Using machine learning in geophysical data assimilation

(some of the issues and some ideas)

Alberto Carrassi

Dept of Physics and Astronomy, U of Bologna - IT

Dept of Meteorology & National Centre for Earth Observation, U of Reading - UK

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

University of Reading

Workshop: SM & FT 2022

Frontiers in Computational Physics

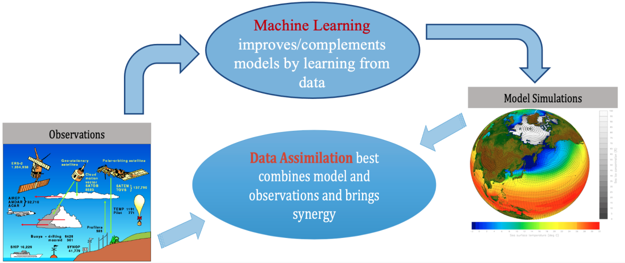
Bari, IT, 20 December 2022

DA from model-driven to *(a bit more)* data-driven

Model-driven DA in geosciences

- In geosciences we possess a "good knowledge" of the laws governing the system.
- The DA ability to combine model and data has been pivotal to the success of DA.
- Using the model, information propagates from observed to unobserved regions.

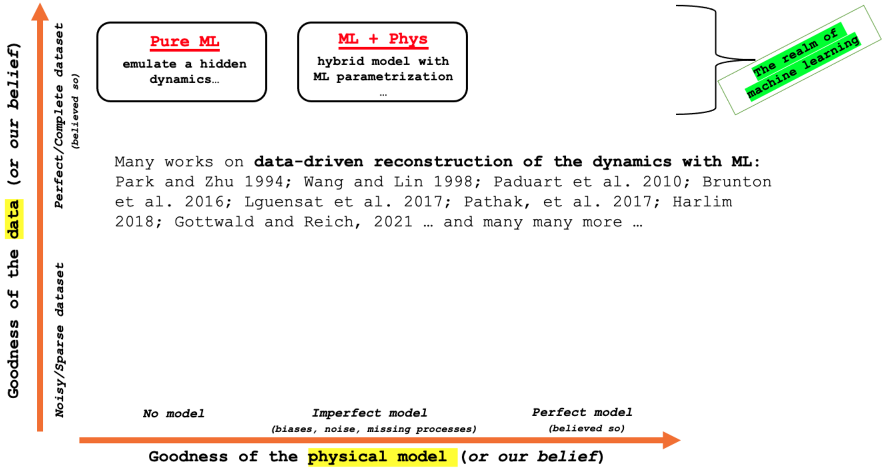
- But models are not perfect and neither complete.
- Recently, machine learning tools have shown formidable in retrieving hidden dynamics only from data.



