Using machine learning to correct model error in data assimilation and forecast applications

Alban Farchi[†*], Patrick Laloyaux[‡], Massimo Bonavita[‡], and Marc Bocquet[†]

[†] CEREA, joint laboratory École des Ponts ParisTech and EDF R&D, Université Paris-Est, Champs-sur-Marne, France
[‡] ECMWF, Shinfield Park, Reading, United Kingdom
[*] alban.farchi@enpc.fr

Introduction

Recent developments in machine learning (ML) have demonstrated impressive skills in reproducing complex spatiotemporal processes. However, contrary to data assimilation (DA), the underlying assumption behind ML methods is that the system is fully observed and without noise, which is rarely the case in numerical weather prediction. In order to overcome this issue, it is possible to embed the ML problem into a DA formalism characterised by a cost function similar to that of the weak-constraint 4D-Var [2]. In practice ML and DA are combined to solve the problem: DA is used to estimate the state of the system while ML is used to estimate the full model.

In realistic systems, the model dynamics can be very complex and it may not be possible to reconstruct it from scratch. An alternative could be to learn the model error of a physical model using the same approach combining DA and ML. In this poster, we test the feasibility of this method using a quasi geostrophic (QG) model.

A Bayesian framework for ML and DA

The system evolution is governed by the state equation

$$\mathbf{x}_{k+1} = \mathcal{M}^{\mathsf{t}}_k(\mathbf{x}_k), \tag{1}$$

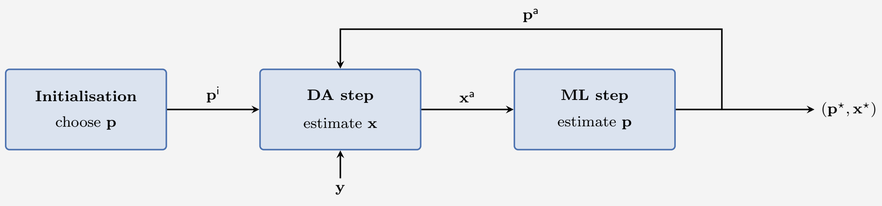
where $\mathcal{M}^{\mathsf{t}}$ is the resolvent of the unknown true dynamical model. In realistic systems, the true system state $\mathbf{x}^{\mathsf{t}}$ is only known through *sparse* and *noisy* observations $\mathbf{y}$ via

$$\mathbf{y}_k = \mathcal{H}_k(\mathbf{x}^{\mathsf{t}}_k) + \mathbf{v}^{\mathsf{t}}_k. \tag{2}$$

The goal is to provide a *surrogate* model $\mathcal{M}^{\mathsf{ml}}$ to emulate $\mathcal{M}^{\mathsf{t}}$. $\mathcal{M}^{\mathsf{ml}}$ is typically a neural network (NN) and depends on a set of parameters $\mathbf{p}$ (weights and biases). A Bayesian framework for this problem is to minimise the cost function [2]

$$\mathcal{J}(\mathbf{p}, \mathbf{x}_{0:N_{\mathsf{t}}}) = \frac{1}{2}\sum_{k=0}^{N_{\mathsf{t}}} \left\| \mathbf{y}_k - \mathcal{H}_k(\mathbf{x}_k) \right\|_{\mathbf{R}^{-1}} + \frac{1}{2}\sum_{k=0}^{N_{\mathsf{t}}-1} \left\| \mathbf{x}_{k+1} - \mathcal{M}^{\mathsf{ml}}_k(\mathbf{p}, \mathbf{x}_k) \right\|_{\mathbf{Q}^{-1}}. \tag{3}$$

Recognising that the parameters $\mathbf{p}$ and the system trajectory $\mathbf{x}_{0:N_{\mathsf{t}}}$ are of different nature, [2, 3] suggest to use a coordinate descent to minimise this cost function.



We want to use this method to correct the error of an original, *physical* model $\mathcal{M}^{\circ}$. In this case, we have to replace $\mathcal{M}^{\mathsf{ml}}$ by the *hybrid surrogate* model $\mathcal{M}^{\mathsf{t}} + \mathcal{M}^{\mathsf{ml}}$ in eq. (3). If the true trajectory $\mathbf{x}^{\mathsf{t}}$ were known (dense, noiseless observations) then the NN could be trained with $\mathbf{x}^{\mathsf{t}}_k \mapsto \mathbf{x}^{\mathsf{t}}_{k+1} - \mathcal{M}^{\circ}_k(\mathbf{x}^{\mathsf{t}}_k)$. With sparse and noisy observations, we need to use:

- the *analysis* $\mathbf{x}^{\mathsf{a}}_k$ in place of the truth $\mathbf{x}^{\mathsf{t}}_k$;
- the *analysis increments* $\mathbf{x}^{\mathsf{a}}_{k+1} - \mathcal{M}^{\circ}_k(\mathbf{x}^{\mathsf{a}}_k)$ in place of $\mathbf{x}^{\mathsf{t}}_{k+1} - \mathcal{M}^{\circ}_k(\mathbf{x}^{\mathsf{t}}_k)$.

