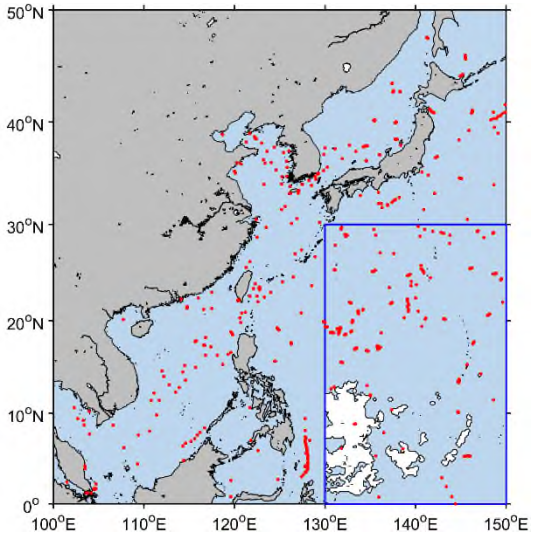
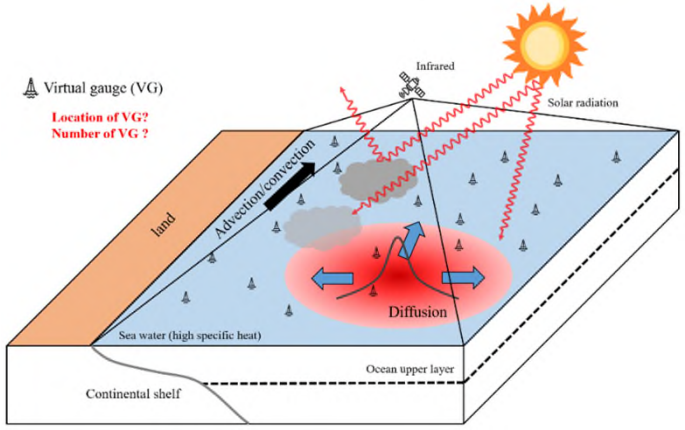
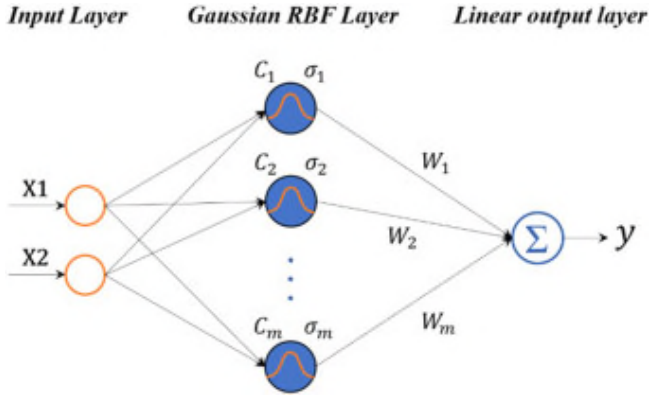
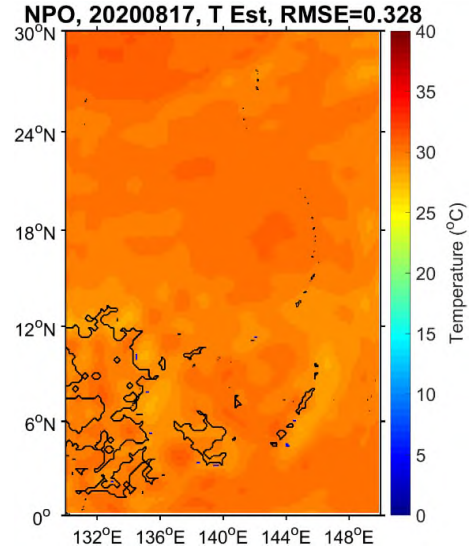
機器學習法可將物理過程整合
用以準確重建雲下及非雲下衛星反演海表溫嗎?
Can machine learning integrate physical processes to accurately reconstruct satellite-derived sea surface temperature under cloud and cloud-free areas?

中央氣象署「衛星環境監測技術發展與應用服務」子計畫









114年第三十九屆天氣分析與預報研討會，2025/09/04

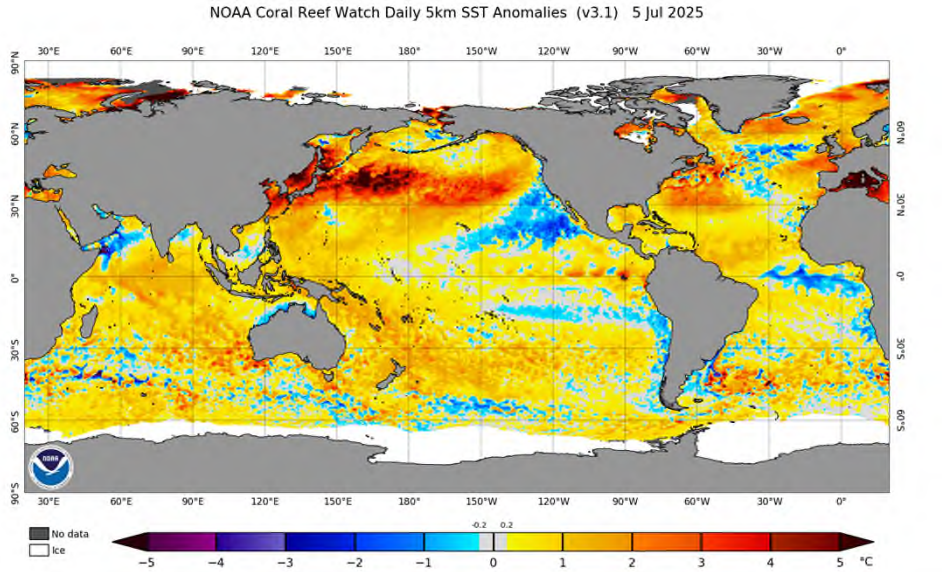
39th Conference on Weather Analysis and Forecasting

國立臺灣海洋大學

鄭佑建，王重皓，楊智傑，李明安，吳俊宏

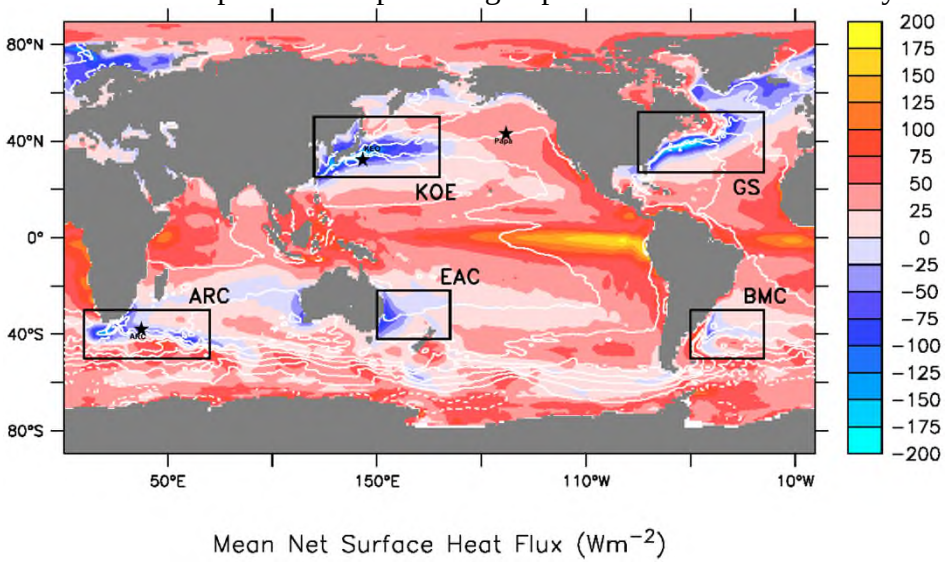
Introduction

SSTA

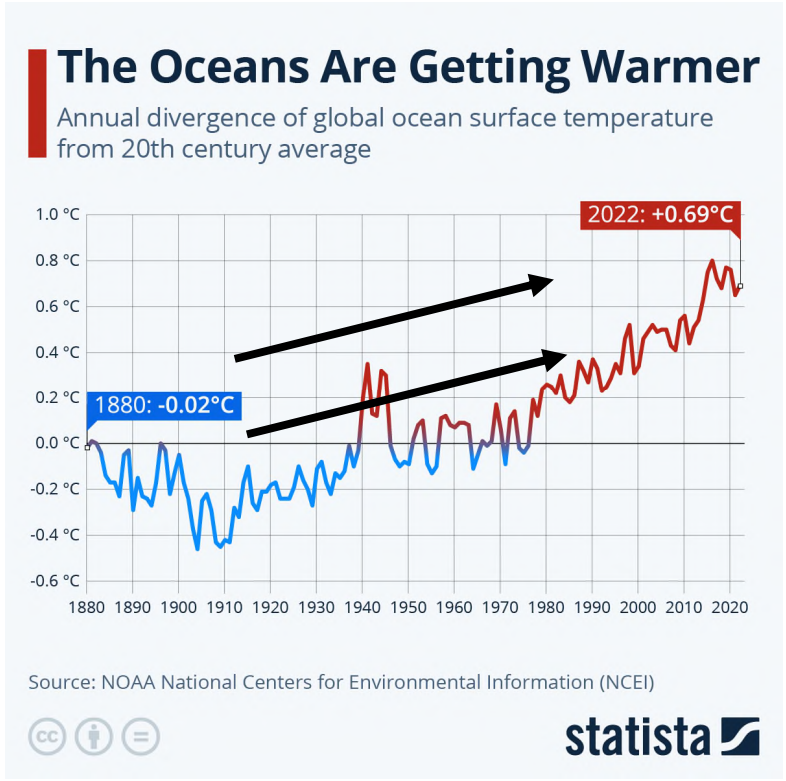


Source: https://www.ospo.noaa.gov/products/ocean/sst/anomaly/

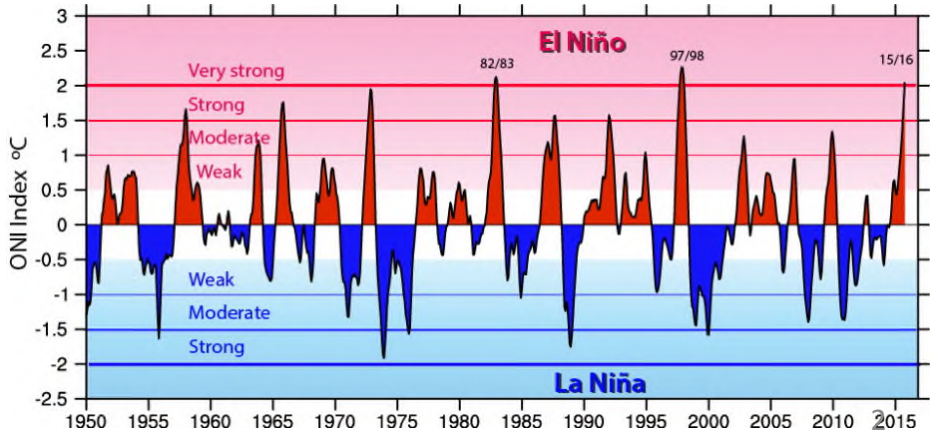
Heat flux



Source: https://www.pmel.noaa.gov/ocs/air-sea-fluxes

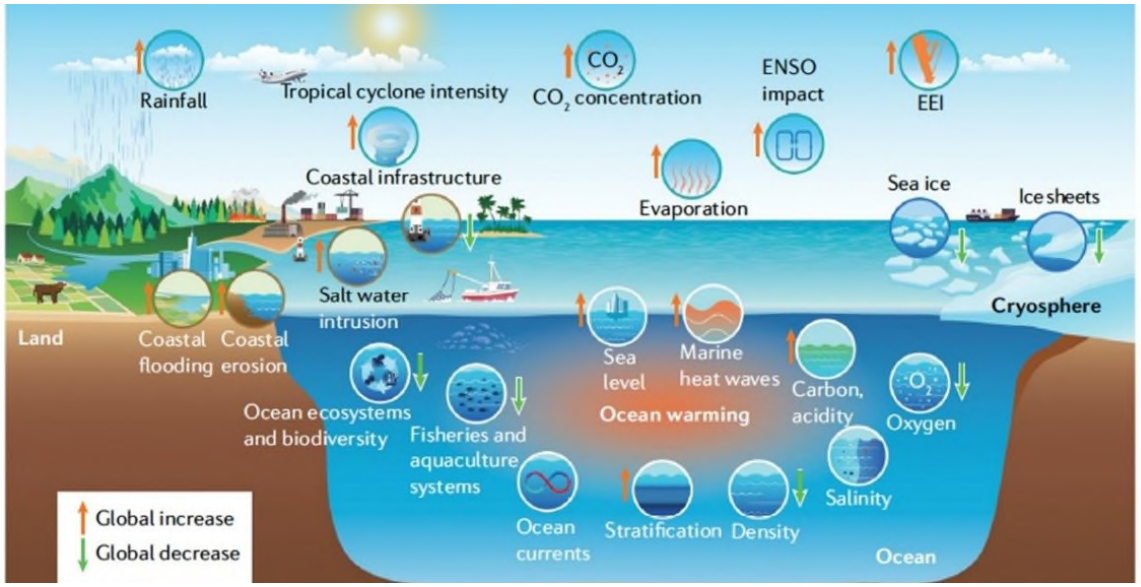


Ocean Niño Index



Source: National Center for Atmospheric Research (NCAR)

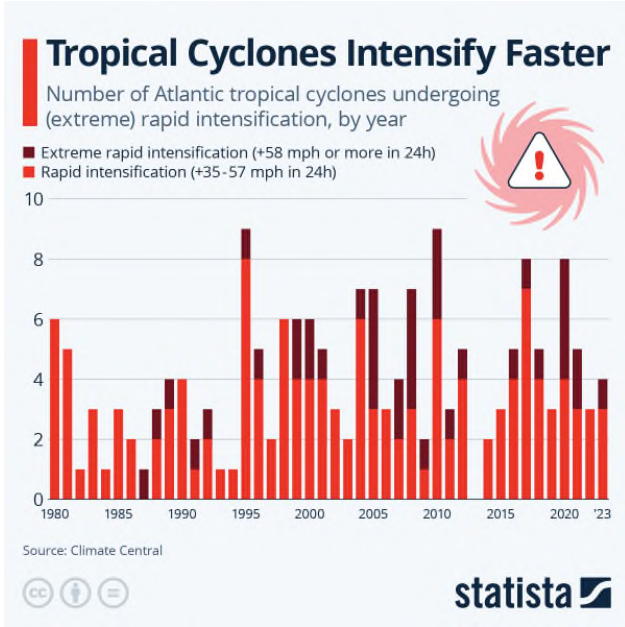
Introduction



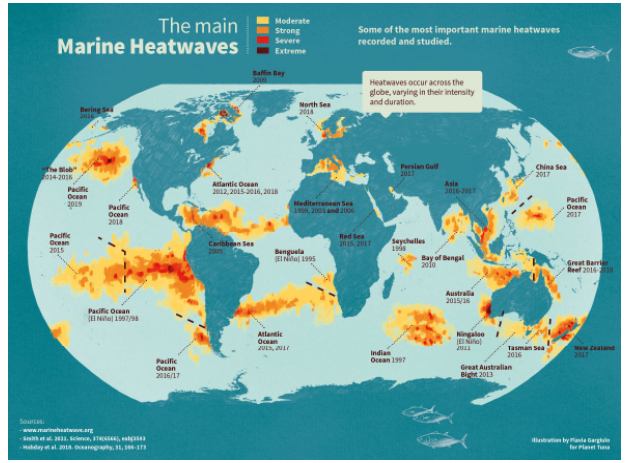
Source: https://www.mercator-ocean.eu/fr/a-new-comprehensive-assessment-of-ocean-warming-highlights-future-climate-risks-with-participation-of-mercator-ocean-expertise/

Obtaining the SST data is a fundamental and important task!

Tropical cyclone



Marine heat wave

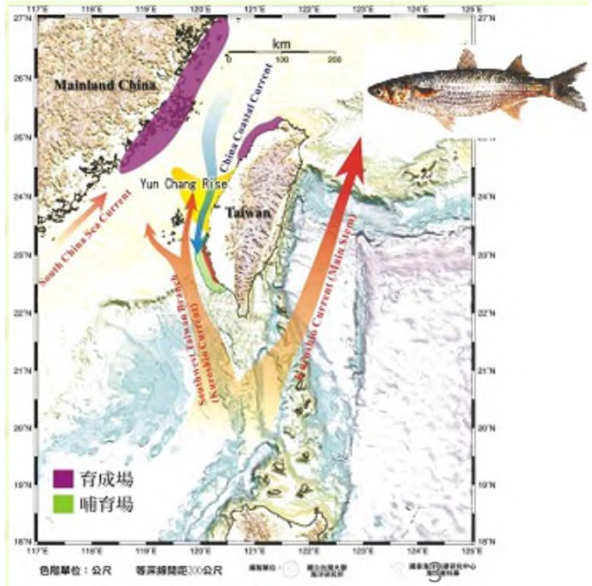


Source: https://www. planettuna.com

Coral reef bleaching

Source: people4ocean Sun Care

Fishing ground



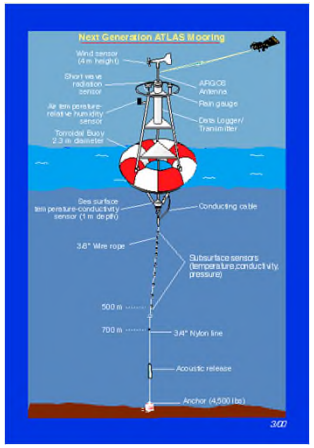
Source: https://www.sdec.ntpc.edu.tw/epaper/10206/1.htm

Observation of SST

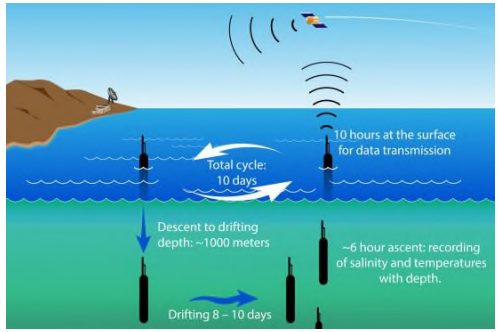
In-Situ data

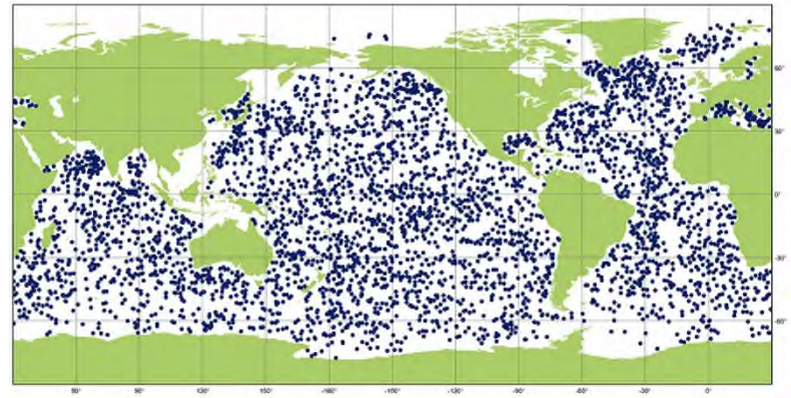
Shipped-loading CTD

Buoy (mooring anchorage)



ARGO

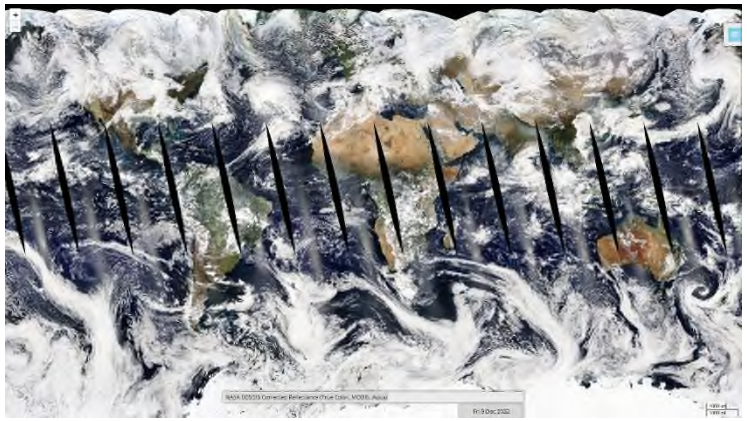




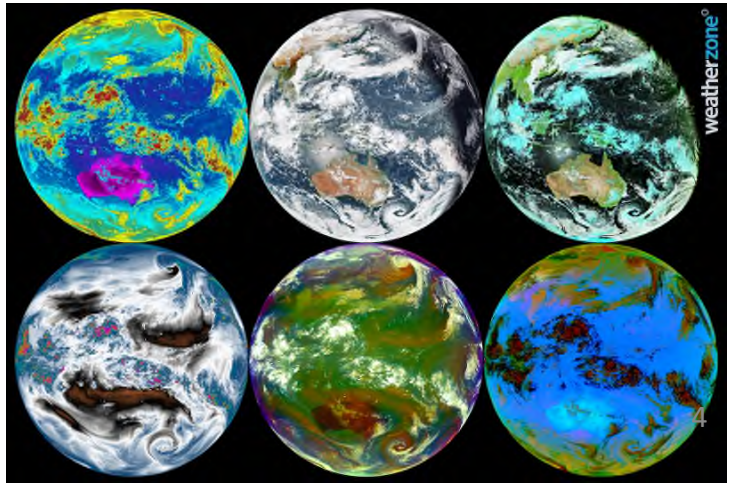
Source: https://argo.ucsd.edu/

Satellite

Polar orbiting satellites (MODIS, Aqua)

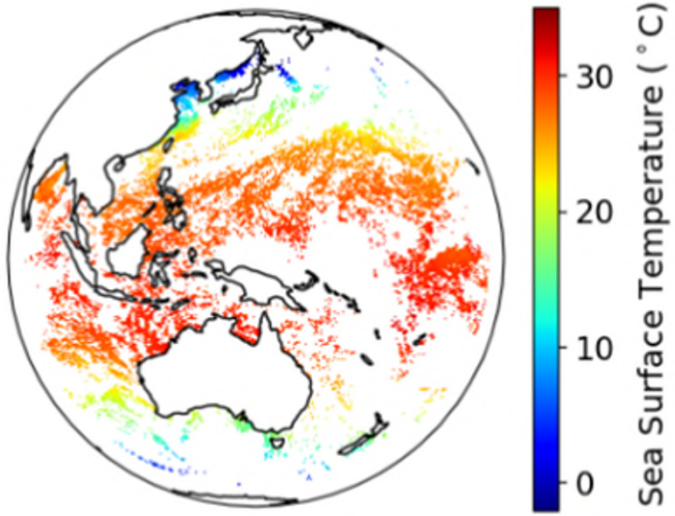


Geostationary satellites (Himawari 8/9)

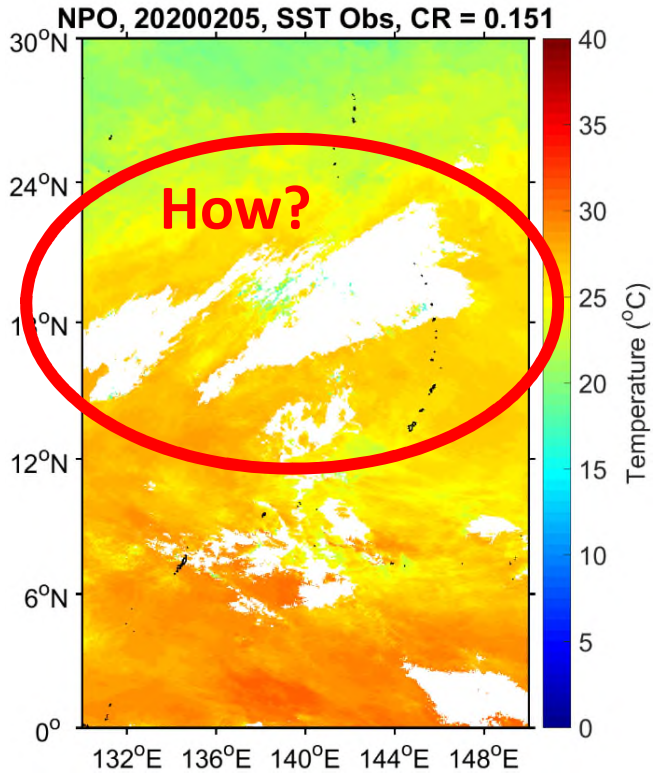


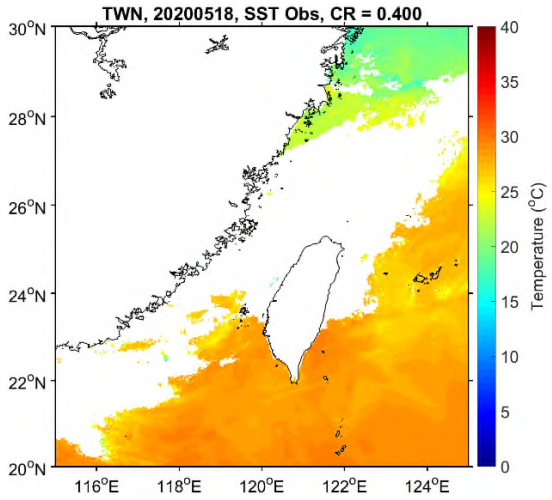
Challenge: cloud coverage

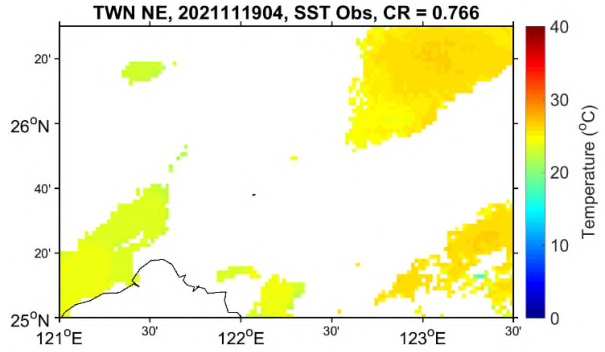
The amount of available data can be less than 30%!



