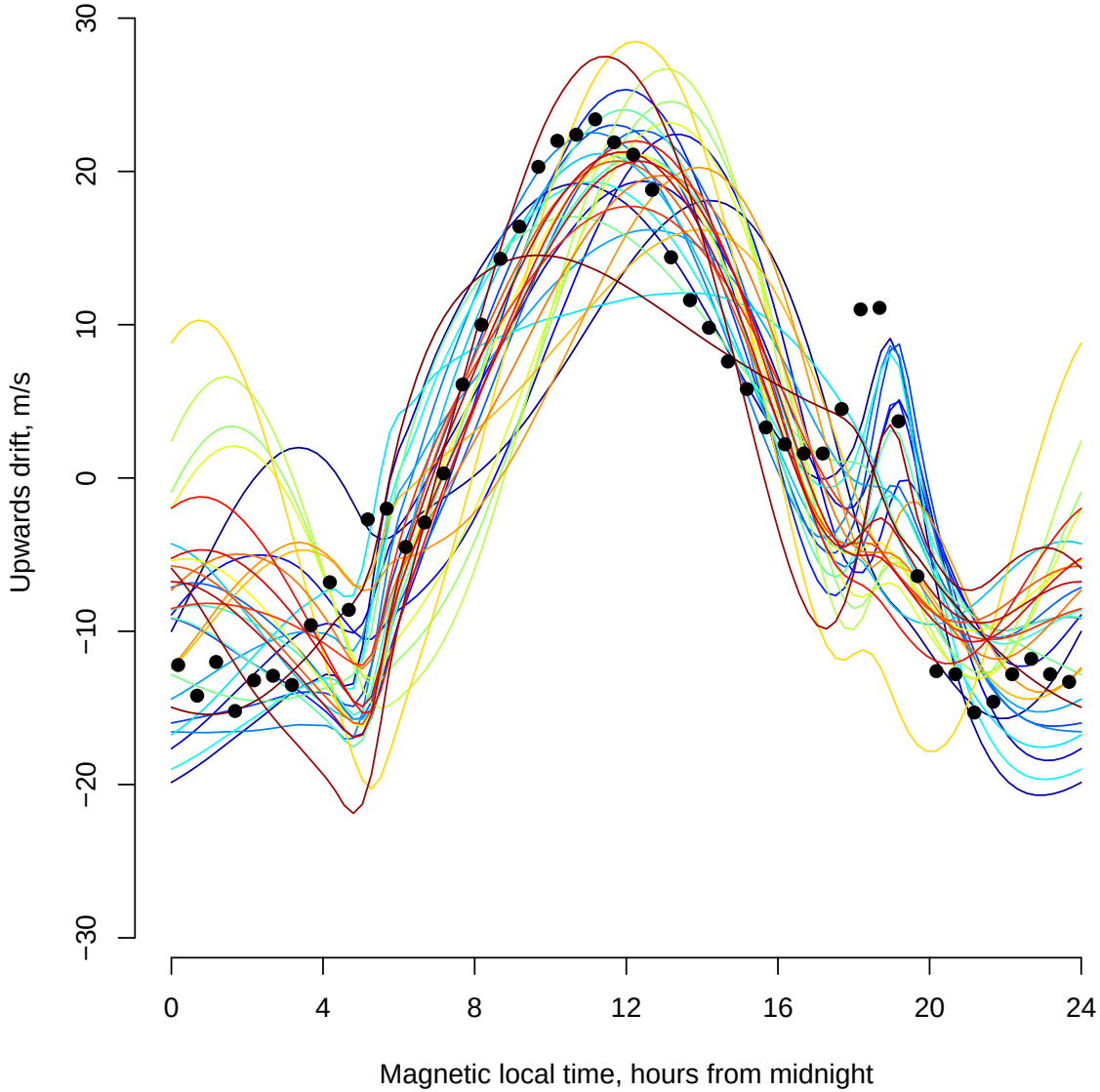
3. A set of regressors that is the pairwise product of regressors in θ and regressors in s:

$$\mathbf{h}(\theta, s) = \mathbf{h}^{\theta}(\theta) \otimes \mathbf{h}^{s}(s).$$

Consequence: In the emulator implementation, the kronecker product representation of the residual variance is conformable with the kronecker product representation of the regression matrix, leading to an algebraic reorganisation that is numerically very efficient.

