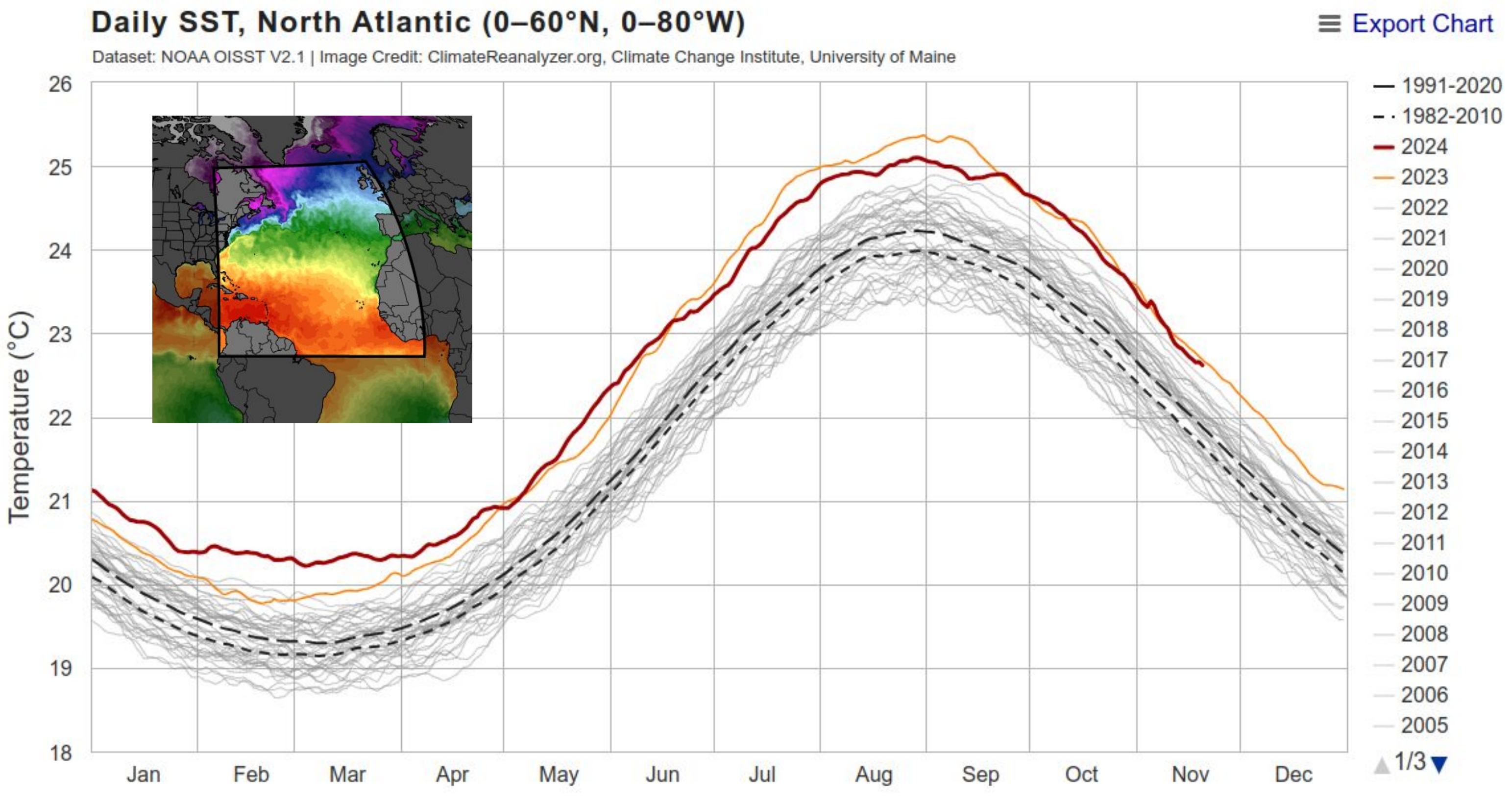
Reconstruction et estimation d′incertitude des images satellites utilisant l′apprentissage profond

Alexander Barth,
Julien Brajard, Aida Alvera-Azcárate, Bayoumy Mohamed, Charles Troupin, and Jean-Marie Beckers

GHER, Université de Liège, Belgique

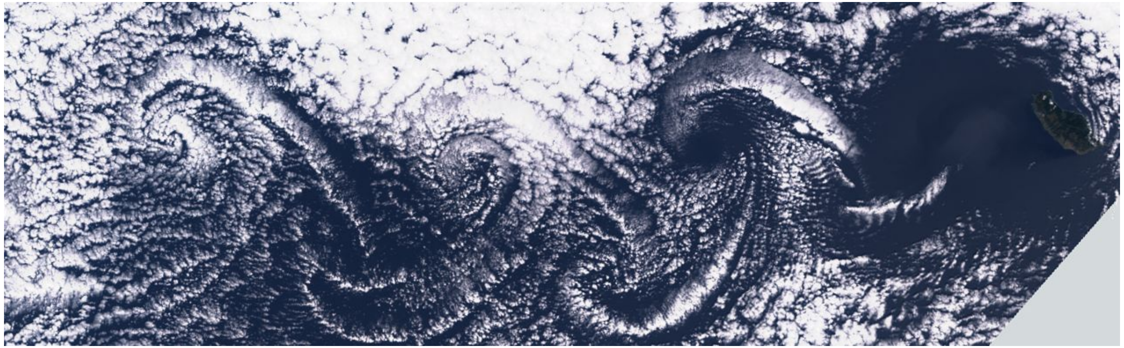


Data: NOAA OISST 2.1, https://climatereanalyzer.org/

Neural network for satellite image reconstruction

- About 75 % of the ocean surface is on average covered by clouds that block sensors in the optical and infrared bands (Wylie et al., 2005)
- For many applications, full images are necessary or at least desirable (even to compute a mean)
- Accuracy of interpolated fields will naturally vary in space and time as the data coverages varies

Objectives



Clouds can be beautiful but…

- Reconstruct missing data (e.g. due to clouds) in satellite images (in-painting)
- Training a neural network
  - From hydrodynamical **model data** (complete; but affected by errors and biases)
  - From **observations** (**incomplete**; still possibly affected by errors and biases; but to a lesser degree)
- **Aim here: training a neural network using incomplete observations**

