Hydrol. Earth Syst. Sci. Discuss.,
https://doi.org/10.5194/hess-2020-464-RC1, 2020

Hydrology and Earth System Sciences
Discussions

Interactive comment on "Technical Note: Temporal Disaggregation of Spatial Rainfall Fields with Generative Adversarial Networks" *by* Sebastian Scher and Stefanie Peßenteiner

Jussi Leinonen (Referee)

jussi.leinonen@meteoswiss.ch

Received and published: 14 December 2020

Review by: Jussi Leinonen

I'm referring a fair bit to my own work in this review. I do believe these are valid comments and suggestions that could benefit the authors' work, but in the interest of transparency, I'm signing this review with my own name.

In this manuscript, the authors demonstrate the use of a generative adversarial network on rainfall disaggregation, that is, given a daily total of rainfall as a 2D field, they generate rainfall fields at an hourly resolution that sum up to the daily total at each

point.

The paper is generally clearly written and the methods are presented adequately, although the short-form "Technical Note" format is probably keeping the authors from providing as much explanation as they could have. In any case, appropriate references are provided where the methods are not described in full.

The authors' approach is promising although I feel there are a few shortcomings that should at least be discussed in the text. First, the resolution of the generated fields is very limited, just 16x16 pixels. With the GAN architecture the authors are using, there doesn't seem to be a way to use the trained network to generate rainfall fields of larger extent. If the network is used repeatedly on adjacent 2D tiles of a larger rainfall field, one cannot enforce continuity between the neighboring tiles. Similarly, the generated field is limited to a single day and there is no way to ensure continuity of the rainfall field between the last hour of a given day and the first hour of the next.

In our recent paper (doi:10.1109/TGRS.2020.3032790, or https://arxiv.org/abs/2005.10374), coauthors and I addressed these two issues by using a fully convolutional architecture to enable the use of the network with different-sized inputs after training, and a recurrent layer to handle the time continuity. Looking at your code in GitHub, I think that the fully convolutional architecture, in particular, would be quite easy to implement and would allow you to generate disaggregated rainfall maps for the entire Sweden with the current training dataset. If you want, I could send you a pull request as an example on how to modify the generator and the critic - let me know!

In any case, I think the paper is publishable with minor revisions as it demonstrates the potential of the method in this application quite effectively and I think this is enough of an incremental advance at this stage, but giving more thought to the abovementioned issues would have made it a better paper. Hopefully these issues can be addressed in later work by the authors.

Specific comments:

Figures: The rainfall maps shown in many figures are really tiny, making these figures quite a strain to the eye to look at. Of course in the PDF one can zoom in but it should be possible to look at the figures in print as well. I'm not sure how to best improve them given that you have 24 frames, but maybe it would be possible to show fewer examples in larger size?

Line 130: Using the day of the year as an input sounds quite susceptible to overfitting as the network can more easily just memorize some input combinations (especially as you're discarding non-raining days). Maybe this is why you're seeing a degraded performance after adding it?

Lines 144-151: This discussion only seems to be about Fig. 4? This is not mentioned here, and also nothing about Fig. 3 is discussed.

Lines 148-149: "In rows 3-4" - minor suggestion, maybe you should number the rows in the figures?

Fig. 3: Where is the prominent feature in the middle of the real precipitation coming from? This seems to be almost stationary throughout the day. Is this real precipitation or a radar artifact? (Also Fig. B1 seems to have a similar feature)

Lines 174-175: Due to the softmax activation, the generator output cannot go to 0 either, so your network also cannot generate zero precipitation (except where the daily total is 0).

Fig. 6a: There seem to be noticeable increases in the generated fractions at hours 2 and 23. I wonder if these could be due to edge artifacts of the convolutional network.

Line 260: Why set β_1 in Adam to 0? This basically disables this feature of the Adam optimizer. No problem if it works for you, but did you follow some particular implementation?

Interactive comment on Hydrol. Earth Syst. Sci. Discuss., https://doi.org/10.5194/hess-2020-464, 2020.