Geosci. Model Dev. Discuss., referee comment RC1
https://doi.org/10.5194/gmd-2021-353-RC1, 2021

Comment on gmd-2021-353

Anonymous Referee #1

Referee comment on "DINCAE 2.0: multivariate convolutional neural network with error estimates to reconstruct sea surface temperature satellite and altimetry observations" by Alexander Barth et al., Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2021-353-RC1, 2021

Review of

DINCAE 2: multivariate convolutional neural network with error estimates to reconstruct sea surface temperature satellite and altimetry observations

The paper presents an update on the previous version of DINCAE, a convolutional autoencoder method for in-painting of sparse satellite data. DINACE2 presents some improvements over the previous version, most notably in performance (vs DINCAE1), speed (presumably due to being rewritten to Julia from Python) and an option to treat ungridded data like satellite altimetry observations. It also introduces an extra refinement step in the cost function to increase its depth, and an intermediate loss term is included in the total loss to compensate for the vanishing gradients of the deep network. When treating sparse data, the error variance estimation of DINCAE2 is more reliable than that from variational interpolation method DIVAnd. The results are solid, the paper is well written, the figures are clear. I recommend publication after minor revision. I do have some comments which might be worth discussing further.

Specific comments:

page 4, eq4: In the comment to equation 4, the authors state that CAE refinement leads to a deeper network and thus to potential worsening of the vanishing gradient problem. They attempt to mitigate this by including intermediate loss term into the total loss function. They state that by doing so, the vanishing gradient problem is reduced. Can the authors perhaps illustrate more clearly that this is indeed the case? Is there a way to say a bit more about this, so that the reader does not have to take the author's word for this?

Also, wouldn't an arbitrary number of refinements further exacerbate the vanishing gradient problem? Which term would be dominant in this case – adding further refinement steps versus including further intermediate losses to the total loss?

Page5: when handling missing data there is an interpretation throughout the text that setting the missing value to zero corresponds to an infinitely large error. This is undoubtedly true for variables which are normalized by their variance, as those in this paper. However there are a number of other scalings where variables are not normalized by their variance. In these cases, it seems to me, the authors interpretation is not the most appropriate. I would propose an independent interpretation that setting the missing values to zero simply numerically means that there will be no back-propagation of error from those missing data – thus the training can continue without any impact from the missing data. This interpretation does not have anything with the specific variable normalization at hand.

A cosmetic remark. The hyperparameters were tuned using Bayesian optimization, which seems adequate. Let's say hyperparameter optimization gives you an optimal network. Separate instances of training this same optimal network (with the same fixed optimal hyperparameters) would provide separately trained versions of this same network. We can use these set of the same network to create a set of predictions. What is the error variance of this set of predictions? I would expect that this error variance is on the order of a 5%, and hence MUCH larger than the stated precision of the RMS errors (4 decimal places) in Table 2. Long story short, perhaps 4 decimal placed is an overkill of precision.