The Food and Environment Research Agency

Modelling variability using emulators

Marc Kennedy, Helen Owen
The Food and Environment Research Agency

Outline

- Sources of variability in model inputs X
- Monte Carlo uncertainty analysis
  - Quantifies impacts of this variability on output Y
- Emulation of spray drift code
  - For practical implementation
- Adding variability in bystander contamination

Sources of variability

- Important inputs subject to variation
  - Boom height
  - Wind speed (turbulence effects)
  - Wind angle
- This variation induces variability in the output
- How do we quantify output variability?
  - Probabilistic methods available to propagate the effects of input variability through the model

Monte Carlo analysis

- Simplest approach:
  - Randomly simulate an input value X from probability distribution, representing variability
  - Run model to get the corresponding output Y = f(X)
  - Repeat many times
- Resulting sample of Y values represents the output distribution
  - Implied by variability in inputs
- But accurate results typically needs thousands of model runs…not practical for complex BREAM models

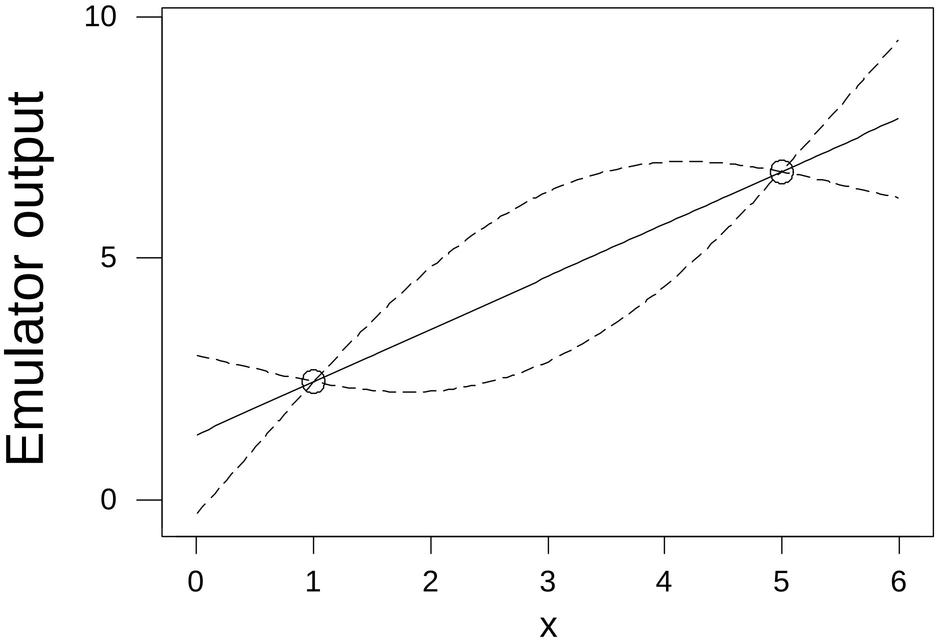
Emulation

- We build a statistical approximating function, known as an ***emulator*** of the computer code
  - Cheap surrogate for the true model
  - Includes uncertainty due to the approximation
  - Correctly reproduces training data
- Training runs
  - Run model for a representative sample of input values, to learn about the code input/output relationship

Multiple outputs emulated independently

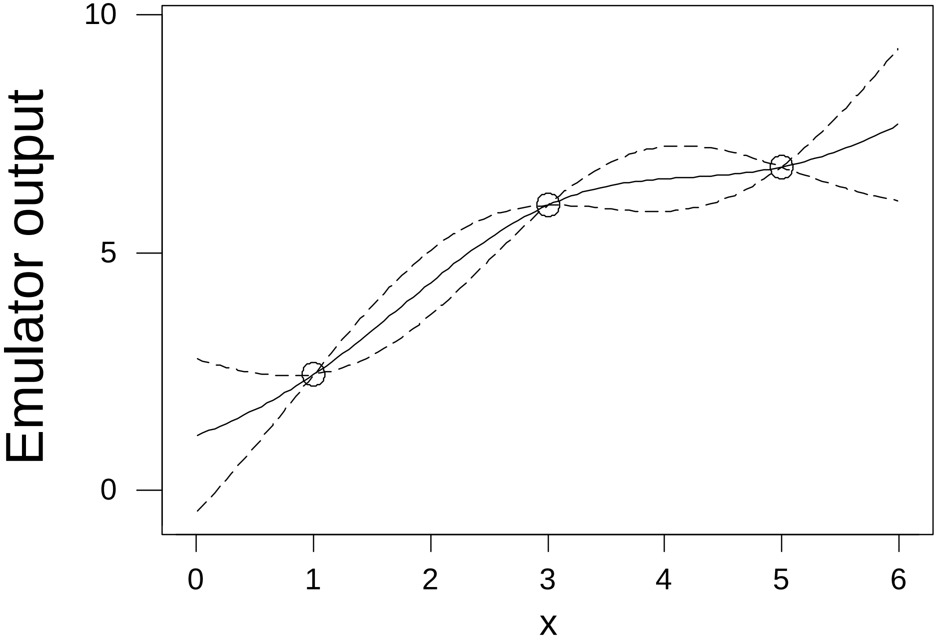
Simple illustration: 2 code runs

- Consider one input and one output
- Emulator estimate interpolates data
- Emulator uncertainty grows between data points