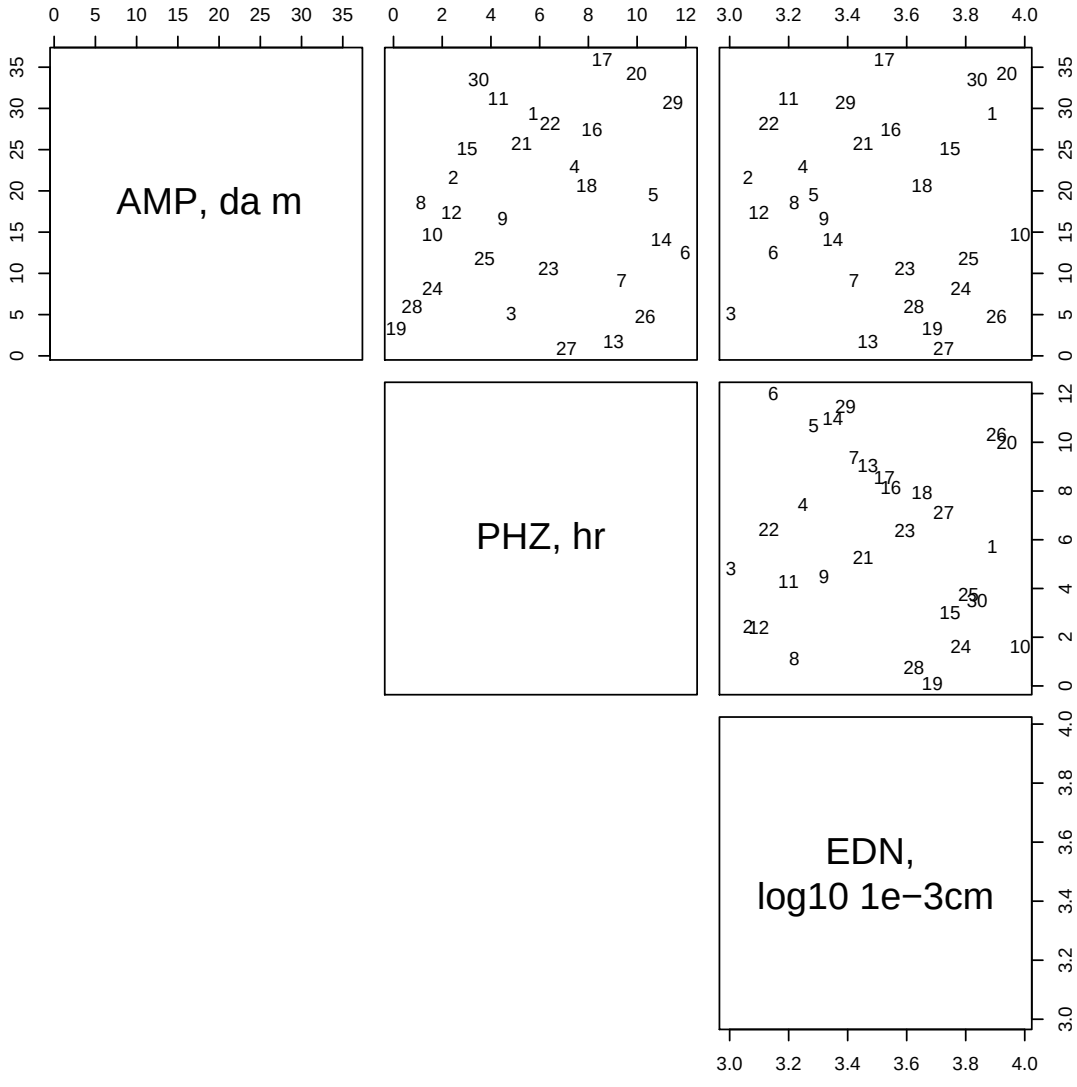
A really messy multivariate emulation

NCAR's TIE-GCM simulator of the upper atmosphere.



