Constrained Deep learning for Bias Correction (CDBC) of Satellite Radiances in Data Assimilation

[1]**HAN Wei** ,[2]SUN Haofei, [1]YIN Ruoying
1 CMA Earth SystemModeling and Prediction Centre(CEMC)
2 University of Chinese Academy of Sciences

March 15–23, 2023, Tromsø, Norway

Outline

- **Background**
  - The bias correction is an ill-posed problem : separate **observation bias** and **model bias** from O-B
  - Bias correction is **VERY important** in operational NWP: **BC, CBC; VarBC, CVarBC**
  - Satellite observation biases: Noninear dependence (time, angle, orbit, …),Non-Gaussian distribution
  - Could Deep learning “learn” nonlinear observation biases from O-B?

- **Constrained Deep learning for Bias Correction(CDBC)**
  - Linear Regression, Deep Learning and Constrained Deep Learning
  - CDBC to FY-4A GIIRS Bias Correction
  - Impact on analyses and forecasts

- **Summary and discussion**

Constrain the satellite data bias correction: a brief review

- **Using "UNBIASED" observations**
  - Radiosonde mask (Eyre 1992),Radiosonde profile (Joiner and Rokke 2000;Kozo et al.,2005)
  - GPS RO temperature sounding (Zou et al.,2014)
- **VarBC using all other un-corrected observations**
  - Derber and Wu 1998; Dee 2004; Auligne et al.2007; Zhu et al.,2013)
- **Anchor channel method**
  - AMSUA Ch14 (McNally,2007; Di Tomaso and Bormann 2011)
  - IASI ozone channel (Han and McNally, 2010)
- **Constrained BC(CBC) and Constrained VarBC(CVarBC)**
  - CBC(Han 2014, ITSC-19;, Faulwetter et al,2023, ITSC-24)
  - CVarBC(Han and Bormann,2016; Bell et al. 2023,ITSC-24)
  - Using priori information and physical model constraints

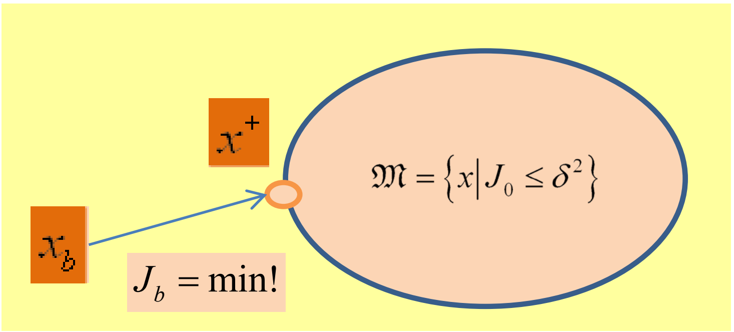
$$\delta J = \left\langle \frac{\partial J}{\partial \mathbf{y}}, \mathbf{y} - \mathbf{H}\mathbf{x}_b - \mathbf{b} \right\rangle$$

FSO: Forecast sensitivity to observation
Over or under bias correction could lead to negative impact

Constrained Variational Bias Correction (CvarBC)

$$\|\mathbf{b}\| \le \|\mathbf{e}\|_{calibration} + \|\mathbf{e}\|_{RT\,model} + \|\mathbf{e}\|_{other}$$

$$\begin{aligned} 2J(\mathbf{x},\boldsymbol{\beta}) \;=\; & (\mathbf{x}_b-\mathbf{x})^T\mathbf{B}_x^{-1}(\mathbf{x}_b-\mathbf{x}) \\ & + (\boldsymbol{\beta}-\boldsymbol{\beta}_b)^T\mathbf{B}_\beta^{-1}(\boldsymbol{\beta}-\boldsymbol{\beta}_b) \\ & + [\mathbf{y}-H(\mathbf{x})-h(\mathbf{x},\boldsymbol{\beta})]^T\mathbf{R}^{-1}[\mathbf{y}-H(\mathbf{x})-h(\mathbf{x},\boldsymbol{\beta})] \end{aligned}$$