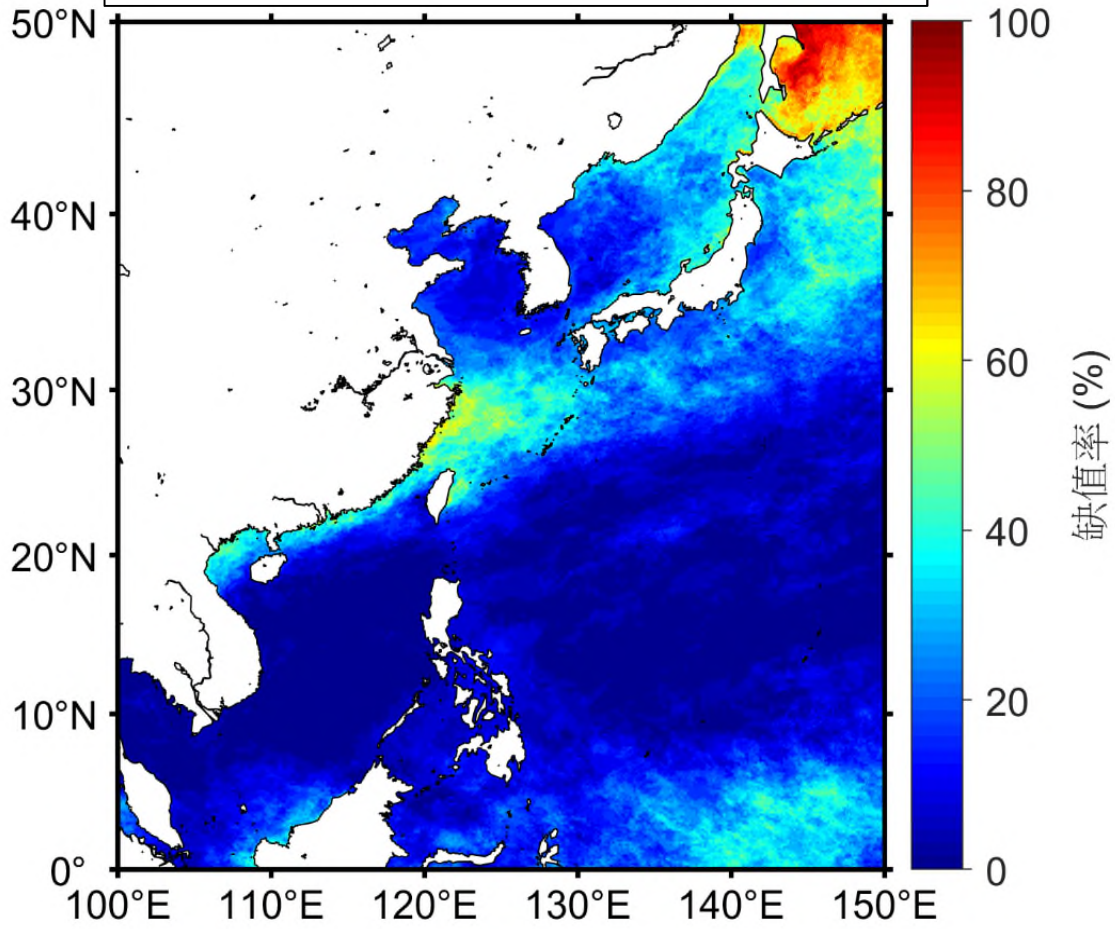
[1] Govekar et al., 2024







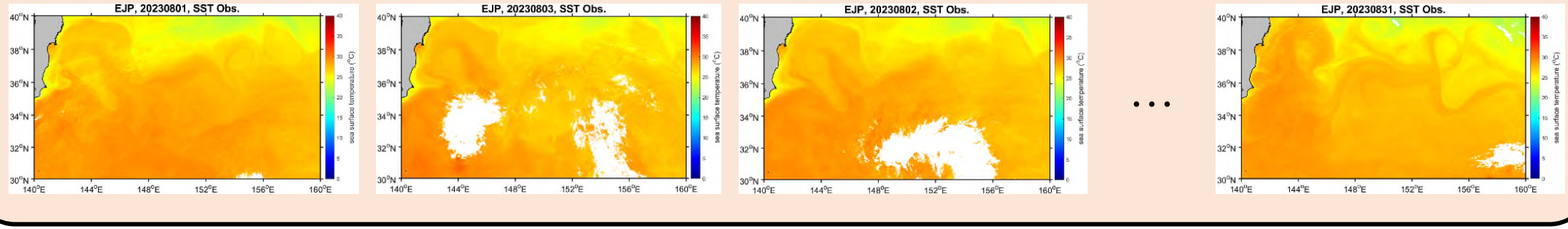
Cloud ratio from 3/1 to 3/31 in 2020



Two solution：
A. Ocean modeling (driven force)
B. Data analyzing

Data Interpolation & Empirical Orthogonal Functions (DINEOF)

[2] Beckers et al., 2003



Step1

Rearrange the data

&

Fill missing data ($SSTA_{i,j}$) with 0

$$x = \begin{bmatrix} SSTA_{1,1} & SSTA_{1,2} & \cdots & SSTA_{1,n} \\ SSTA_{2,1} & SSTA_{2,2} & \cdots & SSTA_{2,1} \\ \vdots & & \ddots & \vdots \\ SSTA_{m,1} & SSTA_{m,2} & \cdots & SSTA_{m,n} \end{bmatrix}$$

Step2

Singular Value Decomposition (SVD)

&

Empirical Orthogonal Functions (EOF)

SVD

$$x_0 = USV^T$$

EOF

U_k S_k V_k

(K : number of optimal modes)

Step3

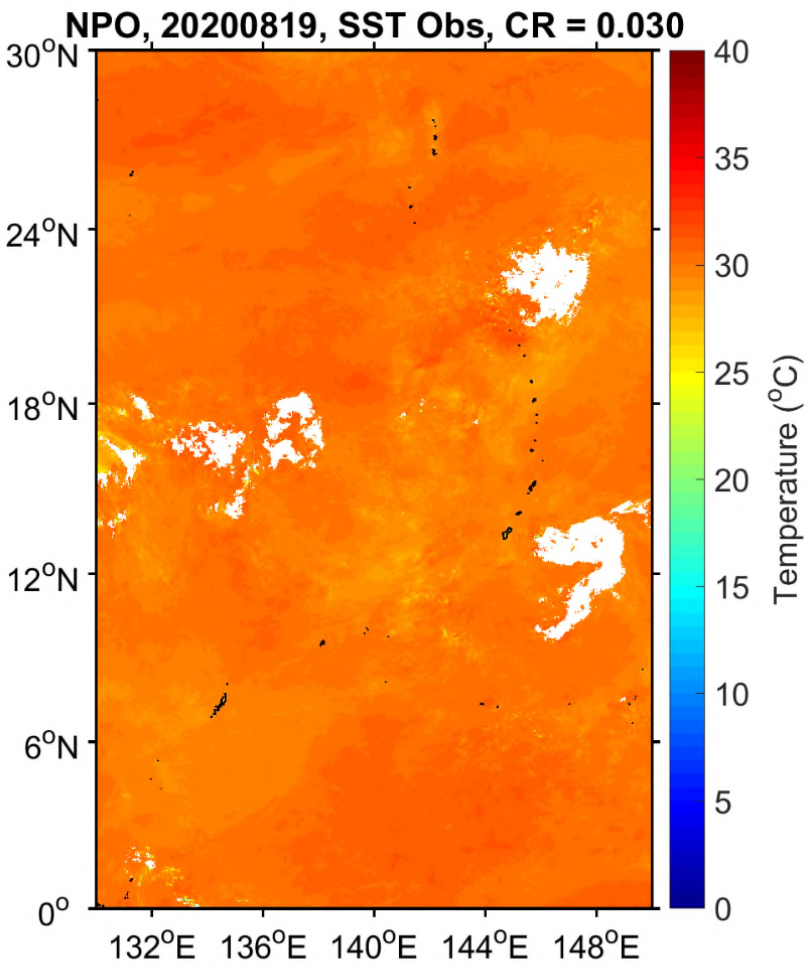
Replace the missing value with reconstruction results

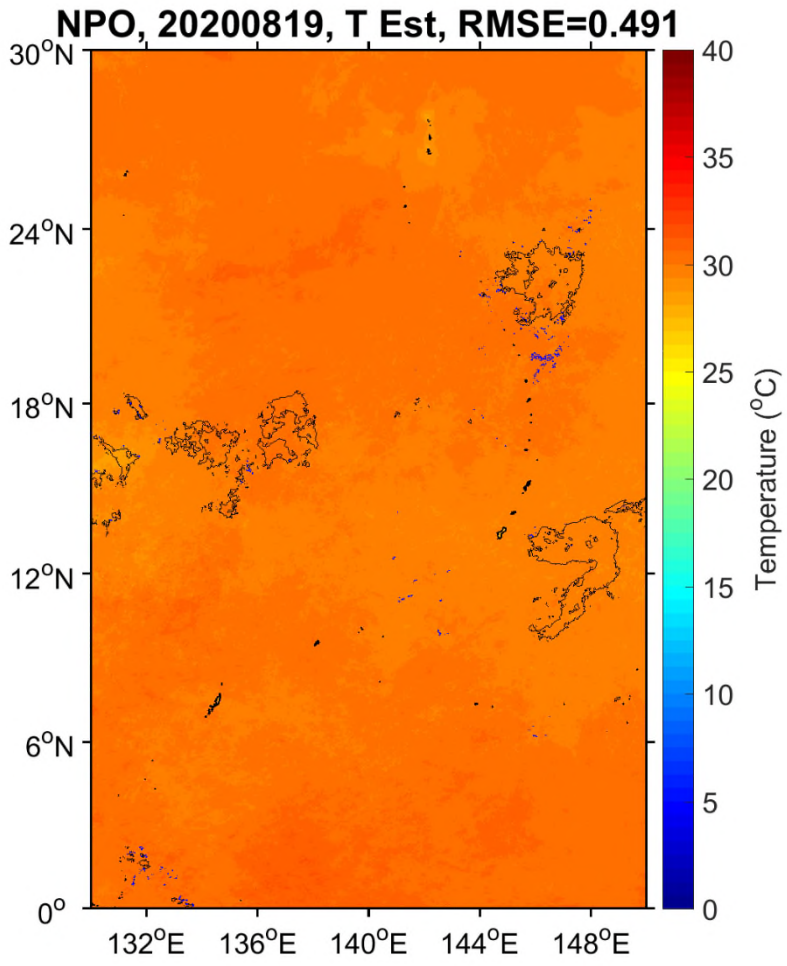
$$(x_{rec})_{i,j} = U_k S_k V_k^T$$

$$x_{rec} = x_0 + (x_{rec})_{i,j}$$

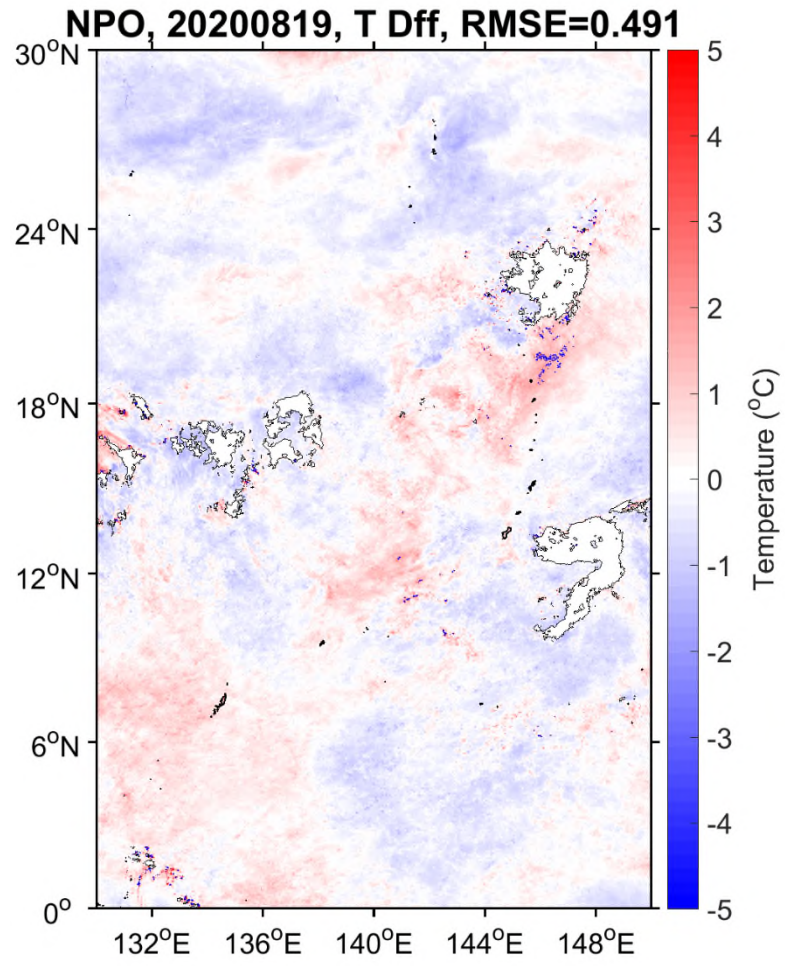
iteration

Reconstruction by DINEOF



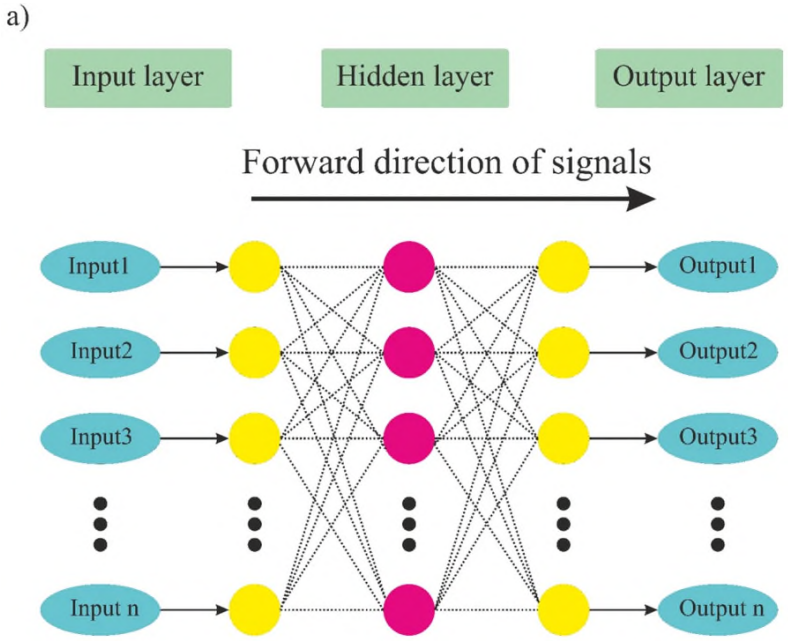


(DINEOF – Obs.)

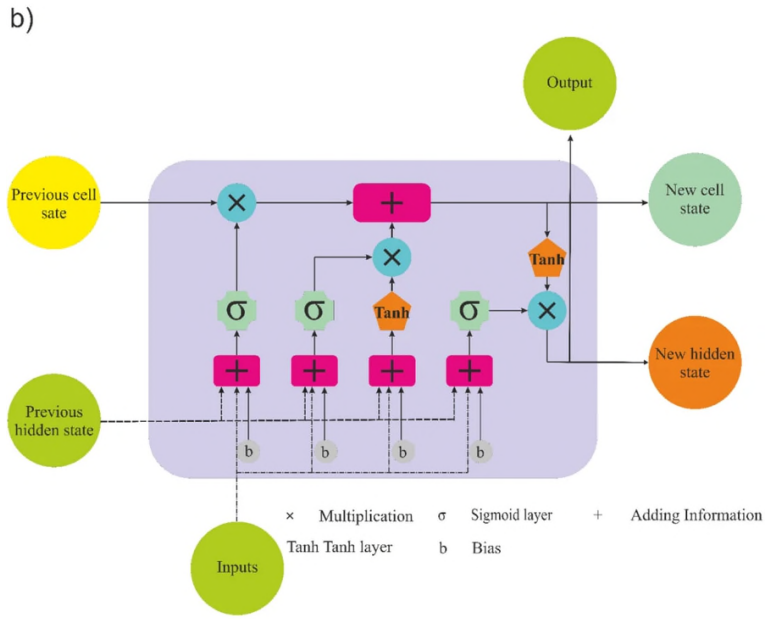


SST prediction/analysis using Artificial Intelligence approaches -Neural networks

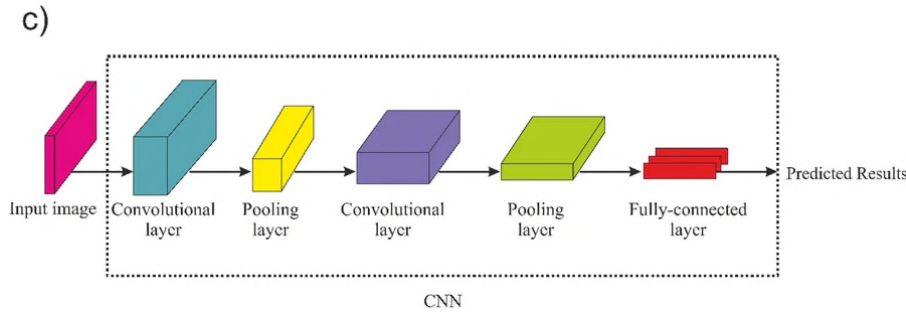
Data-driven approaches with advantages for nonlinear problems



MLP (BPNN) …



RNN, LSTM, …



CNN

How to apply these data-driven approaches while no SST data can be obtained under cloud area?

[3] Haghbin et al., 2023

Data INterpolation Convolutional Auto-Encoder

DINCAE

standard optimal interpolation equations

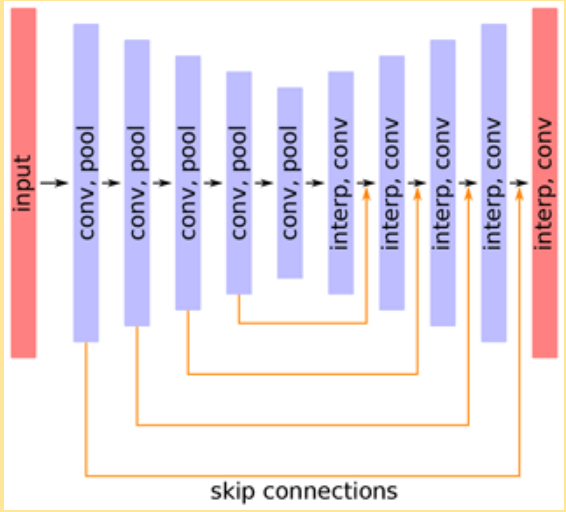
(Bretherton et al., 1976; Buongiorno Nardelli, 2012)

$$P^{a-1}x^a = P^{f^{-1}}x^f + H^T R^{-1} y^o$$

$$P^{a-1} = P^{f^{-1}} + H^T R^{-1} H$$

+

Convolutional Auto-Encoder



Geosci. Model Dev., 13, 1609–1622, 2020
https://doi.org/10.5194/gmd-13-1609-2020

Geoscientific Model Development
EGU

DINCAE 1.0: a convolutional neural network with error estimates to reconstruct sea surface temperature satellite observations [4]

Alexander Barth[1], **Aida Alvera-Azcárate**[1], **Matjaz Licer**[2], **and Jean-Marie Beckers**[1]

[1]GeoHydrodynamics and Environment Research (GHER), University of Liège, Liège, Belgium
[2]National Institute of Biology, Marine Biology Station, Piran, Slovenia

Correspondence: Alexander Barth (a.barth@uliege.be)

Received: 8 May 2019 – Discussion started: 25 June 2019
Revised: 25 February 2020 – Accepted: 28 February 2020 – Published: 27 March 2020

Layer number	Type	Output size	Parameters
1	input	112 × 112 × 8	
2	conv. 2D	112 × 112 × 16	no. filters = 16, kernel size = (3,3)
3	pooling 2D	56 × 56 × 16	pool size = (2,2), strides = (2,2)
4	conv. 2D	56 × 56 × 24	no. filters = 24, kernel size = (3,3)
5	pooling 2D	28 × 28 × 24	pool size = (2,2), strides = (2,2)
7	conv. 2D	28 × 28 × 36	no. filters = 36, kernel size = (3,3)
8	pooling 2D	14 × 14 × 36	pool size = (2,2), strides = (2,2)
9	conv. 2D	14 × 14 × 54	no. filters = 54, kernel size = (3,3)
10	pooling 2D	7 × 7 × 54	pool size = (2,2), strides = (2,2)
11	fully connected layer	529	
12	dropout layer	529	dropout rate for training = 0.3
13	fully connected layer	2646	
14	dropout layer	2646	dropout rate for training = 0.3
15	nearest-neighbor interpolation	14 × 14 × 54	
16	concatenate output of 15 and 8	14 × 14 × 90	
17	conv. 2D	14 × 14 × 36	no. filters = 36, kernel size = (3,3)
18	nearest-neighbor interpolation	28 × 28 × 36	
19	concatenate output of 18 and 5	28 × 28 × 60	
20	conv. 2D	28 × 28 × 24	no. filters = 24, kernel size = (3,3)
21	nearest-neighbor interpolation	56 × 56 × 24	
22	concatenate output of 21 and 3	56 × 56 × 40	
23	conv. 2D	56 × 56 × 16	no. filters = 16, kernel size = (3,3)
24	nearest-neighbor interpolation	112 × 112 × 16	
25	concatenate output of 24 and 1	112 × 112 × 26	
26	conv. 2D	112 × 112 × 2	no. filters = 2, kernel size = (3,3)

Data INterpolation Convolutional Auto-Encoder

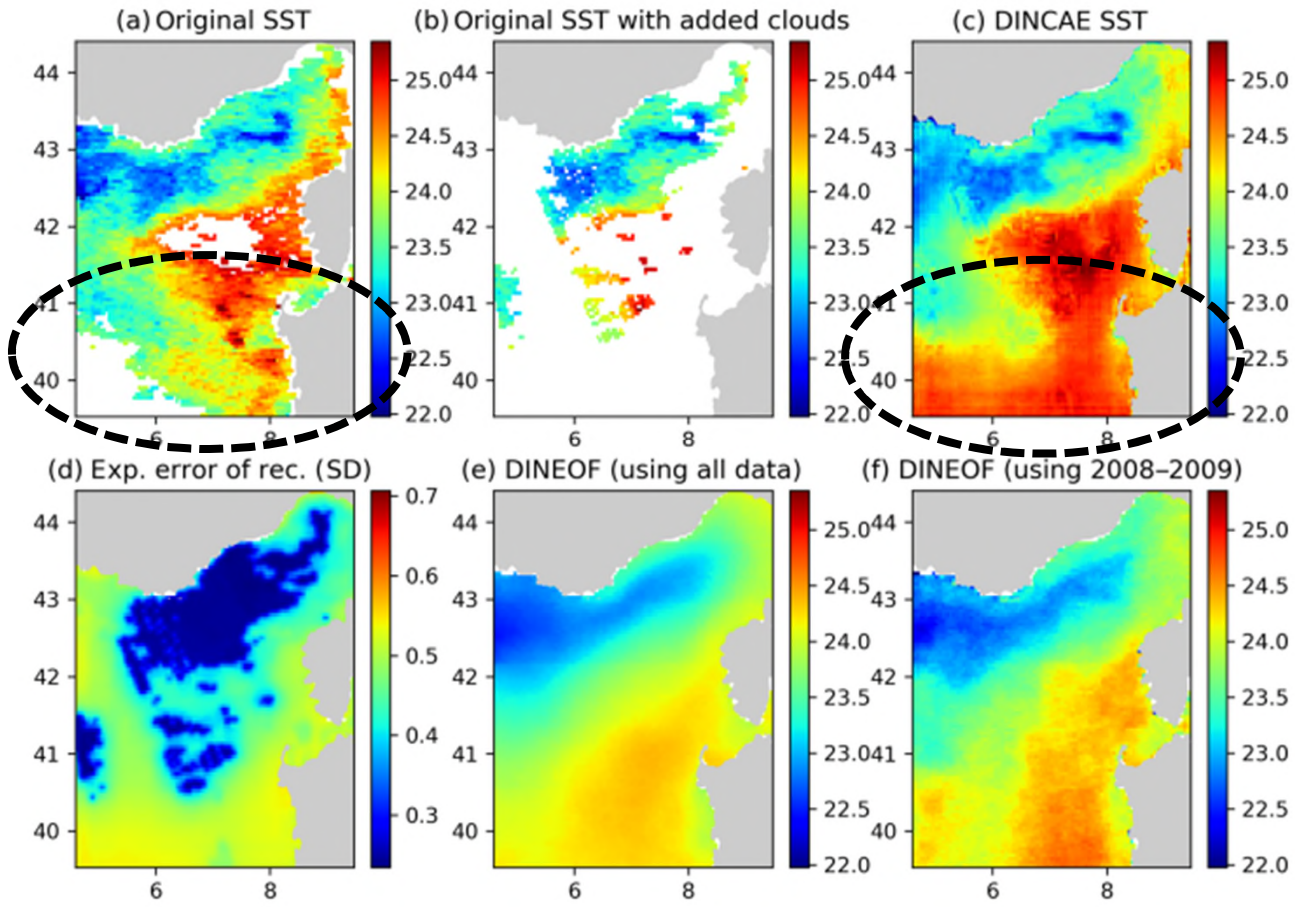
8 inputs, 26 layers, 2 outputs

Inputs

(1)	SST anomalies scaled by the inverse of the error variance (the scaled anomaly is zero if the data are missing)
(2)	inverse of the error variance (zero if the data are missing)
(3)	scaled SST anomalies and inverse of error variance of the previous day
(4)	scaled SST anomalies and inverse of error variance of the next day
(5)	longitude (scaled linearly between −1 and 1)
(6)	latitude (scaled linearly between −1 and 1)
(7)	cosine of the day of the year divided by 365.25
(8)	sine of the day of the year divided by 365.25