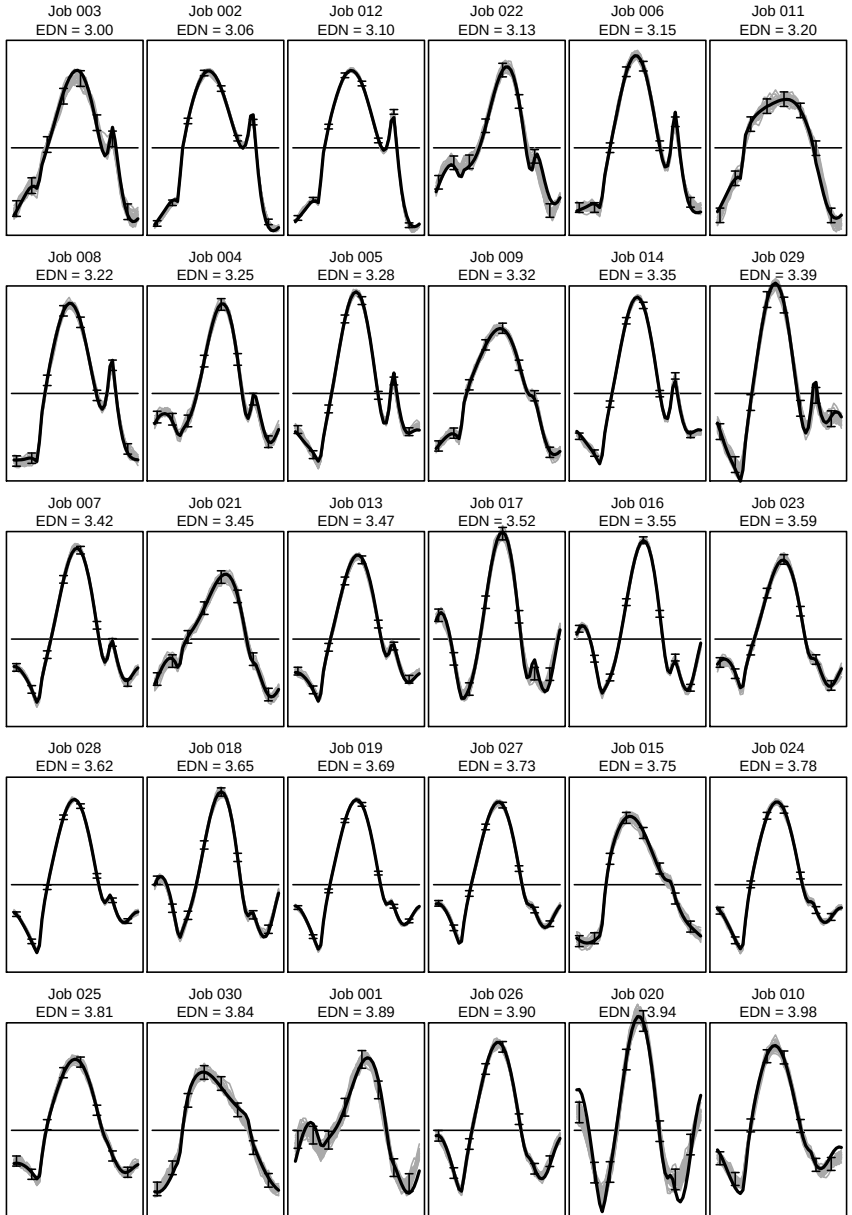
A really messy multivariate emulation

2D projections of the design matrix.



