Using emulators to combine information from different climate simulators

Jonathan Rougier

Dept. Mathematical Sciences

Durham University, UK

J.C. Rougier, D.M.H. Sexton, J.M. Murphy and D. Stainforth, 2005, *Emulating the HadAM3 simulator using ensembles from different but related experiments*, in preparation.

Computer Experiments

- Statistics research groups in the UK and US are concerned with *model-based inference for complex systems*, also known as *computer experiments* because the model is usually implemented by a computer.

Computer Experiments

- Statistics research groups in the UK and US are concerned with *model-based inference for complex systems*, also known as *computer experiments* because the model is usually implemented by a computer.
- We need models for systems where
  - The system is too complex to infer future behaviour from past behaviour;
  - Future behaviour takes place in a different regime to past behaviour,

  where ‘past’ and ‘future’ are used in the loosest sense. Climate has both of these challenges.

Computer Experiments

- Statistics research groups in the UK and US are concerned with *model-based inference for complex systems*, also known as *computer experiments* because the model is usually implemented by a computer.
- We need models for systems where
  - The system is too complex to infer future behaviour from past behaviour;
  - Future behaviour takes place in a different regime to past behaviour,

  where ‘past’ and ‘future’ are used in the loosest sense. Climate has both of these challenges.
- The features that make computer experiments different from ‘standard’ experiments:
  - Large number of uncertain quantities (“parameters”) in the model specification;
  - Highly non-linear model response in certain regions of the parameter-space;
  - Long model-evaluation times.

The simulator

- The code that we evaluate is the *simulator*:

$$\texttt{Model} + \texttt{Treatment} + \texttt{Solver} = \texttt{Simulator}$$

The simulator

- The code that we evaluate is the *simulator*:

$$\texttt{Model} + \texttt{Treatment} + \texttt{Solver} = \texttt{Simulator}$$

- We think of the simulator as a deterministic function

$$x \longrightarrow g(x)$$

where the variables x comprise, eg, (i) uncertain model parameters; (ii) initial conditions, and (iii) future forcing functions (and perhaps past forcing functions as well).

The simulator

- The code that we evaluate is the *simulator*:

$$\texttt{Model} + \texttt{Treatment} + \texttt{Solver} = \texttt{Simulator}$$

- We think of the simulator as a deterministic function

$$x \longrightarrow g(x)$$

  where the variables x comprise, eg, (i) uncertain model parameters; (ii) initial conditions, and (iii) future forcing functions (and perhaps past forcing functions as well).

- The components of the vector $g(x)$ correspond to operationally-defined quantities in the underlying system.

The simulator

- The code that we evaluate is the *simulator*:

$$\texttt{Model} + \texttt{Treatment} + \texttt{Solver} = \texttt{Simulator}$$

- We think of the simulator as a deterministic function

$$x \longrightarrow g(x)$$

  where the variables x comprise, eg, (i) uncertain model parameters; (ii) initial conditions, and (iii) future forcing functions (and perhaps past forcing functions as well).

- The components of the vector $g(x)$ correspond to operationally-defined quantities in the underlying system.

- Many of the components of the vector x are not so well-defined; it is a moot question whether we can proceed as though there is a 'best' value for x, say x^*, for which $g(x^*)$ is the 'best' representation of the system.

Introducing the ‘emulator’

- *What is it?*

- *Where does it live?*

- *What is it good for?*

Introducing the ‘emulator’

- *What is it?*

  A statistical representation of the simulator, constructed using an ensemble of evaluations. Technically, a probability function $F_{g(x)}(v) \equiv \Pr\big(g(x) \leq v \mid x\big)$.

- *Where does it live?*

- *What is it good for?*

Introducing the ‘emulator’

- *What is it?*

  A statistical representation of the simulator, constructed using an ensemble of evaluations. Technically, a probability function $F_{g(x)}(v) \equiv \Pr\big(g(x) \le v \mid x\big)$.

- *Where does it live?*

  Between the ensemble and all the experiments we would like to do with the simulator, if only it were fast enough.

- *What is it good for?*

Introducing the 'emulator'

- *What is it?*

  A statistical representation of the simulator, constructed using an ensemble of evaluations. Technically, a probability function $F_{g(x)}(v) \equiv \Pr\big(g(x) \leq v \mid x\big)$.

- *Where does it live?*

  Between the ensemble and all the experiments we would like to do with the simulator, if only it were fast enough.

- *What is it good for?*
  - Inference about the system, taking due account of uncertainty about the variables;
  - Design calculations, to choose good (ie informative) evaluations for the ensemble;
  - Combining information from different but related simulators.

Introducing the ‘emulator’

- *What is it?*

  A statistical representation of the simulator, constructed using an ensemble of evaluations. Technically, a probability function $F_{g(x)}(v) \equiv \Pr\big(g(x) \le v \mid x\big)$.

- *Where does it live?*

  Between the ensemble and all the experiments we would like to do with the simulator, if only it were fast enough.

- *What is it good for?*

  - Inference about the system, taking due account of uncertainty about the variables;
  - Design calculations, to choose good (ie informative) evaluations for the ensemble;
  - Combining information from different but related simulators.

The primary function of the emulator is to interpolate *and extrapolate* the given ensemble of evaluations with an appropriate measure of uncertainty.

The QUMP sensitivity experiment

QUMP $=$ Quantifying Uncertainty in Model Predictions

- *The simulator:* HadAM3 integrated to equilibrium twice, once with pre-industrial CO_2 and once with $2\times$ CO_2; *sensitivity* is the difference in the two equilibrium global mean temperatures.

The QUMP sensitivity experiment

QUMP $=$ Quantifying Uncertainty in Model Predictions

- *The simulator:* HadAM3 integrated to equilibrium twice, once with pre-industrial CO_2 and once with $2\times$ CO_2; *sensitivity* is the difference in the two equilibrium global mean temperatures.
- *The uncertain quantities:* Thirty-one model parameters; namely thirteen factors and eighteen continuous variables. [OHP]

The QUMP sensitivity experiment

QUMP $=$ Quantifying Uncertainty in Model Predictions

- *The simulator:* HadAM3 integrated to equilibrium twice, once with pre-industrial CO_2 and once with $2\times$ CO_2; *sensitivity* is the difference in the two equilibrium global mean temperatures.
- *The uncertain quantities:* Thirty-one model parameters; namely thirteen factors and eighteen continuous variables. [OHP]
- *The ensemble:* Fifty-four single variable perturbations (typically “low” and “high” with everything else at “standard”) plus a limited number of multi-variable perturbations.

The QUMP sensitivity experiment

QUMP $=$ Quantifying Uncertainty in Model Predictions

- *The simulator:* HadAM3 integrated to equilibrium twice, once with pre-industrial CO_2 and once with $2\times$ CO_2; *sensitivity* is the difference in the two equilibrium global mean temperatures.
- *The uncertain quantities:* Thirty-one model parameters; namely thirteen factors and eighteen continuous variables. [OHP]
- *The ensemble:* Fifty-four single variable perturbations (typically “low” and “high” with everything else at “standard”) plus a limited number of multi-variable perturbations.
- *The problem:* Almost no information in the ensemble about interactions between variables, and about non-linear effects.