Outputs

(1)	SST scaled by the inverse of the expected error variance
(2)	logarithm of the inverse of the expected error variance

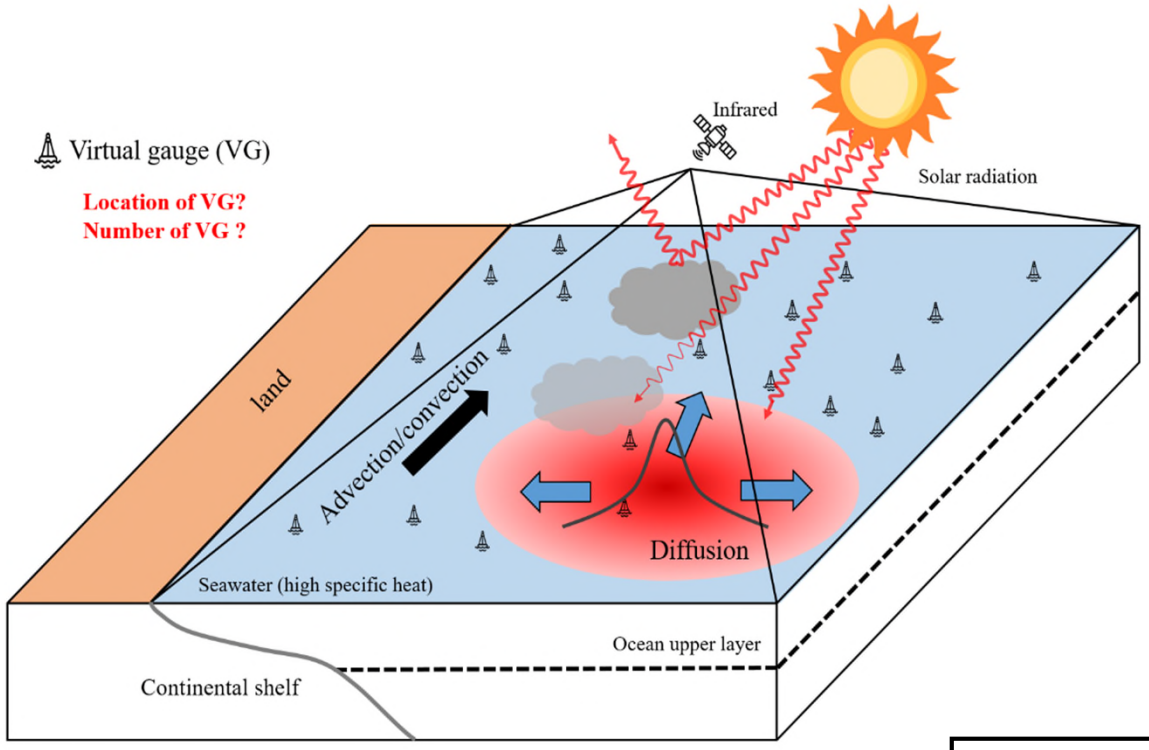


	RMSE	CRMSE	Bias
DINEOF	1.1676	1.1102	−0.3616
DINCAE	1.1362	1.0879	−0.3278

(Barth et al., 2020)

Virtual gauge, physically consistent approach

[5] Young et.al, 2024



- 1-D heat and advection-diffusion equation

[6] Taylor, 1920

$$\frac{\partial T}{\partial t} + u\frac{\partial T}{\partial x} = K_x \frac{\partial^2 T}{\partial x^2} + heat\ source$$

- theoretical analysis

$$T(x - x_o, t - t_o) = \frac{H_o/C_p}{\rho A\sqrt{4\pi K_x (t - t_o)}} e^{-\frac{1}{4}\left[\frac{(x-x_o)-u(t-t_o)]^2}{K_x(t-t_o)}\right.}$$

or

$$T(x - x_o, t - t_o) = T_{max} e^{-\frac{1}{2}\frac{[(x-x_o)-u(t-t_o)]^2}{2K_x(t-t_o)}}$$

$$\Rightarrow \quad T(x - x_o) = T_{max} e^{-\frac{1}{2}\frac{(x-x_c)^2}{\sigma_x^2}} \quad \& \quad T(t - t_o) = T_{max} e^{-\frac{1}{2}\frac{(t-t_c)^2}{\sigma_t^2}}$$

Radial Based Function Neural Network (RBFNN)

- Analytical form:

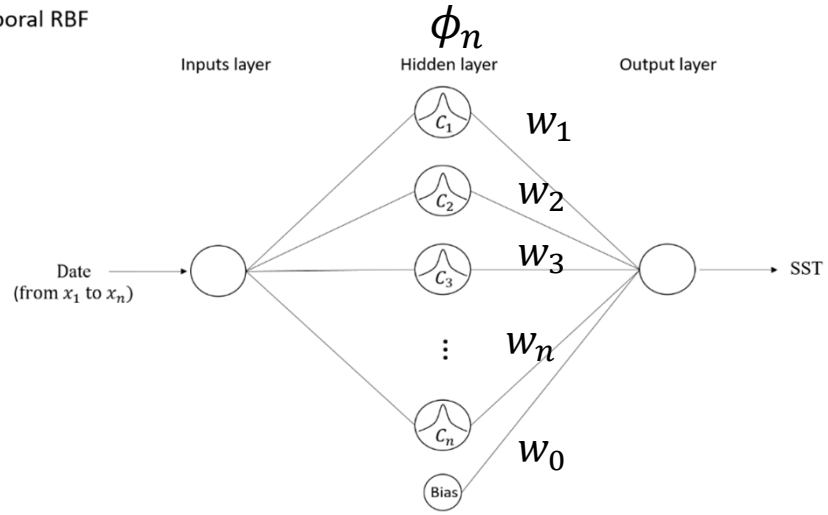
$$T(t-t_o) = T_{max} e^{-\frac{1(t-t_c)^2}{2\ \sigma_t^2}} \quad \& \quad T(x-x_o) = T_{max} e^{-\frac{1(x-x_c)^2}{2\ \sigma_x^2}}$$

Integrate physical rule into AI/ML model

- RBF (Gaussian function)

$$y = rbfnn(x) = \sum_{k=1}^{n} w_k \phi_k + w_0 = \sum_{k=1}^{n} w_k \exp\left(-\frac{\|x - c_k\|^2}{2\sigma_n^2}\right) + w_0$$

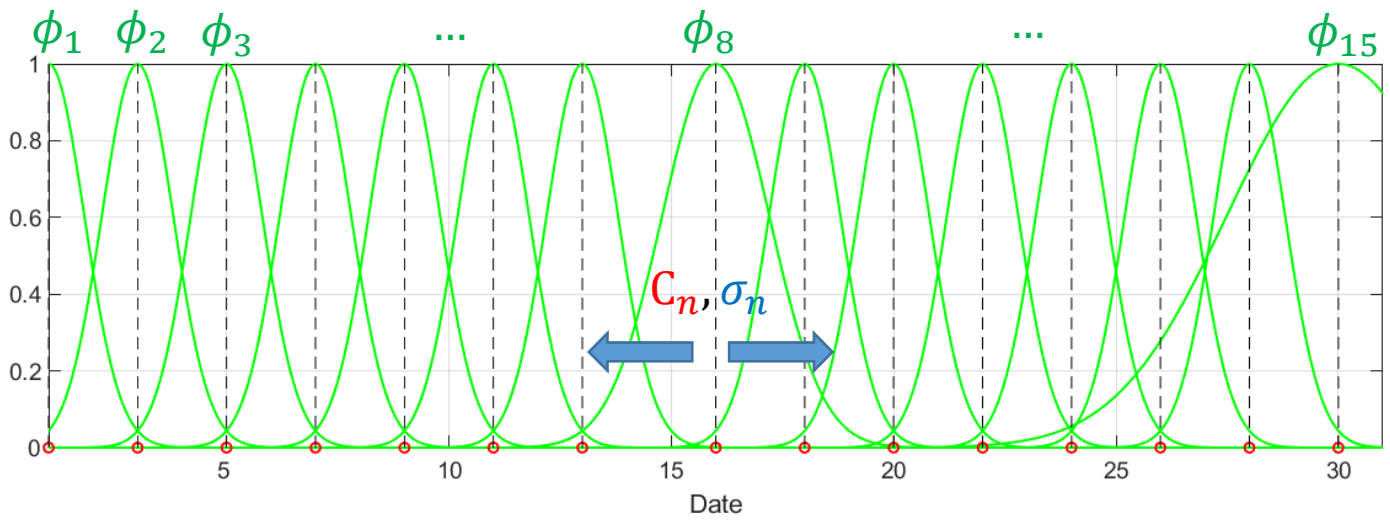
(or SSTr) (or t)



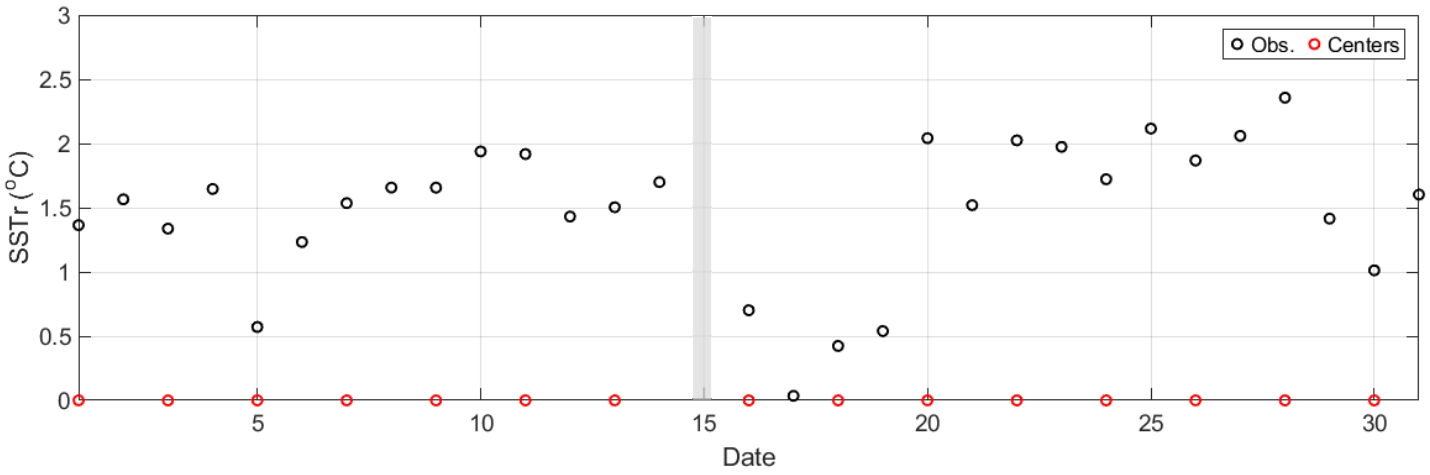
(x : 1 to m, m is **the days of month** or **the location**)

$$[w_1 \quad w_2 \quad \dots \quad w_n] \cdot \begin{bmatrix} \exp\left(-\frac{\|x_1-c_1\|^2}{2\sigma_1^2}\right) & \exp\left(-\frac{\|x_2-c_1\|^2}{2\sigma_1^2}\right) & \dots & \exp\left(-\frac{\|x_m-c_1\|^2}{2\sigma_1^2}\right) \\ \exp\left(-\frac{\|x_1-c_2\|^2}{2\sigma_2^2}\right) & \exp\left(-\frac{\|x_2-c_2\|^2}{2\sigma_2^2}\right) & \dots & \exp\left(-\frac{\|x_m-c_2\|^2}{2\sigma_2^2}\right) \\ \vdots & & \ddots & \vdots \\ \exp\left(-\frac{\|x_1-c_n\|^2}{2\sigma_n^2}\right) & \exp\left(-\frac{\|x_2-c_n\|^2}{2\sigma_n^2}\right) & \dots & \exp\left(-\frac{\|x_m-c_n\|^2}{2\sigma_n^2}\right) \end{bmatrix} = [SSTr_1 \quad SSTr_2 \quad \dots \quad SSTr_m]$$

Radial Based Function Neural Network (RBFNN)



(σ_n= the difference of centers * 0.4)



Step1 Determine the centers & sigma

$$\sum_{k=1}^{n} w_n \exp\left(-\frac{\|t_m - c_n\|^2}{2\sigma_n^2}\right) + w_0 = SSTr_m$$

or

$$[w_1 \quad w_2 \quad \dots \quad w_n] \cdot \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_n \end{bmatrix}$$

$$= [SSTr_1 \quad SSTr_2 \quad \dots \quad SSTr_m]$$

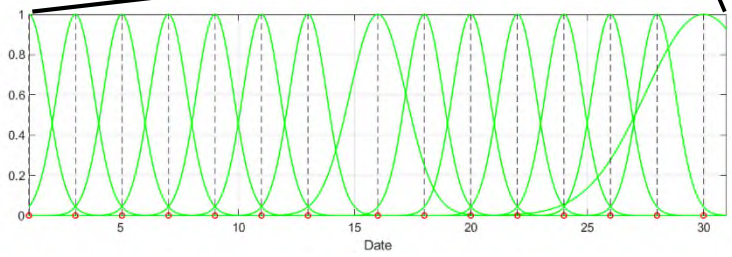
Step2 Pseudo-inverse matrix solution for w

$[w_1 \quad w_2 \quad \dots \quad w_n]$ (solved)

Radial Based Function Neural Network (RBFNN)

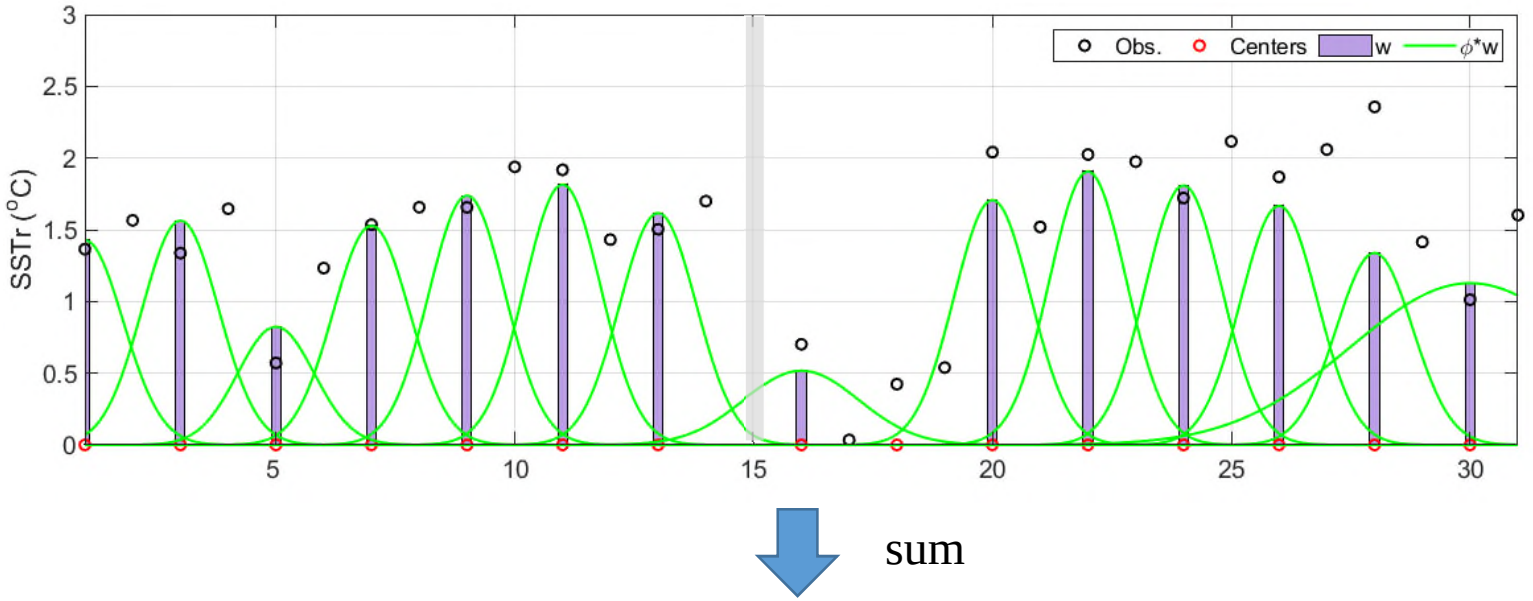
Step3 Estimation (reconstruction)

$$[w_1 \quad w_2 \quad \dots \quad w_n](solved) \cdot \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_n \end{bmatrix}$$

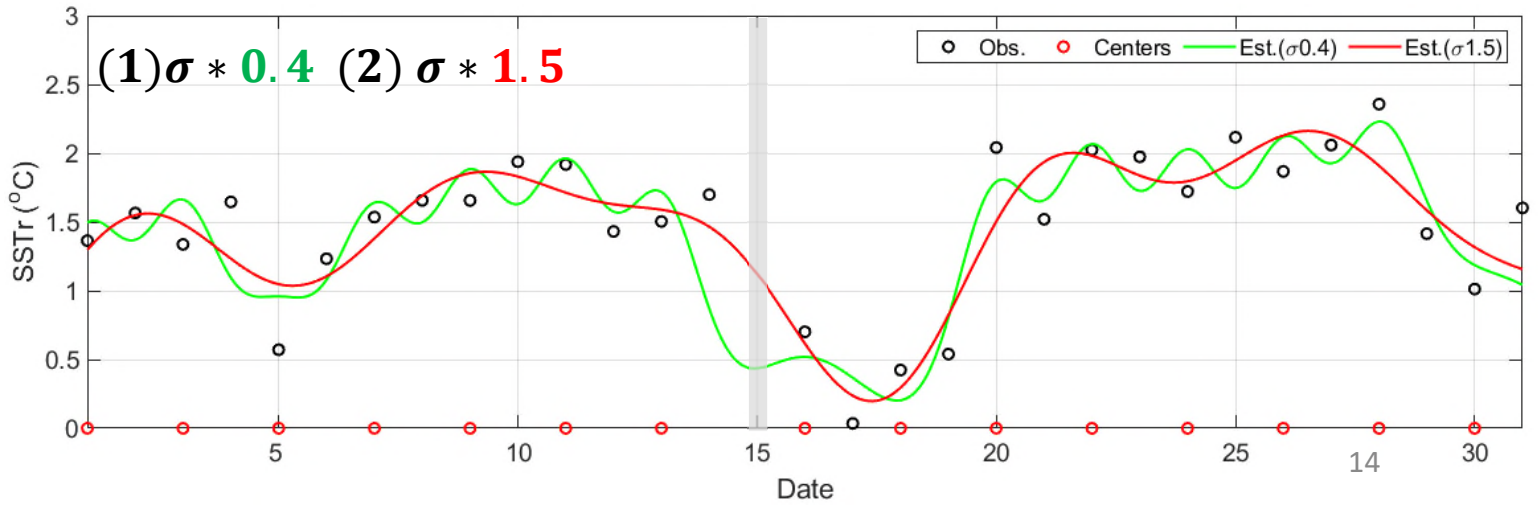


||

$$[SSTrec_1 \quad SSTrec_2 \quad \dots \quad SSTrec_m]$$

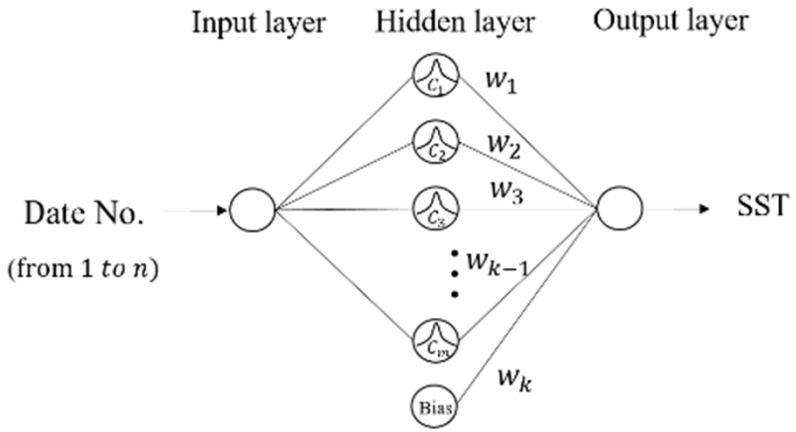


sum

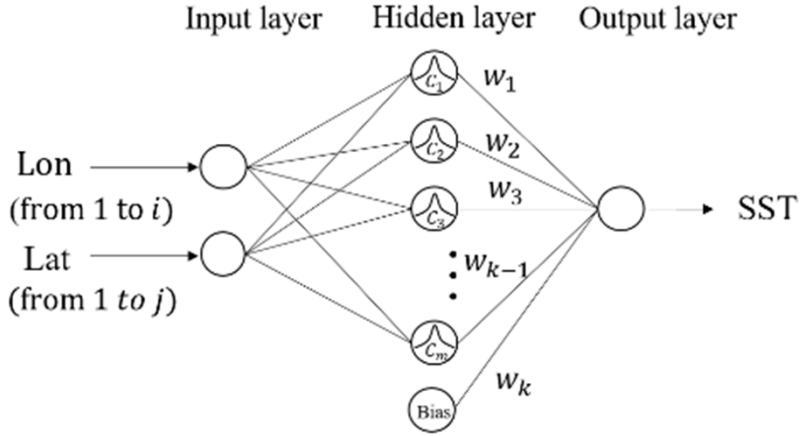


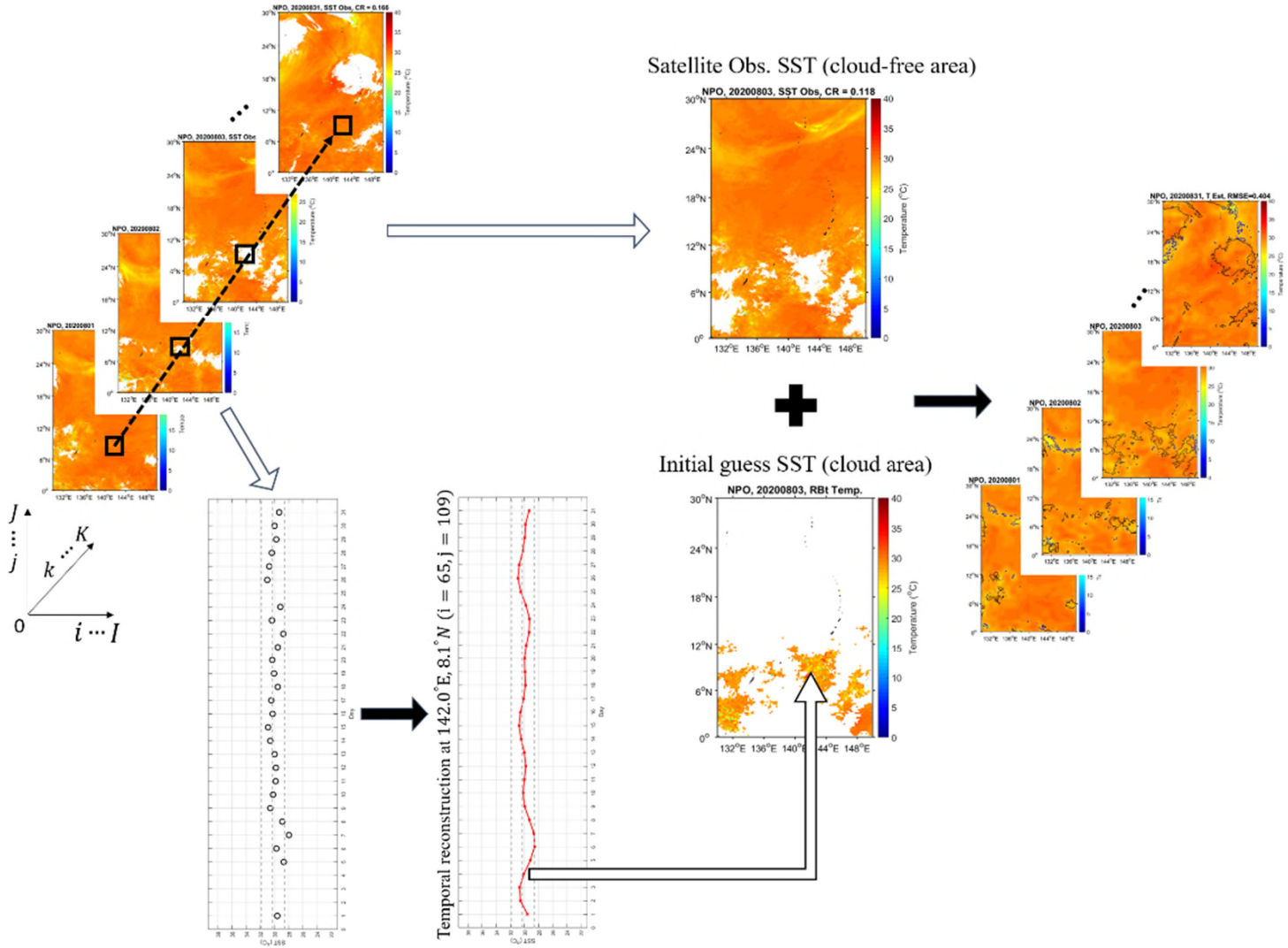
Problem decomposition : TS-RBFNN (Temporal and Spatial learning)