The Bayes' rule or how to combine information

For **Gaussian-distributed errors**:

- prior: $\mathcal{N}(x^f, \sigma^f)$
- observations: $\mathcal{N}(y^o, \sigma^o)$
- posterior: $\mathcal{N}(x^a, \sigma^a)$

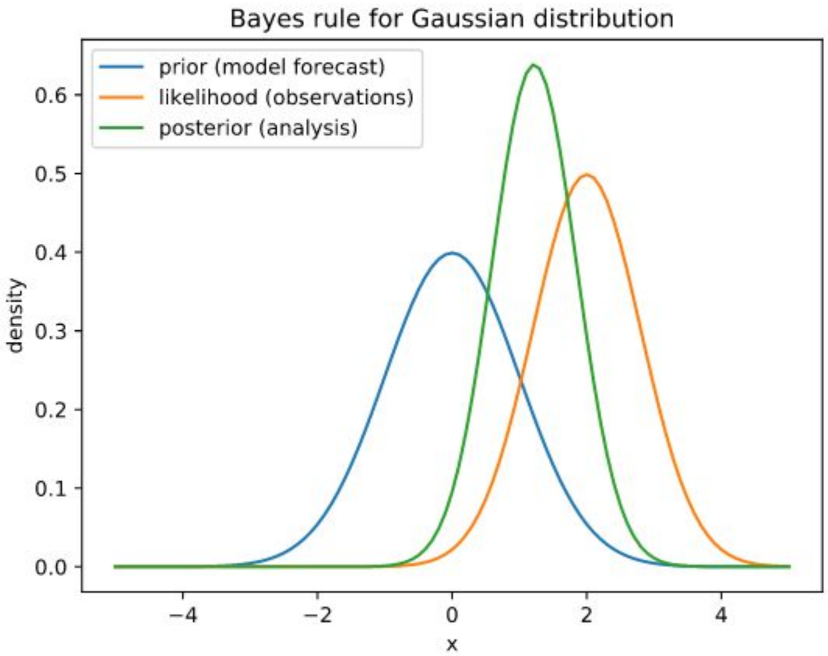
Bayes' rule:

$$p(x|y^o) = \frac{p(x)p(y^o|x)}{p(y^o)}$$

- Mean and variance of posterior, given by:

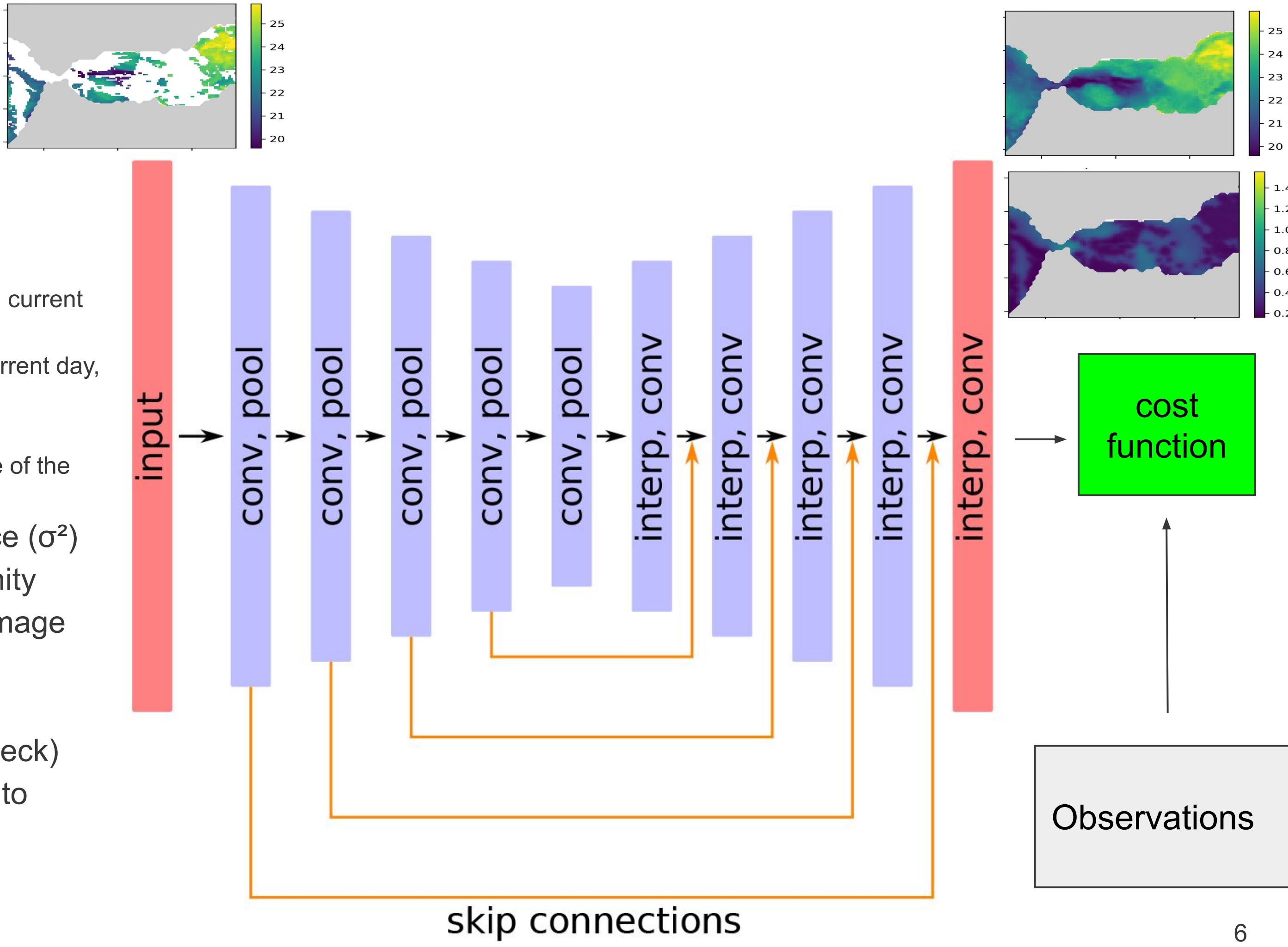
$$\sigma^{a^{-2}} x^a = \sigma^{f^{-2}} x^f + \sigma^{o^{-2}} y^o$$
$$\sigma^{a^{-2}} = \sigma^{f^{-2}} + \sigma^{o^{-2}}$$

