

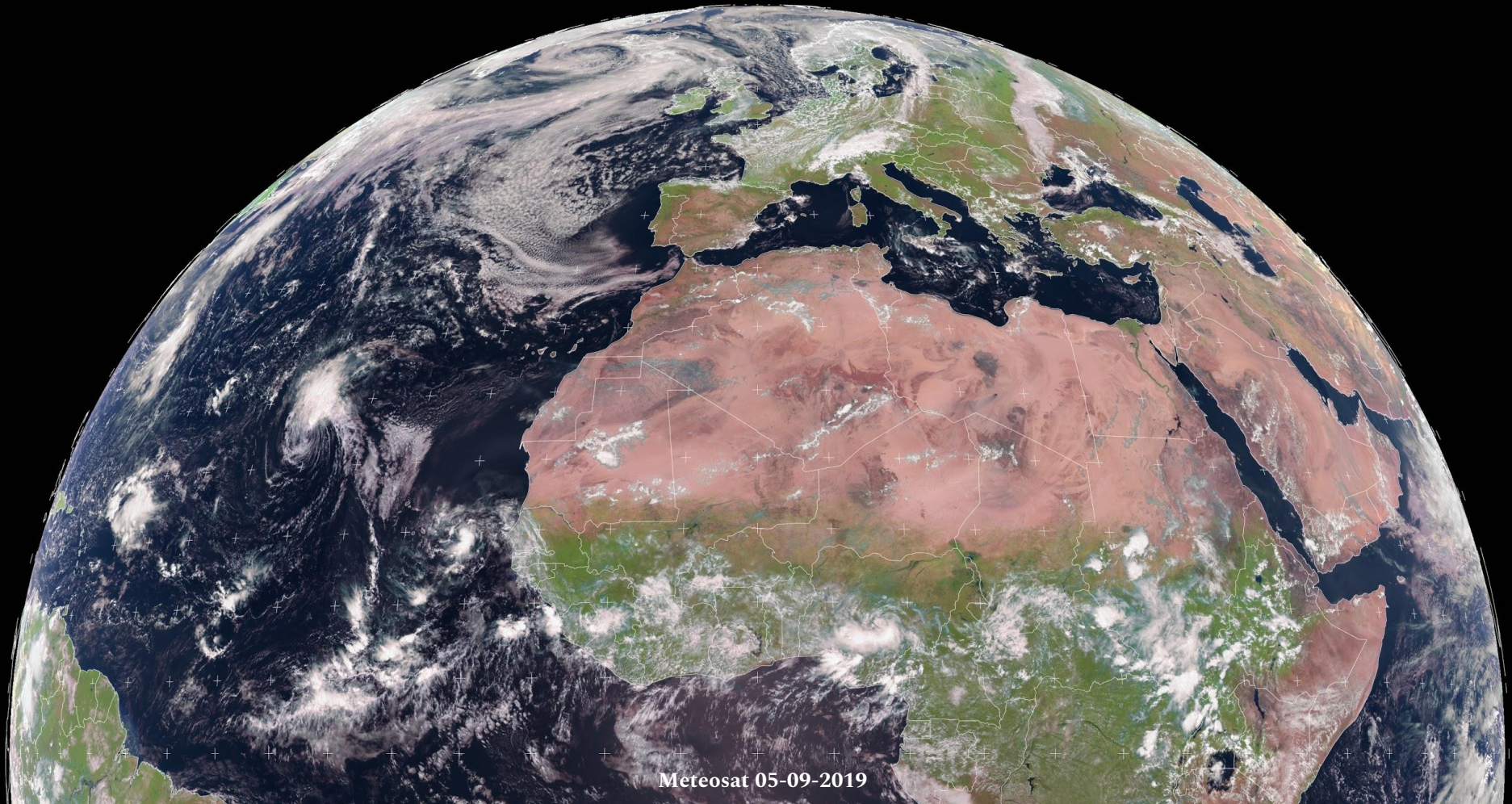
# *DINCAE: reconstruction of missing satellite data with a convolutional auto-encoder*