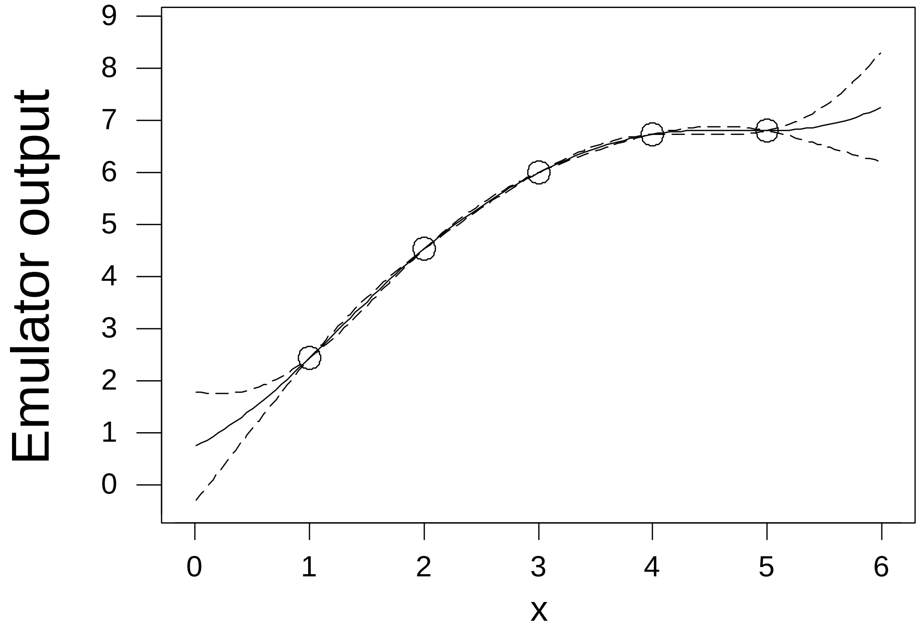
3 code runs

- Adding another point changes estimate and reduces uncertainty



5 code runs

- And so on…



Input variables for the emulators

- **Boom height (0.1 – 1.5 m)**
- **Wind speed (1 – 25 m/s)**
- **Wind angle (10 – 170 degrees)**
- Number of nozzles (9 – 996, in multiples of 12)
- Forward speed (4 – 25 km/h)
- Crop height (0.05 – 2.0 m)
- Distance from source (1 – 15 m)

Interface includes additional inputs to define scenario (to select correct emulator)

Accuracy of the emulators

- Current emulators use 200 code runs (FF nozzle type) to build each emulator
    - Sparse to cover 6 or 7 dimensional input space
    - Checks of accuracy ongoing (cross-validation checks, emulator variance estimates and direct comparisons with model)
- Emulators can be updated using further runs of the code

Then what?

- Given enough training runs we can emulate any model accurately
- Perform Monte Carlo analysis, using emulator in place of code
    - Feasible because emulator is much quicker to evaluate than the original code

Propagating variability through the spray model

Monte Carlo estimate, based on 10000 runs of the emulator with independent input distributions

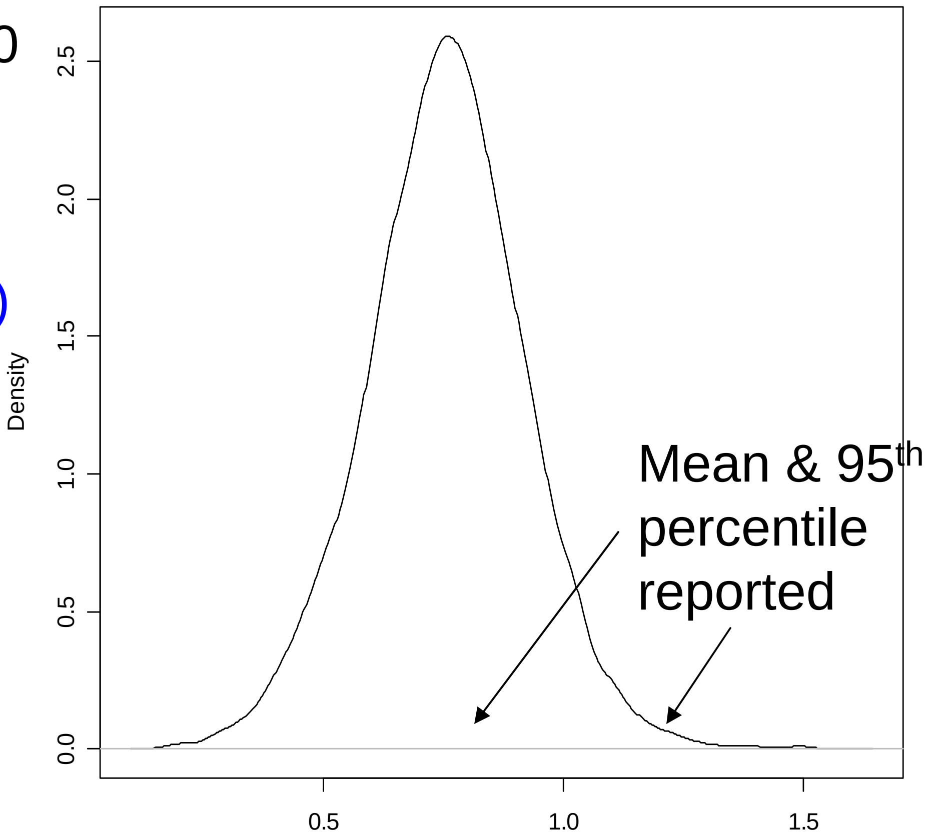
Boom height ~ N(mean, [0.3 x mean]^2)