- **Inverse of the variance are simply added linearly**



Typical UNet

- Input:
  - data/σ² (previous day, current day, following day)
  - 1/σ² (previous day, current day, following day)
  - Longitude, latitude
  - Time (cosine and sine of the year-day/365.25)
- If missing, error variance (σ²) is considered to be infinity
- Compression of input image
- Decompression from a so-called “latent representation” (bottleneck)
- Skip connections allow to bypass the bottleneck



Training

- Partitioned into so-called **mini-batches**
- The entire dataset is used **multiple times (epochs)**
- For every input image, **more data points were masked** (in addition to the cross-validation) by using a **randomly chosen cloud mask during training** (data set augmentation).
- The output of the neural network (for every single grid point *i,j*) is a **Gaussian probability distribution** function characterized by a mean $\hat{y}_{ij}$ and a standard deviation $\hat{\sigma}_{ij}$.

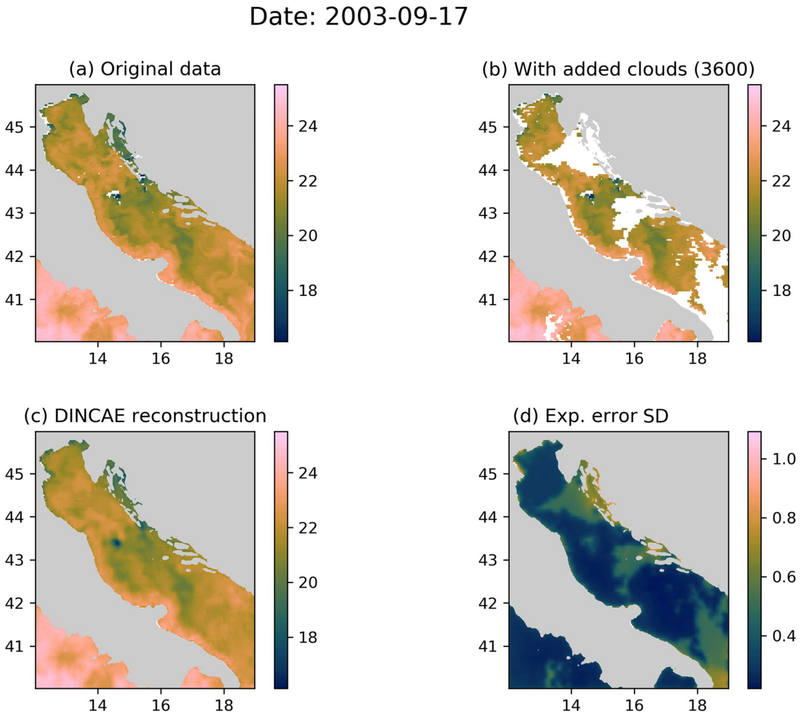
$$J(\hat{\mathbf{y}}, \hat{\sigma}) = -\log\left(p(\mathbf{y}|\hat{\mathbf{y}}, \hat{\sigma})\right) = \frac{1}{2N}\sum_{ij}\left[\left(\frac{y_{ij} - \hat{y}_{ij}}{\hat{\sigma}_{ij}}\right)^2 + \log\left(\hat{\sigma}_{ij}^2\right) + 2\log\left(\sqrt{2\pi}\right)\right]$$

- The first term: **mean square error, but scaled by the estimated error standard deviation.**
- The second term: **penalizes any over-estimation of the error standard deviation.**
- (Remember Gilles Loupe presentation about the overconfidence and underconfidence)

Multivate reconstructions

Dataset used

- MODIS **Sea Surface Temperature**
- **Wind speed** (Cross-Calibrated Multi-Platform, CCMP; gridded surface vector winds)
- **Chlorophyll a** from Ocean Biology Processing Group



Auxiliary parameters	cat skip connections	sum skip connections	sum skip connections and refinement
none	0.66 (0.06–1.02)	0.60 (0.05–0.93)	0.55 (0.04–0.84)
chlor_a	0.64 (0.06–1.00)	0.59 (0.05–0.92)	0.54 (0.04–0.82)
chlor_a, wind_speed	0.65 (0.06–1.00)	0.58 (0.05–0.90)	0.54 (0.04–0.82)
chlor_a, wind_speed, uwnd, vwnd	0.66 (0.06–1.03)	0.57 (0.05–0.88)	0.54 (0.05–0.82)

Unstructured data

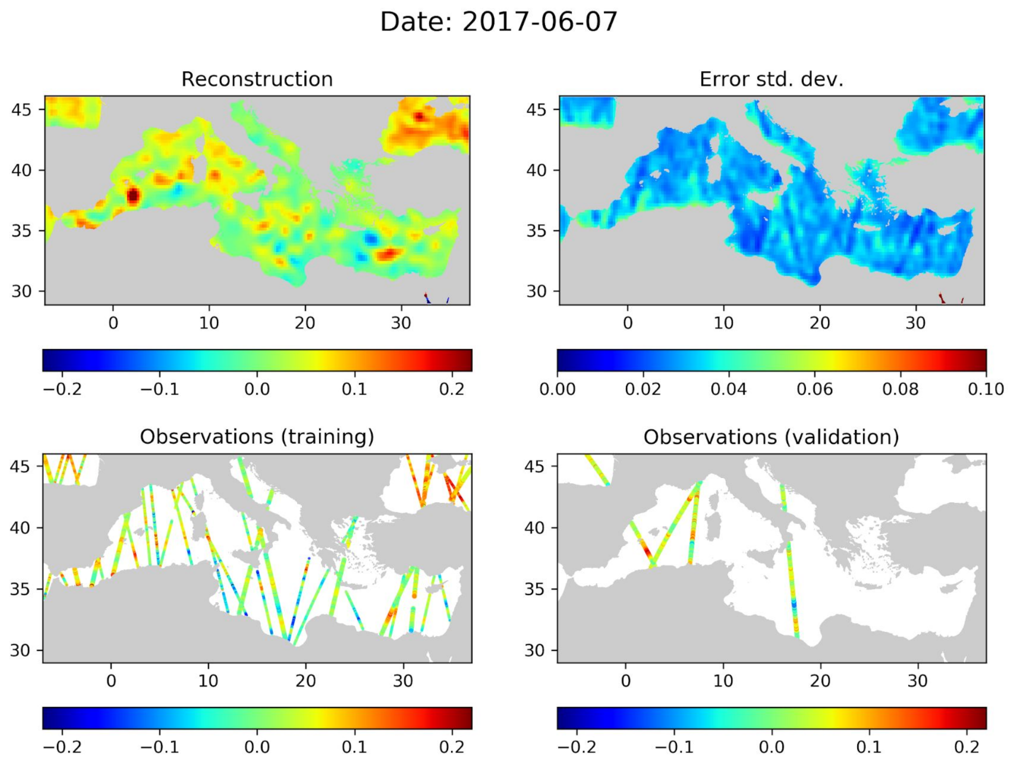
Altimetry data from 1993-01-01 to 2019-05-13 from CMEMS

