Hydrol. Earth Syst. Sci. Discuss.,
https://doi.org/10.5194/hess-2020-128-RC1, 2020

Hydrology and Earth System Sciences
Open Access
Discussions

Interactive comment on "Technical note: "Bit by bit": A practical and general approach for evaluating model computational complexity vs. model performance" *by* Elnaz Azmi et al.

Elena Toth (Referee)

elena.toth@unibo.it

Received and published: 29 May 2020

The Technical Note addresses a very timely research question, covered also in a recent WRR debate on the role of Information Theory for helping to understand the complexity of Earth Systems. In particular, the Note attempts to provide some examples for testing/sustaining some of the ideas presented in the WRR debate, that includes also one contribution by two of the Authors (Weijs and Ruddell, 2020). I find the theme extremely interesting and I believe that proposing a way (using 'Strace') to calculate computational effort without the need to refer to the specific machine where the computations run is indeed brilliant and novel and worthy to be published. On the other

hand, I confess that on one hand I find the Note a bit too theoretical and on the other hand I have some doubts on the soundness of the comparison that is presented.

The note often refers to the Weijs and Ruddell (2020) paper but without well clarifying the relationship between such paper and the present work (what is the content of the previous paper and what is different here). A good part of the theoretical discussion, indeed a truly philosophical one, as such was the 'line' of the WRR debate, is repeated here, in a first part (three pages of introduction) that is definitely too long and too much theoretical for a technical note and may be substantially shortened, referring to the previous publication.

But my main concern is that the comparison of the models is not fair, since they do not make use of the same information and this is instead crucial in a work focussing on information theory. Looking at Table 1, last column, we may see that the data used for running the models are not the same. In fact, the bucket models (Models 02 and 05) do not use any streamflow data in any way for the simulation but only for calibrating the parameters. The same holds for the ANN (Model-08) since only P is provided in input. On the other hand, the autoregressive model (Model-07) uses only past streamflow values as input. It is well known that for a short lead time (the models are here used as simulation models, with lead-time equal to one), the recent measures of the streamflow (Q) is much more informative than the rainfall values, that in real-time flow forecasting become more and more important when the lead-time increases, since Q encapsulates a lot of useful information on the catchment behaviour, and it may be seen as a very good approximation of the catchment conditions. Therefore it was easily predictable that the autoregressive model (Model-07) would have outperformed the other models, independently of its complexity, due to the different input information they use.

Thus, leaving aside the analysis of the performances (expected, due to the setting up of the models), the interesting part of the results is the analysis of the complexity. Section 3.2 and Figure 3 show that, a part from Model-03 and Model-08 (ANN), all the other complexities are very close. The reason for the high computational complexity of Model-03 is the excessively (and not necessary) fine time step. The reason for the high computational complexity of Model-08, that is the Artificial Neural Network, may certainly be inherent in the structure of these kind of models, that tend to have a relatively high number of parameters (but the internal parameters are in some cases not all influential and since, despite the high number of parameters, ANNs generally work very well on independent data, they cannot be blamed of overfitting/overtraining). But in addition, in this case the ANN model is not only fed by the "wrong" input (P instead of Q), but its architecture is also certainly more complex than needed: why using 10 hidden nodes? If it were used, as it should, in a way that is consistent with the regressive model, it should be fed by the last streamflow values rather than (or in addition to) some past rainfall (needed especially if considering longer lead-times) and it would perform much better than now. And probably a few hidden nodes would be more than enough (as proved in many previous works where such models rival with more complex conceptual models in forecasting/updating mode), so its complexity would be less.

Due to the potential of the bit-by-bit concept, and the utility to be able to measure computational complexity through 'Strace', I do encourage the Authors to perform and present a more fair comparison and then focussing and explaining the differences, in performances and complexity, found in models that use indeed the same input information content (and have the most parsimonious structure that is possible).

SPECIFIC COMMENTS

Abstract: ll.12-20: may be summarised.

Pages 2 to 4 may be summarised in one page, referring to Wejis and Ruddell (2020) for the philosophical discussion.

Eq. 1: I would suggest to move eq 1 inside Table 1 (Model-07 row)

ll. 155-158: actually I would have found very interesting an evaluation of out-of-sample

performance of the models, since this is indeed crucial for data-driven models and it would be very useful to understand what each model is able to infer on the behaviour of the basin on independent data, to analyse their generalisation ability.

Second part of Section 2.4.1: I think that more detail on the meaning and computation of entropy is necessary, since it is a 'niche' not widely known to the readers.

ll. 266-269: can you explain the differences in computational complexity between Models 00 and 01? I would have expected their complexity to be practically null for both, since they do not need to make computations at each step. . .

Interactive comment on Hydrol. Earth Syst. Sci. Discuss., https://doi.org/10.5194/hess-2020-128, 2020.