Temporal step:

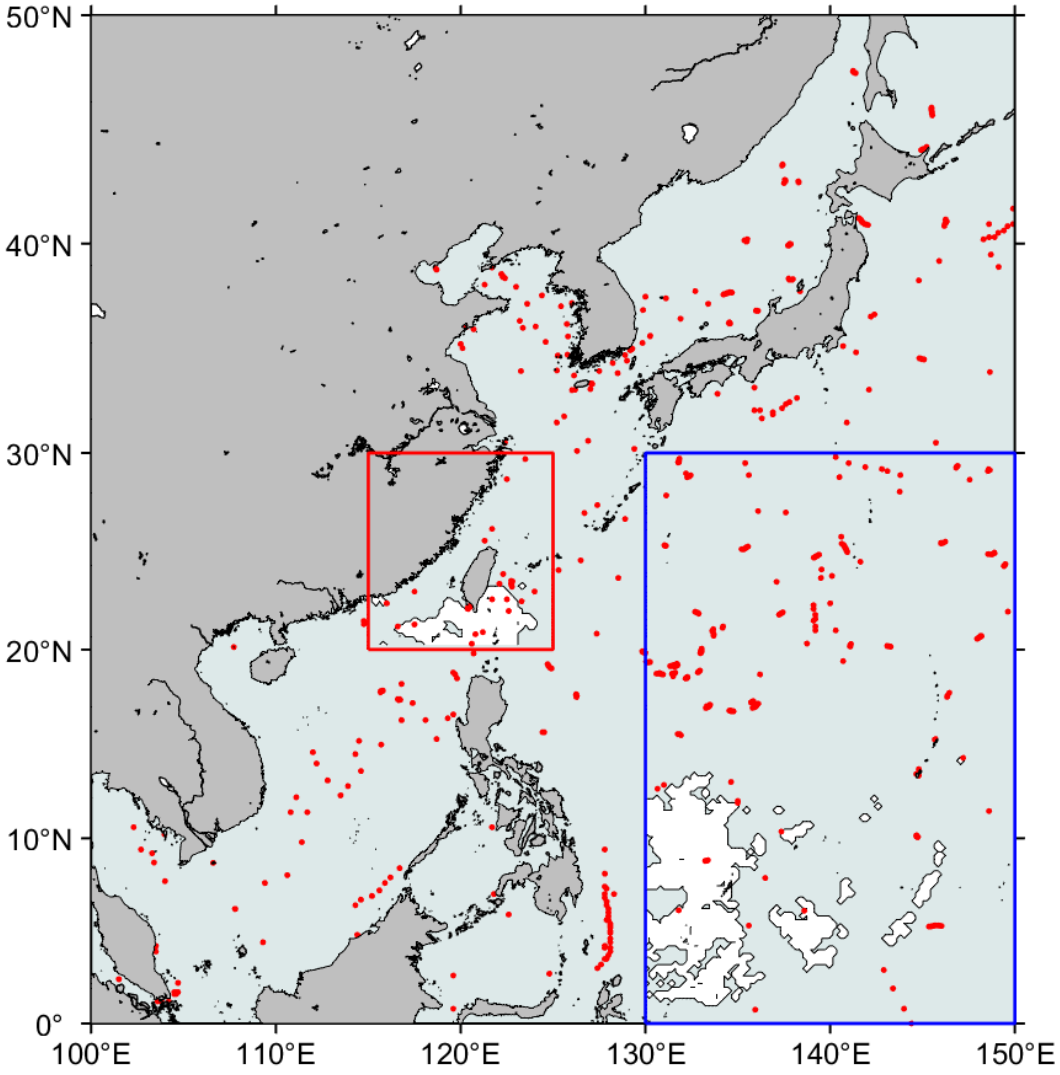


Spatial step:





Study area & time



- Location

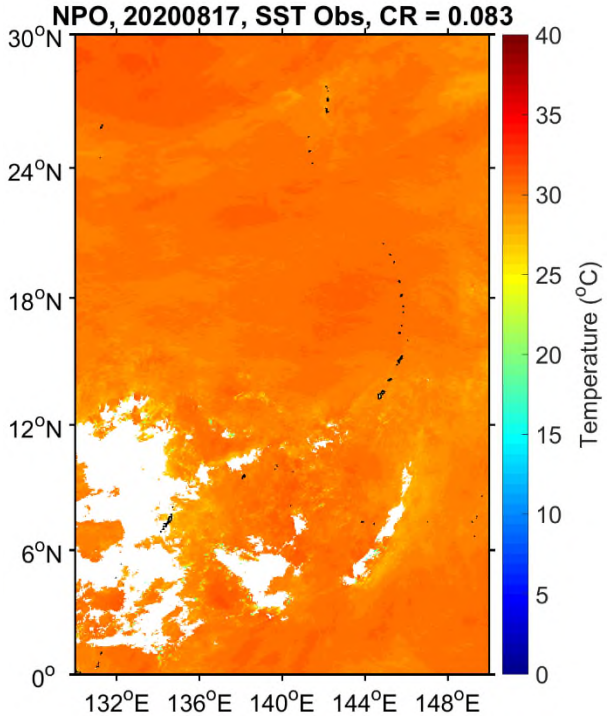
1. Northwestern Pacific ocean (NPO) :

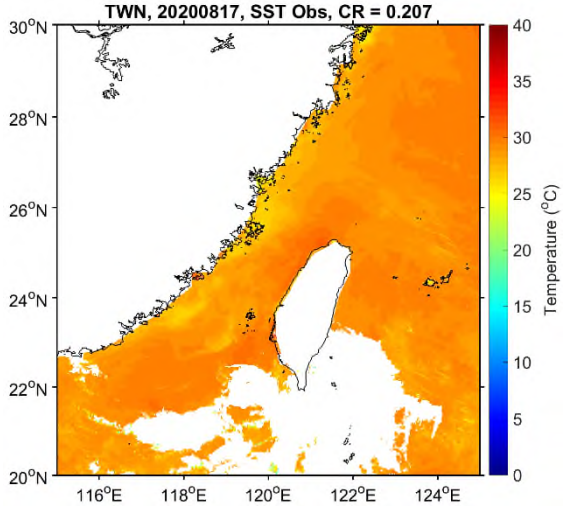
   0˚-30˚N，130˚E-150˚E

2. Areas around Taiwan (TWA) :

   20˚N-30˚N，115˚E-125˚E

- Time : 2019/11, 2020/02, 05, 08





The retrieved daily-mean SST from Himawari-8

Using signals of band-13, 15 & satellite zenith angle from Himawari-8 to construct the MCSST & NLSST formula(Lee et al., 2005; 2020)

- MCSST(Multi-Channel Sea Surface Temperature) =

$$A1 \times \mathbf{T4} + A2 \times (\mathbf{T4}-\mathbf{T5}) + A3 \times (\mathbf{T4}-\mathbf{T5}) \times (\mathrm{Sec}(\boldsymbol{\theta})-1) + A4$$

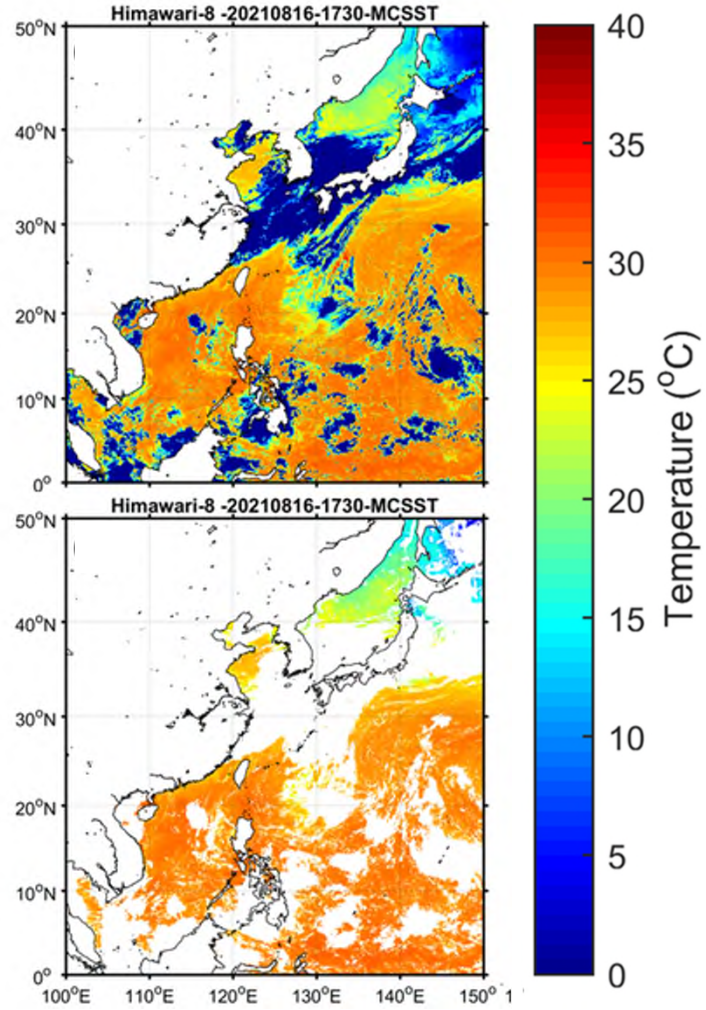
Note: T4 , band-13 (10.4 μm)

T5 , band-15 (12.4 μm)

θ , satellite zenith angle

- Cloud filtering : OTSU method (Otsu, 1979)

Spatial Resolution : 0.02° × 0.02°



Statistical analysis

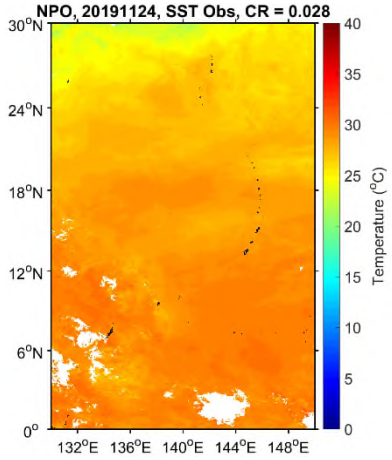
NPO / (TWA) :

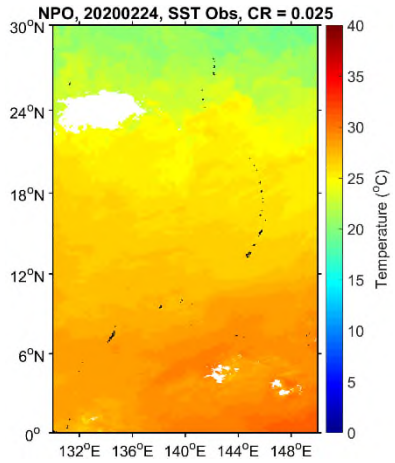
- Cloud cover
- Cloud Ratio (time / space)
- Mean
- Standard deviation

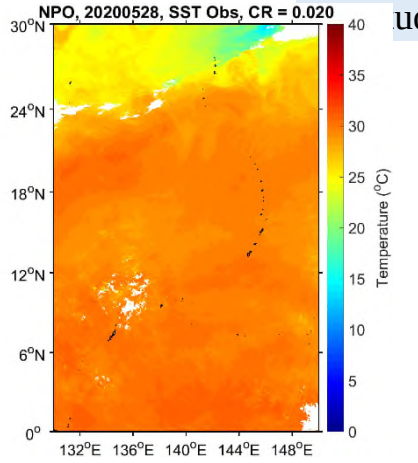
udy area nalysis

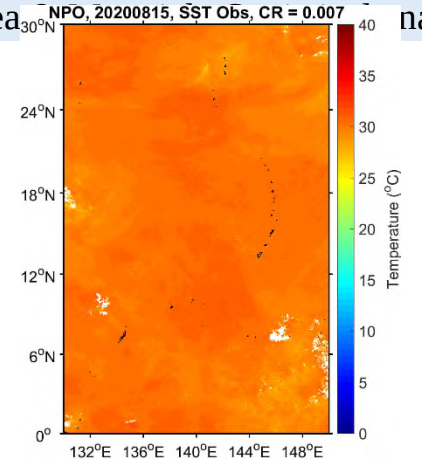
Cloud cover

0%-5%
(mild)

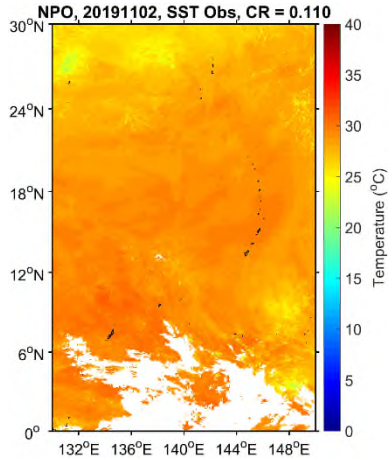


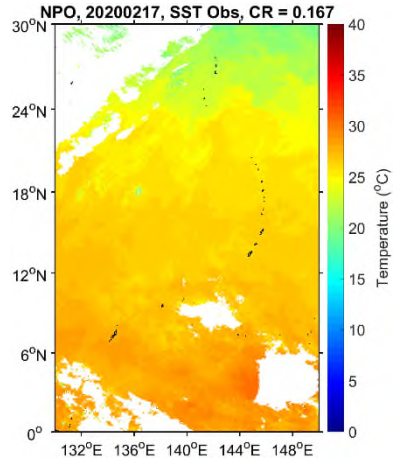


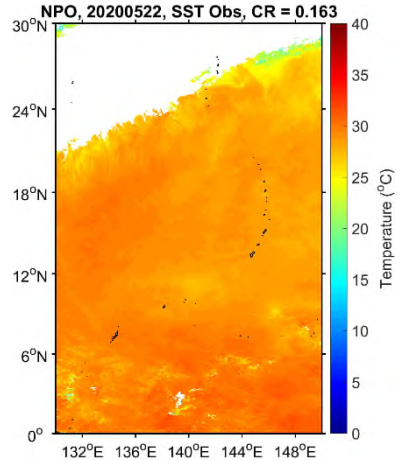


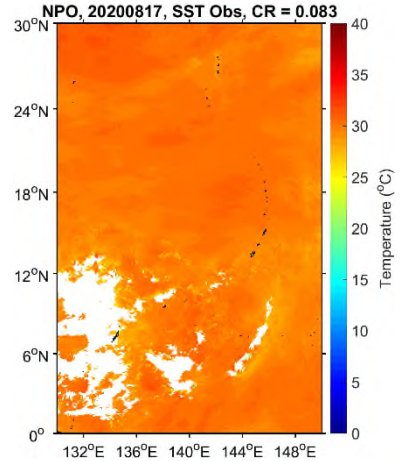


5%-20%
(moderate)

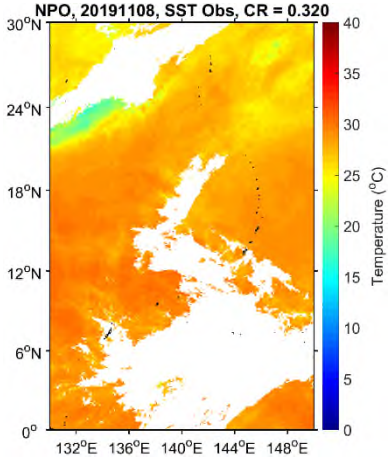


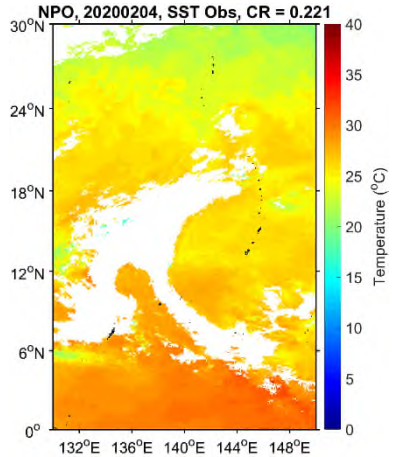


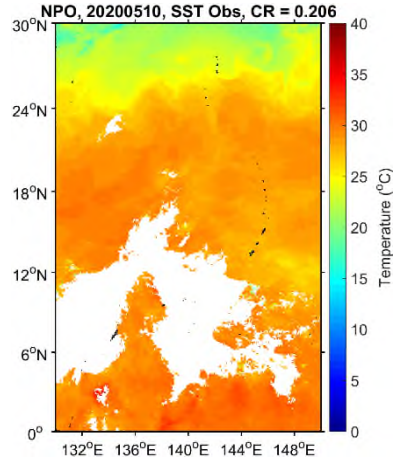


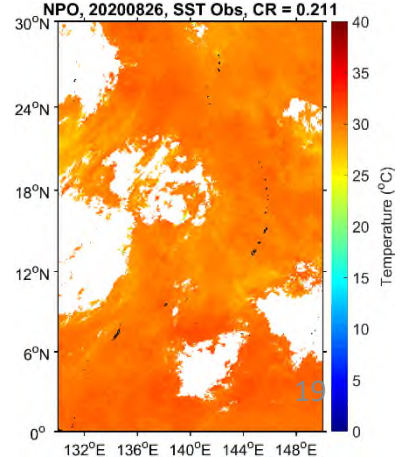


20% ↑
(severe)



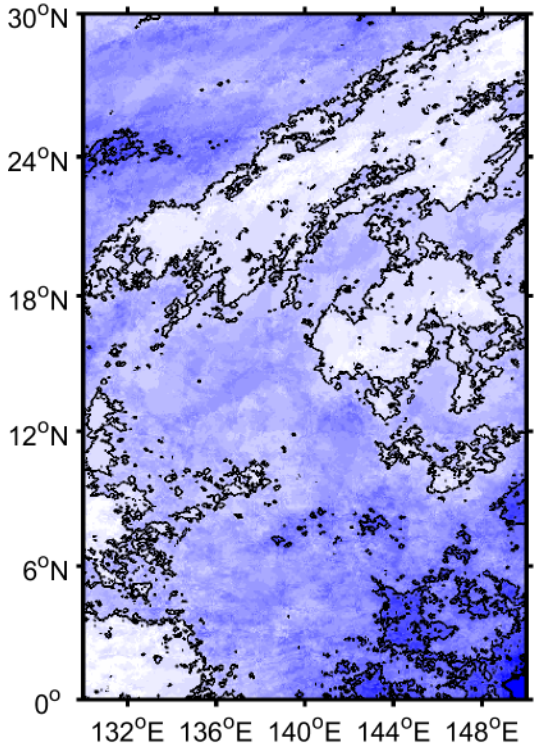






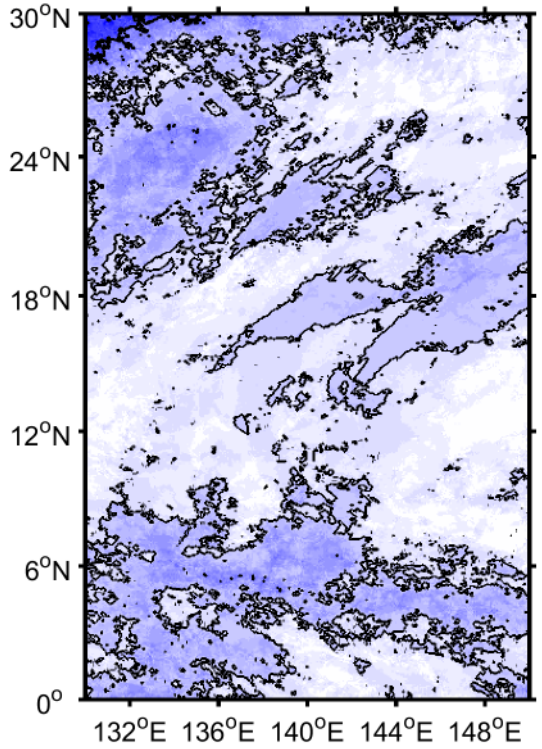
Cloud Ratio in time (NPO)

Nov. 2019



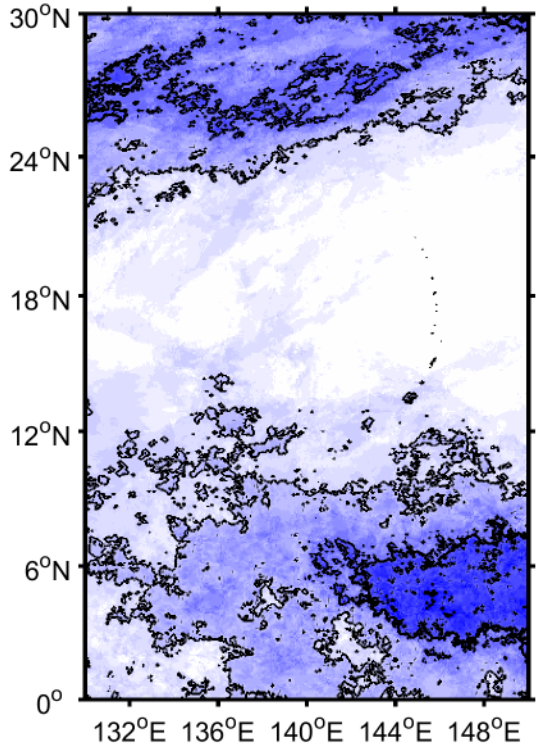
14.6%

Feb. 2020



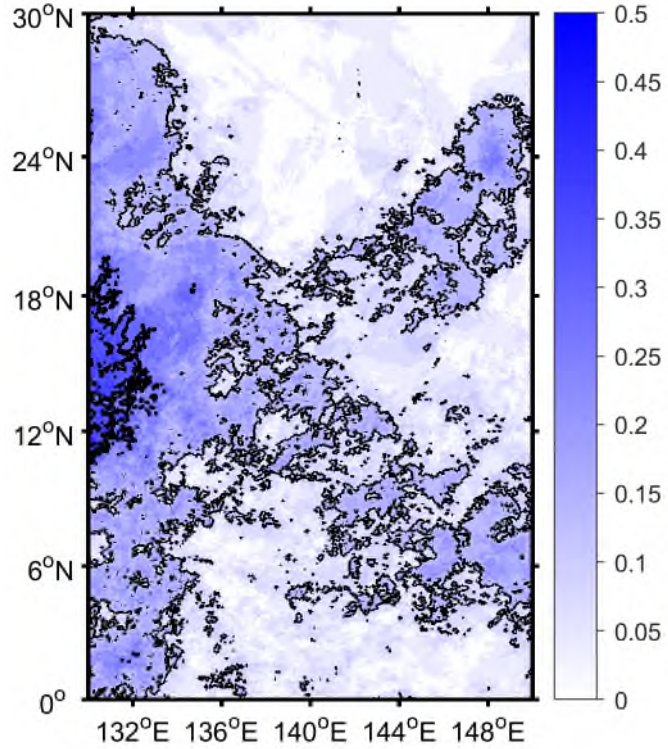
8.5%

May 2020



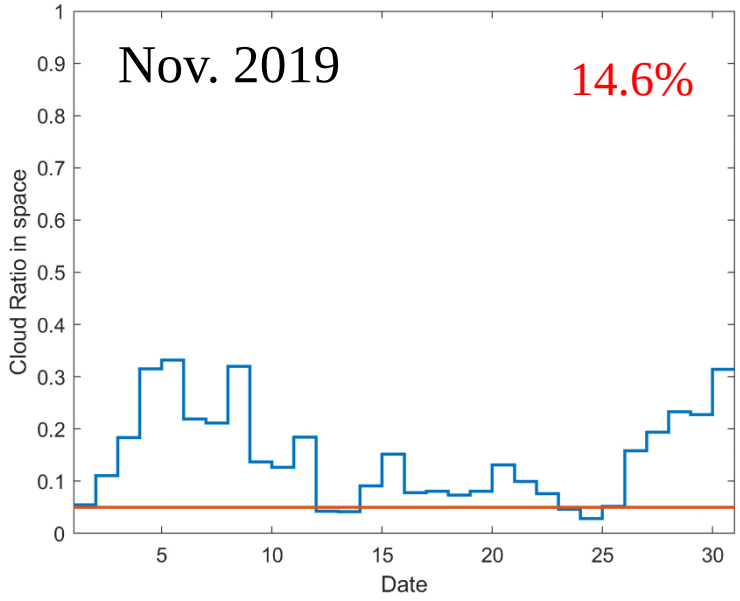
11.7%

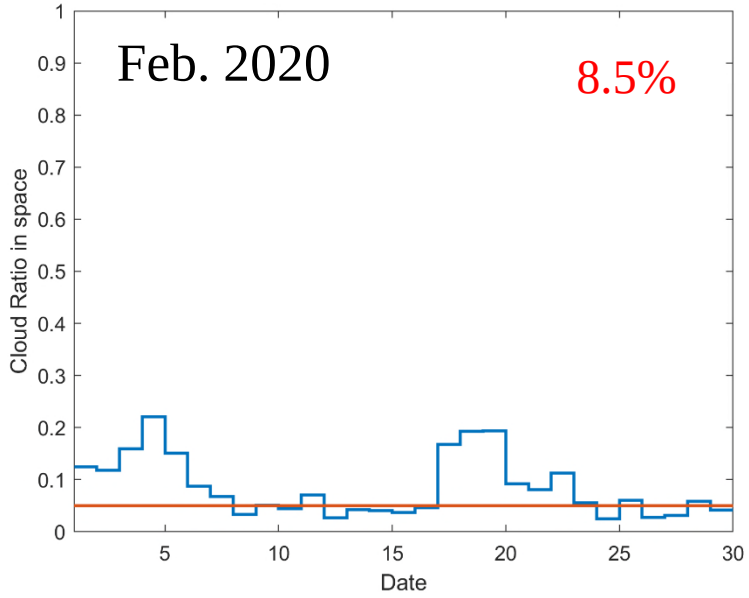
Aug. 2020

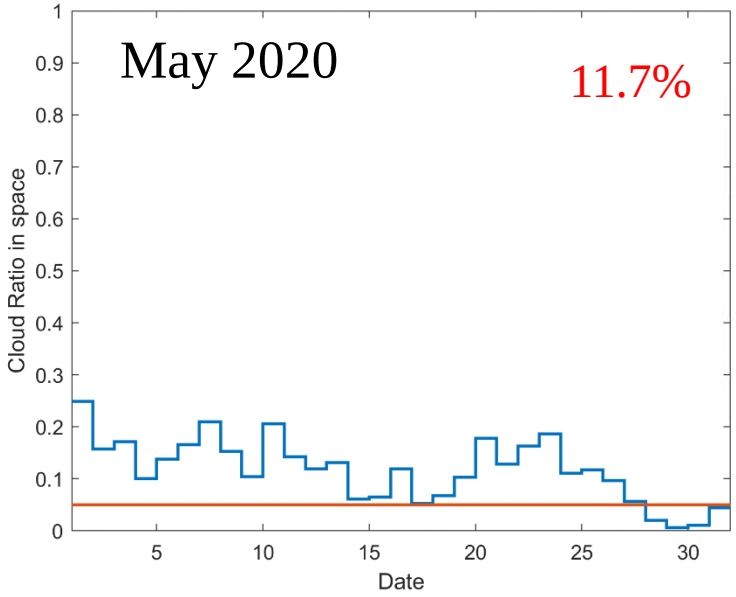


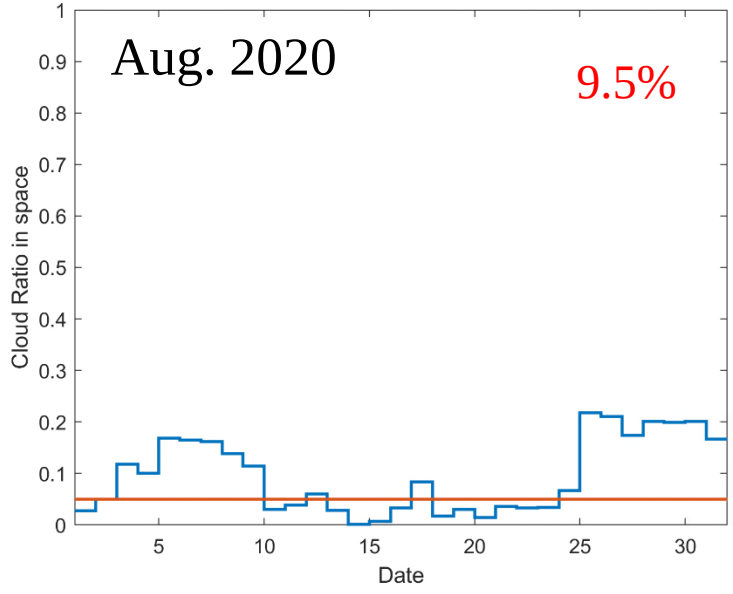
9.5%

Cloud Ratio in space (NPO)



