LIEGE université

Training-Free Data Assimilation with GenCast

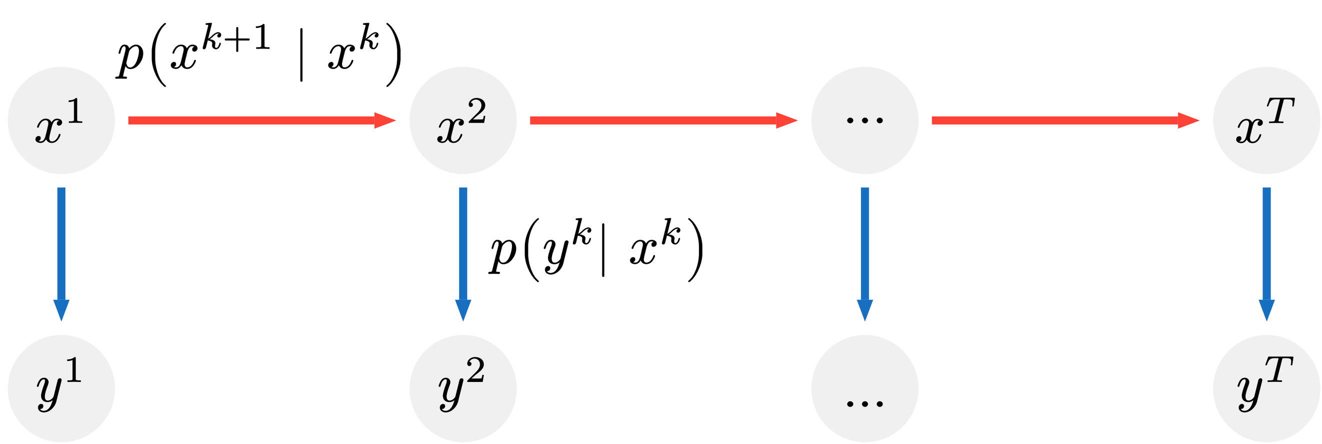
Thomas Savary, François Rozet and Gilles Louppe

Climate Change AI

TL;DR Data assimilation refers to a set of algorithms used to estimate the state of a dynamical system by combining model predictions with observations. In this work, we show that diffusion-based emulators can be efficiently applied to this task without additional training.

Problem statement

One of the goal of data assimilation, known as filtering, is to estimate the state of a discrete time Markovian dynamical system from past and present observations $y^{1:k}$, that is to approximate the posterior distribution $p(x^k \mid y^{1:k})$.



To do so, we assume a pretrained diffusion model that defines the transition law

$$x^{k+1} \sim p(x^{k+1} \mid x^k),$$

together with an observation operator H and covariance Σ_y specifying the likelihood of the observations

$$p(y^k \mid x^k) = \mathcal{N}(y^k \mid H(x^k), \Sigma_y).$$

Methodology

- **Particle filter approximation**

Particle filter approximates $p(x^k \mid y^{1:k})$ by a discrete measure $\mu_x^k = \sum_{i=1}^N w_i^k x_i^k$ such that the following converges weakly

$$\sum_{i=1}^N w_i^k g(x_i^k) \xrightarrow[N\to+\infty]{} \int g(x^k) p(x^k \mid y^{1:k}) dx^k.$$

They can handle strongly nonlinear dynamics but suffer from particle degeneracy.

- **Sampling from the optimal proposal**

Degeneracy is caused by weights variance, which is minimized by sampling particles from the optimal proposal $p(x^{k+1} \mid x^k, y^{k+1})$ using the posterior score $\nabla_{x_t^{k+1}} \log p(x_t^{k+1} \mid x^k, y^{k+1})$ during the diffusion process

$$dx_t^{k+1} = \left[f_t x_t^{k+1} - \frac{1+\eta^2}{2} g_t^2 \nabla_{x_t^{k+1}} \log p(x_t^{k+1} \mid x^k, y^{k+1}) \right] dt + \eta g_t dw_t.$$

Thanks to Bayes' rule, the posterior score can be decomposed into two terms as

$$\nabla_{x_t^{k+1}} \log p(x_t^{k+1} \mid x^k) + \nabla_{x_t^{k+1}} \log p(y^{k+1} \mid x_t^{k+1}, x^k).$$

The first one is known using the pretrained denoiser whereas the second one is computed following Rozet et al, 2024.