The QUMP sensitivity experiment

QUMP $=$ Quantifying Uncertainty in Model Predictions

- *The simulator:* HadAM3 integrated to equilibrium twice, once with pre-industrial CO_2 and once with $2\times$ CO_2; *sensitivity* is the difference in the two equilibrium global mean temperatures.
- *The uncertain quantities:* Thirty-one model parameters; namely thirteen factors and eighteen continuous variables. [OHP]
- *The ensemble:* Fifty-four single variable perturbations (typically “low” and “high” with everything else at “standard”) plus a limited number of multi-variable perturbations.
- *The problem:* Almost no information in the ensemble about interactions between variables, and about non-linear effects.
- *The solution:*
  - Augment the QUMP ensemble with further evaluations of HadAM3 (expensive!);
  - Incorporate information from elsewhere, *eg the CPNET experiment*.

The CPNET sensitivity experiment

CPNET $=$ climate*prediction*.net

- *The simulator:* HadAM3 (PC version) in a single 45-year integration: calibration (15 years), control (15 years), $2 \times CO_2$ (15 years). *Sensitivity* is extrapolated using a fitted exponential curve.

The CPNET sensitivity experiment

CPNET $=$ climate*prediction*.net

- *The simulator:* HadAM3 (PC version) in a single 45-year integration: calibration (15 years), control (15 years), $2 \times$ CO_2 (15 years). *Sensitivity* is extrapolated using a fitted exponential curve.
- *The uncertain quantities:* Six continuous model parameters. [OHP]

The CPNET sensitivity experiment

CPNET $=$ climate*prediction*.net

- *The simulator:* HadAM3 (PC version) in a single 45-year integration: calibration (15 years), control (15 years), $2 \times$ CO_2 (15 years). *Sensitivity* is extrapolated using a fitted exponential curve.
- *The uncertain quantities:* Six continuous model parameters. [OHP]
- *The ensemble:* Several thousand multi-variable perturbations in a full factorial design with three levels for each variable.

The CPNET sensitivity experiment

CPNET $=$ climate*prediction*.net

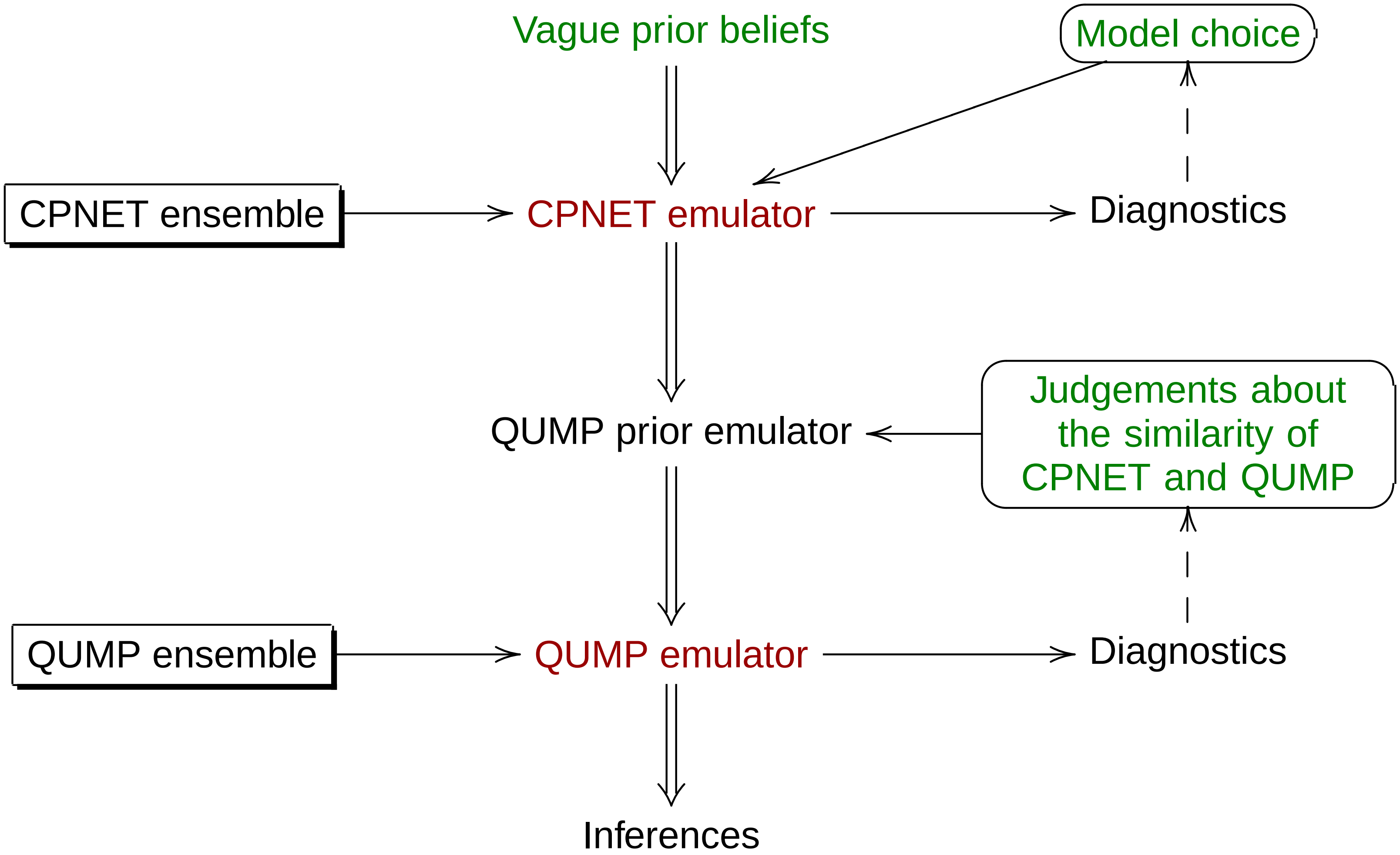
- *The simulator:* HadAM3 (PC version) in a single 45-year integration: calibration (15 years), control (15 years), $2 \times$ CO_2 (15 years). *Sensitivity* is extrapolated using a fitted exponential curve.
- *The uncertain quantities:* Six continuous model parameters. [OHP]
- *The ensemble:* Several thousand multi-variable perturbations in a full factorial design with three levels for each variable.
- *The problem:* (i) Approximation for sensitivity, relative to QUMP definition; (ii) No information at all about the impact of the other twenty-five QUMP variables.