Alexander Barth, Aida Alvera-Azcárate, Matjaz Licer, and Jean-Marie Beckers

Phi-Week  
9-13 September 2019



NACIONALNI INŠTITUT ZA **BIOLOGIJO**  
NATIONAL INSTITUTE OF **BIOLOGY**

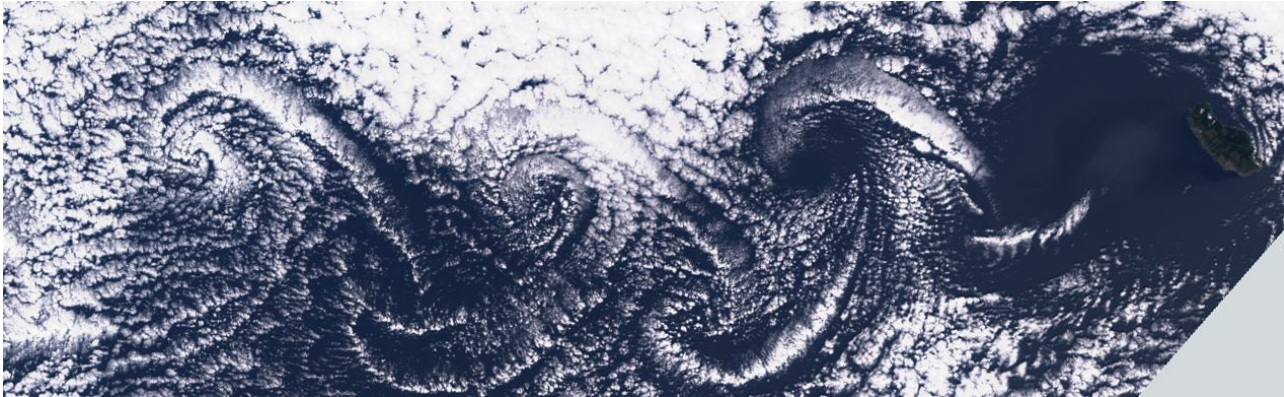


Meteosat 05-09-2019



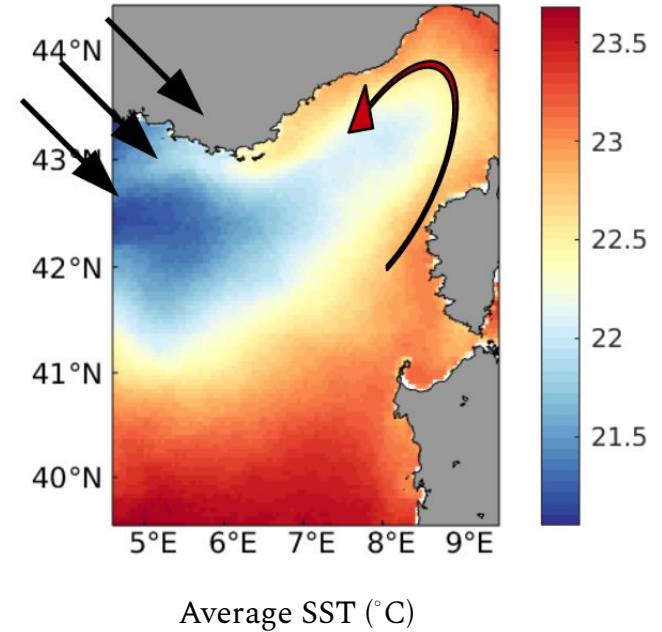
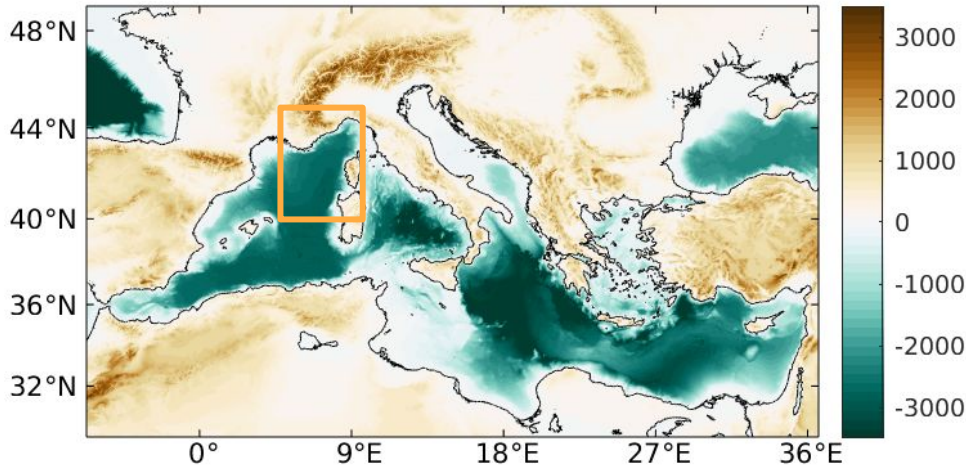
# Objectives

- To derive a methodology to reconstruct missing information in satellite data
  - Based on neural networks
  - Making use of ~four decades of sea surface temperature measurements
  - Able to retain small scale variability
- To assess the benefit of using neural networks in comparison with other state-of-the-art methodologies
  - DINEOF (Data Interpolating Empirical Orthogonal Functions)



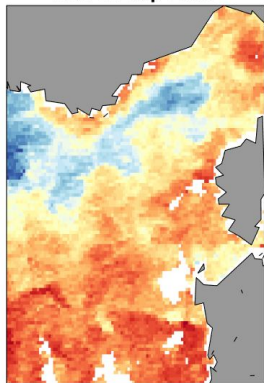
# Data used

- Daily Advanced Very High Resolution Radiometer (AVHRR) Sea Surface Temperature (SST) data
- 4 km spatial resolution
- Liguro-Provençal basin (western Mediterranean Sea)
- 1 April 1985 to 31 December 2009 (25 years)
- 47 % of missing data