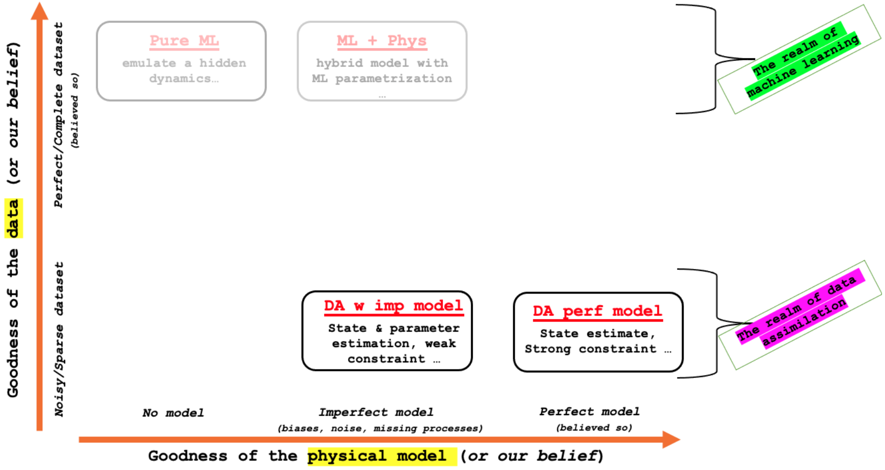
Data-driven DA

How making DA and ML joining forces

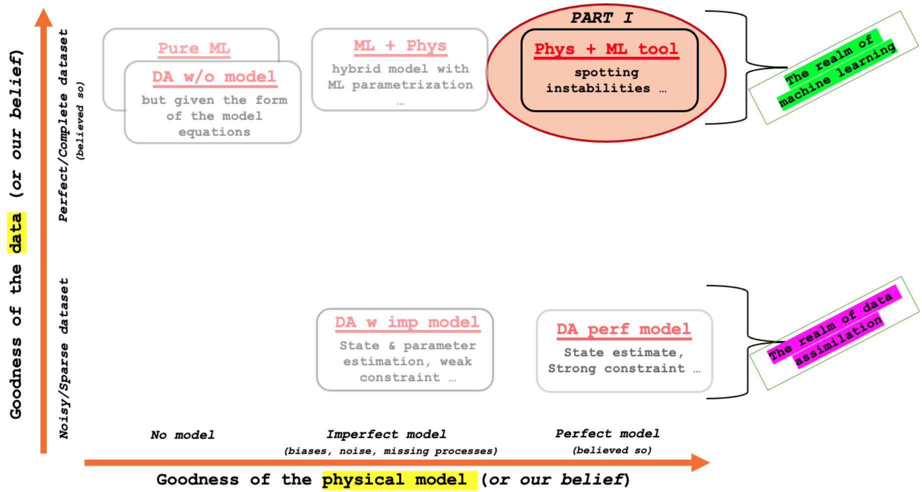
ML and DA: their (different?) *"realms"* and *"goals"*



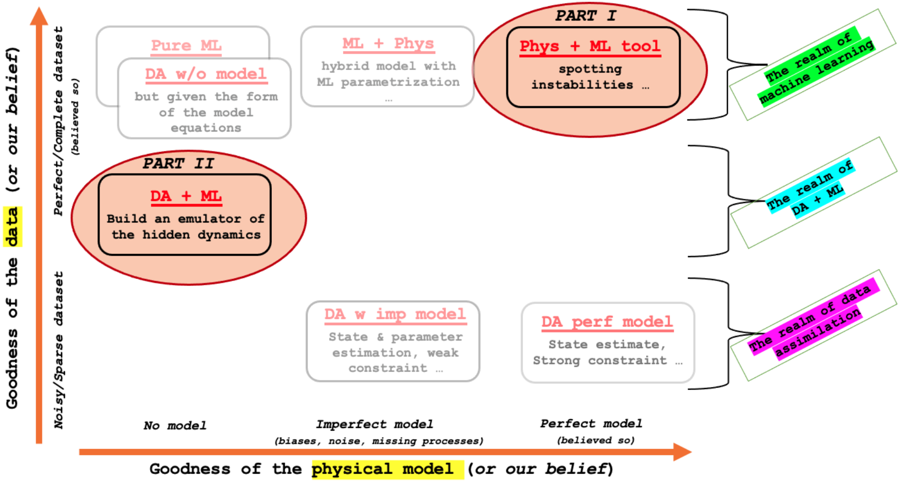
ML and DA: their (different?) *"realms"* and *"goals"*