The CPNET sensitivity experiment

CPNET $=$ climate*prediction*.net

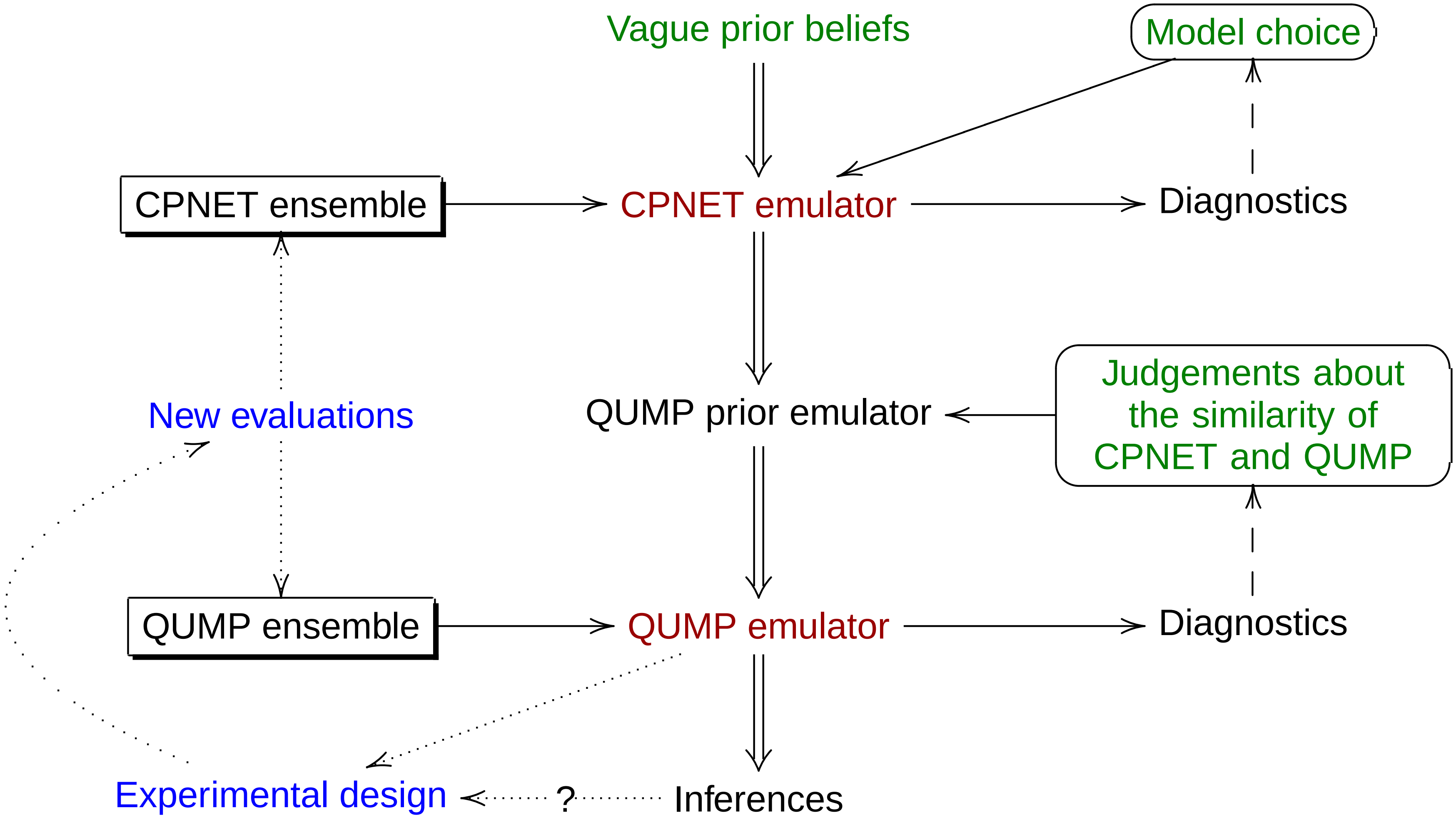
- *The simulator:* HadAM3 (PC version) in a single 45-year integration: calibration (15 years), control (15 years), $2 \times$ CO_2 (15 years). *Sensitivity* is extrapolated using a fitted exponential curve.
- *The uncertain quantities:* Six continuous model parameters. [OHP]
- *The ensemble:* Several thousand multi-variable perturbations in a full factorial design with three levels for each variable.
- *The problem:* (i) Approximation for sensitivity, relative to QUMP definition; (ii) No information at all about the impact of the other twenty-five QUMP variables.
- *The solution:* Combine the CPNET and QUMP ensembles together, taking advantage of the strengths of each ensemble:
  - *QUMP:* Some information about thirty-one uncertain variables; 'standard' definition of sensitivity;
  - *CPNET:* Detailed information about six of the most important variables.

The vision . . .



Vague prior beliefs
Model choice
CPNET ensemble
CPNET emulator
Diagnostics
QUMP prior emulator
Judgements about the similarity of CPNET and QUMP
QUMP ensemble
QUMP emulator
Diagnostics
Inferences

The vision . . .



Vague prior beliefs
Model choice
CPNET ensemble
CPNET emulator
Diagnostics
Judgements about the similarity of CPNET and QUMP
New evaluations
QUMP prior emulator
QUMP ensemble
QUMP emulator
Diagnostics
Experimental design
?
Inferences

Model choice

The emulator for CPNET sensitivity can be thought of as a Bayesian treatment of OLS regression, with a vague (non-informative) prior. We have to follow exactly the same steps that we would follow when constructing a viable regression:

$$t(y) = \sum_i \beta_i \, g_i(x) + \epsilon_i(x)$$

- *Choice of regressor functions, $g_i(\cdot)$*
- Choice of transformation of the response, $y \to t(y)$
- Regression diagnostics based on estimated residuals

$$\hat{\epsilon}_i \triangleq t(y_i) - \sum_i \hat{\beta}_i \, g(x_i).$$

Model choice

The emulator for CPNET sensitivity can be thought of as a Bayesian treatment of OLS regression, with a vague (non-informative) prior. We have to follow exactly the same steps that we would follow when constructing a viable regression:

$$t(y) = \sum_i \beta_i \, g_i(x) + \epsilon_i(x)$$

- *Choice of regressor functions,* $g_i(\cdot)$

  Log-transform of ‘long-tailed’ continuous variables; constant, linear, quadratic, and all two-way interactions for regressors; small number of important additional regressors selected by stepwise methods;

- Choice of transformation of the response, $y \to t(y)$

- Regression diagnostics based on estimated residuals

$$\hat{\epsilon}_i \triangleq t(y_i) - \sum_i \hat{\beta}_i \, g(x_i).$$

Model choice

The emulator for CPNET sensitivity can be thought of as a Bayesian treatment of OLS regression, with a vague (non-informative) prior. We have to follow exactly the same steps that we would follow when constructing a viable regression:

$$t(y) = \sum_i \beta_i \, g_i(x) + \epsilon_i(x)$$

- *Choice of regressor functions,* $g_i(\cdot)$

  Log-transform of 'long-tailed' continuous variables; constant, linear, quadratic, and all two-way interactions for regressors; small number of important additional regressors selected by stepwise methods;
- Choice of transformation of the response, $y \to t(y)$

  The Box-Cox transformation strongly supports $t(y) = 1/y$; [picture]
- Regression diagnostics based on estimated residuals [picture]

$$\hat{\epsilon}_i \triangleq t(y_i) - \sum_i \hat{\beta}_i \, g(x_i).$$

The similarity of CPNET and QUMP

QUMP has more variables than CPNET, so we can think of the QUMP regressors being a superset of the CPNET regressors. *The primary route by which we pass information from CPNET to QUMP is by specifying the degree to which the QUMP emulator coefficients will be similar to their matched coefficients in the CPNET emulator.*

The similarity of CPNET and QUMP

QUMP has more variables than CPNET, so we can think of the QUMP regressors being a superset of the CPNET regressors. *The primary route by which we pass information from CPNET to QUMP is by specifying the degree to which the QUMP emulator coefficients will be similar to their matched coefficients in the CPNET emulator.*

We use the statistical framework

$$\beta_i' - m_i = (1 + \omega_i)(\beta_i - m_i) + (r_y / r_i)\, \nu_i$$

where (we specify the red values):

$\beta_i,\ \beta_i'$	Coefficients for CPNET and QUMP, respectively;
m_i	'Centering' value;
ω_i	Independent mean-zero uncertain quantity with standard deviation σ_ω;
$r_i,\ r_y$	Typical scales for the regressor and transformed response, respectively;
ν_i	Independent mean-zero uncertain quantity with standard deviation σ_ν.