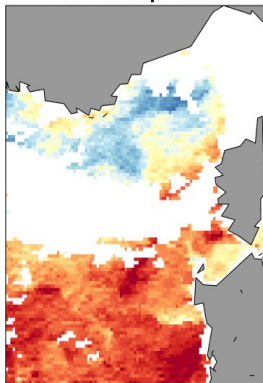


# Challenge: training on gappy data (lots of gaps!)

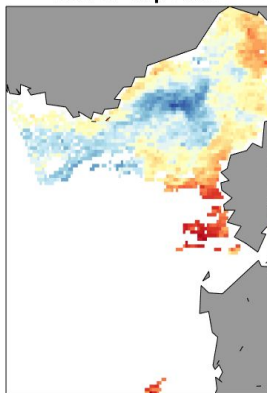
SST 25-Sep-2009



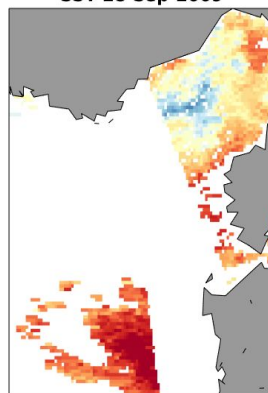
SST 26-Sep-2009



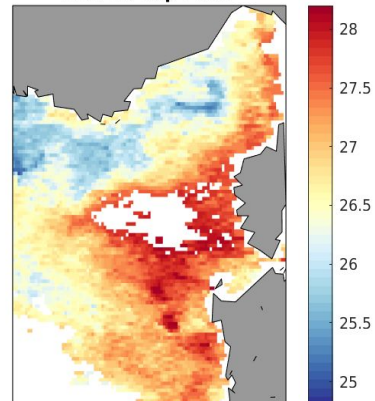
SST 27-Sep-2009



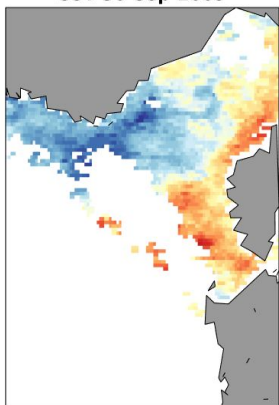
SST 28-Sep-2009



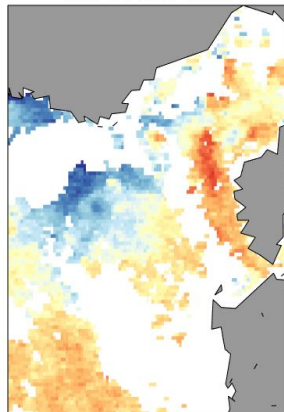
SST 29-Sep-2009



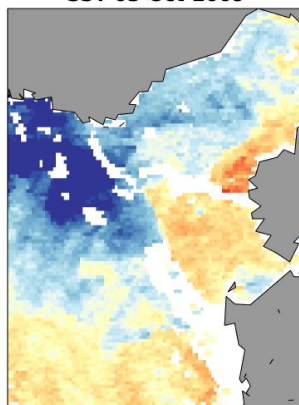
SST 30-Sep-2009



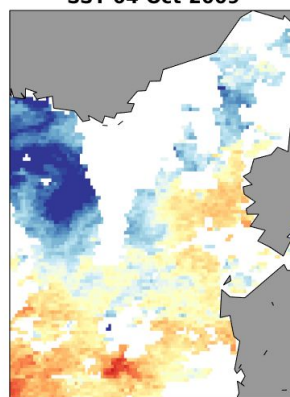
SST 02-Oct-2009



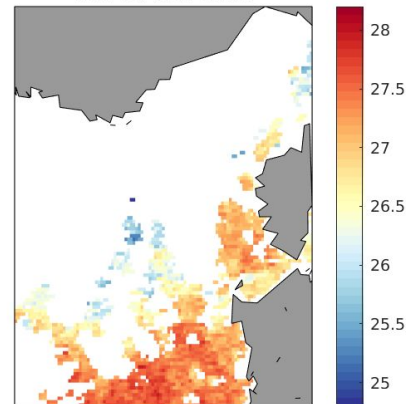
SST 03-Oct-2009