Multiple satellites missions

- 70% training data (determine weight of the networks)
- 20% developpement data (determine structure of the network,...)
- 10% test data (independent validation)

Structure of the network determined by Bayesian optimization

Convolution operator extended to a point cloud



Validation

Reasonable **good match** with the validation data

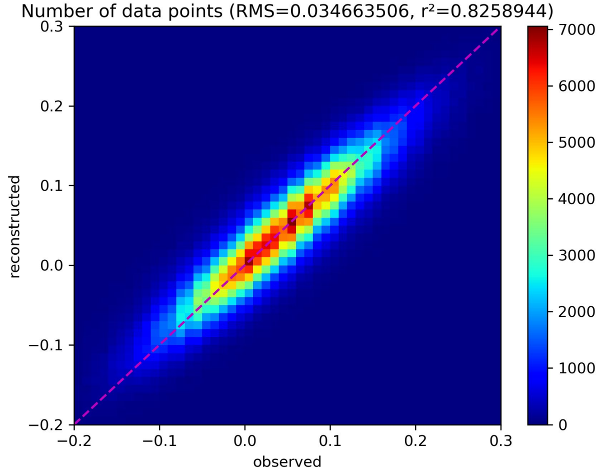
Reliable expected reconstruction errors are notoriously hard to obtain from methods like optimal interpolation

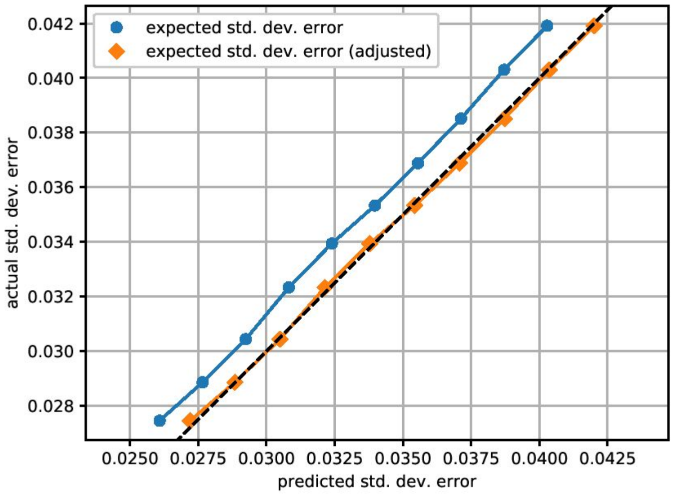
DINCAE also provide the expected error of the reconstruction (per pixel)

The validation data has been **grouped into bins** using the expected error

For every bin the **standard deviation of the actual error** has been computed

The predicted error underestimates the actual error only by 4%





Limitation

- So far, only error per grid cell
- Error of interpolated fields is also correlated in space and time
- Spatial and temporal correlation of error is important when computing e.g. averages

Rather than producing a single reconstruction,
producing an **ensemble of possible reconstruction** coherent with the available data

Denoising diffusion model

- Relatively new type of generative neural network
- Often used to generate images based on texts