Correcting the error of $\mathcal{M}^{\circ}$ is likely to be an easier inference problem than learning the full model $\mathcal{M}^{\mathsf{t}}$. We will focus on the first DA–ML cycle.

The quasi-geostrophic model

The QG model expresses the conservation of (non-dimensional) potential vorticity q for two layers of constant potential temperature in the $x - y$ plane:

$$\frac{\mathrm{d}q_1}{\mathrm{d}t} = \frac{\mathrm{d}q_2}{\mathrm{d}t} = 0. \tag{4}$$

The potential vorticity q is related to the *stream function* ψ through

$$q_1 = \Delta\psi_1 - F_1(\psi_1 - \psi_2) + \beta y, \tag{5}$$
$$q_2 = \Delta\psi_2 - F_2(\psi_2 - \psi_1) + \beta y + R(x, y). \tag{6}$$

The domain is periodic in the x direction, and fixed boundary conditions are used for q in the y direction. We use a discretisation of 40×20 points.

The control vector is ψ on both layers, which gives a total of $N_{\mathsf{x}} = 1600$ variables. The evolution of ψ s characterised by a wave, slowly moving towards the west, with a mean period around 16 days. The model is chaotic, with a doubling time of errors around 250 hours.

Model error is introduced as *perturbed parameters* (layer depths and orography) and doubled *integration time step*. The idea is to mimic errors due to deficiencies in the physical parametrisation, limited spatial resolution, or erroneous time integration. See fig. 1 for an illustration.

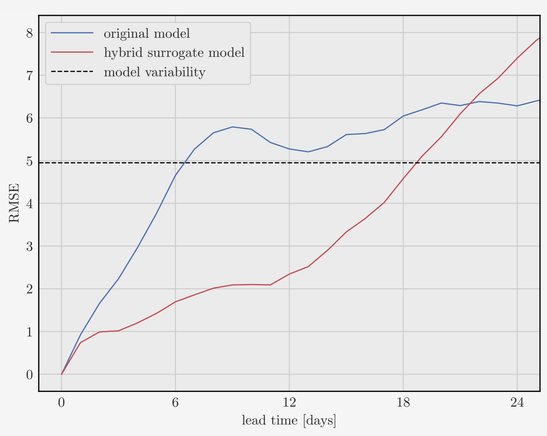


Figure 2: Typical forecast skill of the original and the hybrid surrogate models.