Wind Speed ~ N(mean, 0.185 x mean + 0.0068)

Wind angle ~ N(90, 10^2)

These represent variability in real conditions during a single spray event

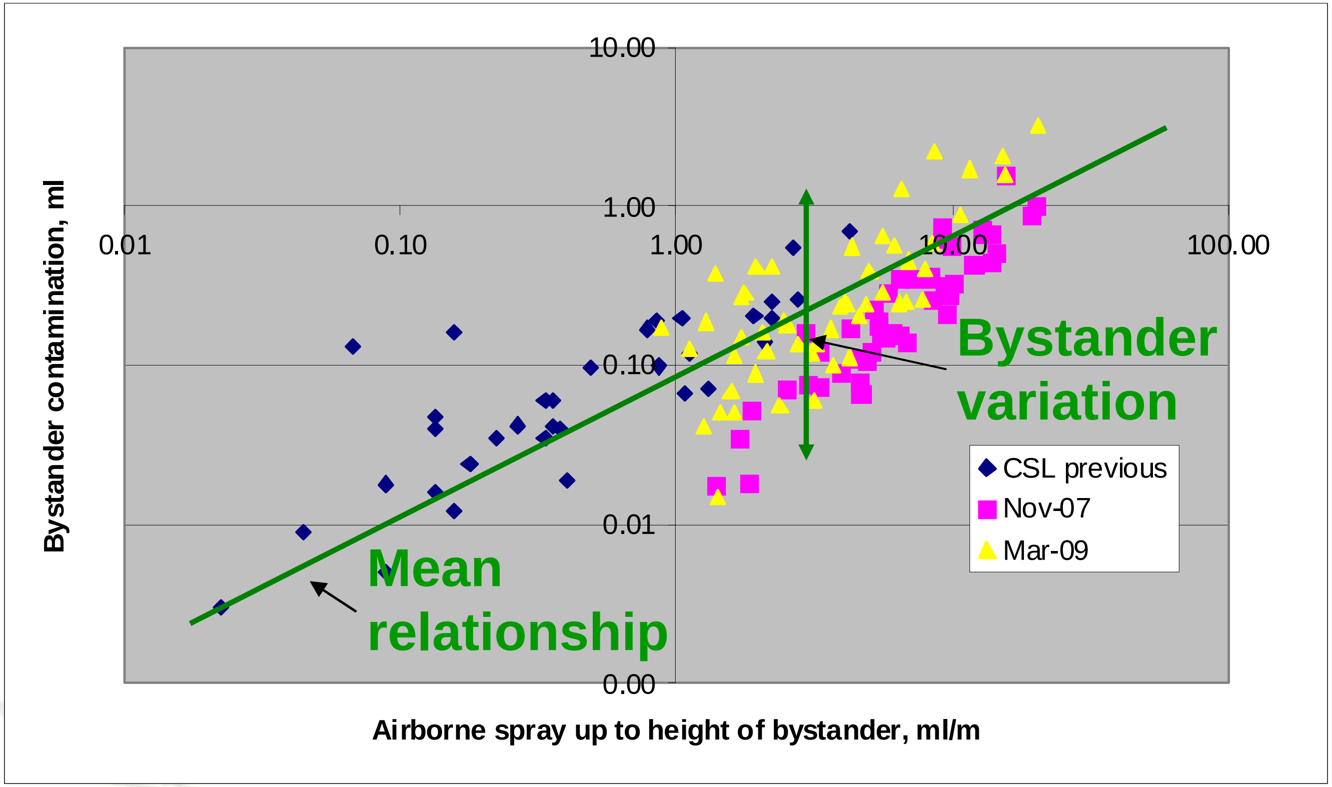
Distribution of airborne spray



Including extra (bystander) variability

MC simulated (airborne spray) outputs distributed along the x-axis

Extra variability around the 'mean bystander deposit' for a given spray level



1. User inputs scenario

2. MC analysis using emulators, assuming variability around the selected input values

3. Use airborne spray vs. bystander contamination data, to derive variability of bystander contamination

4. Calculate (for adult & child) probability distribution of:

Inhalation,

Dermal exposure,

Total exposure = Inhalation + Dermal exposure

5. Mean, 95th percentile of exposure measures output

The Food and Environment Research Agency