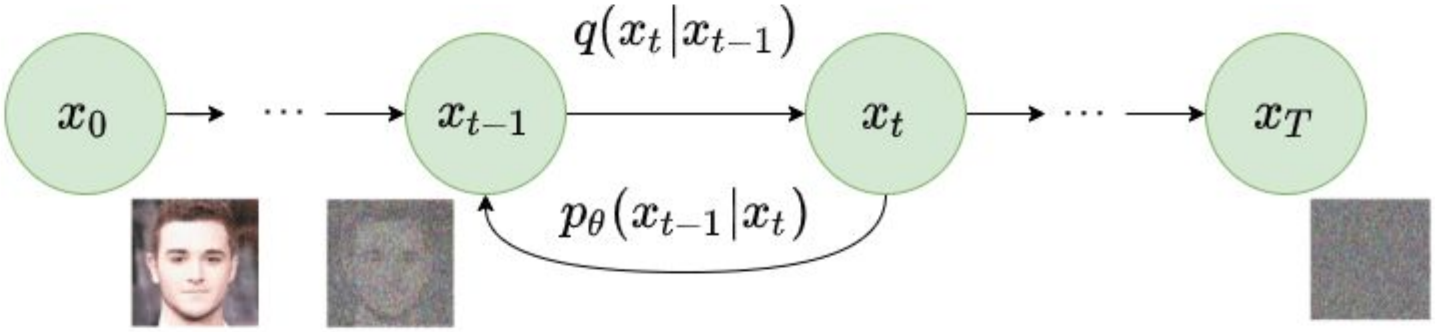
penguin astronaut on Mars

cat as transformer

Santa as a pokemon, pokemon card, attack, fire

Good at generalization

Denoising Diffusion Probabilistic Model



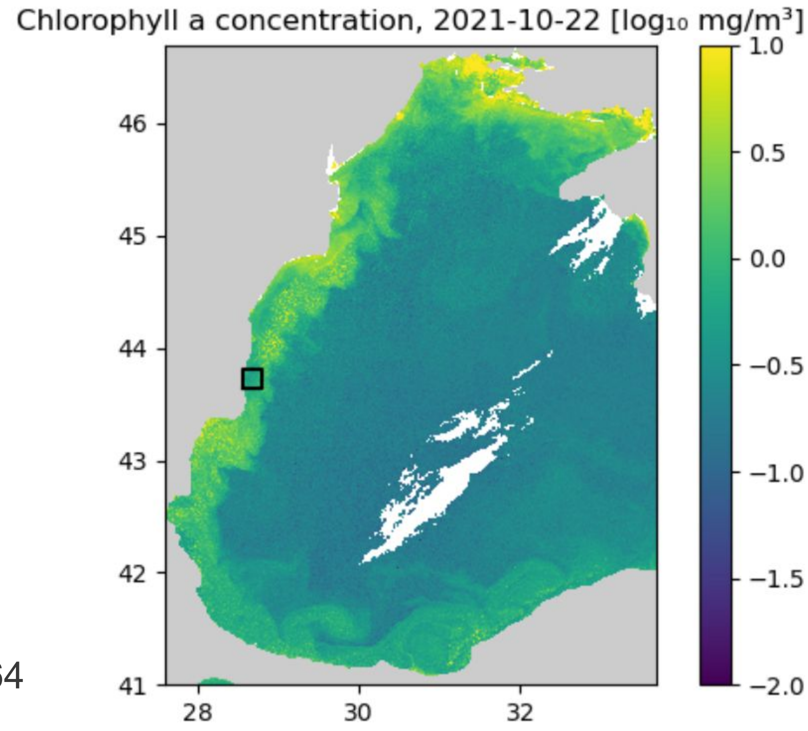
$$\mathbf{x}_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\mathbf{z}_{t-1}$$

β_t variance of noise added, $\mathbf{z}_t$ Gaussian noise, initial image: $\mathbf{x}_0$, t current step

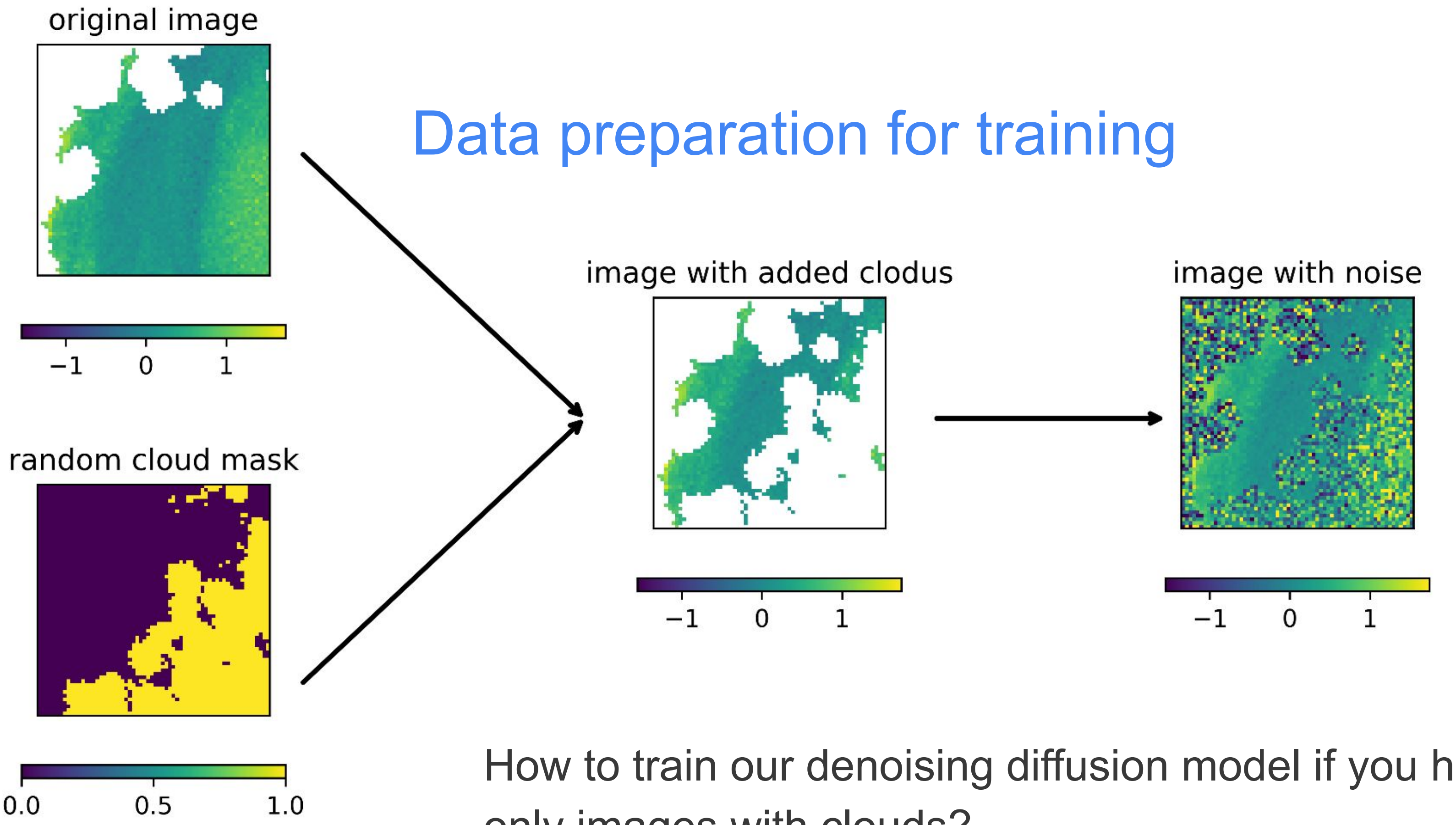
Jonathan Ho, Ajay Jain, Pieter Abbeel, Denoising Diffusion Probabilistic Model, 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada. 2020

Data

- L3 satellite chlorophyll-a concentration of the Black Sea
- spatial resolution of 300 m from the CMEMS
- Ocean and Land Colour Instrument (OLCI) sensor onboard Sentinel-3A and Sentinel-3B.
- Training data from 2016-04-26 to 2021-08-31
- The training data is split horizontally in tiles with 64 x 64 grid cells. We keep only tiles with at least 20% of valid data (i.e. non-clouded pixels) for training.
- Validation data from 2021-09-01 - 2022-08-31
- Test data 2022-09-01 - 2023-08-31
- Units: $\log_{10}$ mg/m³