SST 04-Oct-2009

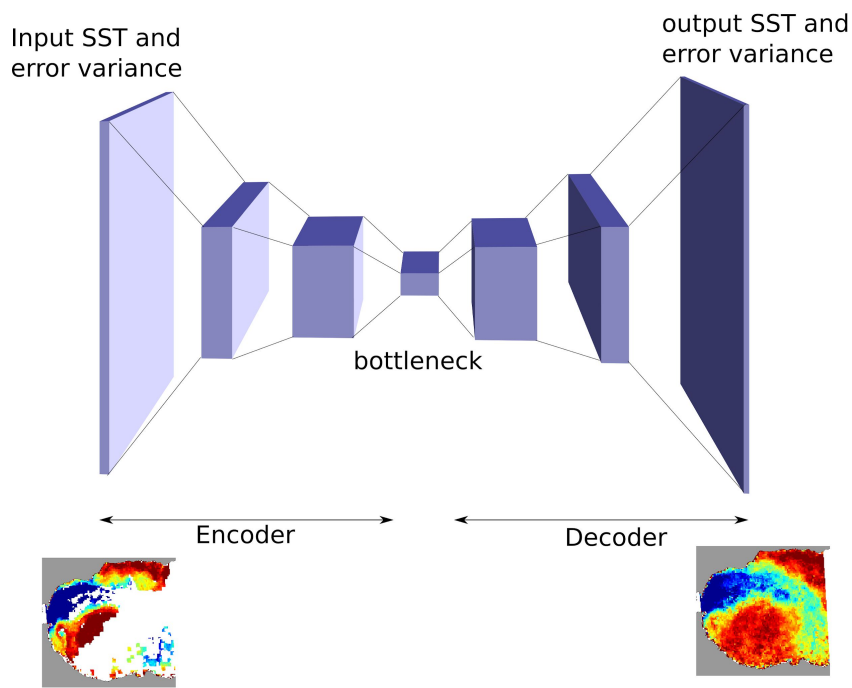


SST 05-Oct-2009



# Methodology

## DINCAE: Data-Interpolating Convolutional Auto-Encoder



**Auto-Encoder:** used to efficiently compress/decompress data, by extracting main patterns of variability

- Similarity to EOFs

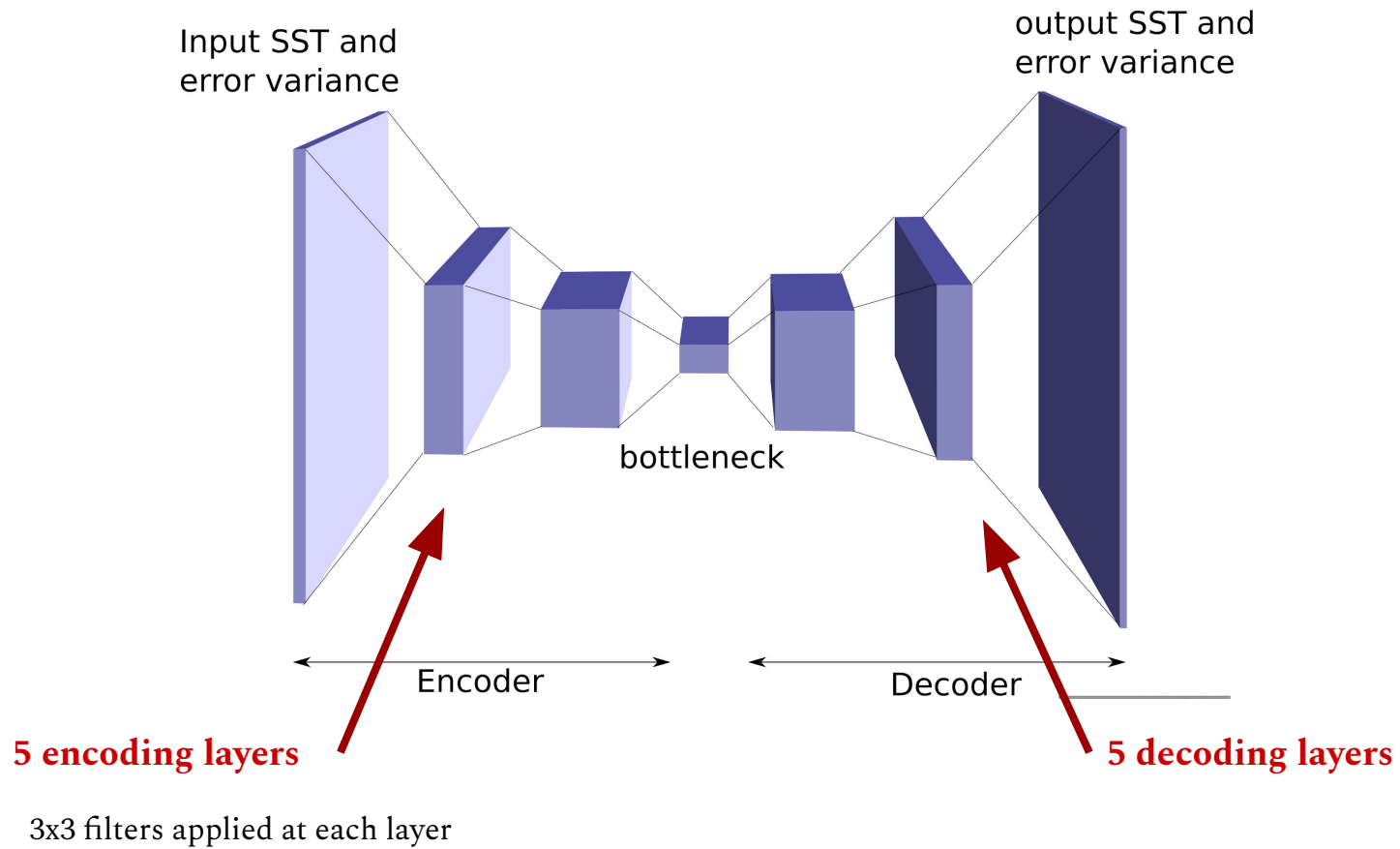
**Convolutional:** works on subsets of data, i.e. trains on local features

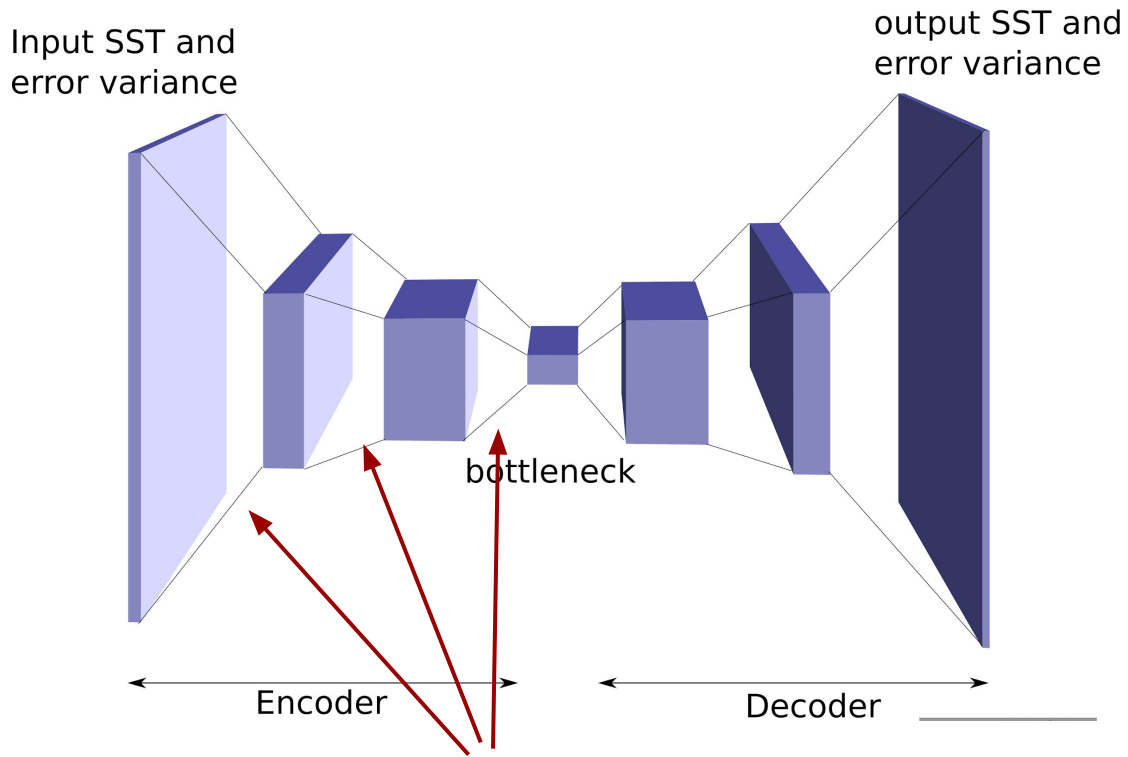
Missing data handled as data with different initial errors

- If missing, error variance ( $\sigma^2$ ) tends to  $\infty$

Input data:

- SST/ $\sigma^2$  (previous day, current day, following day)
- $1/\sigma^2$  (previous day, current day, following day)
- Longitude
- Latitude
- Time (cosine and sine of the year-day/365.25)

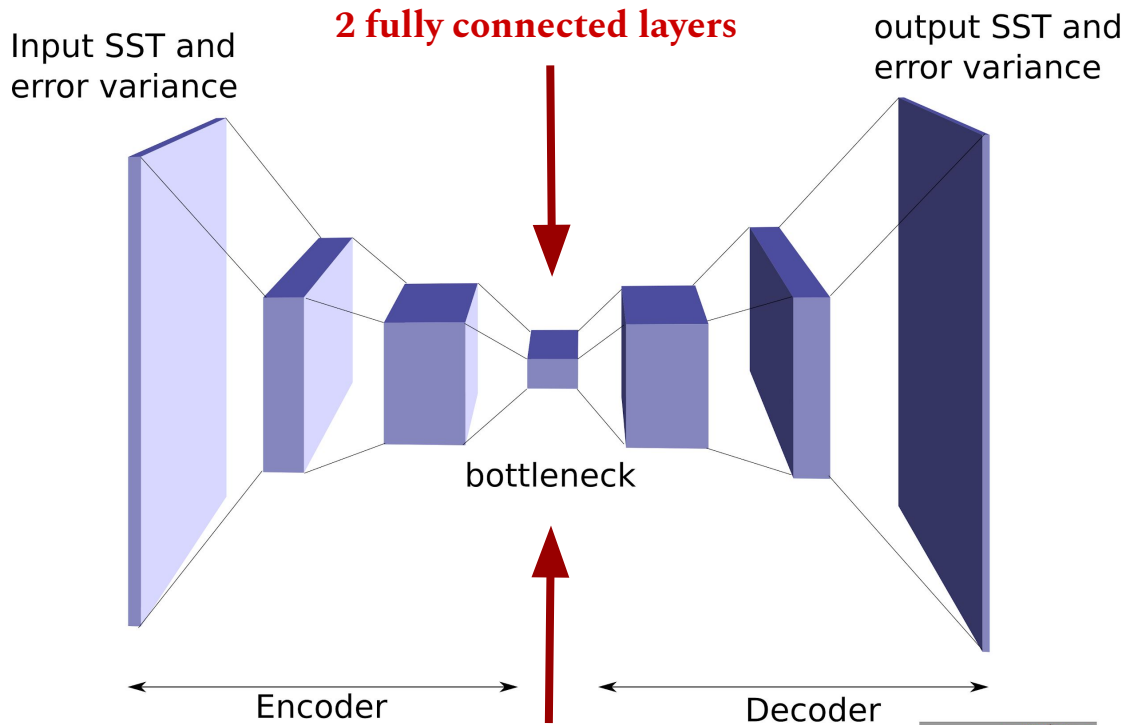




**Average pooling layers**

Reduce size by retaining the average value on 2x2 boxes





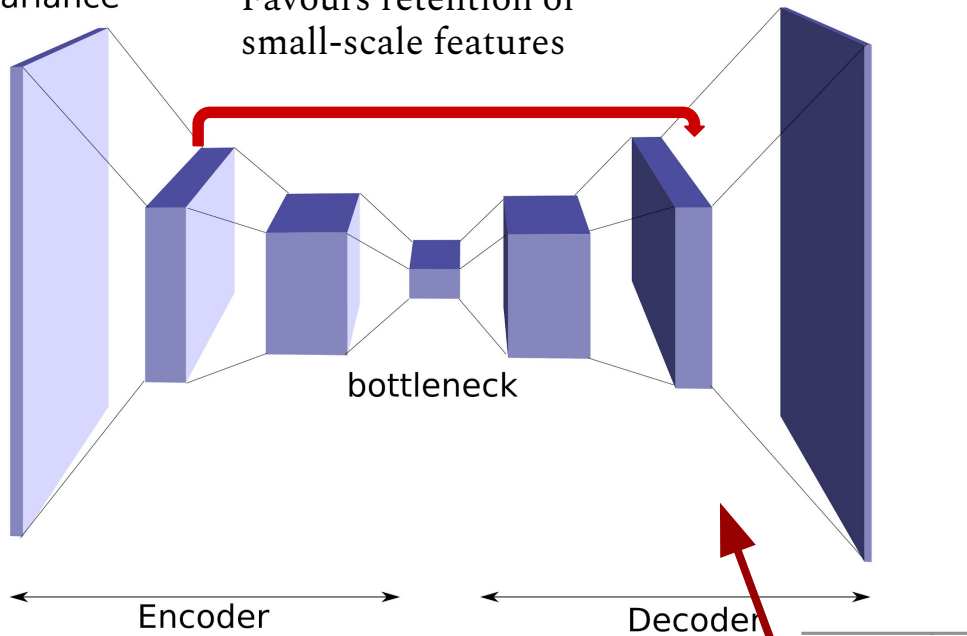
**+ 2 drop-out layers**

Take out 30% of neurons (pixels) to avoid overfitting

Input SST and error variance

**Skip connections:**  
Favours retention of small-scale features

output SST and error variance



**Decoding layers:**  
upscaling by nearest neighbour interpolation

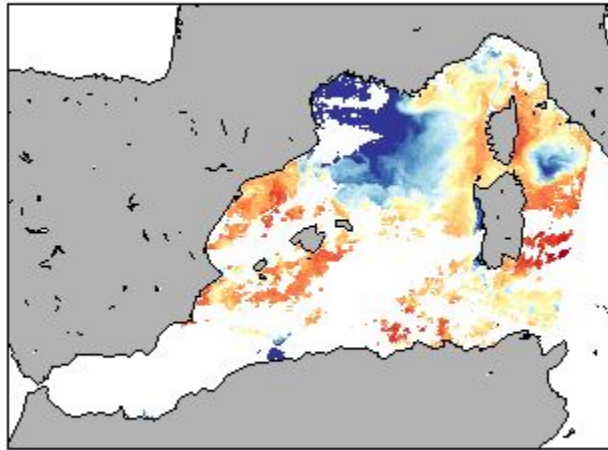
## Baseline method to be improved

**DINEOF** (Data Interpolating Empirical Orthogonal Functions)

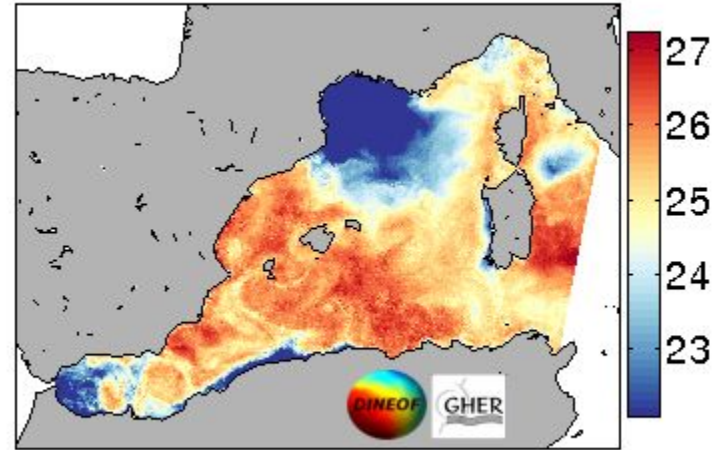