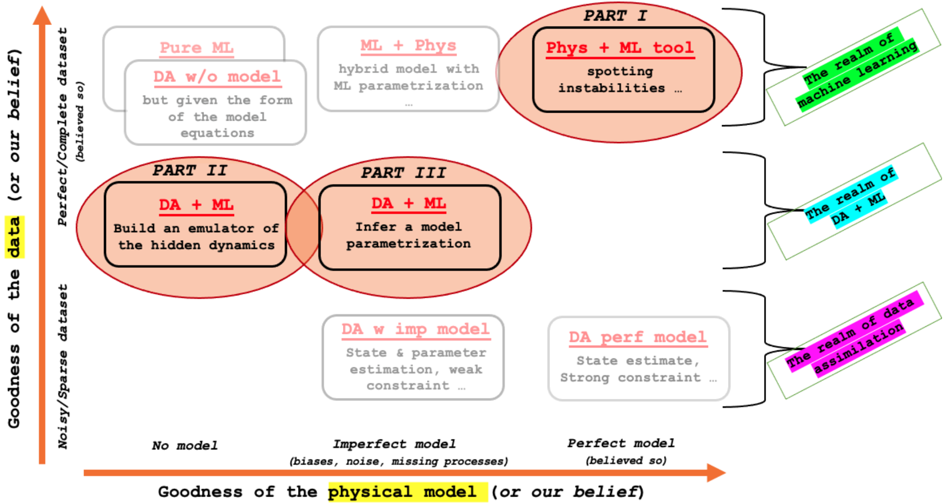
What this *talk talks* about



What this *talk talks* about



What this *talk talks* about



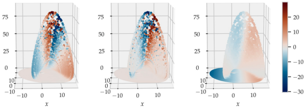
Outline

Part I: "*non intrusive*" ML - A tool supplementing physical models

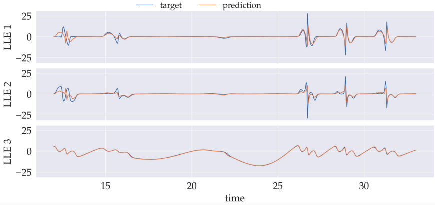
- ▶ We already proved that is possible to estimate the LE using DA (**Chen, et al. 2021**).
- ▶ Here we investigate how to use ML for real-time estimate the LLEs based on the system's state (**Ayers et al., 2022, ArXiv**).

Rossler model

LLE distribution

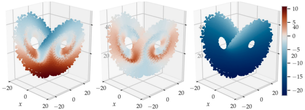


Prediction with Multi Layer Perceptor

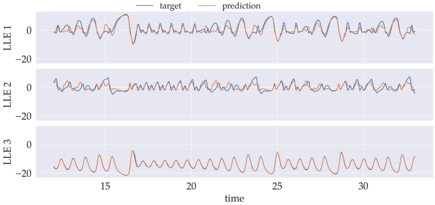


Lorenz 63 model

LLE distribution



Prediction with Convolutional Neural Network

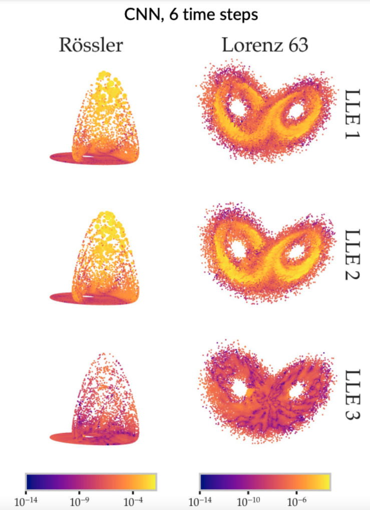


"*non intrusive*" ML - A tool supplementing physical models

- Very good prediction of LLE_3.
- Good prediction of LLE_1
- LLE_2 more difficult to predict
- Accuracy of the prediction strongly depends on "where we are" on the attractor $\Longleftrightarrow$ **Areas of high heterogeneity are difficult to map**.

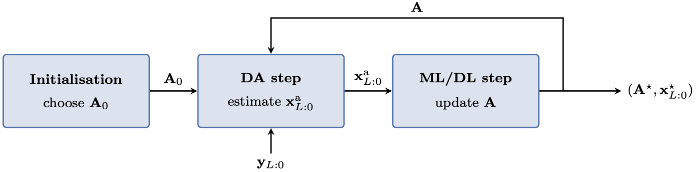
The ML-based information can be used in real-time to operate decision such as increasing spatio-temporal resolution, increasing/reducing ensemble size or locate/remove observations.

Ayers et al., (submitted).



