Should I Trust that Model?

November 3, 2016

Val Snow, guest contributor and member of SESYNC's Core Modelling Practices [1] synthesis team.

Should I Trust that Model?

By Val Snow

This blog post originally appeared [2] in the Integration and Implementation Insights blog (http://I2Insights.org [3]) as "Should I Trust that Model," and is reposted with the author's permission.

How do those building and using models decide whether a model should be trusted? While my thinking has evolved through modelling to predict the impacts of land use on losses of nutrients to the environment – such models are central to land use policy development – this under-discussed question applies to any model.

In principle, model development is a straightforward series of steps:

- Specification: what will be included in the model is determined conceptually and/or quantitatively by peers, experts and/or stakeholders and the underlying equations are decided
- Coding: the concepts and equations are translated into computer code and the code is tested using appropriate software development processes
- Parameterisation: here the values that go into the equations are determined by a variety of methods
- Testing: the model is compared against data using any of a wide range of metrics, the comparisons are examined and the fitness of the model for the intended purpose or scope is decided. Bennett and colleagues (2013) give an excellent position on the variety of statistical approaches that can be used for this purpose.

In reality, of course, these steps do not take place in an orderly progression, there are many loops backward, some of the parameterisation and testing occurs in parallel to the coding and the first step is often re-visited many times.

It is mostly assumed that assessment of 'trust' or 'confidence' in a particular model should be based on the metrics or statistics resulting from the comparison of the model outputs against experimental datasets. Sometimes, however, the scope of the testing data and whether the model has been published in a good journal are also taken to imply confidence in the model. These criteria largely refer to that last testing step and this focus is

understandable. Of the steps above, testing is the one mostly readily documented against accepted standards with the results made available externally. However, even with a quantitative approach to testing, Bennett and colleagues note that the actual values of the statistics that are considered to be acceptable are a subjective decision.

While I agree with the approach and need for quantitative testing, the testing results themselves have very little to do with my confidence or trust in a model. My confidence will evolve over time as I become more familiar with the model. By the time I am prepared to make any statements about the specific reasons for my degree of trust, the reasons for that trust will largely have become tacit knowledge – and that makes it very difficult for me to explain to someone else why I have confidence (or not) in that model.

Here I have attempted to tease out the factors that influence my confidence in a model. I should note that my trust in the models I have been involved in developing, or that I use at an expert level, can fluctuate quite widely and wildly over time so, for me, the process of developing trust is not a linear process and is subject to continual revision. I assess four key areas concerning the model using a range of questions, as follows:

Area 1. The nature of the problem domain: Are the 'correct' outputs even measureable? How mature is the science community's understanding and agreement of the conceptual and quantitative processes that must be included in the model? What constraints and deliberate assumptions have been included? Will these assumptions likely constrain error or allow (or even encourage) it to blossom?

Area 2. Software development and parameterisation: Who did the work and do I have a favourable opinion of their other modelling activities? What documented software development processes did they use? Do they use a reliable version control system and can I compare older versions of the model to the current version? Is the documentation sufficient and sufficiently well-presented that I can, for the large part, understand the workings of the model and its implementation assumptions? If I need more detail can I (or can I get someone else to) dive into the code to understand more detail? How open/transparent does the process appear to be? Can it be readily reviewed by others?

Area 3. Developer's testing: What have the developers done with respect to testing? Does it feel robust (eg., basic things like not reusing data used for parameterisation, but also have they delved into and explained reasons for poor performance)? Have they relied mostly on reporting statistical values or are there extensive graphs that are appropriate for the domain of the model?

Area 4. User's experience: Is the model user interface set up in such a way that I can investigate the model's behaviour as inputs, settings and parameters are changed? When I do this investigation, how often does the model 'surprise' me? How many of those are "Wow!" surprises (meaning I thought the model would be unlikely to behave well but it did), how many are surprising surprises (the model outputs can be rationalised and even make sense once investigated) and how many are "Really!?!" surprises (the model outputs do not make sense in any way that I can explain and/or they seem to be in conflict with the developer's testing or documentation)? When I get the last type of surprise: is the model constructed in such a way that I can understand the extent to which that surprise will flow through to outputs that matter or are the effect of any such surprises likely to be minimised or cancelled out by the way the model is constructed?

These questions are how I develop trust in a model. Do my questions align with your criteria or have I missed critical points? Do you have a completely different process for developing trust in a model? My approach is probably strongly tuned by my experience with mechanistic or process-based models (where the model is intended to represent an expert's opinion of how the system works rather than being driven by data). Given that, if you work with a different type of model, does your approach to developing trust work differently? Might you place more reliance on comparison to data? I'd value your thoughts.

Reference:

Bennett, N. D., Croke, B. F. W., Guariso, G., Guillaume, J. H. A., Hamilton, S. H., Jakeman, A. J., Marsili-Libelli, S., Newham, L. T. H., Norton, J. P., Perrin, C., Pierce, S. A., Robson, B., Seppelt, R., Voinov, A. A., Fath, B. D., Andreassian, V., (2013). Characterising performance of environmental models. *Environmental Modelling & Software*, 40: 1–20. Online (DOI): 10.1016/j.envsoft.2012.09.011 [4]

Biography: *Val Snow is a systems modeller at AgResearch in New Zealand and comes from a soil physics and*

agricultural science background. Her research focuses on the development and use of simulation models to support technological innovation in pastoral agricultural systems and assessment of the impacts of land use. Application areas include land use policy, future farming systems, greenhouse gas mitigation and climate change adaptation.

Associated Project:
[Core Modeling Practices](#) [5]
Audience:
[Researcher](#) [6]
[Educator](#) [7]
[Cyberinfrastructure](#) [8]

Source URL: https://www.sesync.umd.edu/blog/thu-2016-11-03-1432/should-i-trust-that-model

Links
[1] http://www.sesync.org/project/enhancing-socio-environmental-research-education/model-process-practices
[2] https://i2insights.org/2016/10/06/trust-in-model/
[3] http://i2insights.org
[4] http://www.sciencedirect.com/science/article/pii/S1364815212002435
[5] https://www.sesync.umd.edu/project/enhancing-socio-environmental-research-education/model-process-practices
[6] https://www.sesync.umd.edu/audience/researcher
[7] https://www.sesync.umd.edu/audience/educator
[8] https://www.sesync.umd.edu/audience/cyberinfrastructure