A reconstruction method based on the EOF basis from the dataset  
~15 years of development & improvements

<http://www.dineof.net/DINEOF/>

Original data



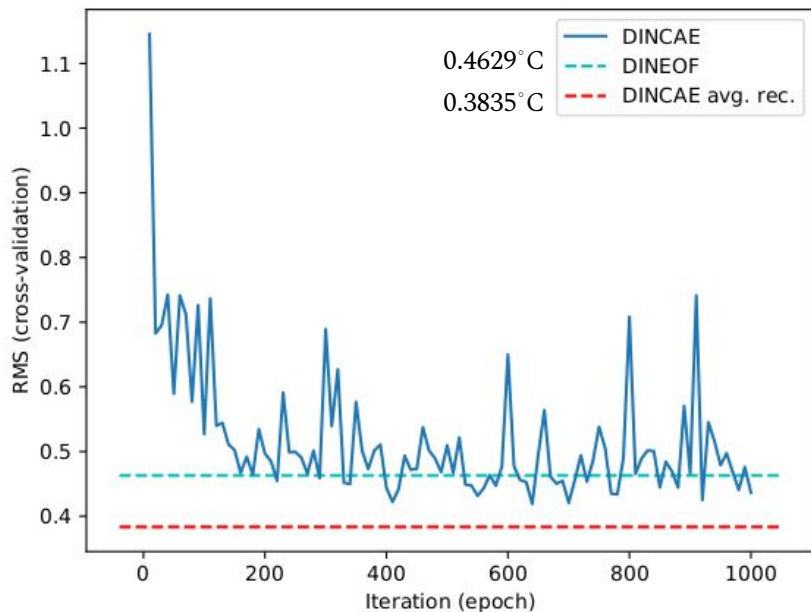
08-Sep-2019



# Results

Cross-validation: data removed from the last 50 images of the times series (with cloud mask from first 50 images)

Averaging epochs 200 to 100 improved DINCAE results



Reconstruction results -full time series-  
compared to WOD in situ data (under clouds)

RMS (DINEOF) 1.1676°C

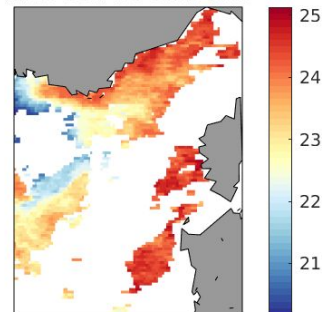
RMS (DINCAE) 1.1362°C



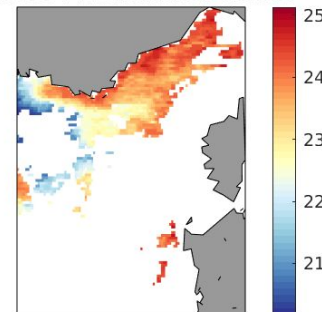
# Results

Reconstruction examples

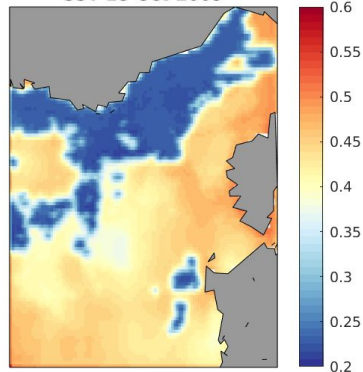
True SST 18-Oct-2009



True SST with added clouds

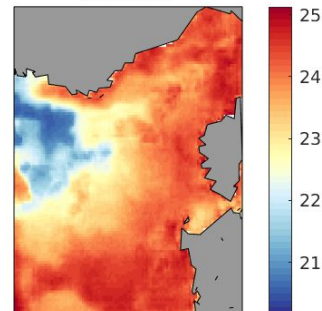


SST 18-Oct-2009

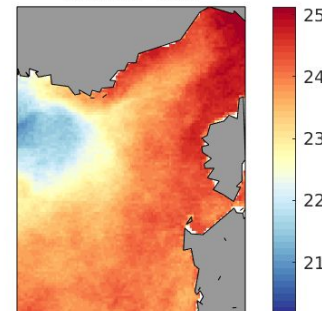


Expected error (std. dev.)