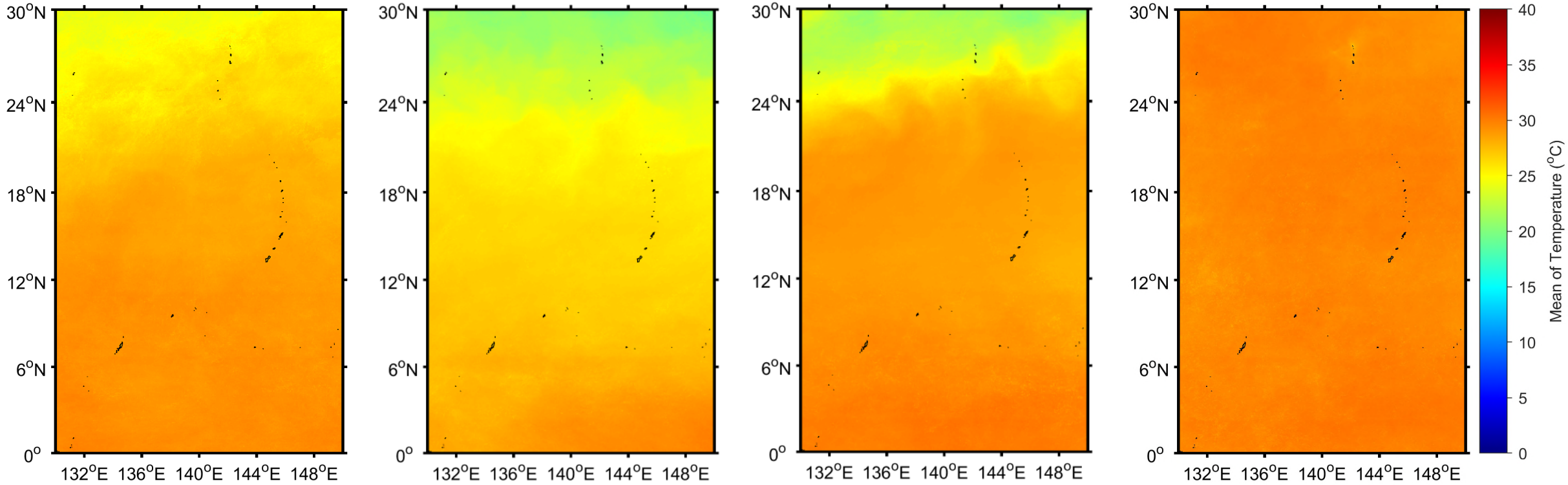






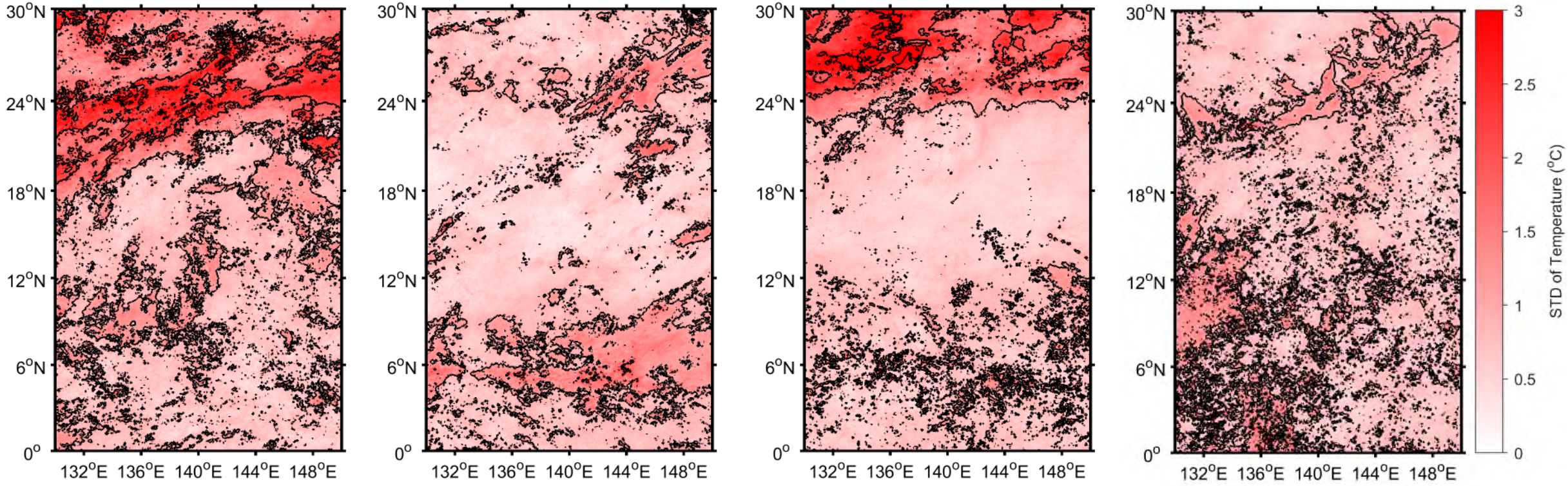
Mean (NPO)

Nov. 2019 Feb. 2020 May 2020 Aug. 2020



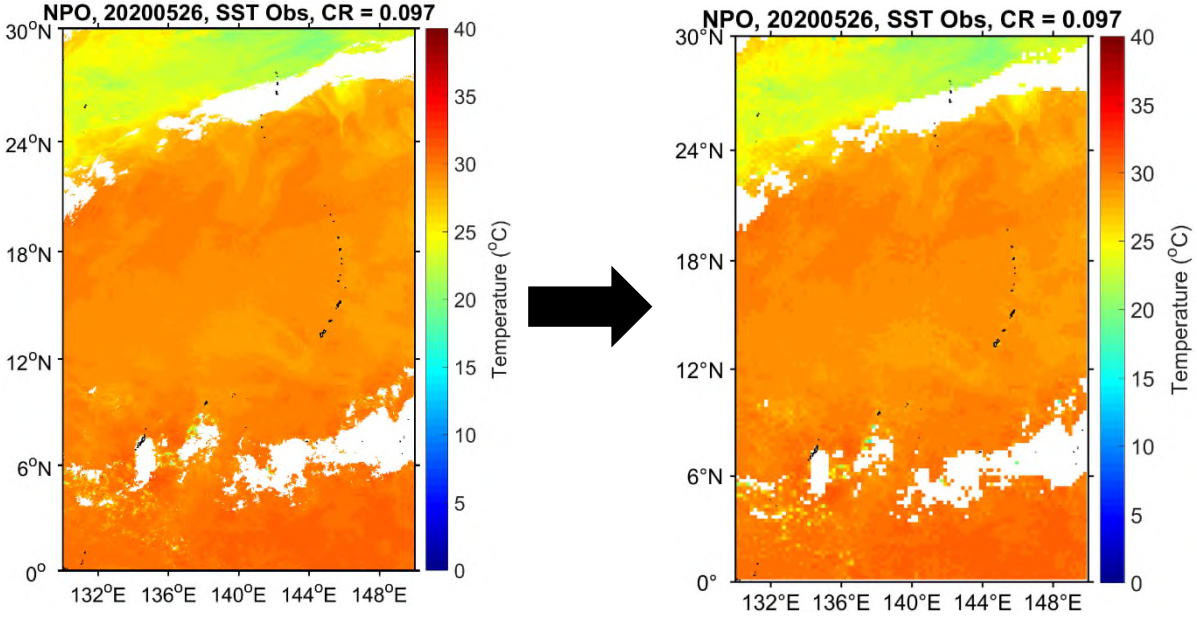
Standard Deviation (NPO)

Nov. 2019 Feb. 2020 May 2020 Aug. 2020



Model setup

1. Resample (low resolution)



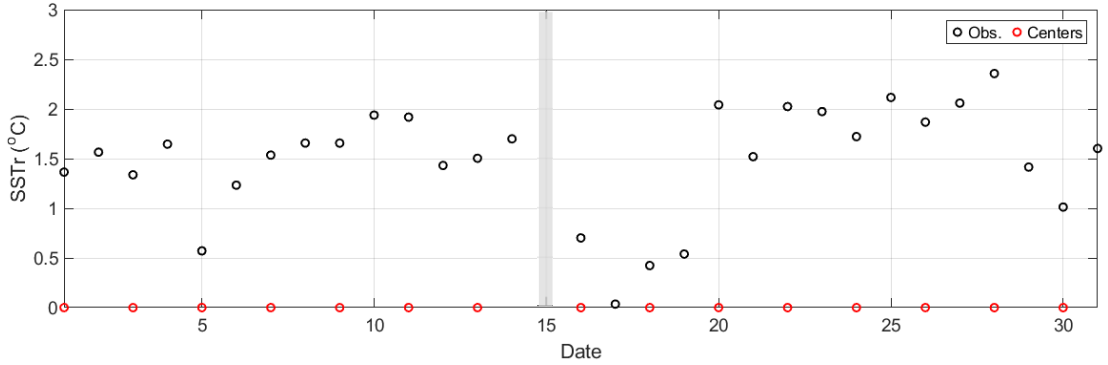
Spatial Resolution :
$0.02° \times 0.02°$

Spatial Resolution :
$0.2° \times 0.2°$

2. Temporal reconstruction

c_n : a half of the days with clear sky

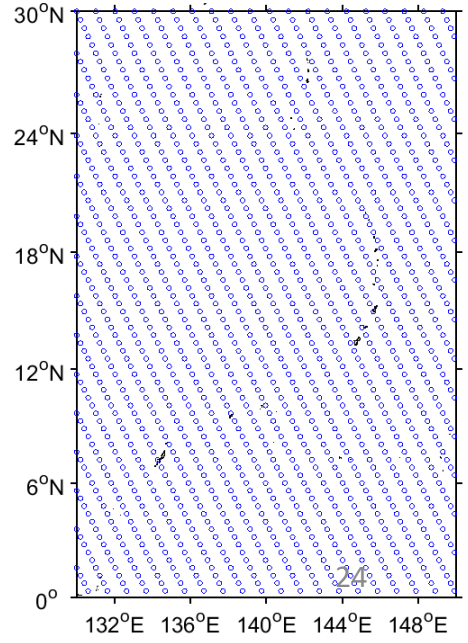
σ_n : the difference of centers * 1.5



3. Spatial reconstruction

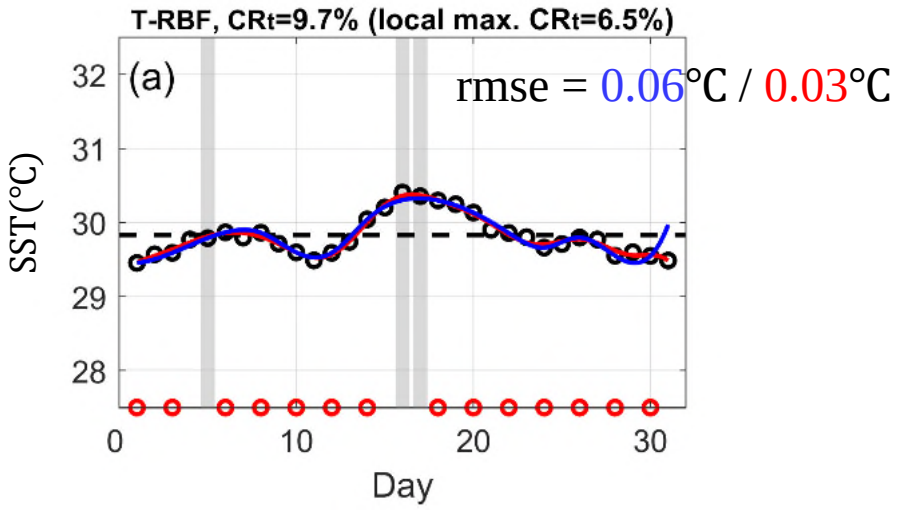
$c_{i,j}$ = 0.1*data grids

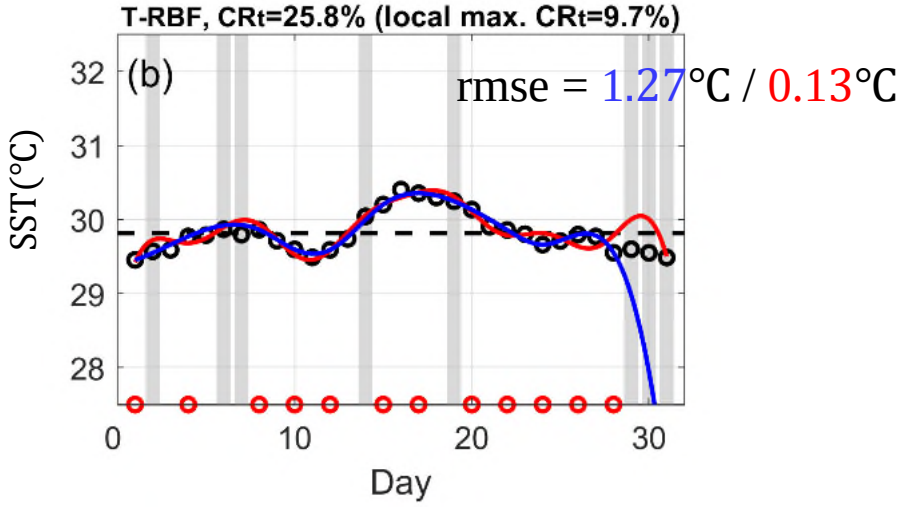
$\sigma_{i,j}$ = 2.0°



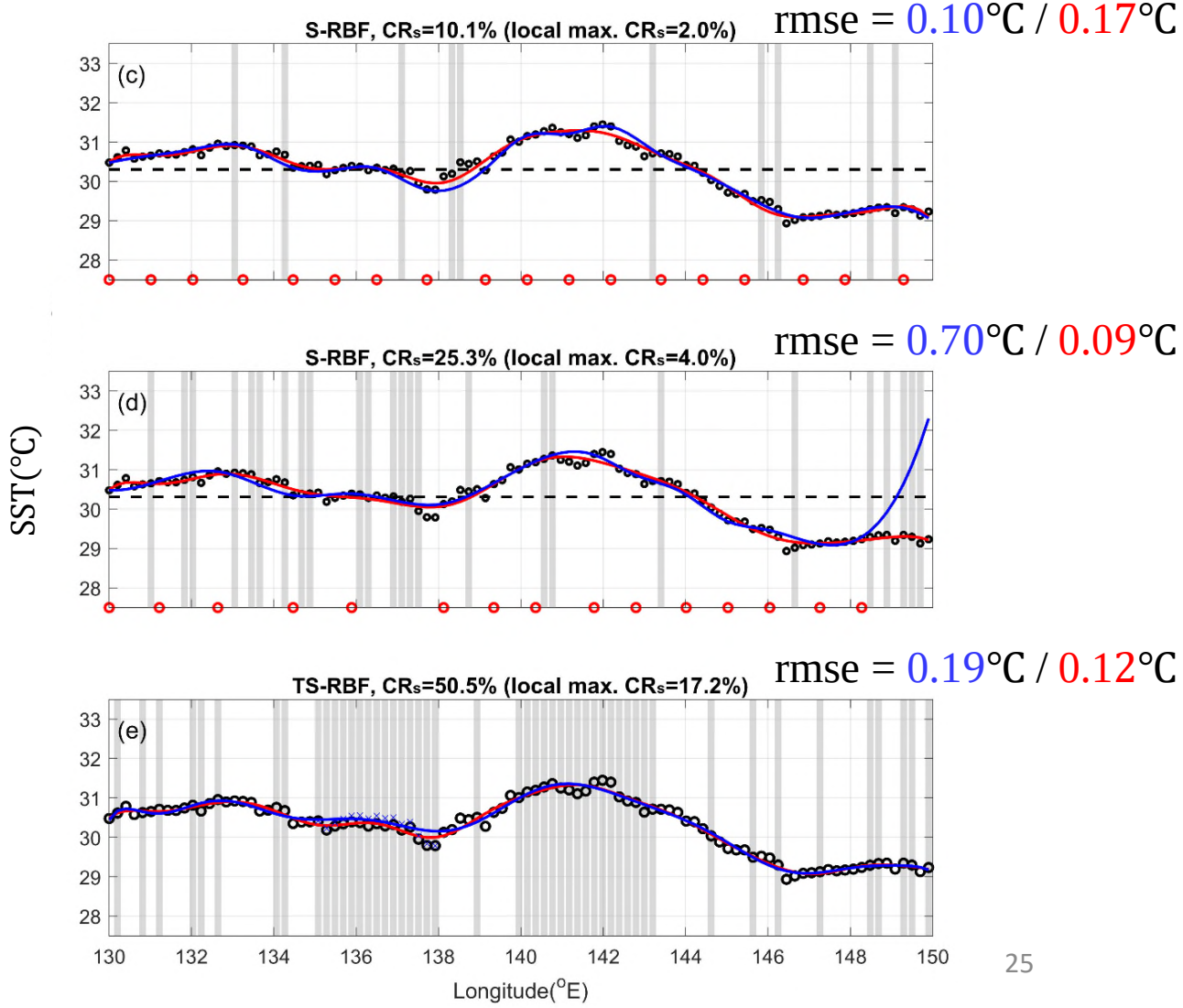
Model validation – artificial clouds

- Temporal reconstruction
  (2020/08, 130.09°E, 17.19°N)

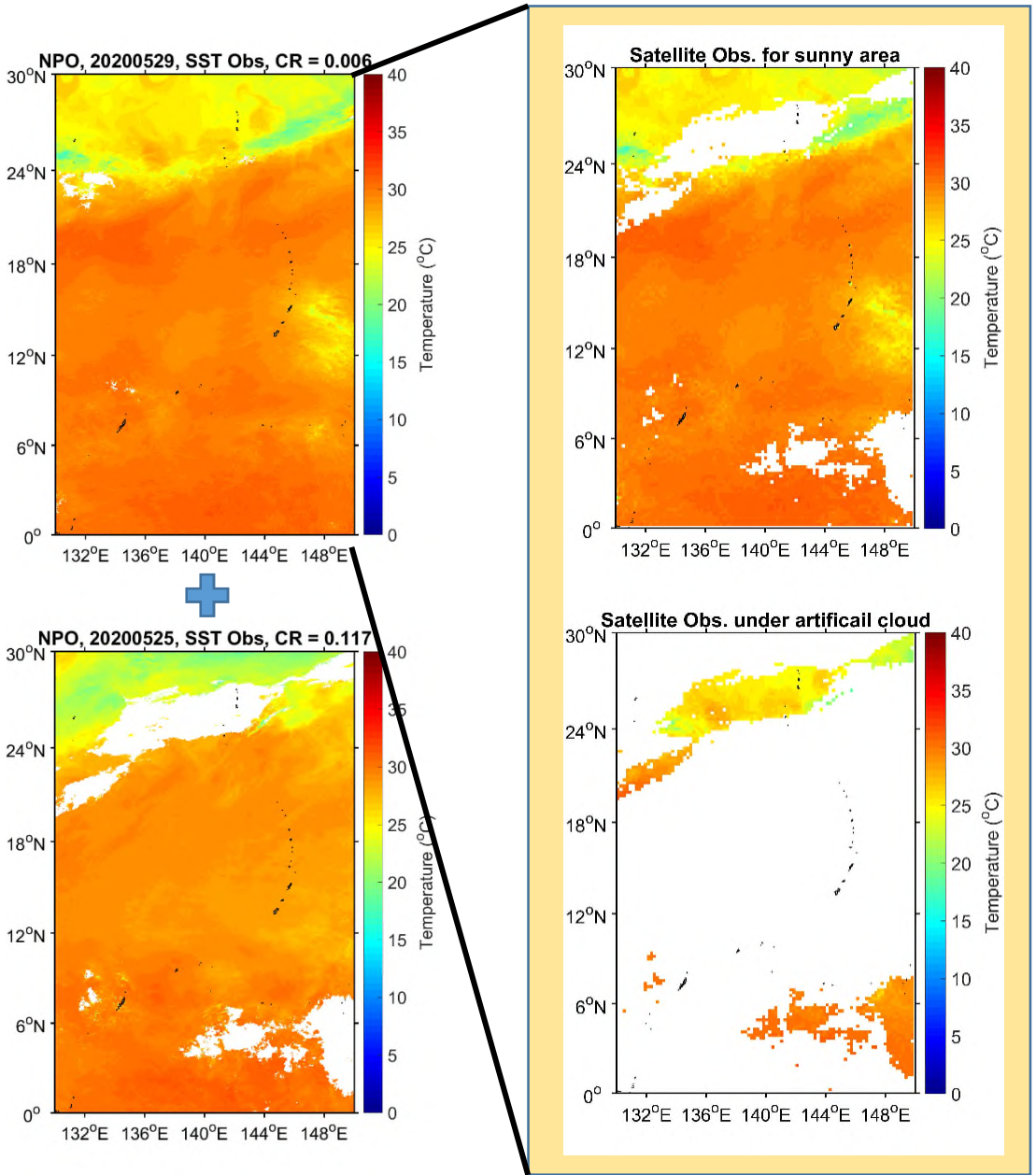




- Spatial reconstruction (2020/08/14, along 25.11°N)