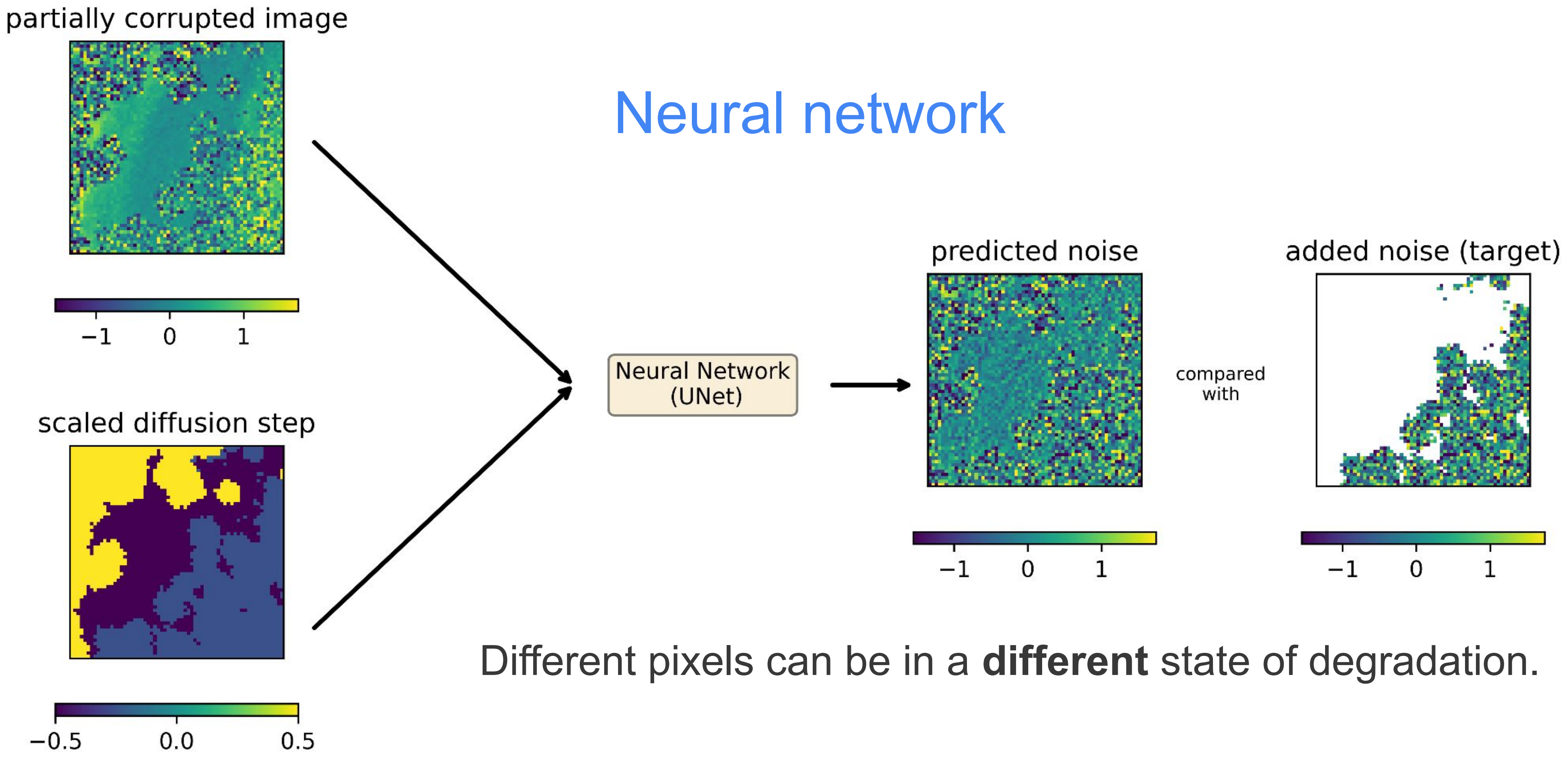


Data preparation for training



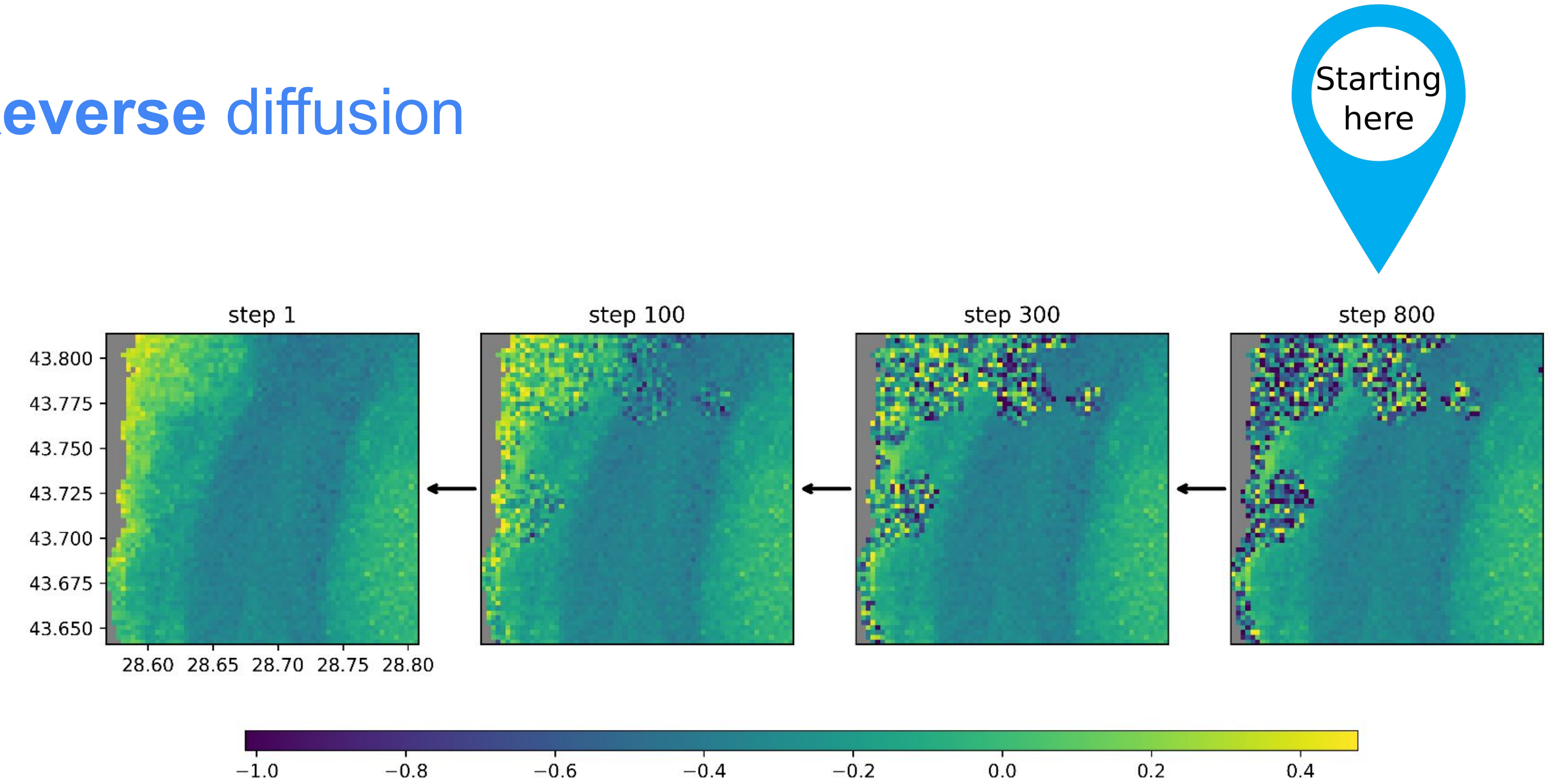
How to train our denoising diffusion model if you have only images with clouds?