Outline

Combining ML with DA



DA+ML for two complementary goals

1. Build a **full model** of the observed process.
2. Infer the model error and build an **hybrid physical/data-driven model**.

Outline

Emulating a model by combining DA and ML

Emulation of the resolvent **combining** DA and ML:

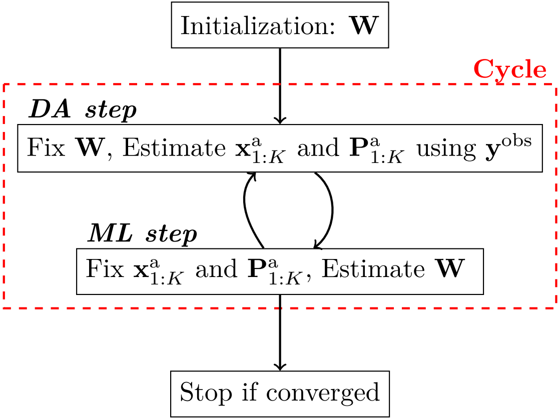
$$\mathbf{x}_{k+1} = \mathcal{M}(\mathbf{x}_k) \approx \mathcal{G}_{\mathbf{W}}(\mathbf{x}_k) + \boldsymbol{\epsilon}_k^{\mathrm{m}},$$

where $\mathcal{G}_{\mathbf{W}}$ is a neural network parameterised by $\mathbf{W}$ and $\boldsymbol{\epsilon}_k^{\mathrm{m}}$ is a stochastic noise.

- For the DA part we use the Finite-Size Ensemble Kalman Filter (EnKF-N).
- For the ML part we train a neural net

Brajard *et al*, 2020

Proposed DA+ML algorithm



Proposed DA+ML algorithm

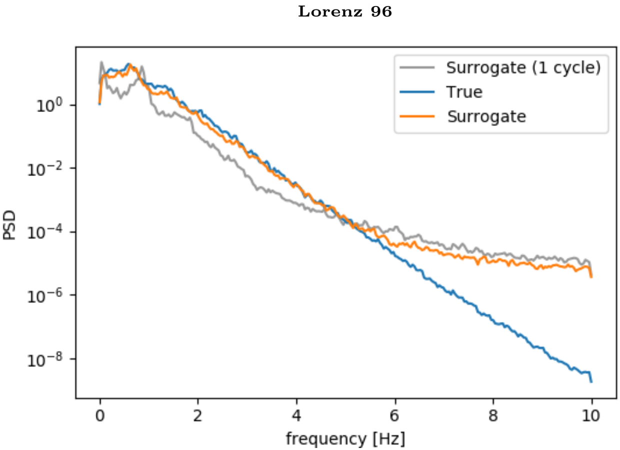
- Step 1 - Data Assimilation: estimate the state field $\mathbf{x}_{1:K}$ (the analysis) and associated (analysis) error covariance, $\mathbf{P}_k$, based on the fixed model parameters $\mathbf{W}$ and using sparse and noisy data, $\mathbf{y}$.

- Step 2 - Machine learning: using $\mathbf{x}_{1:K}$ and $\mathbf{P}_k$ from DA estimate $\mathbf{W}$
  - The neural network can be expressed as a parametric function $\mathcal{G}_{\mathbf{W}}(\mathbf{x}_k) = \mathbf{x}_k + f_{\mathrm{nn}}(\mathbf{x}_k, \mathbf{W})$ where f_{nn} is a neural network and $\mathbf{W}$ its weights; f_{nn} is composed of convolutive layers.
  - The determination of the optimal $\mathbf{W}$ is done in the *training phase* by minimising the loss function:

$$L(\mathbf{W}) = \sum_{k=0}^{K-N_{\mathrm{f}}-1} \sum_{i=1}^{N_{\mathrm{f}}} \left\| \mathcal{G}_{\mathbf{W}}^{(i)}(\mathbf{x}_k) - \mathbf{x}_{k+i} \right\|_{\mathbf{P}_k^{-1}}^2 ,$$