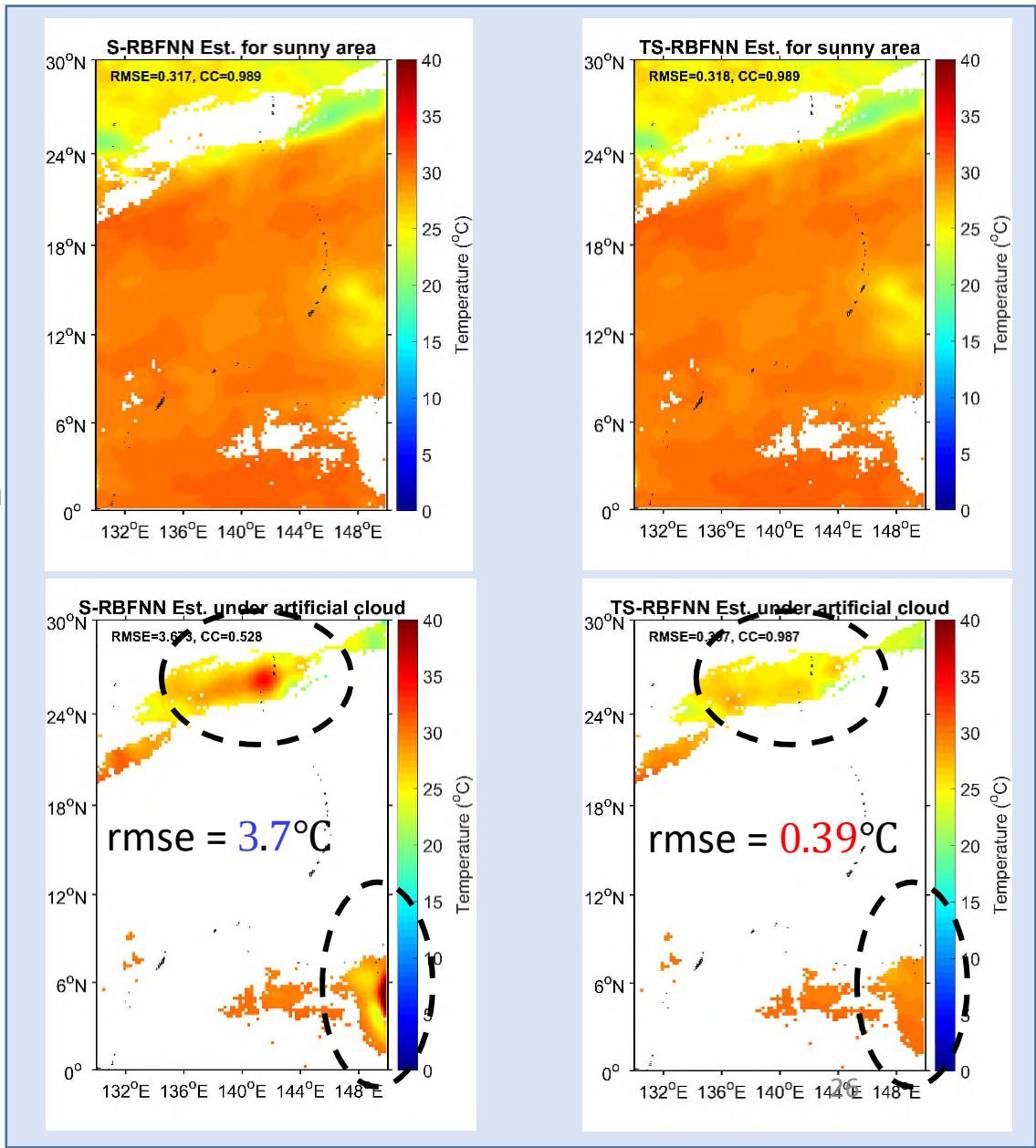
Artificial clouds



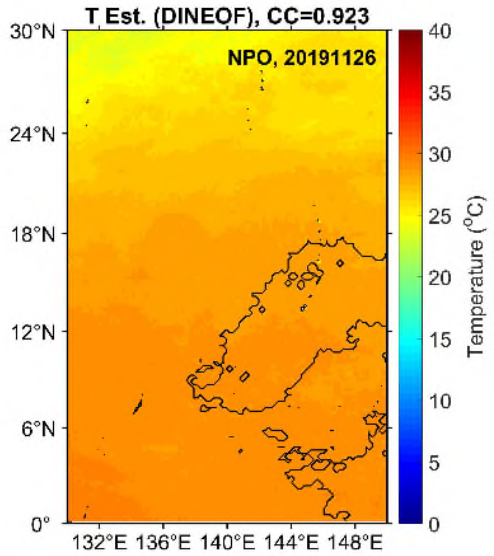
reconstruction

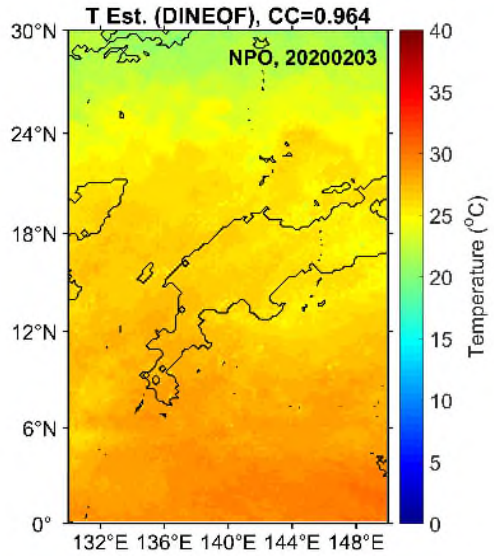
Case1 S-RBFNN

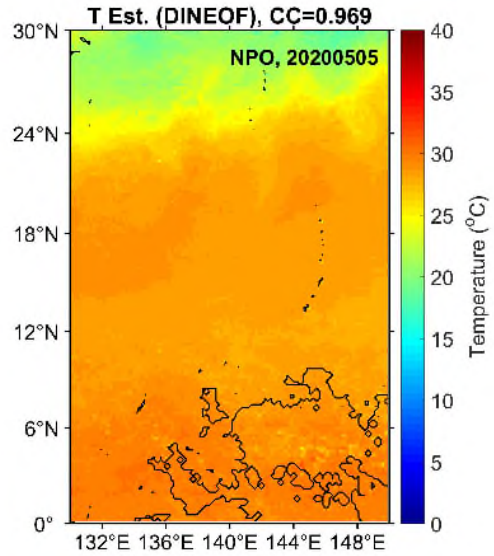
Case2 TS-RBFNN

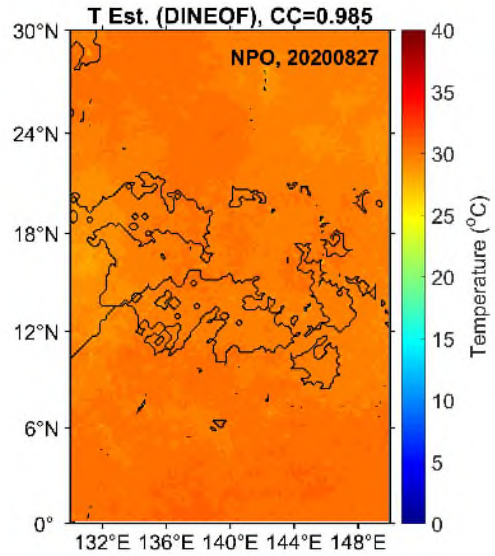


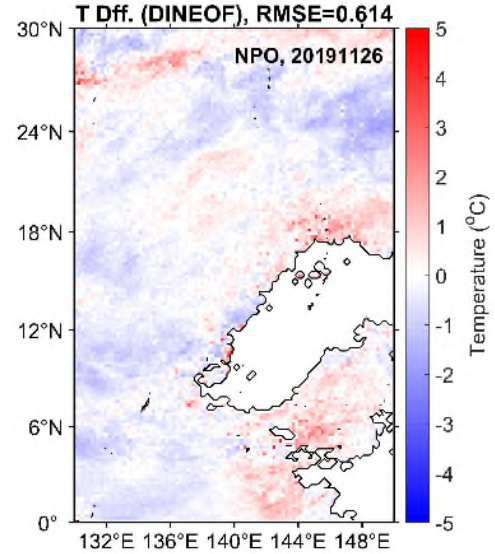
Results – DINEOF (NPO)

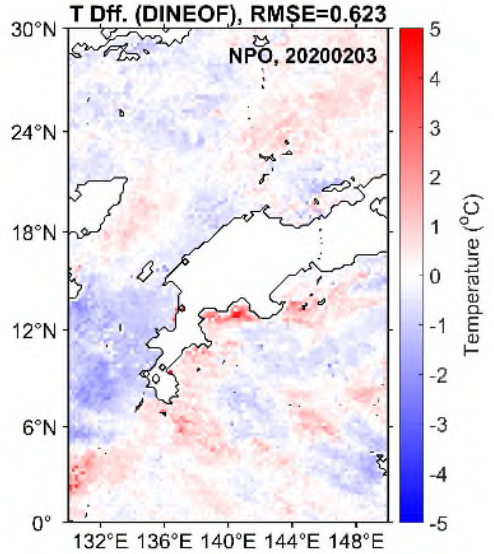


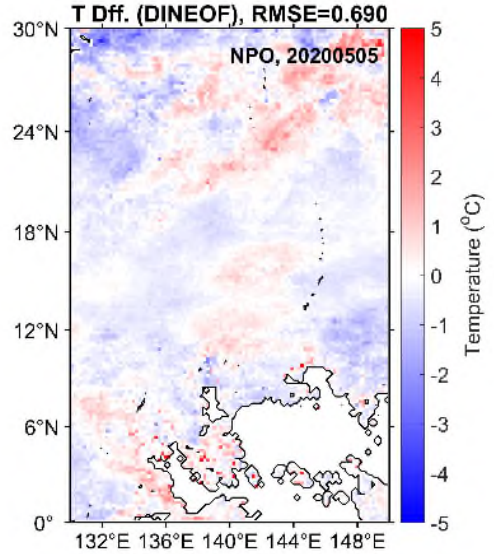


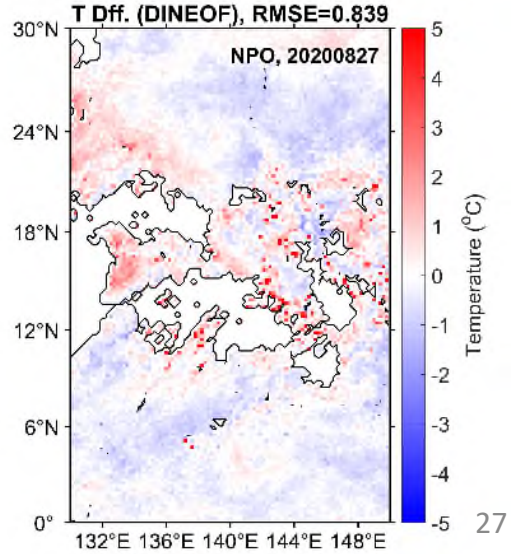




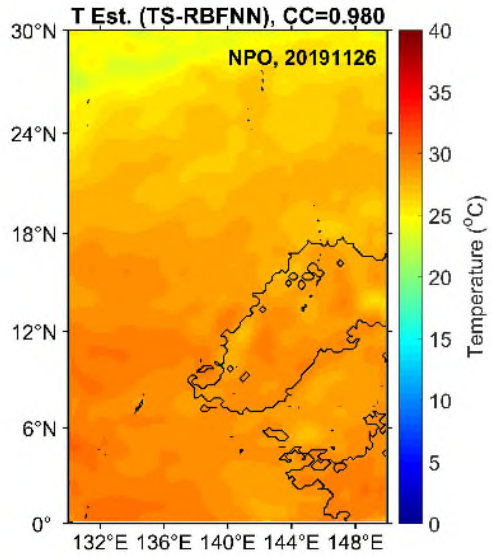


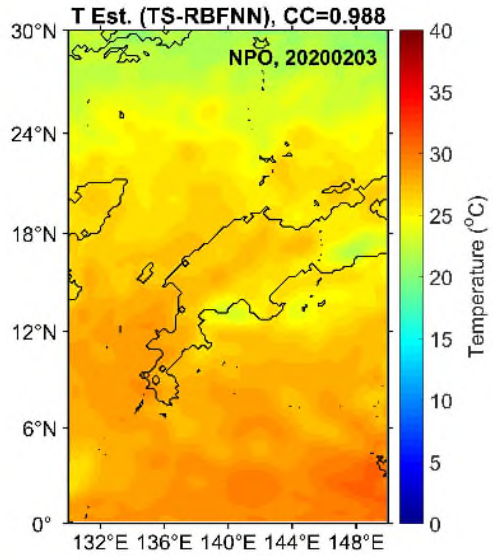


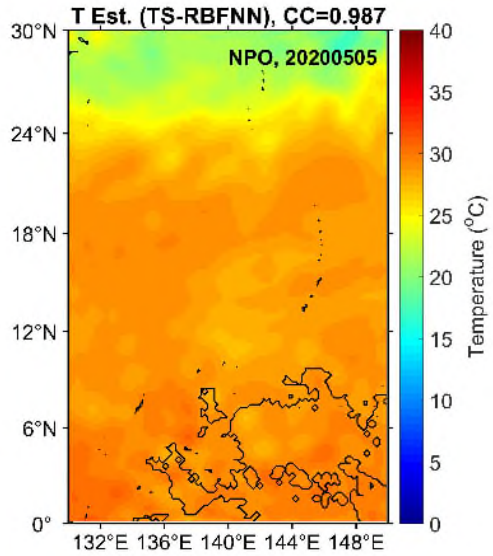


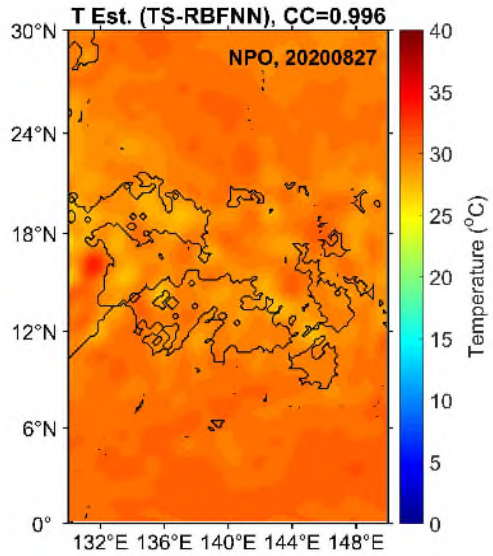


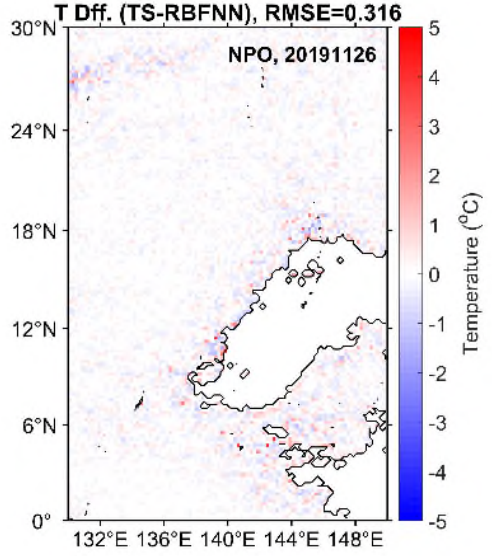
Results – TS-RBFNN (NPO)

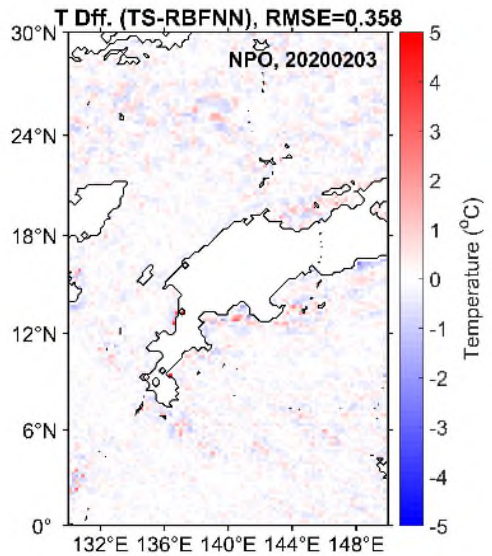


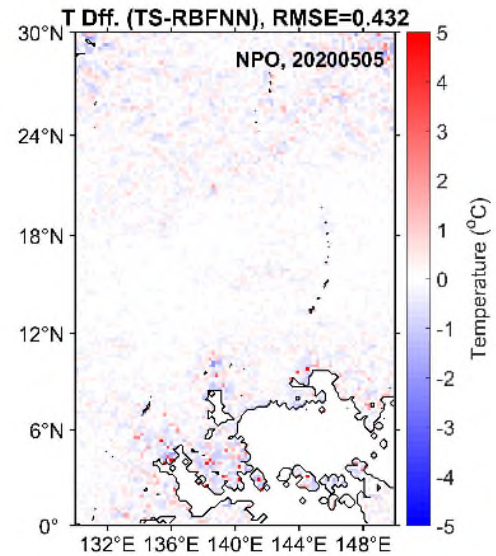


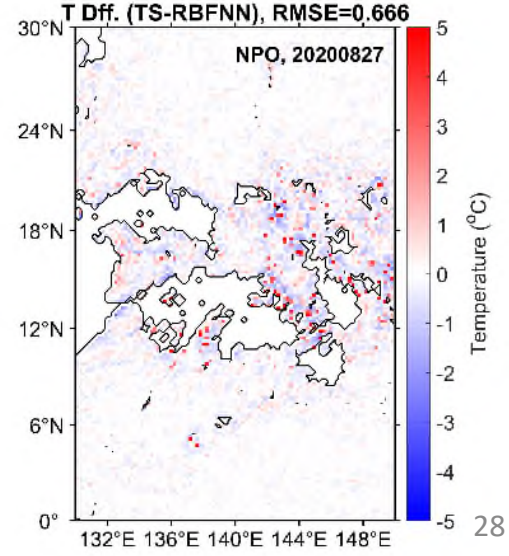






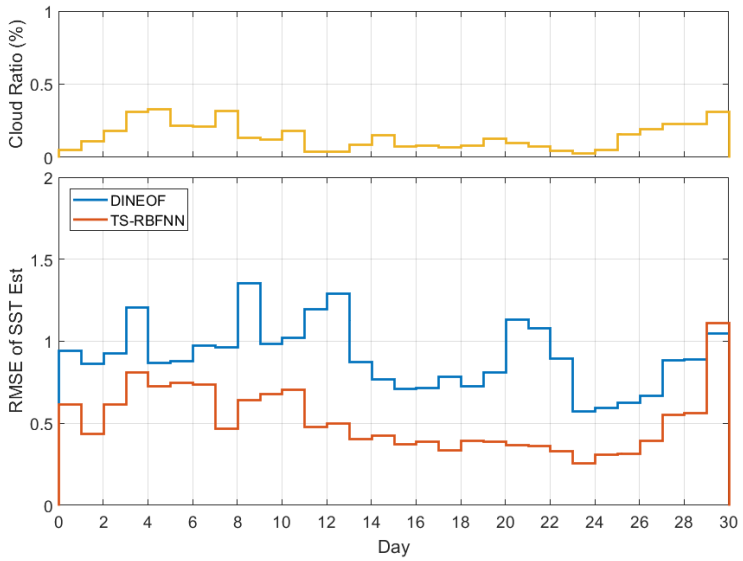




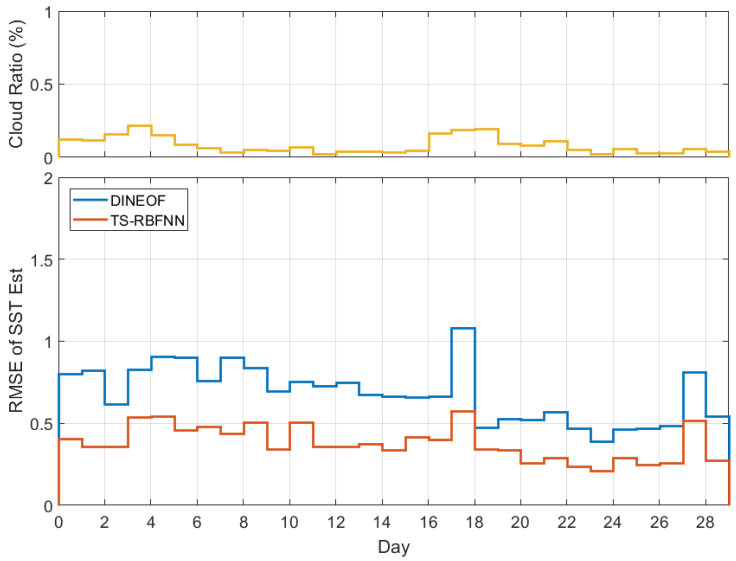


Comparison of performance (DINEOF vs. TS-RBFNN)

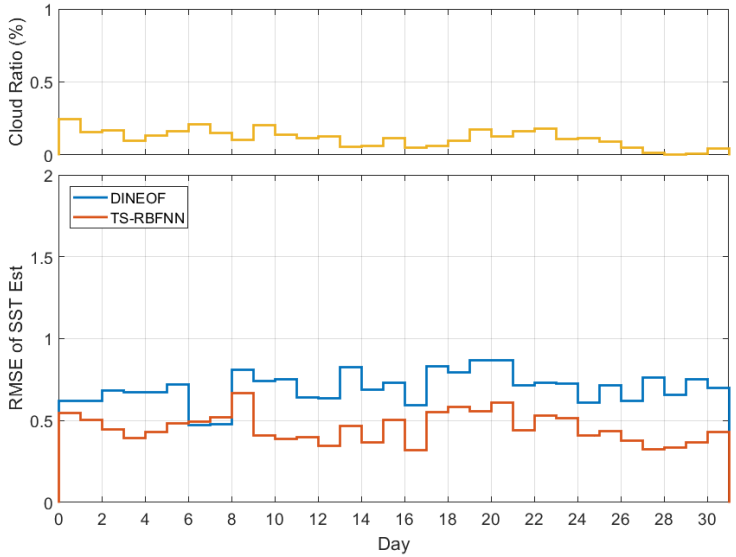
(a) 201911



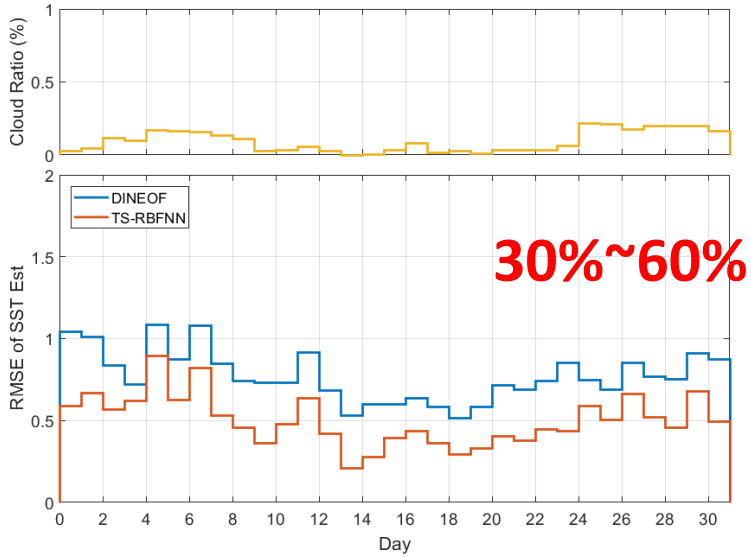
(b) 202002



(c) 202005



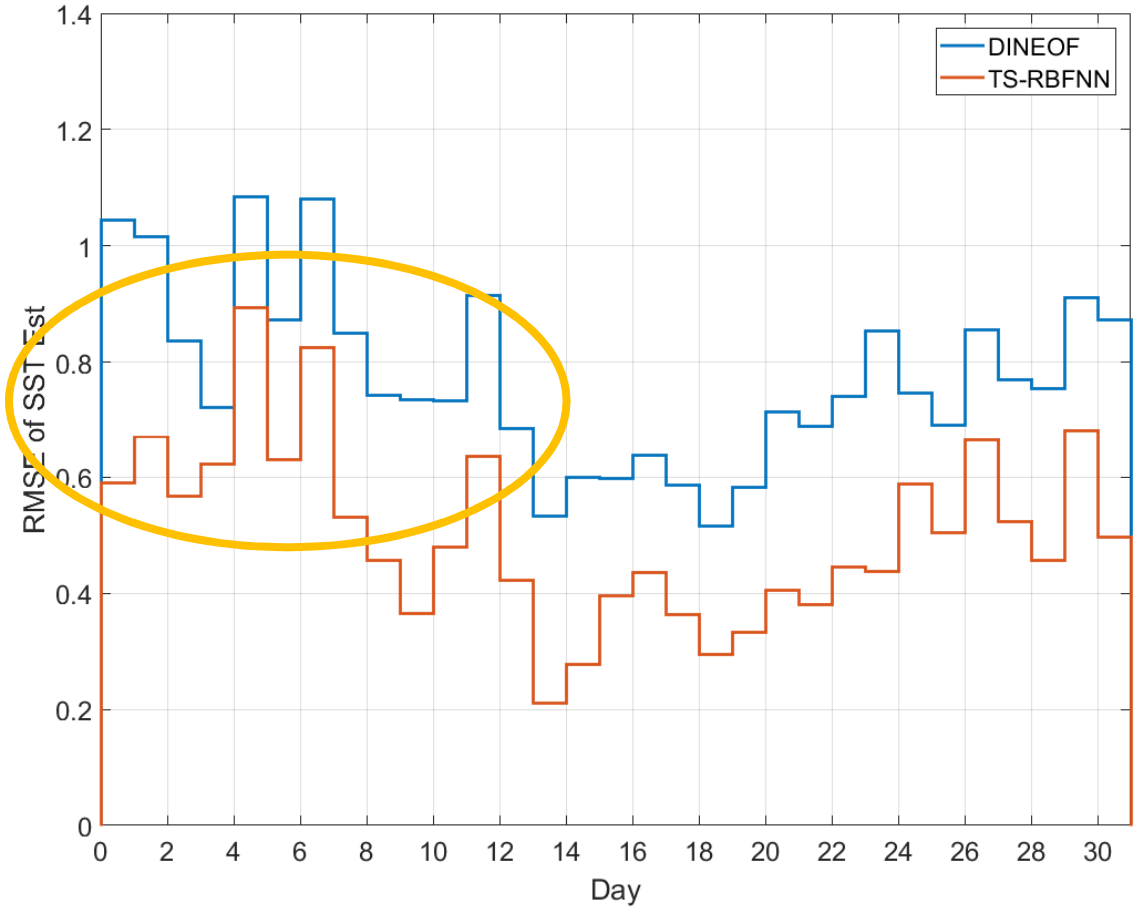
(d) 202008

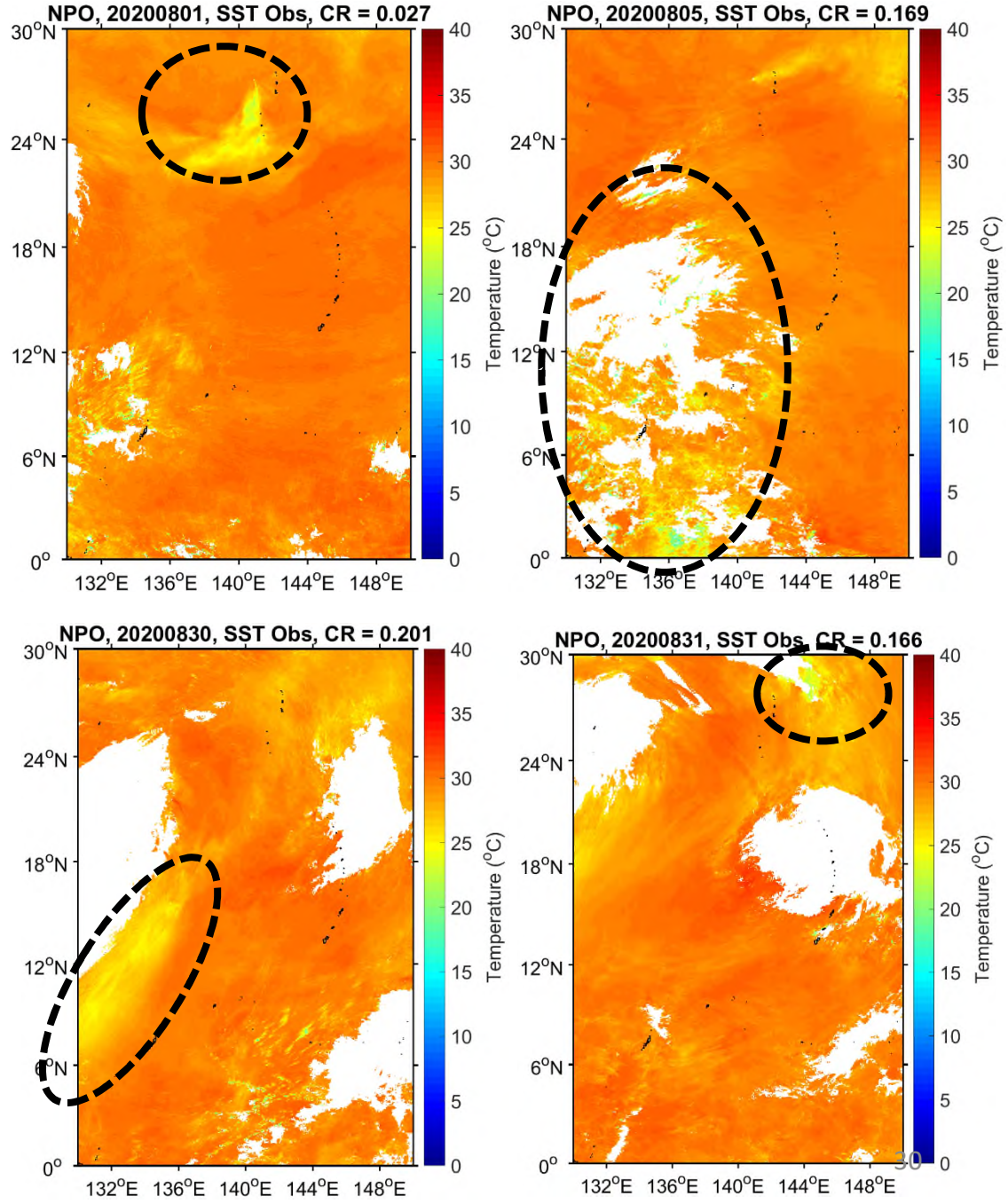


30%~60% improvement

Future work

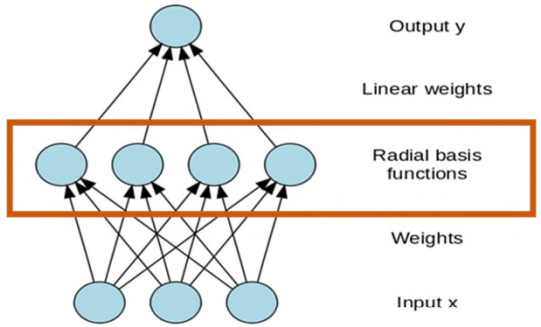
Can this approach be further improved?
in terms of accuracy, efficiency, implication