CAE SST



DINEOF SST

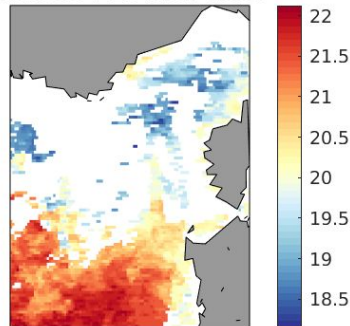


Small-scale variability

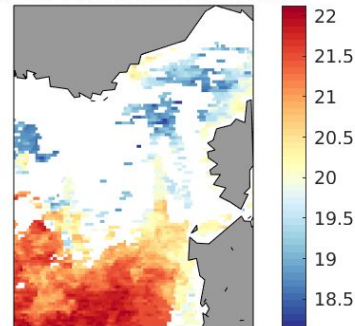
# Results

Reconstruction examples

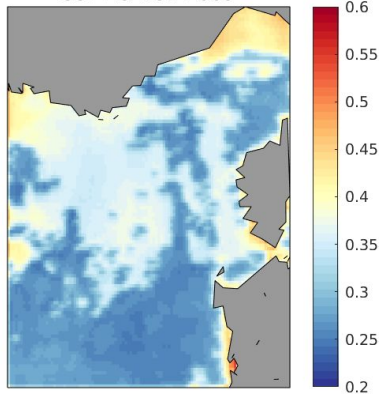
True SST 25-Nov-2009



True SST with added clouds

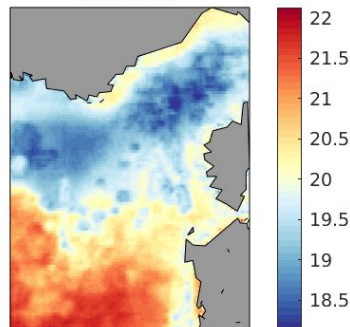


SST 25-Nov-2009

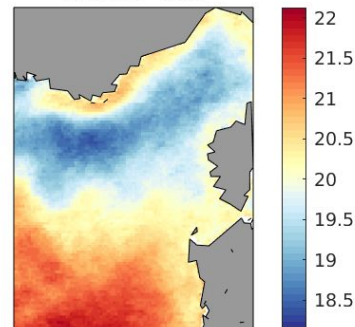


Expected error (std. dev.)

CAE SST



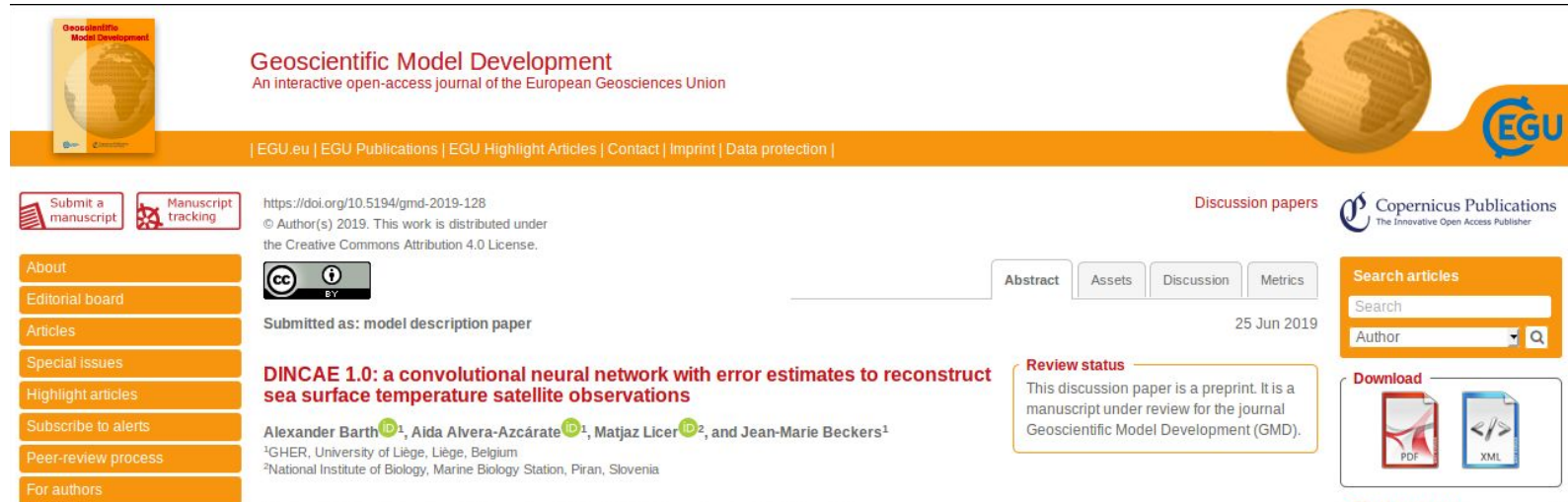
DINEOF SST



Some artifacts appear when too few data

## If you want to know more...

- Manuscript under revision (open review) in GMD



The screenshot shows the journal's homepage for the article "DINCAE 1.0: a convolutional neural network with error estimates to reconstruct sea surface temperature satellite observations". The page features a navigation bar with the journal title and logo, a sidebar with navigation links, and a main content area with a search bar and article details. The article is marked as a "Discussion paper" and is currently under review.

**Geoscientific Model Development**  
An interactive open-access journal of the European Geosciences Union

| EGU.eu | EGU Publications | EGU Highlight Articles | Contact | Imprint | Data protection |

Submit a manuscript | Manuscript tracking

https://doi.org/10.5194/gmd-2019-128  
© Author(s) 2019. This work is distributed under the Creative Commons Attribution 4.0 License.