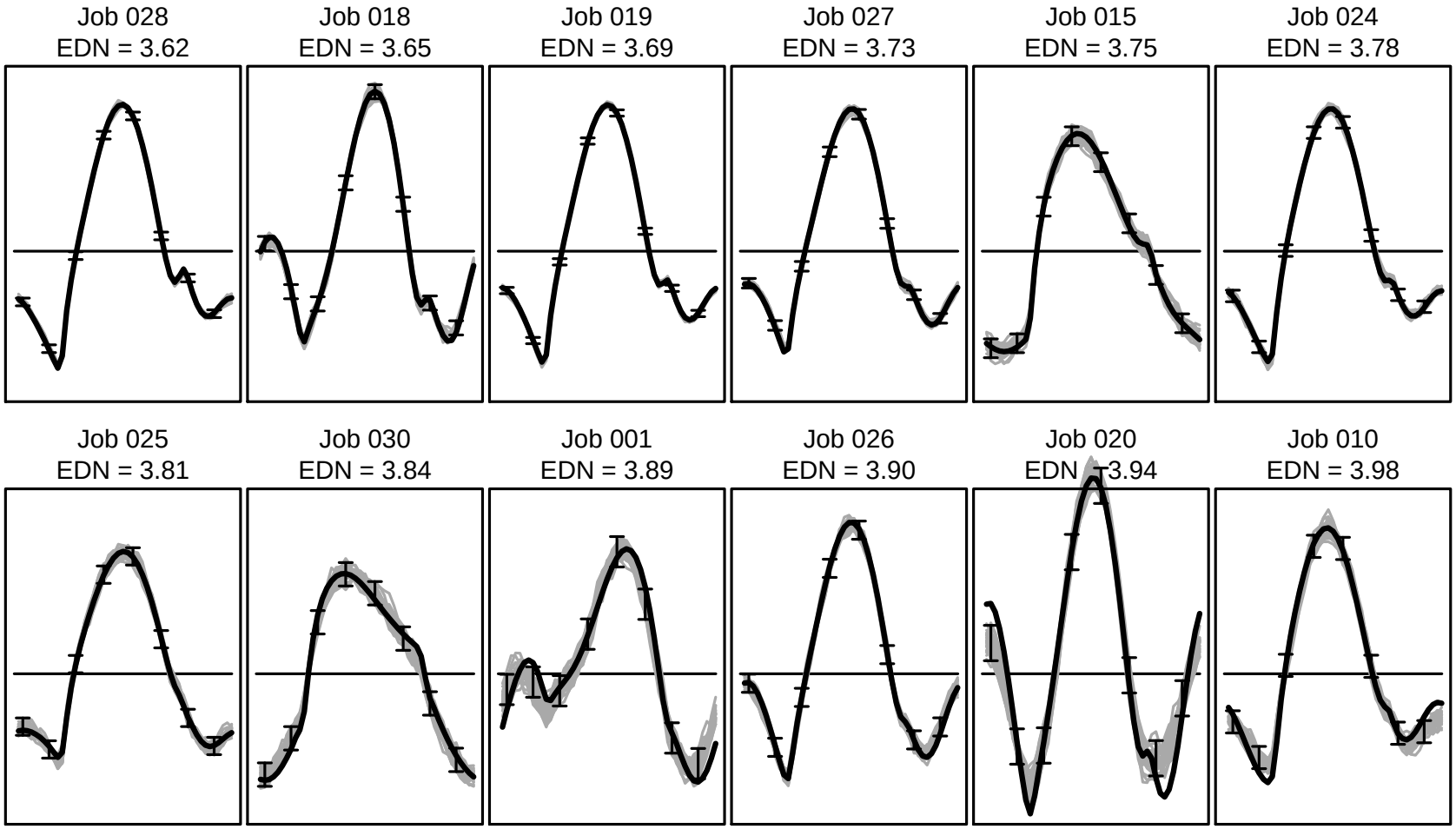
A really messy multivariate emulation

Leave-one-out predictive diagnostic.