Neural network

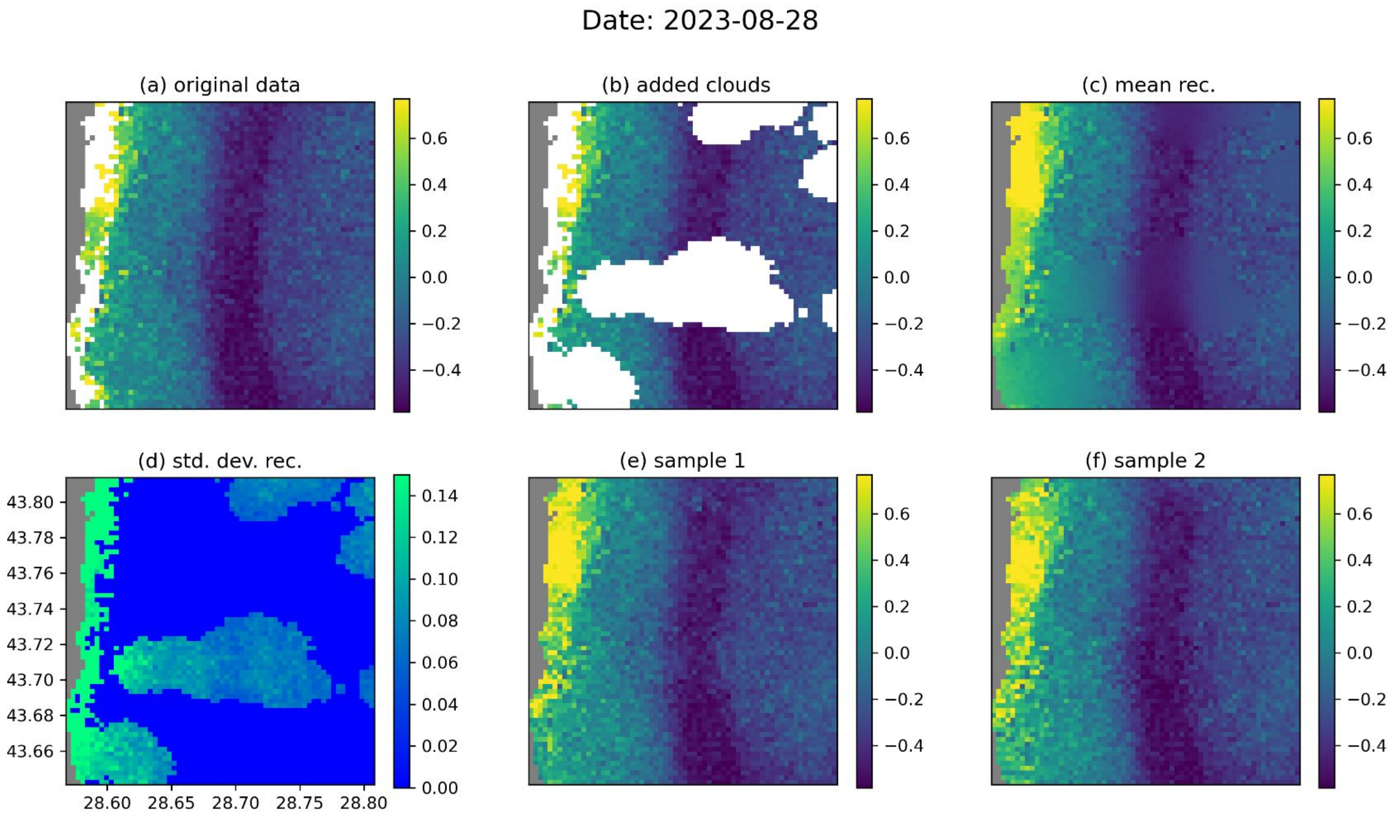


Different pixels can be in a **different** state of degradation.