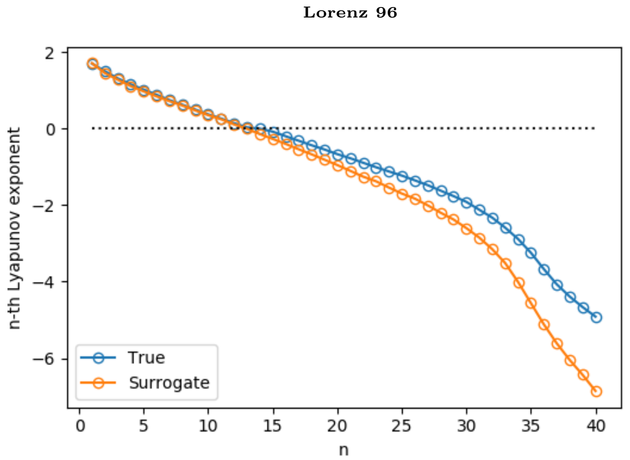
    where N_{f} is the number of time steps corresponding to the forecast lead time on which the error between the simulation and the target is minimised (with "coordinate descent" Bocquet *et al* 2020).
  - $\mathbf{P}_k$ is a symmetric, semi-definite positive matrix estimated by *analysis error covariances* from the DA step.
  - This time-dependent matrix, $\mathbf{P}_k$, plays the role of the surrogate model error covariance matrix and gives different weights to each state during the optimisation process.

Emulating the underlying dynamics: *Power spectrum density*



- After one cycle, some frequencies are favoured (see the peak at ~ 0.8Hz) and indicate that the **periodic signals are learnt first**.
- At convergence, the surrogate model reproduces the spectrum up to 5 Hz but then **adds high-frequency noise**.
- **Low frequencies are better observed** and better reproduced after the DA step.
- The PSD has been computed using a long simulation ($16,000$ time steps), which means that **the surrogate model is very stable**.

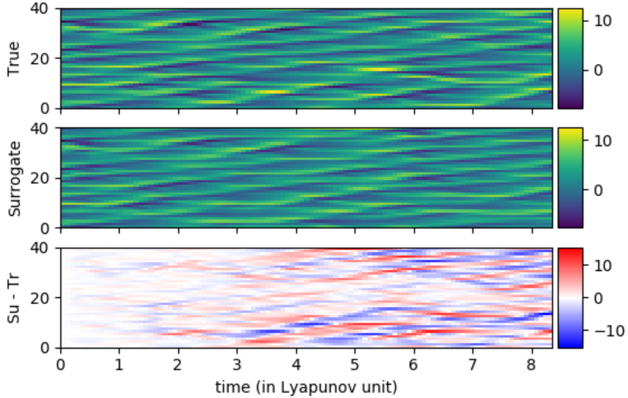
Emulating the underlying dynamics: *Lyapunov spectrum*



- **Accurate unstable spectrum** ⇒ Same **error growth** rate and **Kolmogorov entropy**, as the true model.
- **Less accurate reconstruction of the neutral and quasi-neutral** part of the spectrum.
- This is similar to what found in Pathak *et al* 2017 . Possibly due to the slower convergence (linear vs exponential) of the neutral exps Carrassi *et al* 2022.
- The stable part of the spectrum is shifted toward smaller values ⇒ PDFs contracts faster than in the true model, *i.e.* **surrogate model more dissipative than the truth**.

Forecast skill

Hovmøller plot of the true and surrogate models (in Lyapunov time*, LT)



- The simulations start from the same initial conditions.
- Good prediction until 2 LTs. Error saturation at 4-5 LTs.
- (*): the time for the error to grow by a factor e.