Diagnostics

Our specification is $m_i = 0$, $\sigma_\omega = 1/3$, $\sigma_\nu = 1/18$ for the matched coefficients; for the unmatched coefficients we have $\sigma_\nu = 1/9$ in the simpler framework $\beta_i' = (r_y/r_i)\,\nu_i$.

- *Prior predictive* [picture]

  We predict the evaluations in the QUMP ensemble using our QUMP prior emulator (checking for over- or under-dispersion);

- *Moving coefficients* [picture]

  We examine the way in which the matched coefficients move after updating the QUMP prior emulator with the QUMP ensemble;

- *Leave-one-out* [picture]

  We update our QUMP prior emulator using all but one of the evaluations in the QUMP ensemble and then predict that one; we do this for each evaluation and examine the marginal prediction errors;

- *Leave-150-out* [picture]

  We leave out 150 evaluations from the QUMP ensemble, update the QUMP prior emulator with the rest, and examine the joint prediction errors.

Example application: Prediction for QUMP sensitivity

- Notionally there is a 'best' input x^*, but we are uncertain about its value. So we want to predict $g(x^*)$ where $g(\cdot)$ is the QUMP simulator and x^* has some specified distribution function $F_{x^*}(\cdot)$. *For us, $g(\cdot)$ is uncertain too*, because we have only a finite number of evaluations in our QUMP ensemble. *Prediction is integrating an uncertain function over an uncertain quantity:*

$$\Pr\big(g(x^*) \leq v\big) = \int_x F_{g(x)}(v)\, dF_{x^*}(x).$$

Example application: Prediction for QUMP sensitivity

- Notionally there is a 'best' input x^*, but we are uncertain about its value. So we want to predict $g(x^*)$ where $g(\cdot)$ is the QUMP simulator and x^* has some specified distribution function $F_{x^*}(\cdot)$. *For us, $g(\cdot)$ is uncertain too*, because we have only a finite number of evaluations in our QUMP ensemble. *Prediction is integrating an uncertain function over an uncertain quantity:*

$$\Pr\big(g(x^*) \leq v\big) = \int_x F_{g(x)}(v)\, dF_{x^*}(x).$$

- The QUMP choice for $F_{x^*}(\cdot)$ was independent for all components, uniform for continuous variables and equi-probable for all factors. *This is not a very good choice!* Ignoring this fact, we can compute a probability distribution for $g(x^*)$. [picture]

Example application: Prediction for QUMP sensitivity

- Notionally there is a 'best' input x^*, but we are uncertain about its value. So we want to predict $g(x^*)$ where $g(\cdot)$ is the QUMP simulator and x^* has some specified distribution function $F_{x^*}(\cdot)$. *For us, $g(\cdot)$ is uncertain too*, because we have only a finite number of evaluations in our QUMP ensemble. *Prediction is integrating an uncertain function over an uncertain quantity:*

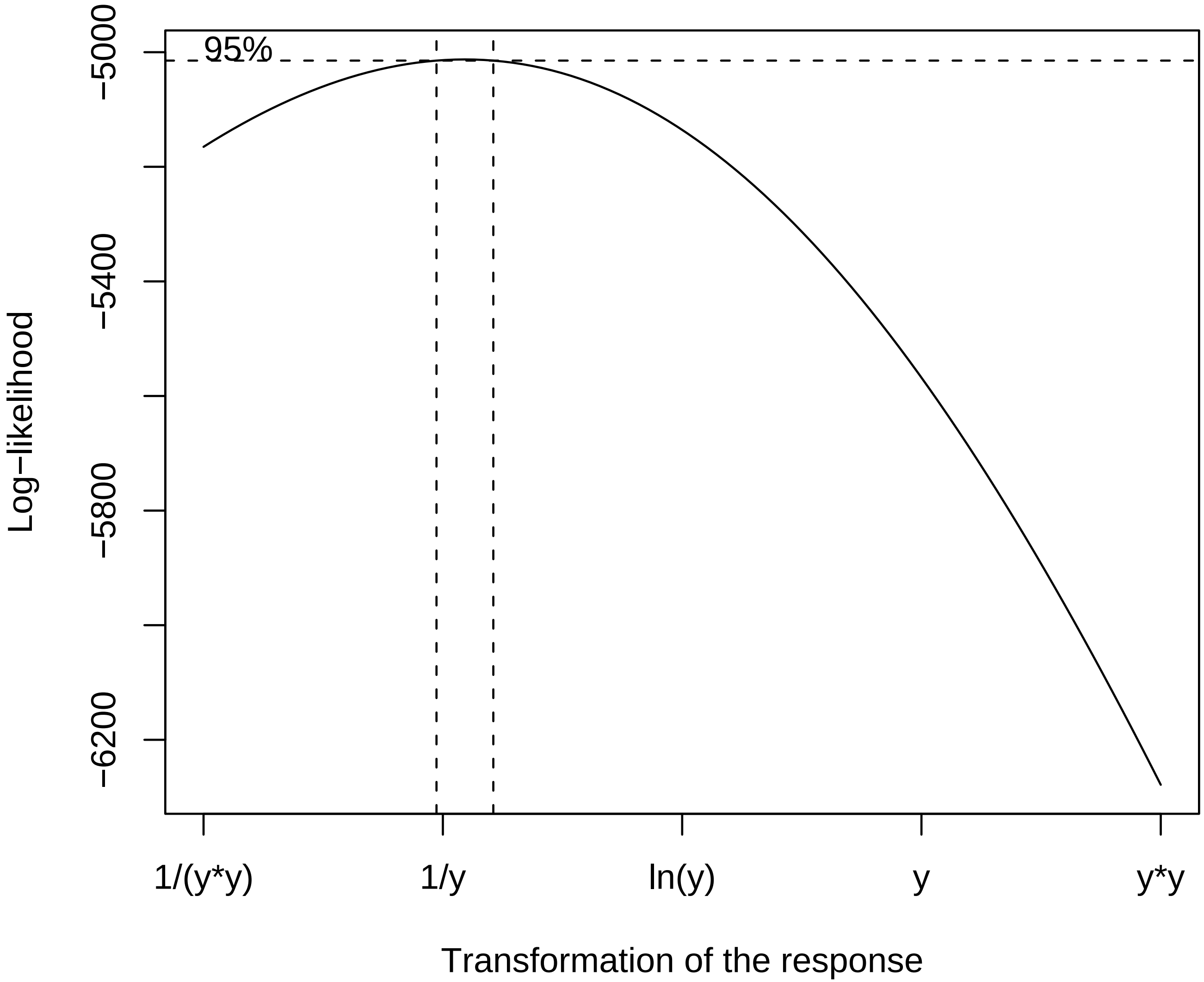
$$\Pr\big(g(x^*) \leq v\big) = \int_x F_{g(x)}(v)\, dF_{x^*}(x).$$