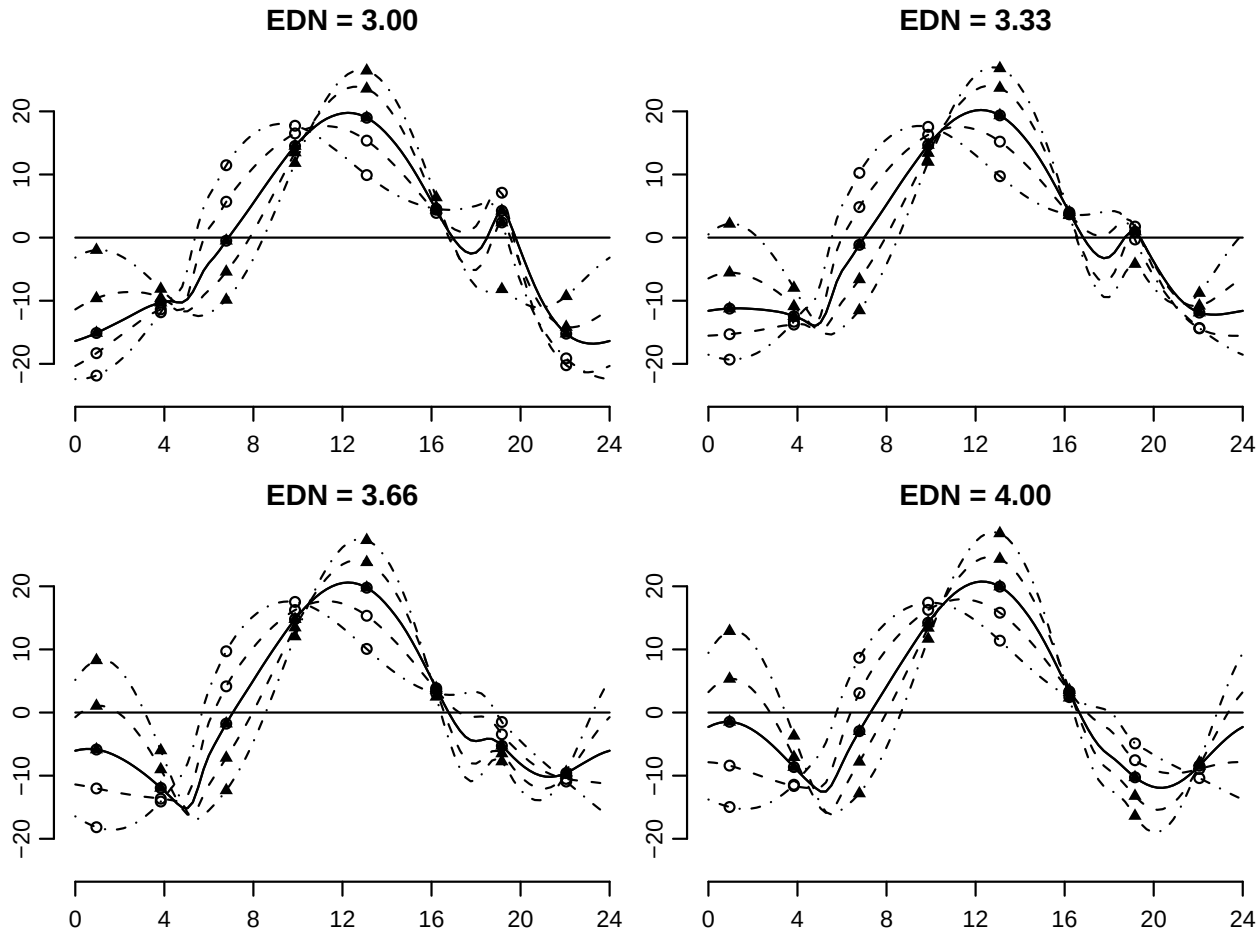
A really messy multivariate emulation

LOO: zoom in on last twelve.



A really messy multivariate emulation

Exploring the simulator behaviour.



The simulator's response to different values of the three inputs (mean function, interpolated with a periodic B-spline). Line styles denote values of AMP: solid = 0, dashed = 18, dot-dashed = 36. Plotting characters denote values of PHZ: open circle = 3, filled triangle = 9. The two solid lines are coincident, because there is no PHZ effect when AMP = 0.

Summary

1. Emulators are useful whenever the cost of spanning the simulator parameter space with evaluations exceeds the computational budget.
2. This often happens with simulators of complex physical systems, like global climate. Such simulators have multivariate outputs with lots of structure (e.g. space and time).
3. *A multivariate emulator is a complicated object, and requires detailed diagnostic checking.* But multivariate emulation, in general, is plagued by $\mathcal{O}(n^3 q^3)$ computations.
4. The Outer Product Emulator offers an $\mathcal{O}(q^3)$ solution: emulators of complex simulators with high-dimensional outputs that can be computed in seconds.
5. This makes it possible to use traditional approaches to statistical model choice and model criticism, based on predictive diagnostics like Leave-One-Out and One-Step-Ahead.

Some references

Bastos, L. and A. O'Hagan (2008). Diagnostics for gaussian process emulators. Technical Report No. 574/07, Department of Probability and Statistics, University of Sheffield. Currently available at `http://mucm.group.shef.ac.uk/Pages/Downloads/Technical_Reports/08-02.pdf`.

Craig, P., M. Goldstein, J. Rougier, and A. Seheult (2001). Bayesian forecasting for complex systems using computer simulators. *Journal of the American Statistical Association 96*, 717–729.

Goldstein, M. and J. Rougier (2006). Bayes linear calibrated prediction for complex systems. *Journal of the American Statistical Association 101*, 1132–1143.

Goldstein, M. and J. Rougier (2009). Reified Bayesian modelling and inference for physical systems. *Journal of Statistical Planning and Inference 139*, 1221–1239. With discussion.

Rougier, J. (2008a). Discussion of 'Inferring climate system properties using a computer model' by Sansó *et al. Bayesian Analysis 3*(1), 45–56.

Rougier, J. (2008b). Efficient emulators for multivariate deterministic functions. *Journal of Computational and Graphical Statistics 17*(4), 827–843.

Rougier, J., S. Guillas, A. Maute, and A. Richmond (2009). Expert knowledge and multivariate emulation: The Thermosphere-Ionosphere Electrodynamics General Circulation Model (TIE-GCM). Forthcoming in *Technometrics*, currently available at `http://www.maths.bris.ac.uk/~mazjcr/TIEGCM.pdf`.

See also `http://mucm.group.shef.ac.uk/`