Outline

Combined DA-ML to infer unresolved scales parametrizations

The objective is to produce a hybrid (physical/data-driven) model

$$\mathbf{x}(t+\delta t) = \mathcal{M}^{\varphi}[\mathbf{x}(t)] + \mathcal{M}^{\mathrm{UN}}[\mathbf{x}(t)],$$

where:

- $\mathbf{x}(t)$ is the state of the dynamical system
- $\mathcal{M}^{\varphi}$ is the physical model (assumed to be known a priori)
- $\mathcal{M}^{\mathrm{UN}}$ is the unresolved component of the dynamics to be inferred from data
- δt is the integration time step

$\mathcal{M}^{\mathrm{UN}}$ is approximated by a data-driven model represented under the form of a neural network whose parameters are $\boldsymbol{\theta}$: $\mathcal{M}_{\boldsymbol{\theta}}[\mathbf{x}(t)]$

Proposed approach

Simplified description of the algorithm:

1. Estimating the state $\mathbf{x}^{\mathrm{a}}_{1:K}$. At each time t_k, we calculate a forecast $\mathbf{x}^{\mathrm{f}}$:

$$\mathbf{x}^{\mathrm{f}}_{k+1} = \mathbf{x}^{\mathrm{f}}(t_k + \Delta t) = (\mathcal{M}^{\varphi})^{N_c}(\mathbf{x}^{\mathrm{a}}_k)$$

   An observation $\mathbf{y}_{k+1}$ is assimilated with **strongly coupled EnKF** to produce an analysis $\mathbf{x}^{\mathrm{a}}_{k+1}$

2. Determining an estimation of the unknown part of the model. We assume that:
   - $\mathbf{x}(t+\Delta t) \approx (\mathcal{M}^{\varphi})^{N_c}(\mathbf{x}(t)) + N_c \cdot \mathcal{M}^{\mathrm{UN}}[\mathbf{x}(t)]$
   - $\mathbf{x}(t) \approx \mathbf{x}^{\mathrm{a}}(t)$

   We consider that $\mathcal{M}^{\mathrm{UN}}(\mathbf{x}_k) \approx \mathbf{z}_{k+1} = 1/N_c \cdot \left(\mathbf{x}^{\mathrm{a}}_{k+1} - \mathbf{x}^{\mathrm{f}}_{k+1}\right) \Longrightarrow$ The "target" (*i.e.* the model error) is estimated using the *analysis increments* (Carrassi and Vannitsem, 2011).

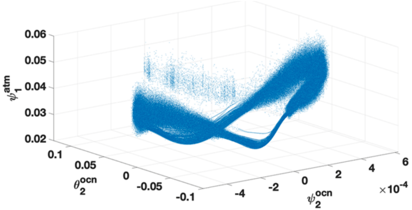
3. Training a neural network $\mathcal{M}_{\boldsymbol{\theta}}$ by minimising the loss $L(\boldsymbol{\theta}) = \sum_{k=0}^{K-1} ||\mathcal{M}_{\boldsymbol{\theta}}(\mathbf{x}^{\mathrm{a}}_k) - \mathbf{z}_{k+1}||^2$

4. Using the hybrid model $\mathcal{M}^{\varphi} + \mathcal{M}_{\boldsymbol{\theta}}$ to produce new simulations (*e.g.* to make forecasts).

Brajard *et al*, 2021

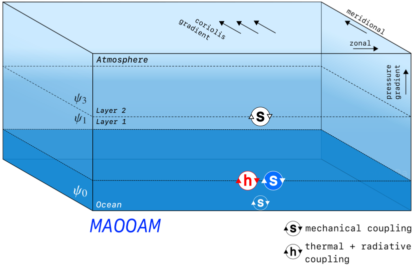
Experiments with the coupled atmosphere-ocean MAOOAM

- **MAOOAM**: Modular arbitrary-order ocean-atmosphere model (Le Cruz *et al*, 2016)
- A two-layer QG atmosphere coupled, thermally and mechanically, to a QG shallow-water ocean layer in the β-plane.
- MAOOAM is resolved in spectral space, for streamfunction and potential temperature, with adjustable resolution.