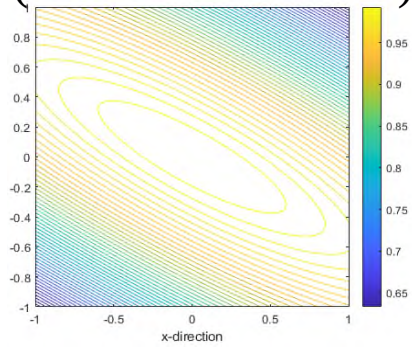


Future work

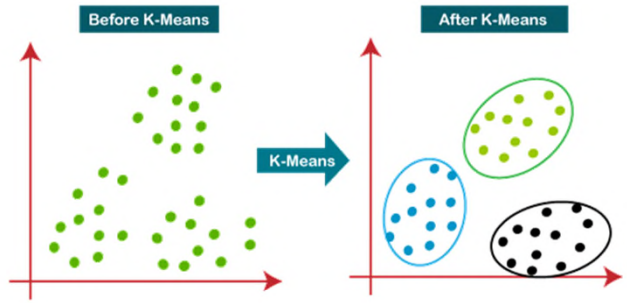
Temporal & Spatial Radial Basis Function Neural Network (TS-RBFNN)



(Bi-directional RBF)

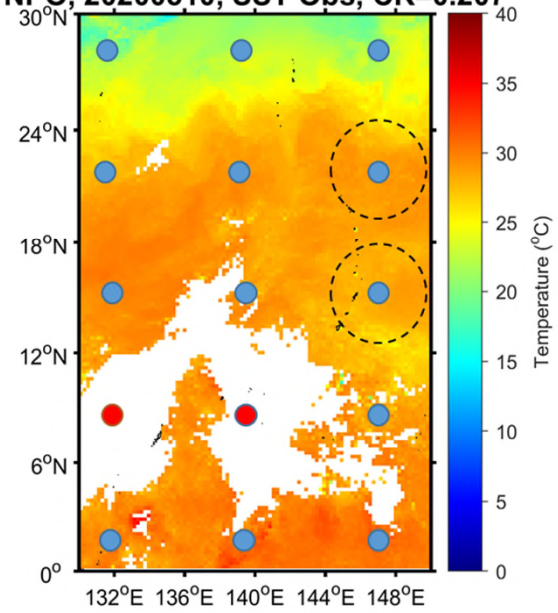


Characteristic of SST variation (based on K-Means)



TS-RBFNN

NPO, 20200510, SST Obs, CR=0.207



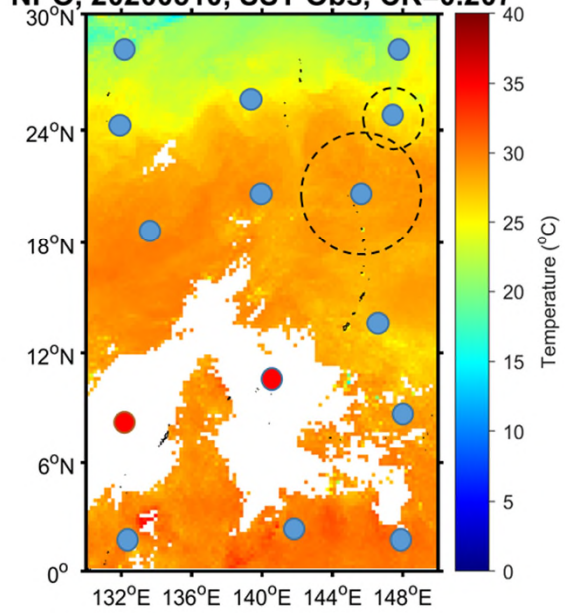
Temporal RBF

Spatial RBF

Improved Accuracy

ATS-RBFNN

NPO, 20200510, SST Obs, CR=0.207



Conclusions

1. According to the physical concept of advection-diffusion process, the classic NN method (Radial Based Function Neural Network) or Physically-Informed Machine Learning(PIML) method have been selected and developed to reconstruct the SST in the Northwestern Pacific Ocean.
2. Variable decomposition is applied to reduce computational cost. Temporal learning first estimates the reasonable SST variation, followed by spatial learning, which smooths the cloud boundary region.
3. The performance on TS-RBFNN is greatly better than DINEOF (30%-60%) due to the effect of local pattern learning.
4. In the future, the outlier will be filtered out (improve data quality) for reducing the impact of abnormal values during model training.
5. Adaptive version of TS-RBFNN will be further developed for improving the efficiency and accuracy.

Reference

[1] Govekar, P., Griffin, C., Embury, O., Mittaz, J., Beggs, H. M., & Merchant, C. J. (2024). Himawari-8 Sea Surface Temperature Products from the Australian Bureau of Meteorology. *Remote Sensing*, *16*(18), 3381. https://doi.org/10.3390/rs16183381

[2] Beckers, J. M., &Rixen, M. (2003). EOF Calculations and Data Filling from Incomplete Oceanographic Datasets. *Journal of Atmospheric and Oceanic Technology*, 20(12), 1839–1856.

[3] Haghbin, M., Sharafati, A., Motta, D. *et al*. Applications of soft computing models for predicting sea surface temperature: a comprehensive review and assessment. *Prog Earth Planet Sci* **8**, 4 (2021).

[4] Barth, A., Alvera-Azcárate, A., Licer, M., &Beckers, J.-M. (2020). DINCAE 1.0: a convolutional neural network with error estimates to reconstruct sea surface temperature satellite observations. *Geoscientific Model Development*, 13(3), 1609–1622.

[5] Young, C.-C.; Cheng, Y.-C.; Lee, M.-A.; Wu, J.-H. (2024). Accurate reconstruction of satellite-derived SST under cloud and cloud-free areas using a physically-informed machine learning approach. Remote Sens. Environ. 313, 114339.

[6] Taylor, G.I. (1922), Diffusion by Continuous Movements. Proceedings of the London Mathematical Society, s2-20: 196-212.

- https://ars.els-cdn.com/content/image/3-s2.0-B9780323855976000033-f13-03-9780323855976.jpg
- https://y0.ifengimg.com/7db61b22778943d0/2013/0807/rdn_5201ab5730598.jpg
- https://w3.oc.ntu.edu.tw/chap4/chap4s5.htm
- https://encrypted-tbn2.gstatic.com/images?q=tbn:ANd9GcQR_Ep8x7kOWlnqNABUzqEnDEHC0wsJarPXhnhjcI1VRQBttZpK
- https://skepticalscience.com//pics/Argo_530w.jpg
- https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSQWmEYyTq5GJoJbbQOpY1iDwIGwwnLrh2niTaJgDzox_7AJkNr
- https://apac.dtn.com/news/himawari-9-satellite-replacing-himawari-8-today/
- https://www.weatherzone.com.au/news/himawari9-satellite-replacing-himawari8-today/981133


Thanks for your listening
NTOU

Outline

1. Introduction
   - Importance
   - Motivation & Challenge
   - Previous study & limitation
   - Deeping learning : DINCAE
2. Methodology
3. Study area & Material
   - Statistical analysis
3. Results
   - Model setup
   - Model validation
   - Model application
4. Future work
5. Conclusion

Introduction

Methodology

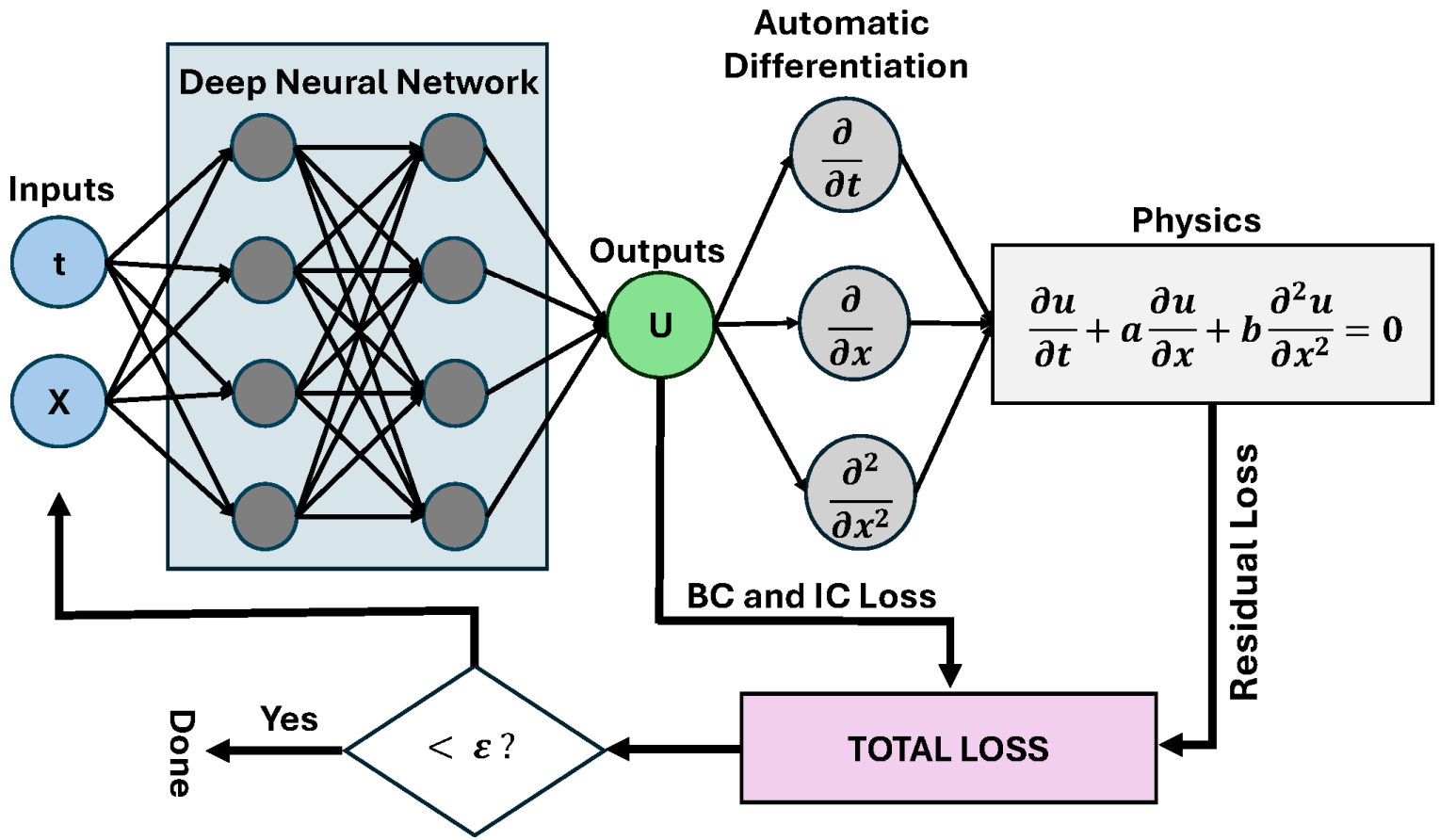
Study area & Material

Results

Future work

Conclusion

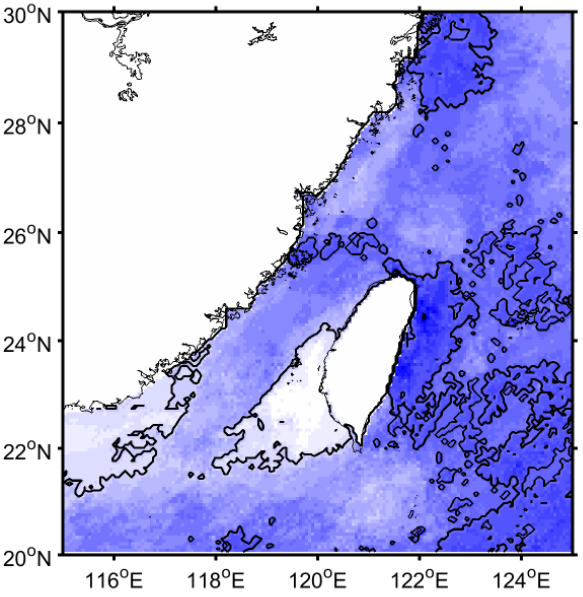
Physic informed neural network



Trahan, C., Loveland, M., & Dent, S. (2024). Quantum Physics-Informed Neural Networks. *Entropy*, *26*(8), 649. https://doi.org/10.3390/e26080649

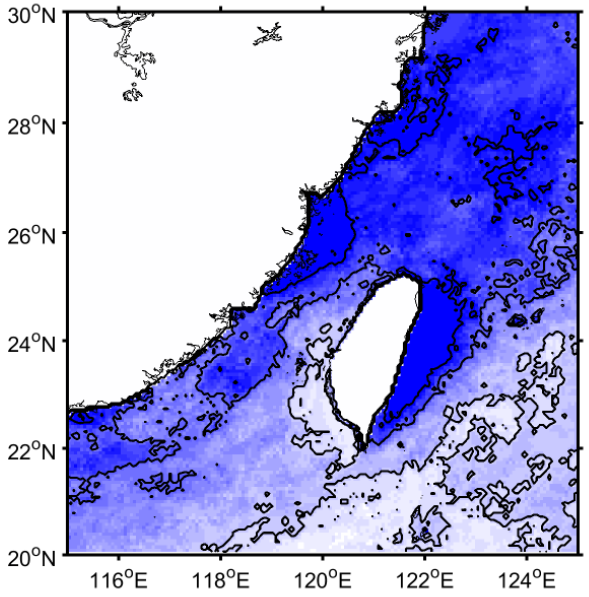
Cloud Ratio in time (TWA)

Nov. 2019



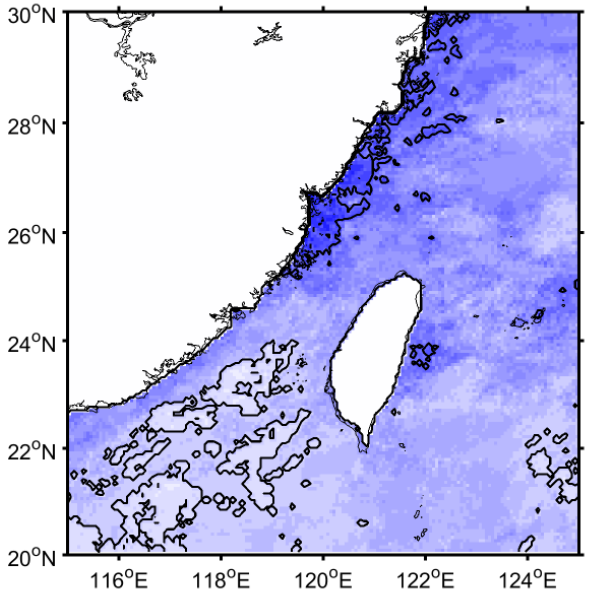
23.6%

Feb. 2020



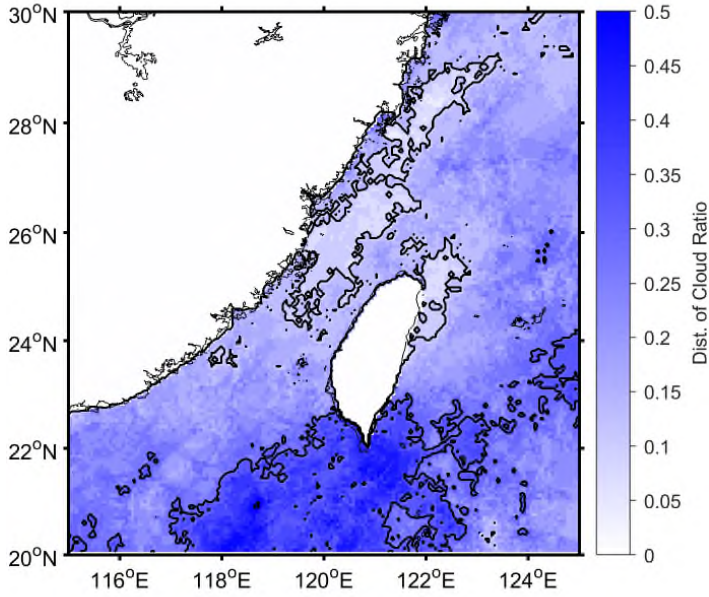
28.1%

May 2020



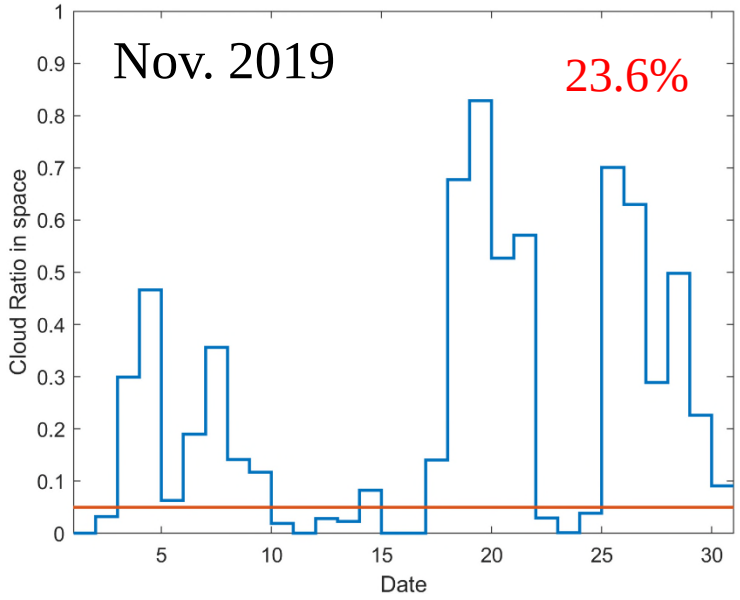
17.1%

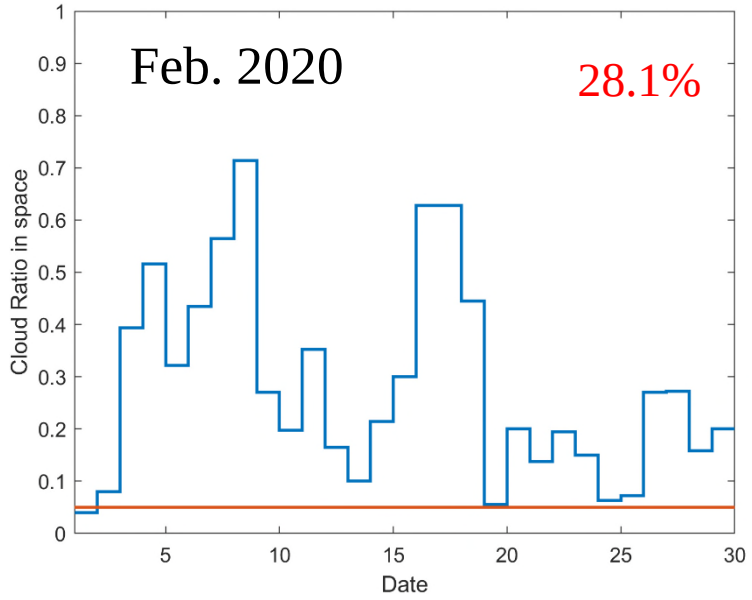
Aug. 2020

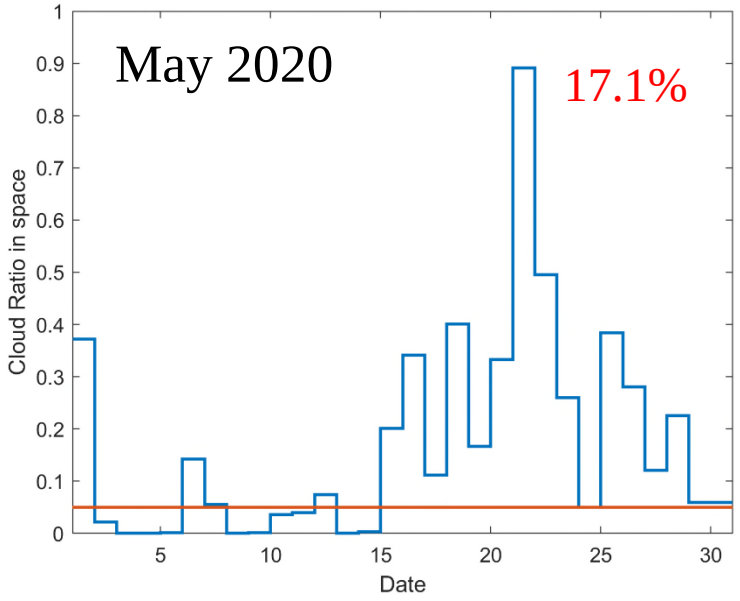


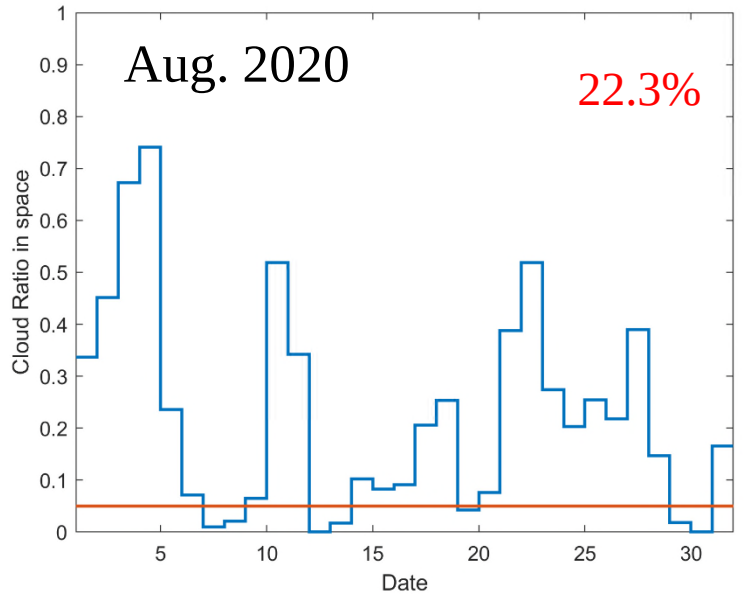
22.3%

Cloud Ratio in space (TWN)