Discussion papers | Copernicus Publications  
The Innovative Open Access Publisher

About | Editorial board | Articles | Special issues | Highlight articles | Subscribe to alerts | Peer-review process | For authors

Submitted as: model description paper

**DINCAE 1.0: a convolutional neural network with error estimates to reconstruct sea surface temperature satellite observations**

Alexander Barth<sup>1</sup>, Aida Alvera-Azcárate<sup>1</sup>, Matjaz Licer<sup>2</sup>, and Jean-Marie Beckers<sup>1</sup>  
<sup>1</sup>GHER, University of Liège, Liège, Belgium  
<sup>2</sup>National Institute of Biology, Marine Biology Station, Piran, Slovenia

Abstract | Assets | Discussion | Metrics

25 Jun 2019

**Review status**  
This discussion paper is a preprint. It is a manuscript under review for the journal Geoscientific Model Development (GMD).

Search articles  
Search  
Author

Download  
PDF | XML

- Code available at:

<https://github.com/gher-ulg/DINCAE>

# Conclusions & future work

A convolutional Autoencoder approach to reconstruct missing data

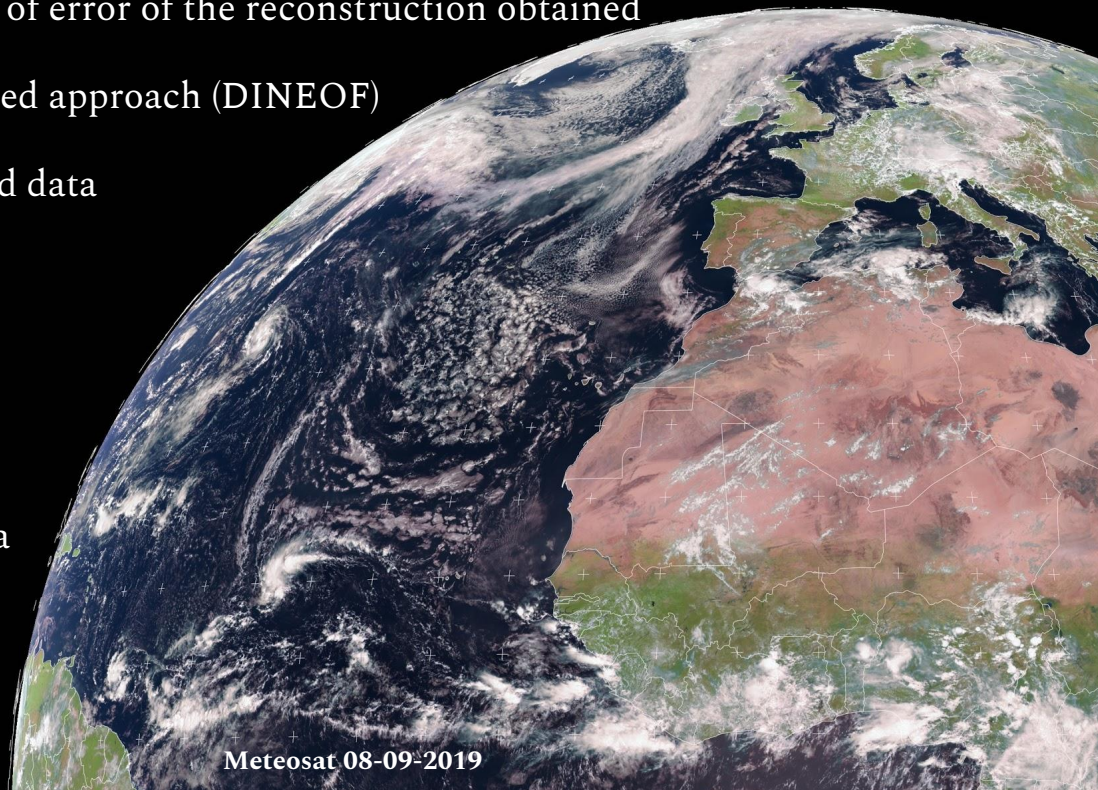
- Missing data handled by including expected error variance in the input data
- Estimation of missing data + estimation of error of the reconstruction obtained

Results similar (& slightly better) than established approach (DINEOF)

Small scale variability retrieved in reconstructed data

**Future work includes:**

- Multivariate analyses
- Work with higher spatial resolution data
- Use of other NN architectures





number	type	output size	parameters
1	input	112 x 112 x 8	
2	conv. 2d	112 x 112 x 16	n. filters = 16, kernel size = (3,3)
3	pooling 2d	56 x 56 x 16	pool size = (2,2), strides = (2,2)
4	conv. 2d	56 x 56 x 24	n. filters = 24, kernel size = (3,3)
5	pooling 2d	28 x 28 x 24	pool size = (2,2), strides = (2,2)
7	conv. 2d	28 x 28 x 36	n. filters = 36, kernel size = (3,3)
8	pooling 2d	14 x 14 x 36	pool size = (2,2), strides = (2,2)
9	conv. 2d	14 x 14 x 54	n. filters = 54, kernel size = (3,3)
10	pooling 2d	7 x 7 x 54	pool size = (2,2), strides = (2,2)
11	fully connected layer	529	
12	drop-out layer	529	drop-out rate for training = 0.3
13	fully connected layer	2646	
14	drop-out layer	2646	drop-out rate for training = 0.3
15	nearest neighbor interpolation	14 x 14 x 54	
16	concatenate output of 15 and 8	14 x 14 x 90	
17	conv. 2d	14 x 14 x 36	n. filters = 36, kernel size = (3,3)
18	nearest neighbor interpolation	28 x 28 x 36	
19	concatenate output of 18 and 5	28 x 28 x 60	
20	conv. 2d	28 x 28 x 24	n. filters = 24, kernel size = (3,3)
21	nearest neighbor interpolation	56 x 56 x 24	
22	concatenate output of 21 and 3	56 x 56 x 40	
23	conv. 2d	56 x 56 x 16	n. filters = 16, kernel size = (3,3)
24	nearest neighbor interpolation	112 x 112 x 16	
25	concatenate output of 24 and 1	112 x 112 x 26	
26	conv. 2d	112 x 112 x 2	n. filters = 2, kernel size = (3,3)