- We implement a **strongly coupled EnKF** (Tondeur *et al* 2020).

a) 2-layer atmosphere coupled to 1-layer ocean configuration
b) includes friction at boundaries for atmosphere and wind stress at A-O boundary
Mechanical component



a) includes thermal and radiative heat transport between atmosphere and ocean as function of T_a and T_o.
Thermal component

Experiments with MAOOAM

1. **Truth**: $n_\mathrm{a} = 20$ and $n_\mathrm{o} = 8$ modes for atmosphere and ocean. **Total dimension** $N_x = 56$.
2. **Truncated**: $n_\mathrm{a} = 10$ and $n_\mathrm{o} = 8$ modes for atmosphere and ocean. **Total dimension** $N_x = 36$.

- The truncated model is **missing 20 high-order atmospheric variables**
- There is not locality in spectral space so the NN is made of 3 layers multi-layer perceptrons

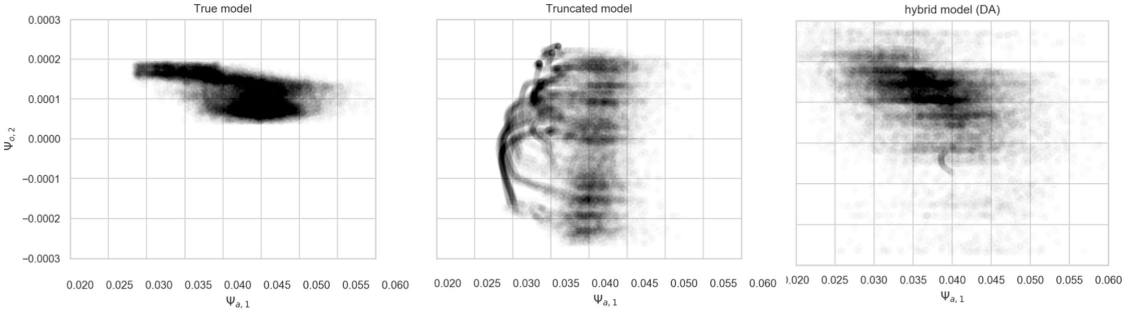
RMSE-f of hybrid and truncated MAOOAM models

RMSE-f(lead time τ)	$\psi_{o,2}$(2 years)	$\theta_{o,2}$(2 years)	$\psi_{a,1}$(1 day)
Truncated	0.23	0.21	0.36
Coupled DA-ML hybrid	**0.10**	**0.06**	**0.28**

- The hybrid models have superior skill than the truncated model.
- The improvement is larger for the ocean that is fully resolved $\Longrightarrow$ **Enhanced representation of the atmosphere-ocean coupling processes thanks to coupled DA**.
- The atmosphere is improved less: the hybrid is not very good in representing the fast processes.

Numerical experiments: atmosphere-ocean model MAOOAM

Reconstruction of the model attractor



- The truncated model visits areas of the phase space that are not admitted in the real dynamics.
- Discrepancies are reduced by the hybrid models.

Outline

Parametrization of sea-ice melt ponds with DA and ML

Sea Ice and Albedo:
The Role of Melting/Melt Ponds on Albedo

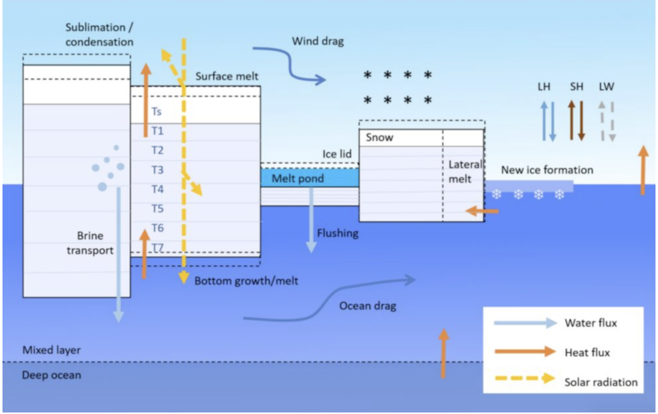
- Each year melting of sea ice occurs, **melt ponds** form on the surface of the ice.
- **The evolution of melt ponds in the summer is one of the main factors affecting the polar climate system.**
- The impact of melt ponds on the climate system will increase as climate change continues.