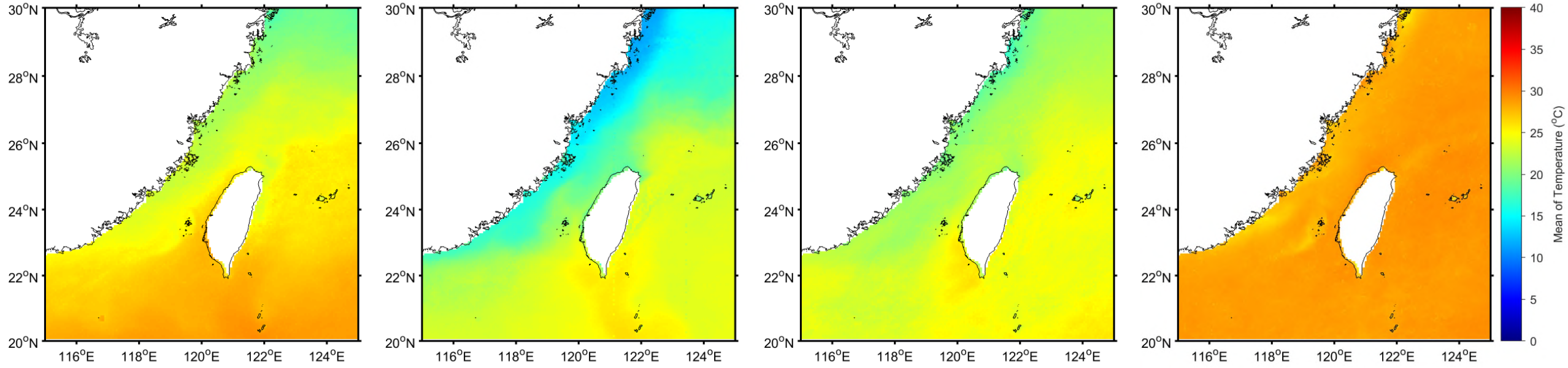






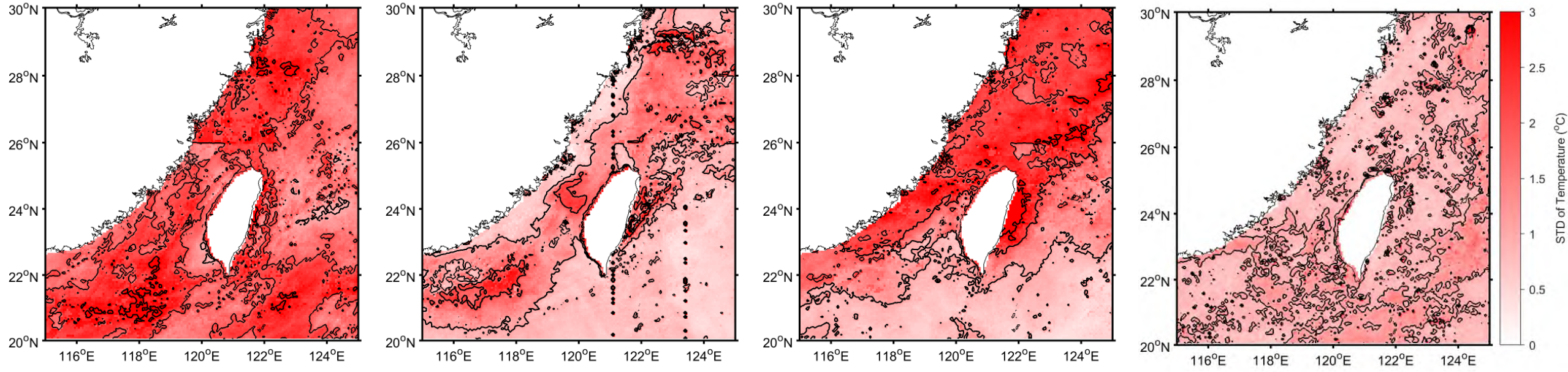
Mean (TWA)

Nov. 2019 Feb. 2020 May 2020 Aug. 2020

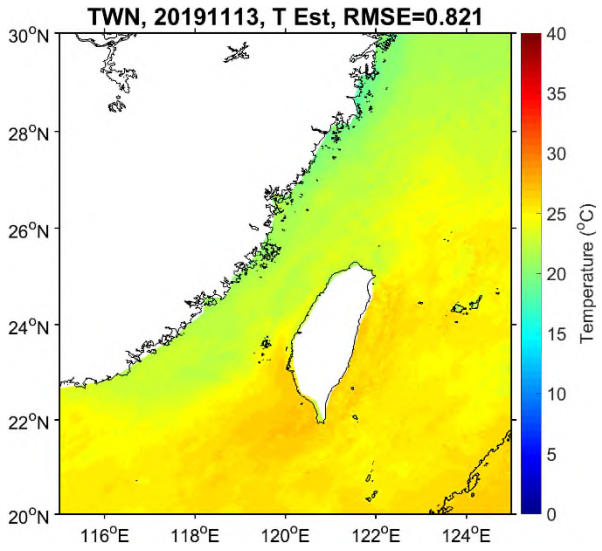


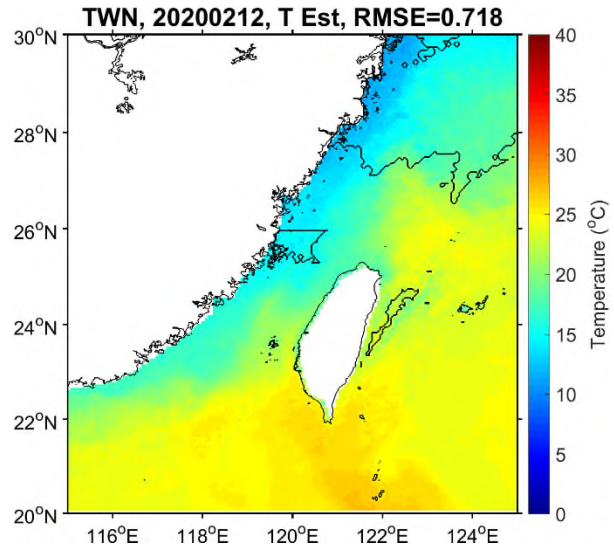
Standard Deviation (TWA)

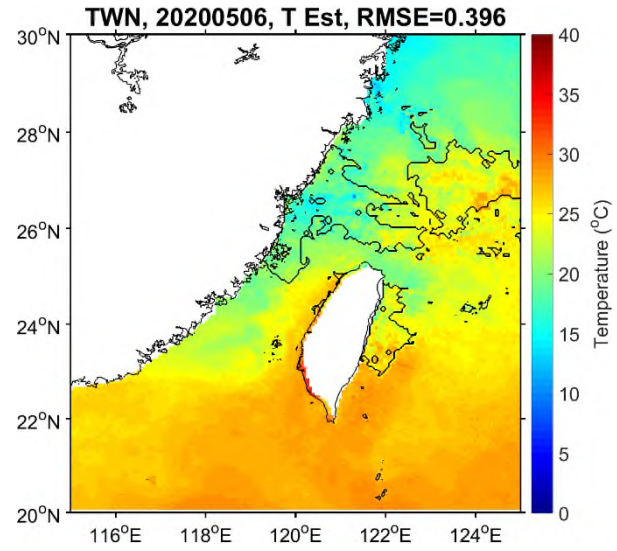
Nov. 2019 Feb. 2020 May 2020 Aug. 2020

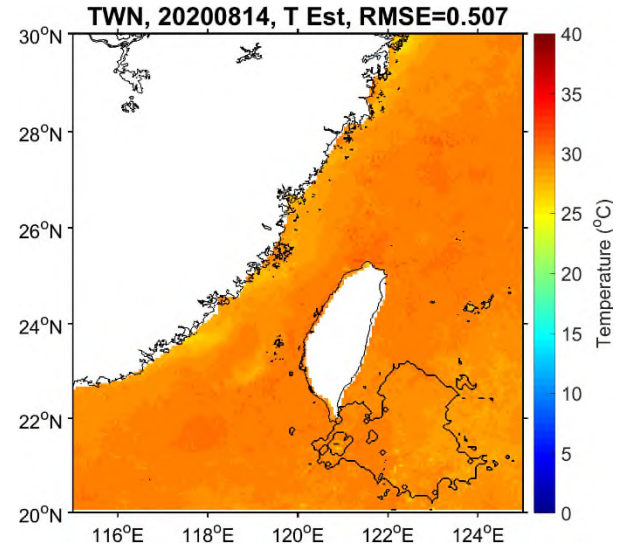


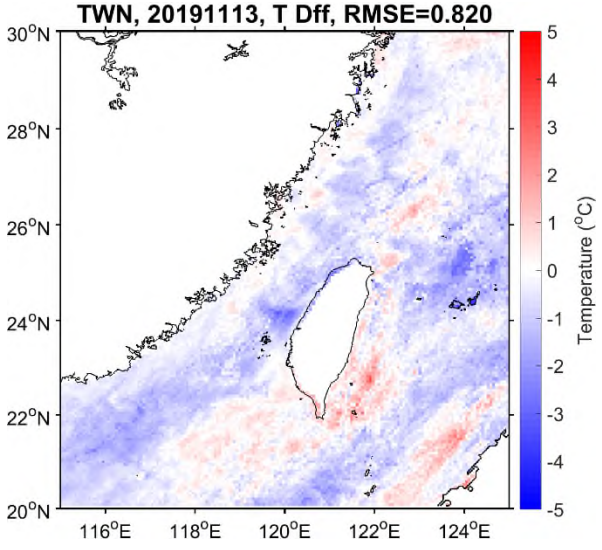
Results – DINEOF (TWA)

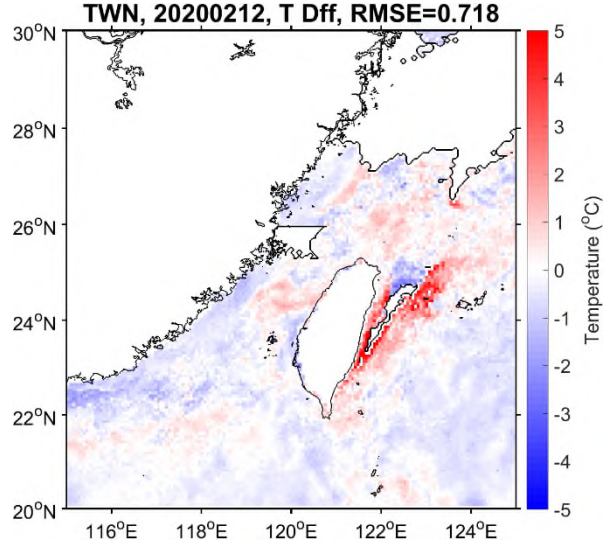


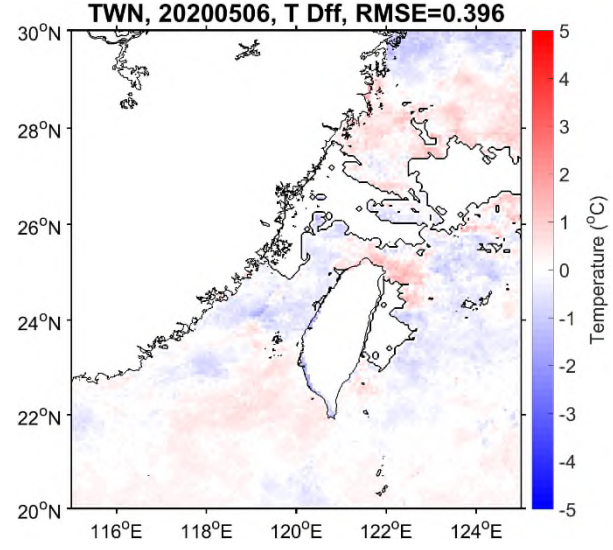


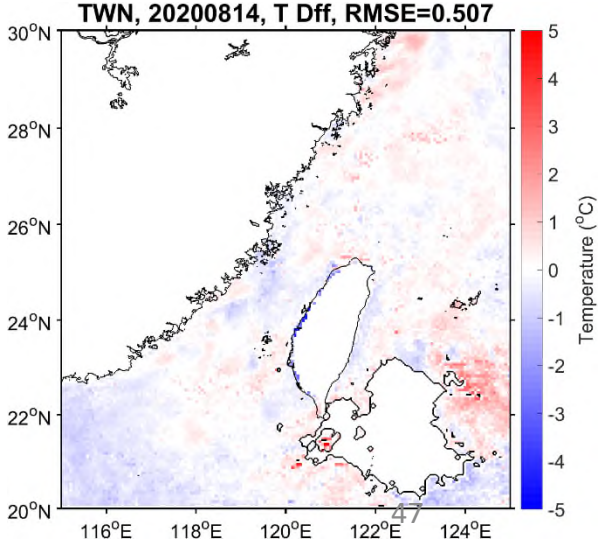




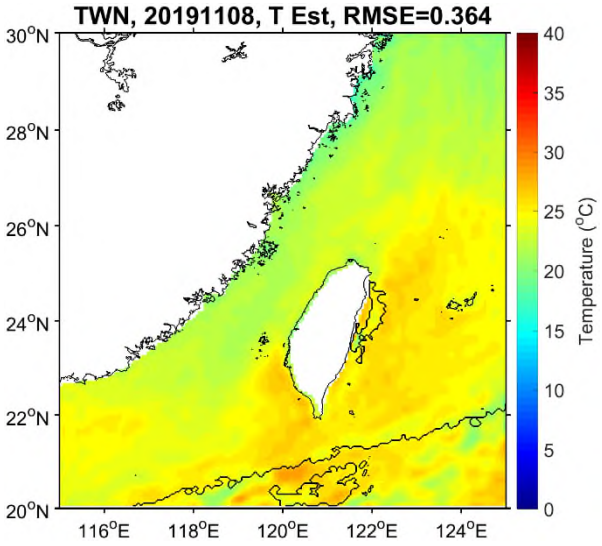


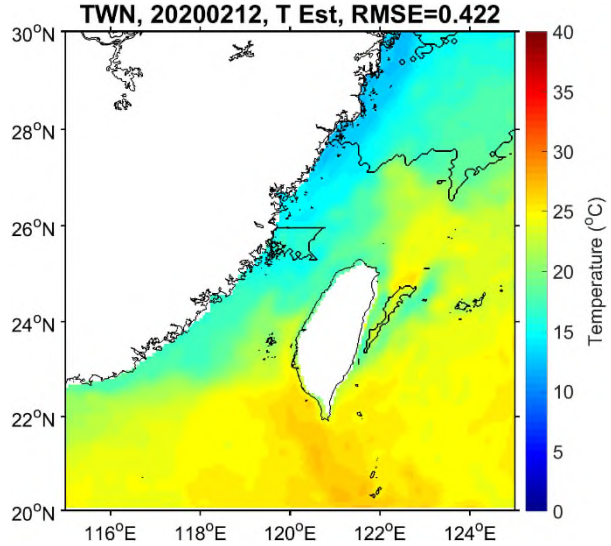


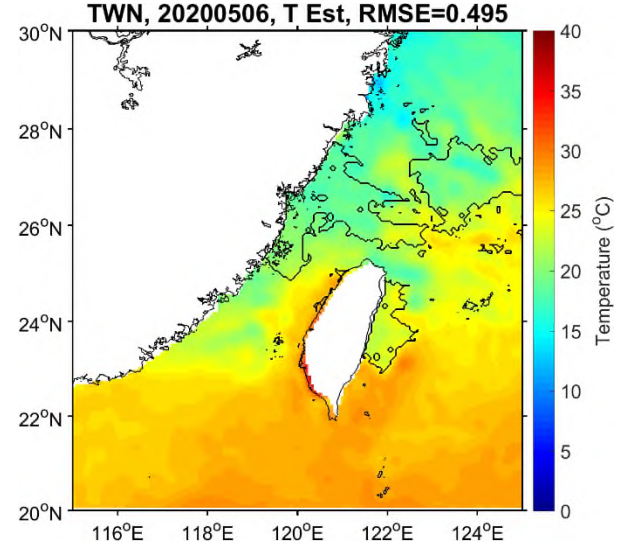


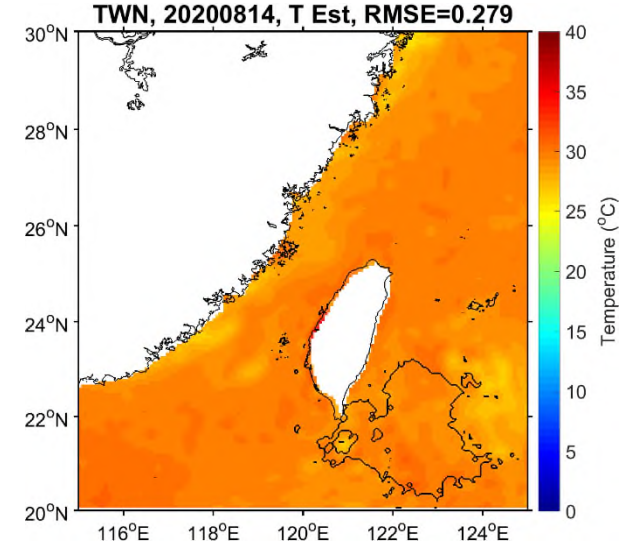


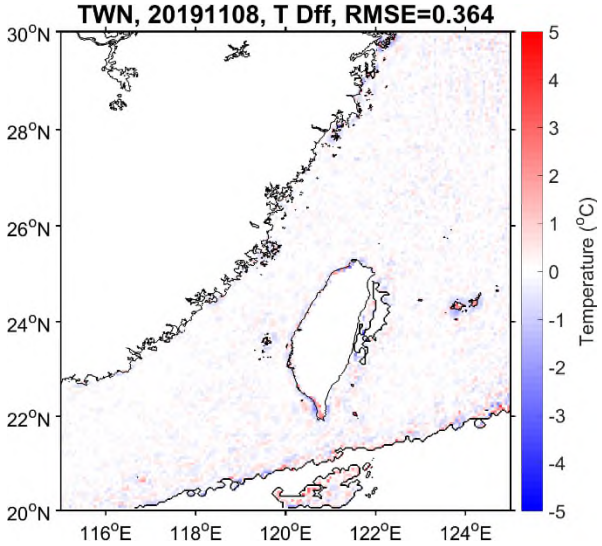
Results – TS-RBFNN (NPO)

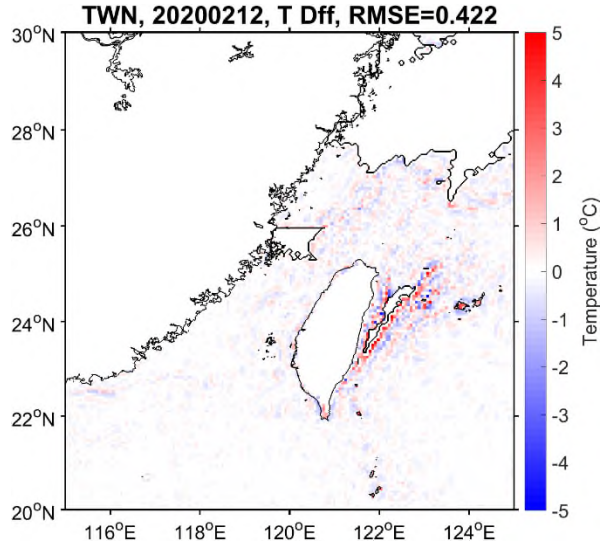


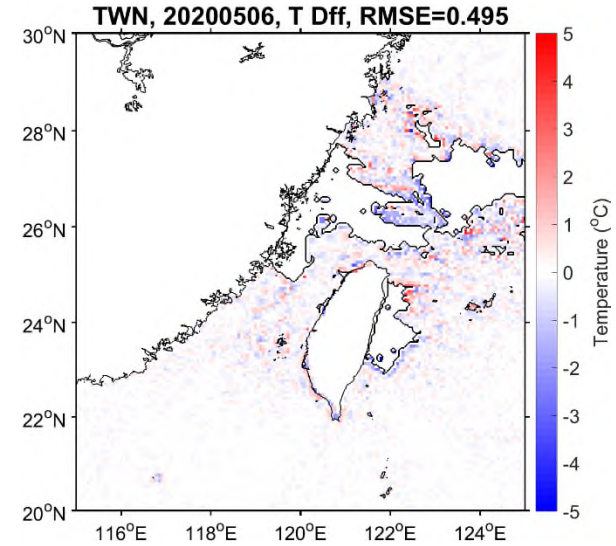


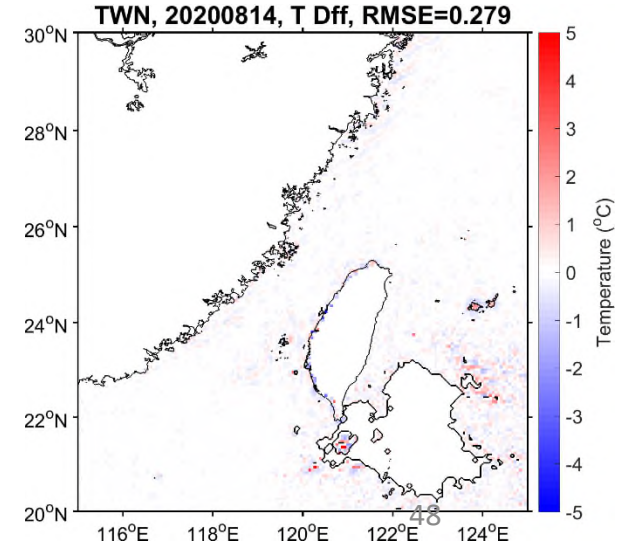






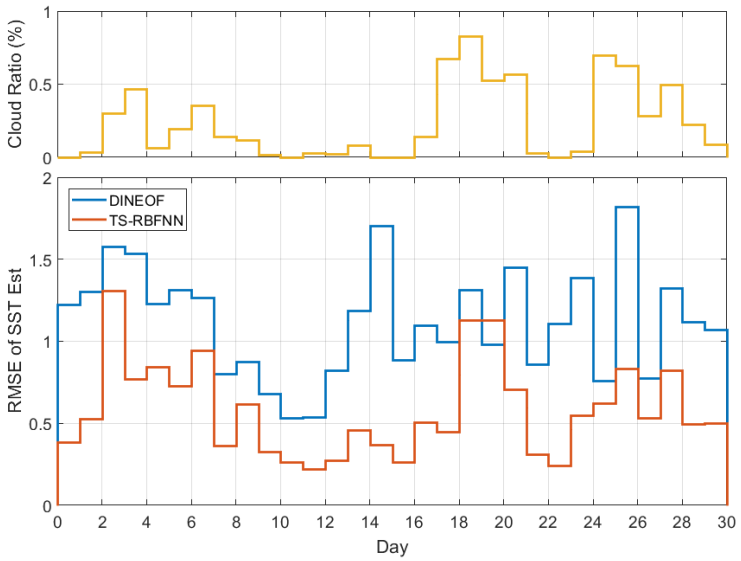




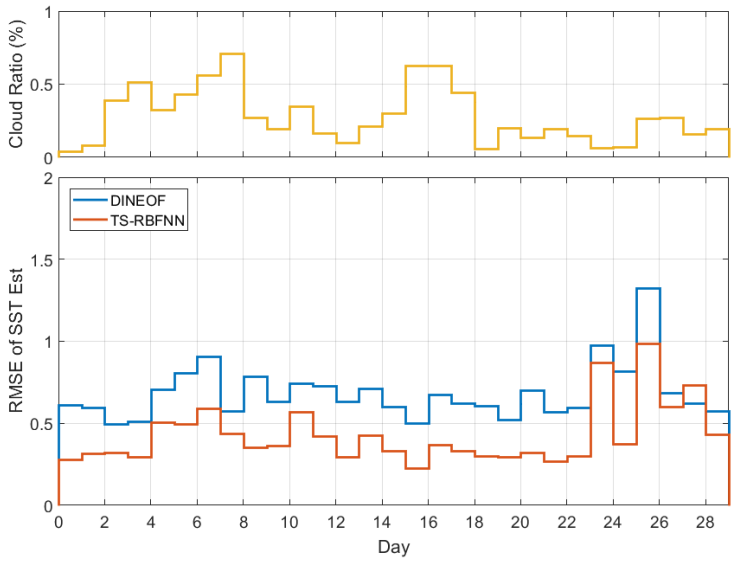


Comparison of performance (DINEOF vs. TS-RBFNN)

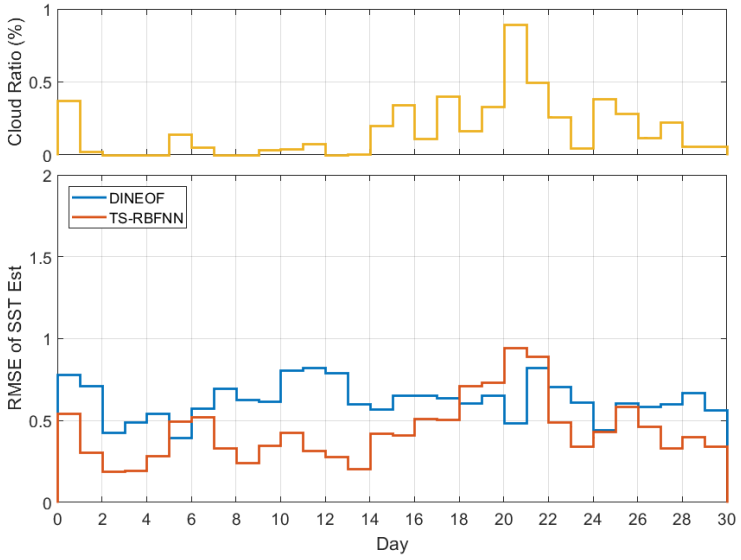
(a) 201911



(b) 202002



(c) 202005



(d) 202008