AOE/CS/ME 6444 Verification and Validation in Scientific Computing
Spring 2020
Instructor: Dr. Chris Roy

Project Statement
Officially Due Wednesday May 6, 2020 at 10pm
(Note: you may have until Tuesday May 12th 10pm to turn it in w/out penalty)

Please upload your project in PDF format to the appropriate assignment section of Canvas. Please use the following file naming convention: VVSC_Lastname_Firstname_**Project**.pdf. If you have problems getting your project into PDF format, then let me know.

This final semester project builds on your earlier semester homework assignments, and adds in the assessment of predictive capability (i.e., total uncertainty quantification) for your application. Each relevant homework assignment should be revised according to the graded comments and incorporated into your project write up. In many cases, you will need to redo the work that you did for your homework in the context of the final simulation predictions that you will be presenting in the semester project since a number of you have changed your application, system, surroundings, or otherwise received important feedback on your homeworks.

New Aspects of Semester Project: Predictive Uncertainty

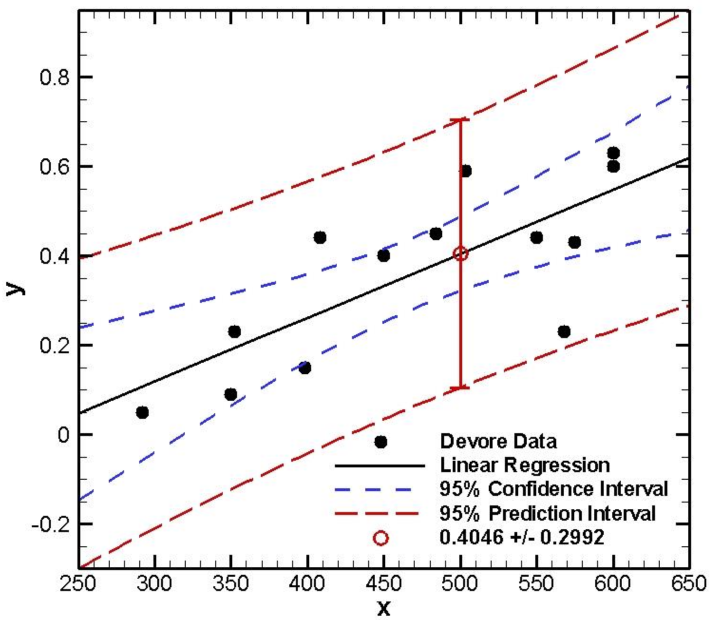
Extrapolation of Model Form Uncertainty

At this point, you should have computed a validation metric for your SRQ in Homework #5 with one model input being aleatory (i.e., random) and the other model inputs being deterministic. Choose one of the deterministic inputs to vary as a control parameter in order to extrapolate the validation metric to some application (i.e., prediction) condition of interest. For example, if you are using the heat conduction code and have computed the validation metric for a Thermal Design Power (TDP) of 125 W, then you may be interested in predicting the model form uncertainty for a TDP of 200 W. Assume (i.e., make up) some model form uncertainty values in a region around the location where you have already computed your validation metric. For example, if the validation metric at TDP = 125 W was d = 5 K, then assume validation metrics of 4 K, 5.5 K, and 5.25 K at TDP values of 100 W, 140 W, and 160 W, respectively. If you used the Modified Area Validation Metric, then you will need to extrapolate the d^+ and d^- values separately. Compute a linear regression of the validation metric(s) as a function of the TDP. Compute 95% prediction intervals about this linear regression for the TDP where you are interested in making a prediction (i.e., TDP = 200 W). The upper limit of this prediction interval will provide an estimated model form uncertainty at the prediction location.

You may want to use the MATLAB curve fitting tools (*cftool*) in order to easily compute these prediction intervals. To make sure you are implementing prediction intervals correctly, you may want to implement the following example problem first for computing prediction intervals at x = 500 based on 13 data points and a linear regression fit (from J. Devore, *Probability and Statistics for Engineering and the Sciences*, 7th Ed., Brooks/Cole, Belmont, CA 2009, page 446).

```
x =
398   292   352   575   568   450   550   408   484   350   503   600   600

y =
0.15  0.05  0.23  0.43  0.23  0.4   0.44  0.44  0.45  0.09  0.59  0.63  0.6
```



Devore Example for Computing Prediction Intervals

Uncertainty Propagation

The following deals with the application of your scientific computing code to a prediction condition of practical interest. First, choose (at least) two model input parameters in your application to be uncertain, with the rest being deterministic. For the first model input parameter, assume that the uncertainty is normally distributed as specified by an appropriate mean value and standard deviation that you must choose. For the second input parameter, assume that the uncertainty is epistemic and is given by an interval with upper and lower bounds that you also must choose. In choosing these parameters for your uncertain model inputs, make sure that they are physically meaningful (i.e., realistic) and that your chosen SRQ is at least somewhat sensitive to the expected range of input values. In an actual application, these uncertain model input distribution parameters should come from expert opinion, prior experience with the system, or experimental measurements. It is fine if your actual application has more than two uncertain inputs as long as you have at least one input that is probabilistic and at least one that this epistemically uncertain and characterized as an interval.