- The QUMP choice for $F_{x^*}(\cdot)$ was independent for all components, uniform for continuous variables and equi-probable for all factors. *This is not a very good choice!* Ignoring this fact, we can compute a probability distribution for $g(x^*)$. [picture]
- There is a vigorous ongoing debate regarding U or $1/U$ for some of the continuous variables in x^*. We can try them both to see whether it matters. [picture] *(It does!)*

Conclusion

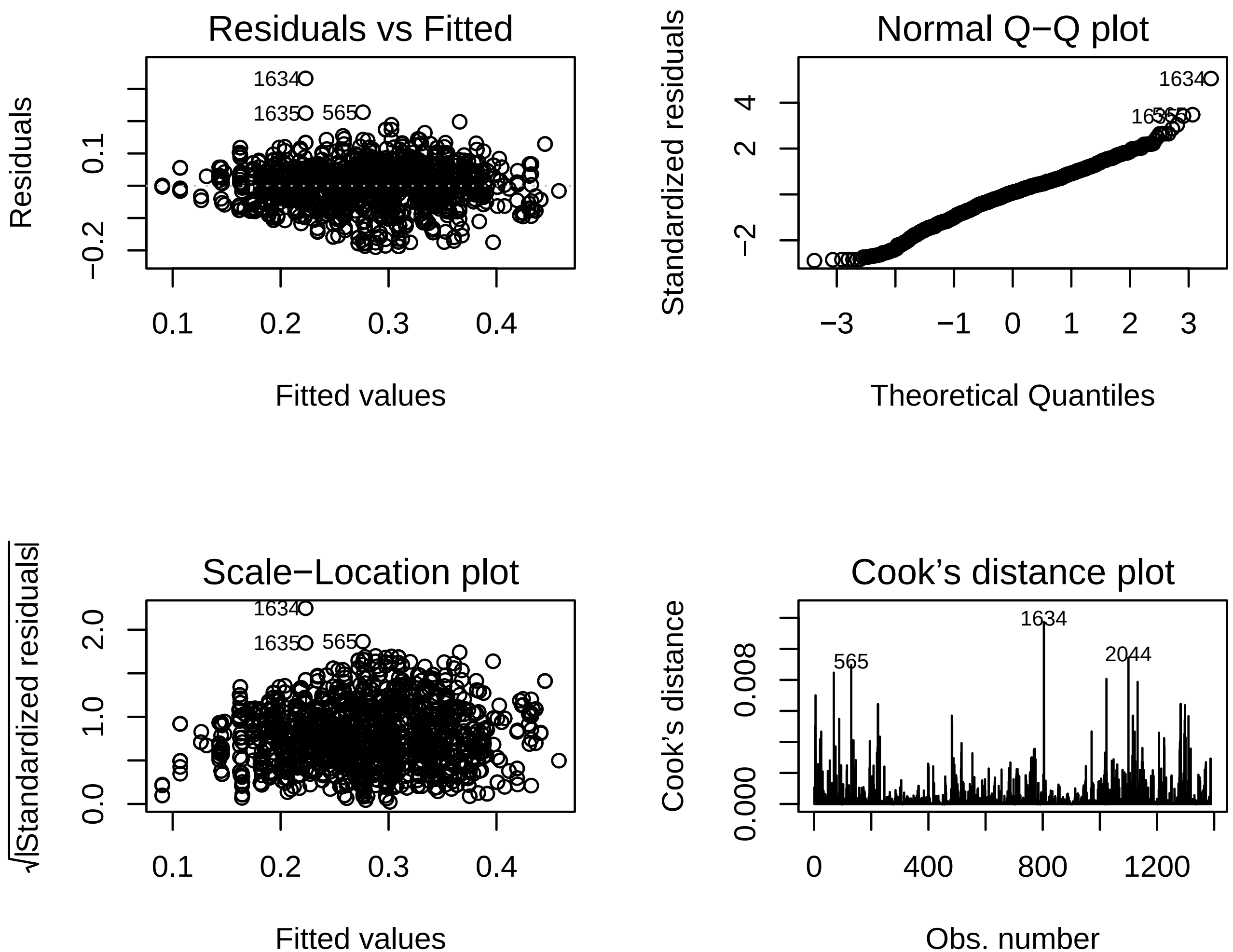
- Emulators are a necessary part of our inference when we work with large simulators; their purpose is to extrapolate our ensemble over the space of uncertain simulator variables, including a measure of uncertainty; as with all extrapolators, the emulator has to be carefully constructed, with lots of diagnostics.
- Emulators separate the business of learning about the simulator from the business of making inferences using the simulator. Inferential decisions like *"What prior on x^*?"* have little role to play when we choose the evaluations in our ensemble. Once we have built out emulator we can try out lots of different priors.
- Emulators provide a way of combining information from related but different experiments, allowing us to specify our judgements regarding the degree to which the experiments are related. These judgements can be subjected to some validation.