- **Computing weights**

Updating the weights in the case of the optimal proposal requires evaluating $p(y^{k+1} \mid x^k)$, which we approximate by

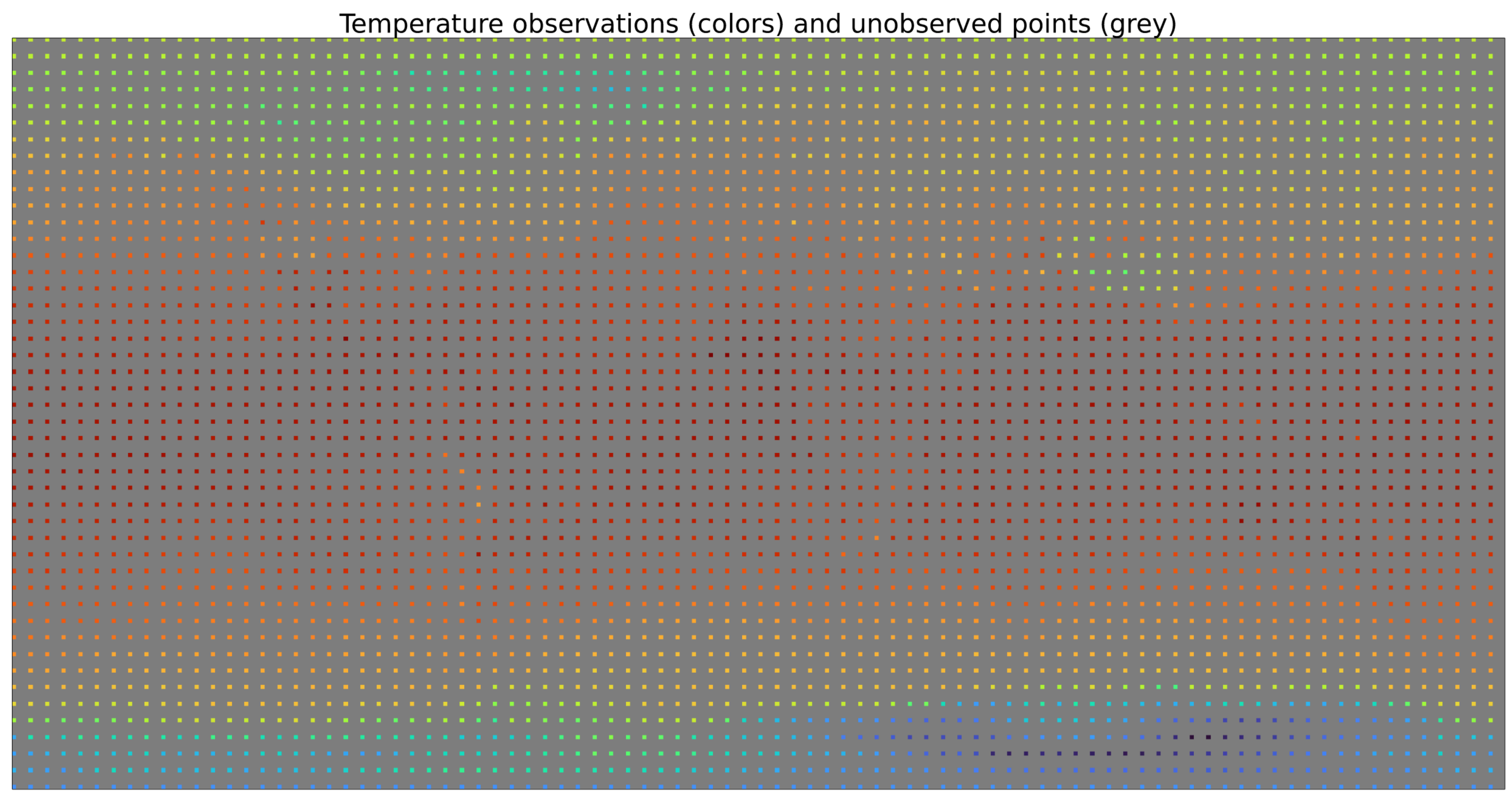
$$\int p(y^{k+1} \mid x^{k+1}) p(x^{k+1} \mid x^k) \, \mathrm{d}x^{k+1} \approx p(y^{k+1} \mid \mathbb{E}[x^{k+1} \mid x^k]).$$

The conditional expectation $\mathbb{E}[x^{k+1} \mid x^k]$ is not known a priori, but can be efficiently estimated using the pretrained diffusion denoiser

$$d_\theta\left(x_{t=1}^{k+1} = \sigma_1 \varepsilon, x^k, t=1\right) \approx \mathbb{E}[x^{k+1} \mid x^k, \sigma_1 \varepsilon] \underset{\varepsilon\sim\mathcal{N}(0,I)}{=} \mathbb{E}[x^{k+1} \mid x^k].$$

Experimental setup

- We use the pretrained GenCast denoiser at 1° resolution with $N = 256$ particles.
- We only observe temperature from the surface to the top of the atmosphere on a regular latitude–longitude grid.



- Observations, initial condition and ground truth are taken from a reference ERA5 trajectory (a global atmospheric reanalysis).

Results