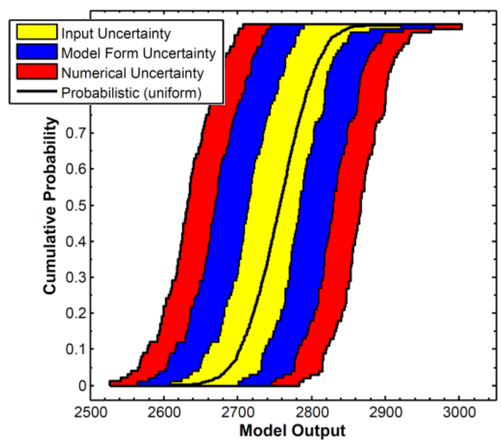
Using the nested sampling procedure discussed in class, compute a p-box for your chosen SRQ. The outer loop will sample over your epistemic uncertain input interval, and your inner loop will sample over your aleatory uncertain input distribution. Use your experience from Homework #5 to help you determine an appropriate number of Latin Hypercube samples over your aleatory input. For the outer loop (over the epistemic variable), use equal partition sampling (discussed in Section 3). If you only have one epistemic input variable, this will mean you break the interval up into N_e sub-intervals and then sample the endpoint of each sub-interval for a total of $N_e + 1$ epistemic samples.

Examine the effects of varying the number of epistemic samples N_e from 5 to 25 (and possibly to 100 if computationally feasible). Use the resulting SRQ values to generate the ensemble of CDFs, then determine the outer limit of these CDFs (i.e., the p-box). You will need to generate a p-box for each choice of epistemic sample size N_e. The grid, iterative tolerance, and digits of precision should be chosen in accordance with your findings from redoing HW#4 below (i.e., you will need to trade-off between speed of the simulations and numerical uncertainty).

Compare the results for the segregated sampling (discussed above) to those found by treating the epistemic uncertainty as a uniform distribution over the same range of values as chosen for the interval. In this latter approach, you should end up with only a single CDF as your simulation outcome rather than a p-box. Explain how treating the interval probabilistically (as a uniform distribution) affects the nondeterministic predictions.

Total Prediction Uncertainty

Find the total uncertainty in your predicted value. To do this, start with the p-box generated in the uncertainty propagation step. Add to that structure an interval on each side corresponding to the validation metric (either AVM or MAVM). Further add in the estimated total numerical uncertainty. Discuss the relative magnitude of uncertainty due to: aleatory inputs, epistemic inputs, model form, and numerical error. Give a recommendation on the best use of resources to reduce this uncertainty. Your results should look something like the figure to the right.



Representation of Total Uncertainty

Note: you will need to do many of the tasks discussed below *<u>before</u>* *estimating the total predictive uncertainty in your SRQ.*

Revisiting of Prior Homework Assignments: V&V

HW#2: Redo your discussion of the physical system, the model equations, and the numerical techniques used to encompass the actual simulations used for your semester project. (A number of you have changed your application, system, or surroundings significantly since HW#2).

HW#3: Redo your code verification studies to account for the values, boundary conditions, and physical models used in the final project application. You do not have to demonstrate full coverage of all options, but be clear about what options are verified and to what level (i.e., you may only see second-order accuracy when the code is formally third-order accurate).

HW#4: Redo your solution verification study at different points in your input uncertainty space to obtain estimates of the uncertainty due to numerical approximations that cover your prediction domain. You are free to choose these points in any sensible manner, but I recommend you at least examine the four combinations of inputs coming from the upper and lower values of your epistemic uncertain input and the $\pm 2\sigma$ values for your aleatory uncertain input. Take the largest estimate of total numerical uncertainty over these points, and then widen your final prediction p-box to account for uncertainty due to numerical approximations. Make sure to include estimates of uncertainty due to discretization error, iterative error, and round-off error. Use this part of the project to help you determine a good trade-off between numerical accuracy and the efficiency of the computations for your uncertainty propagation discussed above.

HW#5: Use a validation metric that you computed from Homework #5 along with the "*Extrapolation of Model Form Uncertainty*" process discussed above to estimate the model form uncertainty at your prediction location. Extend the width of the final prediction p-box to account for estimated model form uncertainty at the prediction location.

Final Project Report and Results

Compile the results of the tasks described above in a formal project report. This report should be as short as possible while still conveying the important information described here. You may reference earlier homeworks and/or use appendices as needed. The length should be comparable to a technical paper at a conference, with a final length of no more than 12 pages (not including the references or appendices). This report should also include a discussion of your VV&UQ results along the way and a final discussion of predictive uncertainty in your chosen SRQ suitable for reporting the results of your study to a decision maker. The clarity and grammatical correctness of your report is important. For report format, you may use any reasonable conference paper or journal format. One possibility is the AIAA conference paper format (see: https://aviation.aiaa.org/TechPresenterResources/ which has a PDF style guide and MSWord and LaTeX templates).