Box-Cox plot for CPNET



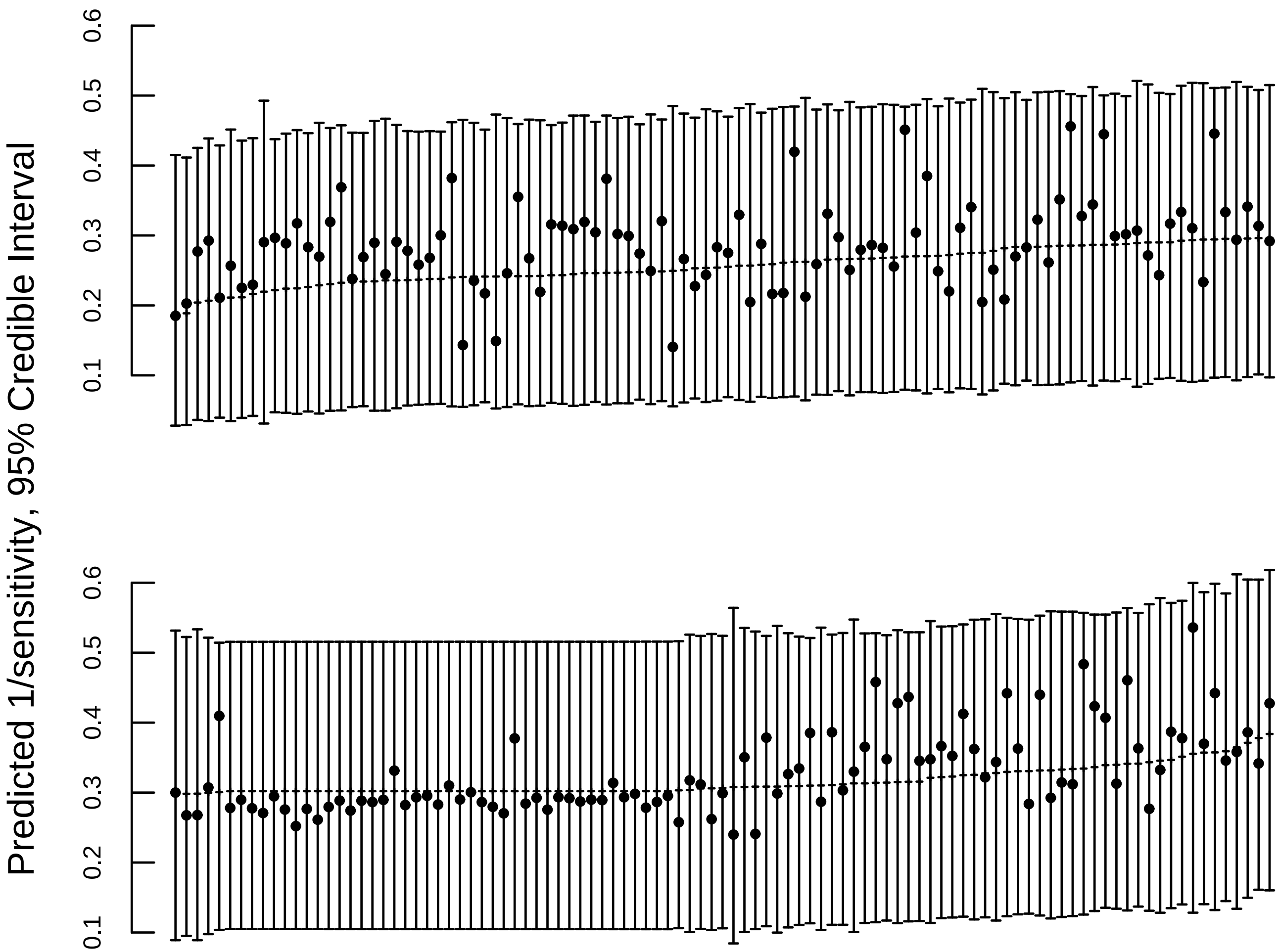
[back]

CPNET regression diagnostics



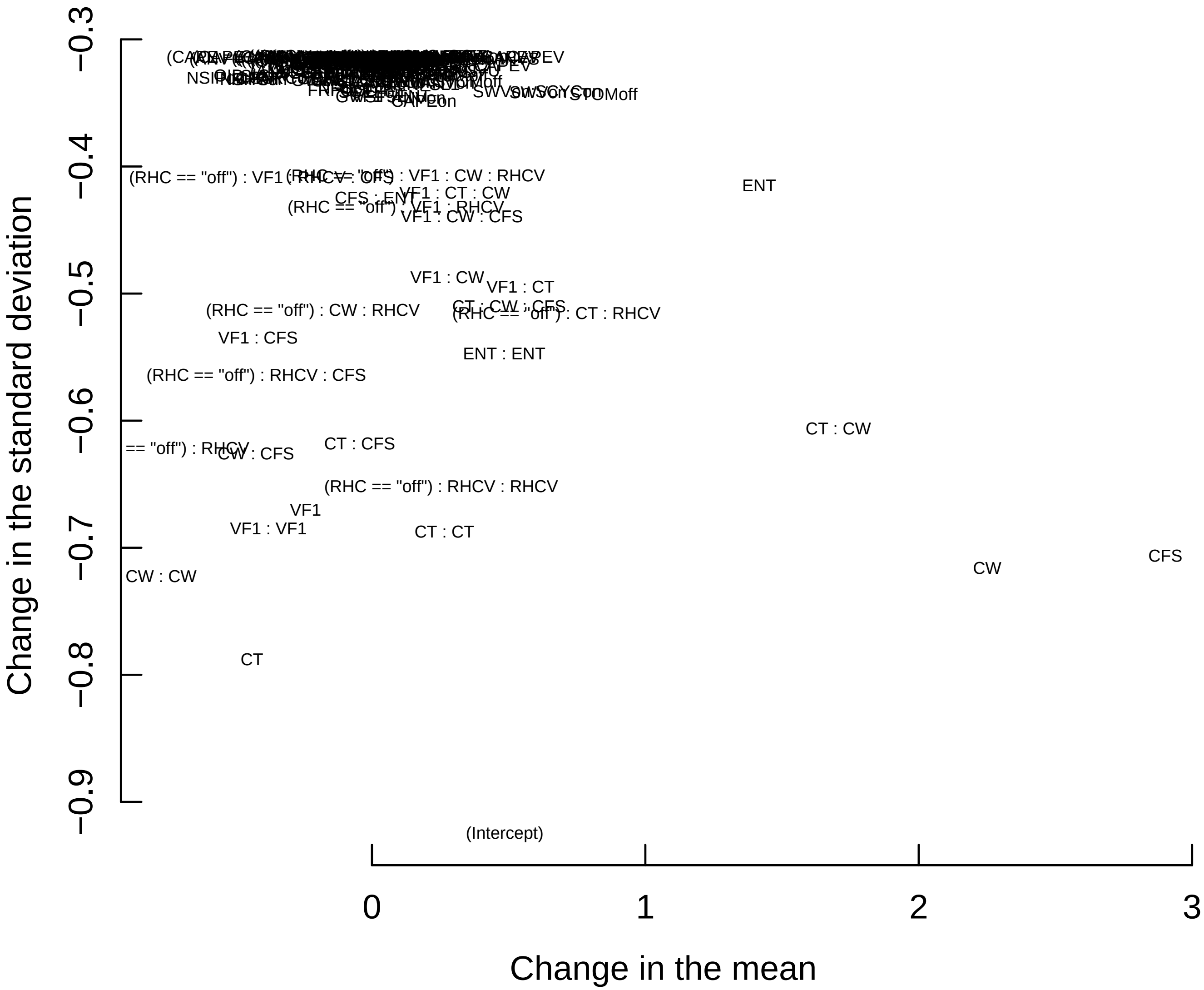
[back]