Melt ponds are spatially irregular, sub-grid scale, and their evolution is dependent on many competing factors. Therefore they have to **parametrised.**

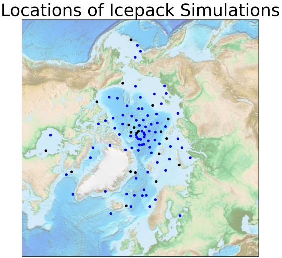
Parametrization of sea-ice melt ponds with DA and ML

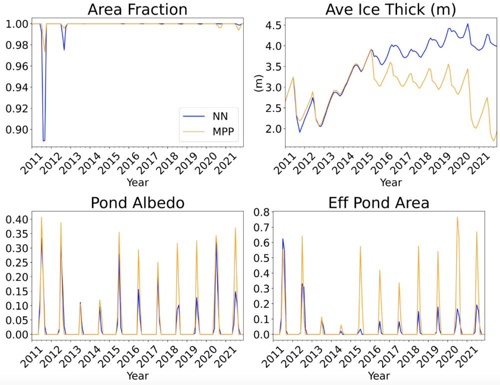
Icepack - A column physics sea ice model

- **Icepack** is a **state-of-the-art column physics model** representing many **crucial sea ice processes** (thermodynamics, ridging, biogeochemistry, and associated area and thickness changes).



Parametrization of sea-ice melt ponds with DA and ML





- Very good accuracy (R2 score ≈> 0.98) when the NN is trained on the model output (*i.e.* with synthetic data)
- Feature selection for model reduction performed using **Shannon mutual information**.
- Moving to training on real data (in situ Sheeba dataset)

Conclusions

- We showed three ways on how using ML in geophysical DA and forecasting: **(1)** to supplement a physical model, **(2)** to infer parametrization of unresolved/poorly known processes, and, **(3)** to fully emulate an hidden dynamics.
- We studied the potential for ML to estimate LLEs: Greater accuracy is associated with local homogeneity of the LLEs on the system attractor.
- We developed a combined **DA+ML** approach whereby DA is instrumental to handle partial and noisy data and then inform the ML algorithm.
- This flow of information **from DA to ML** includes a state-dependent estimate of the uncertainty about the state that is key in the ML optimization step.
- The **DA+ML** approach is **very flexible**: any DA or ML algorithm can be plug-in.

Bibliography

[1] Ayers D, J Amezcua, A Carrassi and V Ohija, 2022. *Supervised machine learning to estimate instabilities in chaotic systems: estimation of local Lyapunov exponents*, *Under Review*, Available at **https://arxiv.org/abs/2202.04944**

[2] Bocquet M, Brajard J, A Carrassi, and L Bertino, 2019. *Data assimilation as a learning tool to infer ordinary differential equation representations of dynamical models*, *Nonlin. Proc. Geophys.*,**26**, 175–193

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

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

[5] Brajard J, A Carrassi, M Bocquet and L Bertino. 2021. *Combining data assimilation and machine learning to infer unresolved scale parametrisation.*, *Phil. Trans A of the Roy Soc.*, **379** (2194). 20200086

[6] Carrassi A, M Bocquet, J Demayer, C Grudzien, P Raanes, and S Vannitsem, 2022. *Data assimilation for chaotic dynamics.* Springer International Publishing, Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications (Vol. IV) (2022): 1-42.

[7] Chen Y., A Carrassi, and V Lucarini. 2021. *Inferring the instability of a dynamical system from the skill of data assimilation exercises.* Nonlinear Processes in Geophysics, **28**, 633-649

[8] Tondeur M, A. Carrassi, S. Vannitsem and M. Bocquet: *On temporal scale separation in coupled data assimilation with the ensemble Kalman filter.* *J. Stat. Phys.*, **44**. 101171, 2020