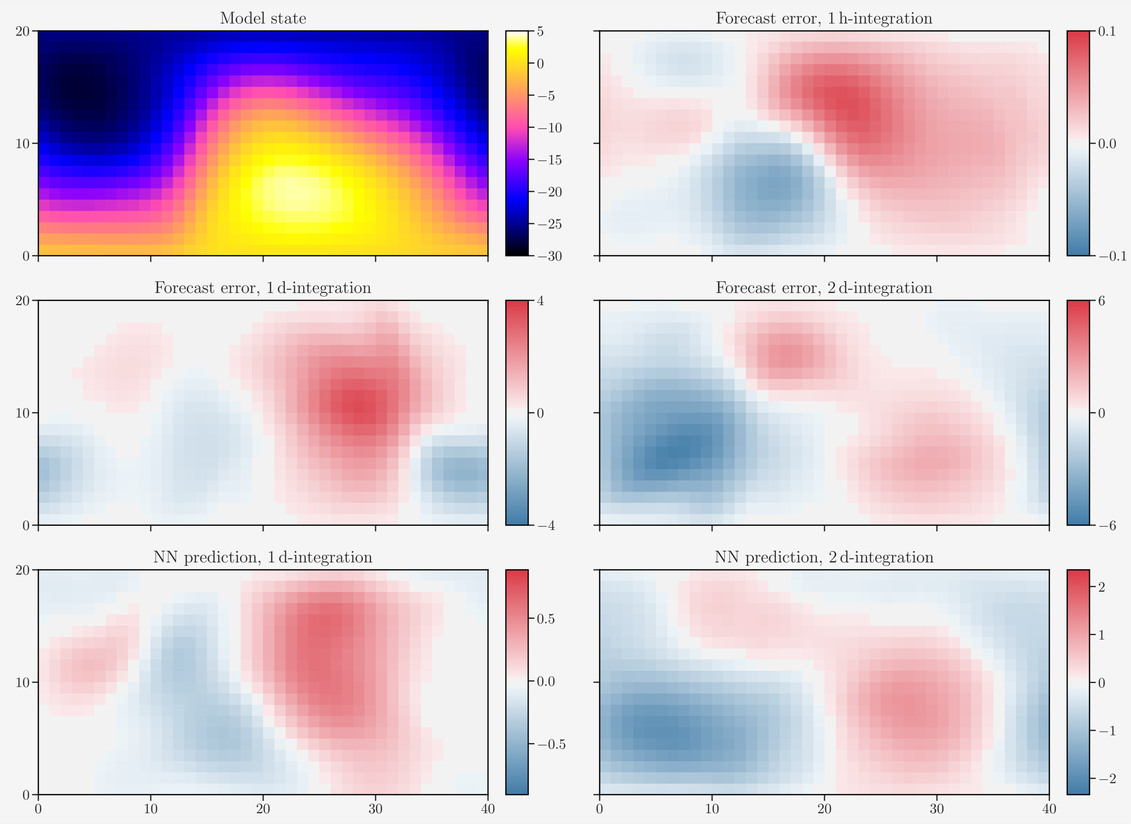


Figure 1: Typical snapshot of the QG model. Top left: model state. Top right, middle left and right: forecast error starting from this model state for several lead times. Bottom left and right: corresponding NN predictions.

Neural network training

We first perform a long cycled DA experiment with $\mathcal{M}^{\circ}$, the perturbed QG model. Observations are available at 50 random locations every 2 hours. The observation noise is about 2 % of the model variability. We use the *strong-constraint 4D-Var* algorithm with consecutive windows of 24 hours. The time-average analysis RMSE is about 5 % of the model variability.

We use the analysis increments to train small NNs. Depending on the sampling frequency of the ML step, the NNs are able to explain 80 % to 90 % of the analysis increments variance, but only 30 % to 85 % of the model error variance. Visually, the NNs predict *correct model error patterns*, but incorrect scaling. Decreasing the sampling frequency of the ML step can help, because it increases the relative accuracy of the analysis increments. See fig. 1 for an illustration.

The accuracy of mid- to long-range forecasts with the hybrid model are then evaluated using the *forecast skill*, *i.e.*, the time evolution of the forecast error starting from the same initial condition. See fig. 2 for an illustration. The hybrid model indeed yields better forecasts than the original model, and the correction is effective until 10 to 16 days. A systematic comparison of different NNs and training setups shows that:

- only short trajectories are necessary to train these small NNs;
- nonlinear NNs noticeably outperform linear NNs;
- fully-connected layers can be (to some extent) replaced by convolutional layers, which are more scalable to high-dimensional problems.

Corrected data assimilation step

Finally, we want to evaluate the potential improvements from the correction in a subsequent 4D-Var experiment. From a technical viewpoint, we assume a *linear error growth* in time. The model integration for a single time step δt is given by

$$\mathbf{x} \mapsto \mathcal{M}^{\circ}(\mathbf{x}) + \tau\delta t\mathcal{M}^{\mathsf{ml}}(\mathbf{x}), \tag{7}$$

where τ is the sampling frequency of the ML step. Depending on τ, using the hybrid model yields up to 25 % *reduction* in the analysis RMSE.

sampling period $1/\tau$	1 d	2 d	4 d	8 d	no correction
RMSE	0.18	0.20	0.21	0.24	0.24

The linear error growth in time is a limiting factor for a more accurate corrected DA. Ideally, the NN should be able to predict the model error for short-term forecasts, in particular shorter than the DA window.

References

[1] M. Bocquet, J. Brajard, A. Carrassi and L. Bertino. 'Data assimilation as a learning tool to infer ordinary differential equation representations of dynamical models'. In: *Nonlinear Process. Geophys.* 26.3 (2019), pp. 143–162.

[2] M. Bocquet, J. Brajard, A. Carrassi and L. Bertino. 'Bayesian inference of chaotic dynamics by merging data assimilation, machine learning and expectation-maximization'. In: *Found. Data Sci.* 2.1 (2020), pp. 55–80.

[3] J. Brajard, A. Carrassi, M. Bocquet and L. Bertino. 'Combining data assimilation and machine learning to emulate a dynamical model from sparse and noisy observations: A case study with the Lorenz 96 model'. In: *J. Comput. Sci.* 44 (2020), p. 101171.

[4] J. Brajard, A. Carrassi, M. Bocquet and L. Bertino. 'Combining data assimilation and machine learning to infer unresolved scale parametrisation'. In: *arXiv:2009.04318* (2020).

[5] A. Farchi, P. Laloyaux, M. Bonavita and M. Bocquet. 'Using machine learning to correct model error in data assimilation and forecast applications'. In preparation. 2020.