Prior prediction errors



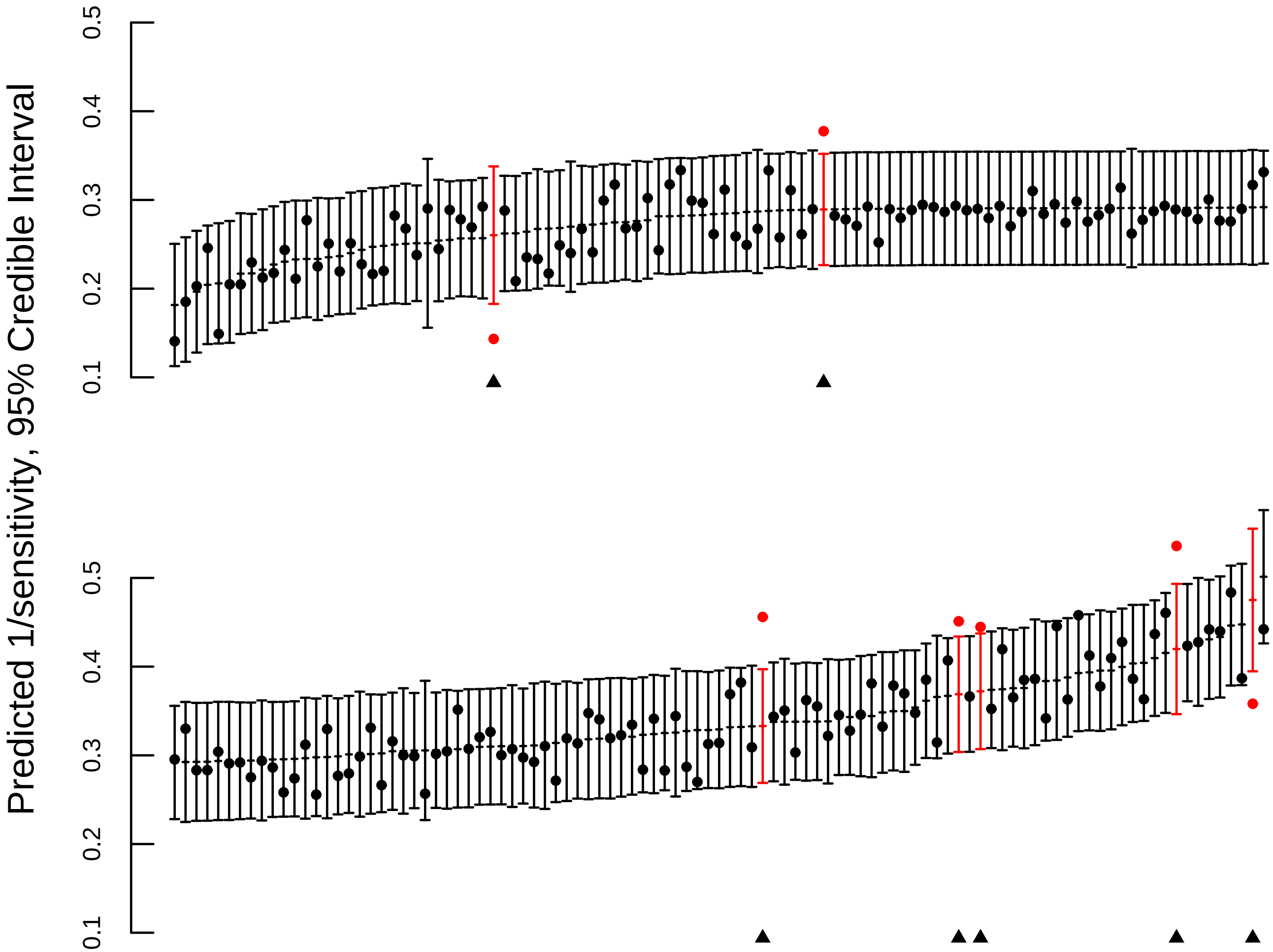
[back]

Moving coefficients



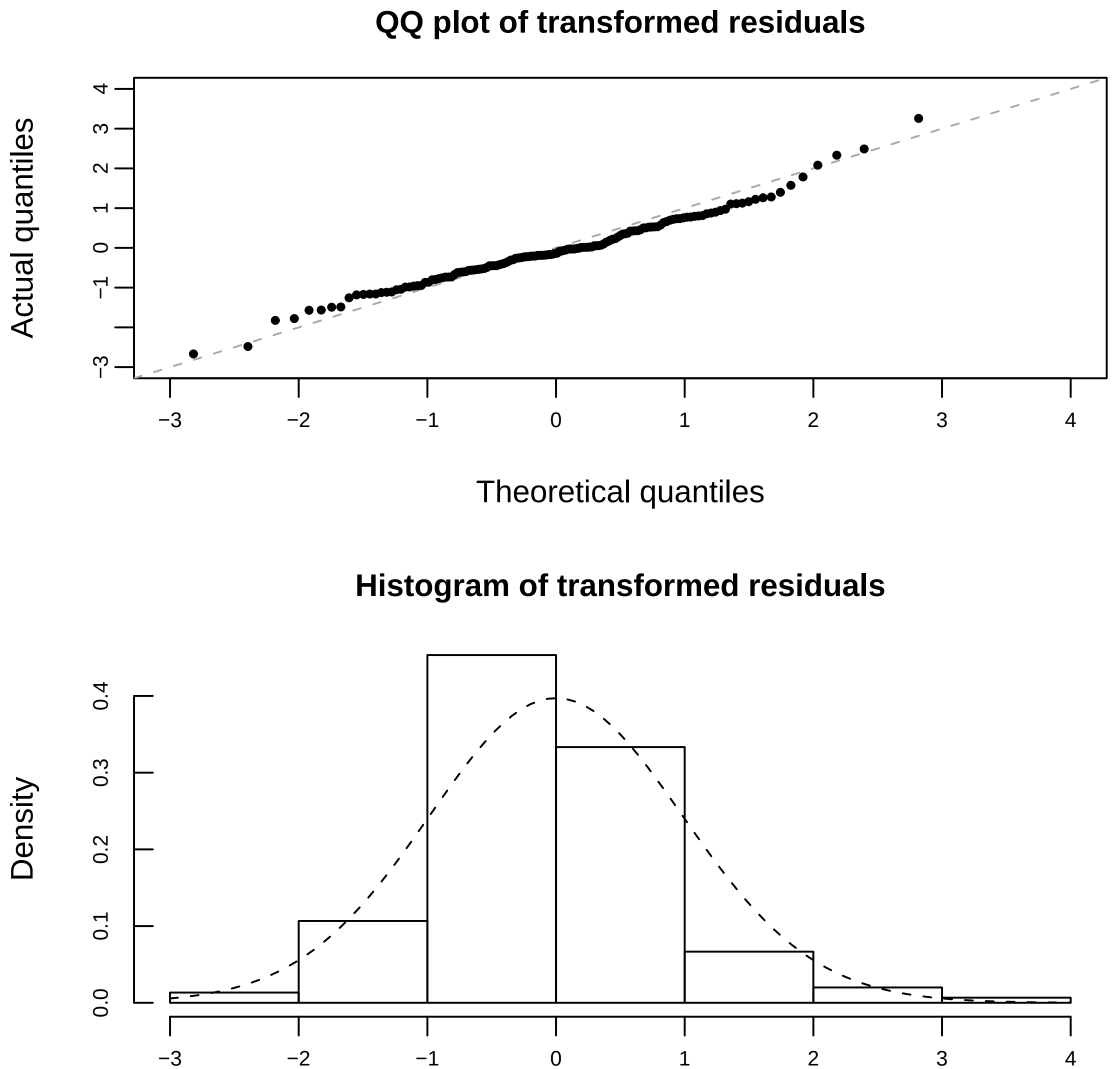
[back]