Reverse diffusion



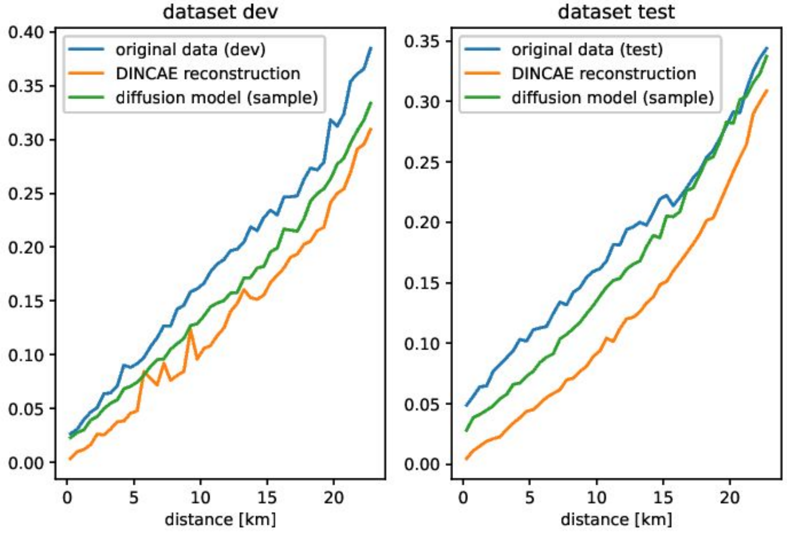
Results



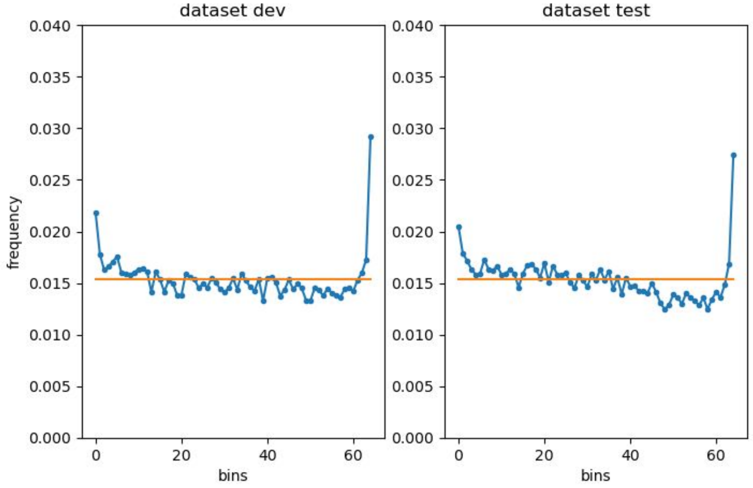
Mean rec:

Mean reconstruction of an ensemble of 64 sample reconstructions

Validation



Variogram



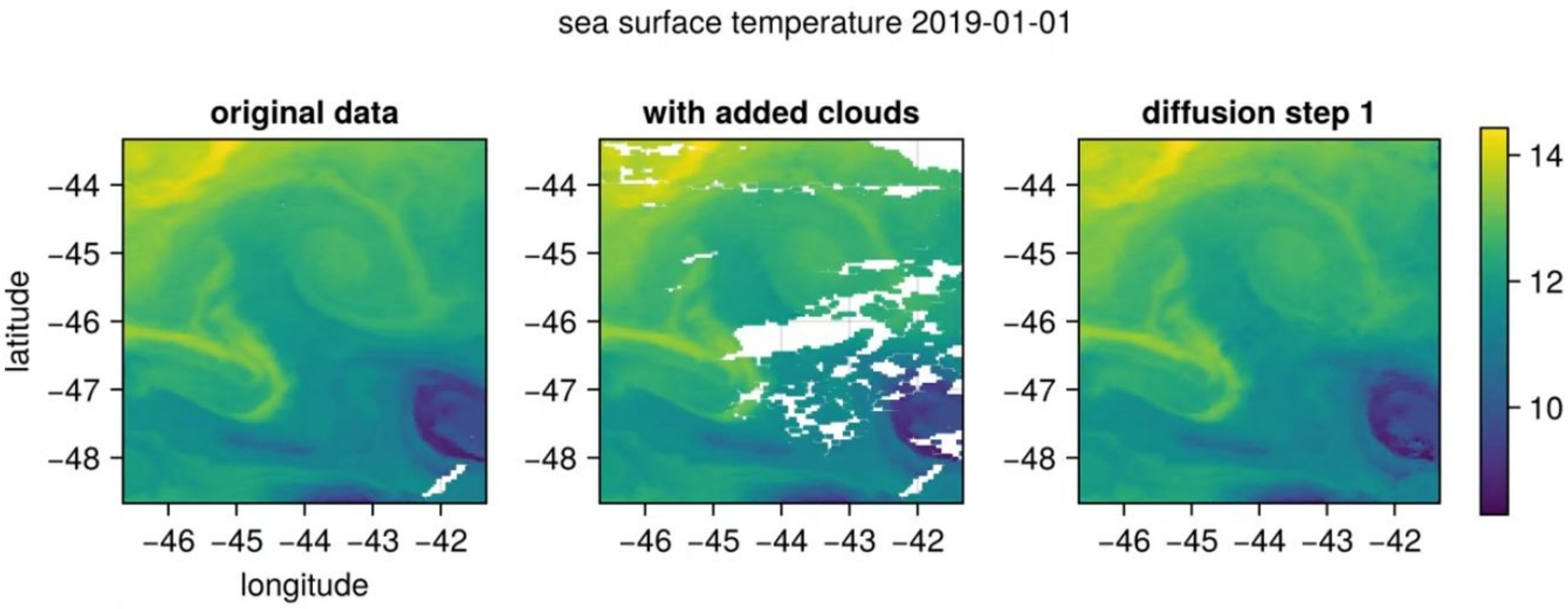
Talagrand diagram

Ideally: orange and green line overlap with blue → variability at all scales is preserved

Ideally: blue line should be flat → probabilities are reliable and indicative of the uncertainty

Reverse diffusion

- Satellite sea-surface temperature (MODIS: Moderate-resolution Imaging Spectroradiometer)
- Training on global dataset
- Strong seasonal cycle
- Larger images (124 x 124, more context for the reconstruction)



Conclusions

DINCAE:
- A convolutional Autoencoder approach to reconstruct missing data
- Missing data handled by including expected error variance in the input data
- Estimation of missing data + estimation of error of the reconstruction obtained
- Reliable error estimates

Diffusion models
- Better accuracy as DINCAE (RMS)
- Reliable ensemble of possible reconstructions

Code on GitHub with links to the papers:
https://github.com/gher-uliege/DINCAE.jl
https://github.com/gher-uliege/DINDiff.jl