Leave-one-out diagnostic



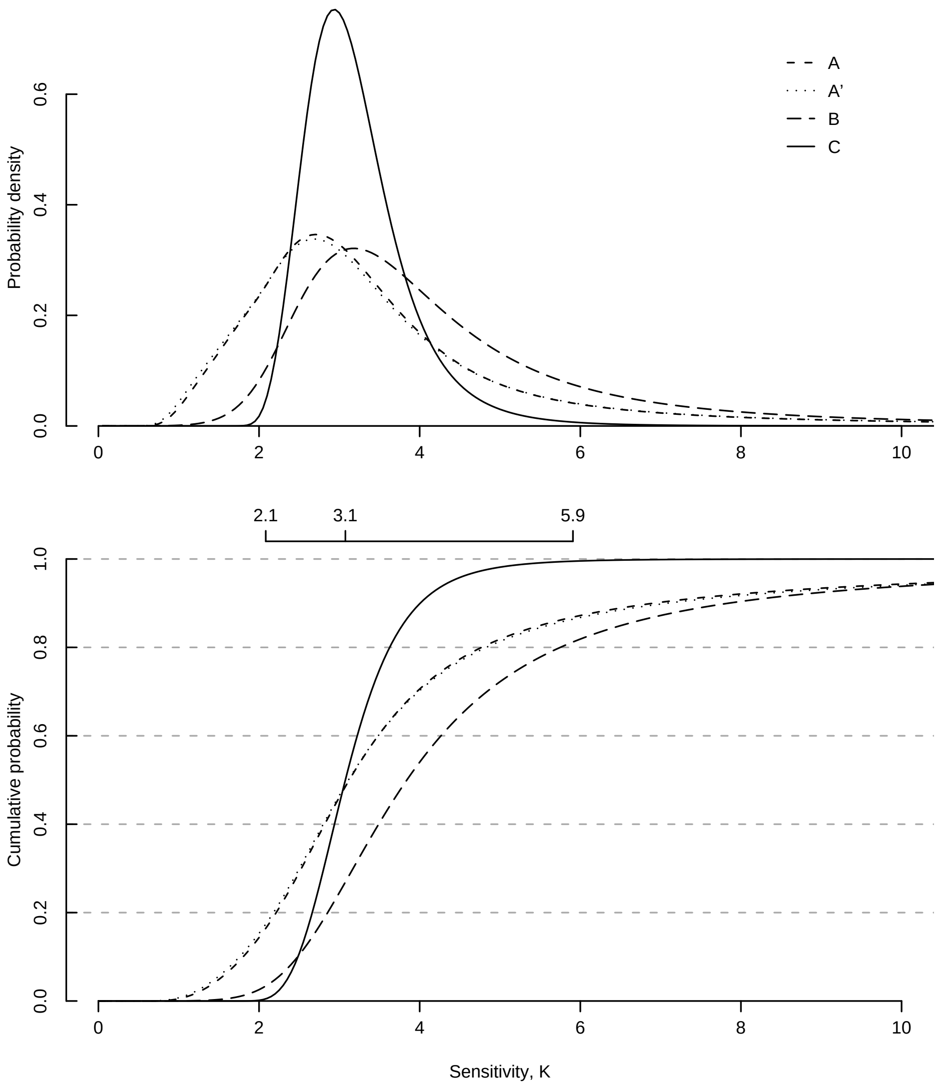
[back]

Leave-150-out diagnostic



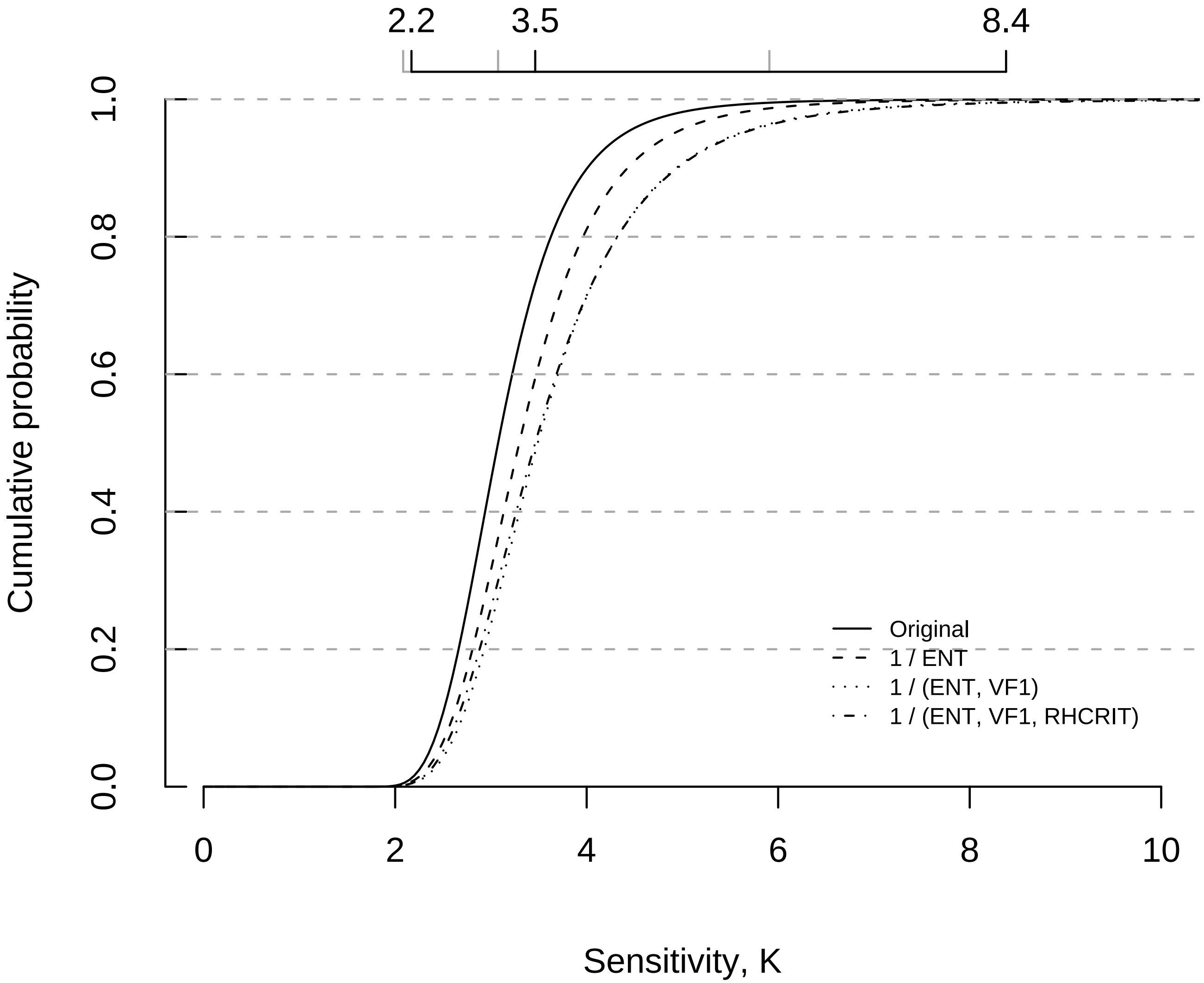
[back]

Predicting QUMP sensitivity



[back]

U or $1/U$?


2.2
3.5
8.4
1.0
0.8
0.6
0.4
0.2
0.0
Cumulative probability
Original
1 / ENT
1 / (ENT, VF1)
1 / (ENT, VF1, RHCRIT)
0
2
4
6
8
10
Sensitivity, K

[back]