$$J_b = \min_{x\in m} \qquad m = \{x | J_0 \le \delta\}$$

$$||\mathrm{h}(\mathrm{x},\beta) - b_0|| \le \delta^2$$

Constrain the total size of bias correction to each channel (Weak Constraint)

$$\begin{aligned} 2J(\mathbf{x},\boldsymbol{\beta}) \;=\; & (\mathbf{x}_b-\mathbf{x})^T\mathbf{B}_x^{-1}(\mathbf{x}_b-\mathbf{x}) \\ & + (\boldsymbol{\beta}-\boldsymbol{\beta}_b)^T\mathbf{B}_\beta^{-1}(\boldsymbol{\beta}-\boldsymbol{\beta}_b) \\ & + [\mathbf{y}-H(\mathbf{x})-h(\mathbf{x},\boldsymbol{\beta})]^T\mathbf{R}^{-1}[\mathbf{y}-H(\mathbf{x})-h(\mathbf{x},\boldsymbol{\beta})] \\ & + \alpha^2[h(\mathbf{x},\boldsymbol{\beta})-\mathbf{b}_0]^T\mathbf{R}_b^{-1}[h(\mathbf{x},\boldsymbol{\beta})-\mathbf{b}_0] \end{aligned}$$

$$\mathbf{d} = \mathbf{y} - H(\mathbf{x})$$

$$\mathbf{P}\boldsymbol{\beta} = h(\mathbf{x},\boldsymbol{\beta})$$

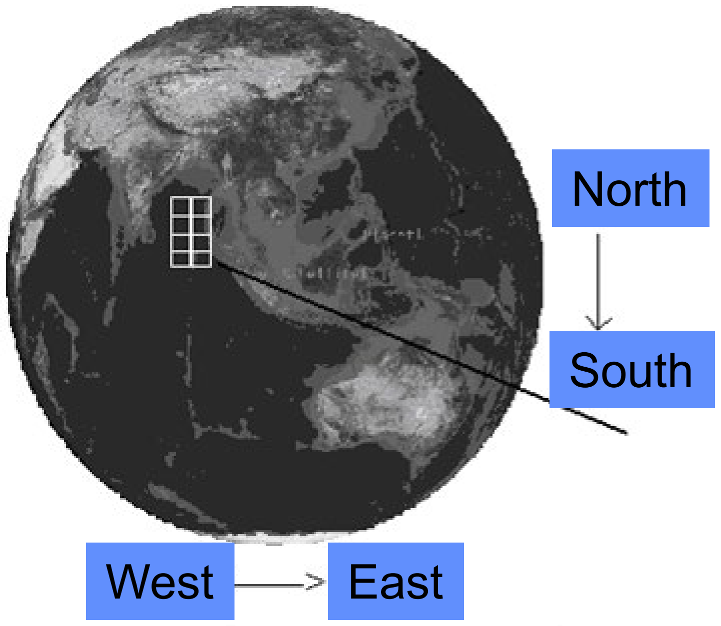
$$\begin{aligned} \nabla_{\boldsymbol{\beta}} J(\mathbf{x},\boldsymbol{\beta}) &= \mathbf{B}_\beta^{-1}(\boldsymbol{\beta}-\boldsymbol{\beta}_b) - \mathbf{P}^T\mathbf{R}^{-1}[\mathbf{d}-\mathbf{P}\boldsymbol{\beta}] + \alpha^2\mathbf{P}^T\mathbf{R}_b^{-1}[\mathbf{P}\boldsymbol{\beta}-\mathbf{b}_0] \\ &= (\mathbf{B}_\beta^{-1} + \mathbf{P}^T\mathbf{R}^{-1}\mathbf{P} + \alpha^2\mathbf{P}^T\mathbf{R}_b^{-1}\mathbf{P})\boldsymbol{\beta} - (\mathbf{B}_\beta^{-1}\boldsymbol{\beta}_b + \mathbf{P}^T\mathbf{R}^{-1}\mathbf{d} + \alpha^2\mathbf{P}^T\mathbf{R}_b^{-1}\mathbf{b}_0) \end{aligned}$$

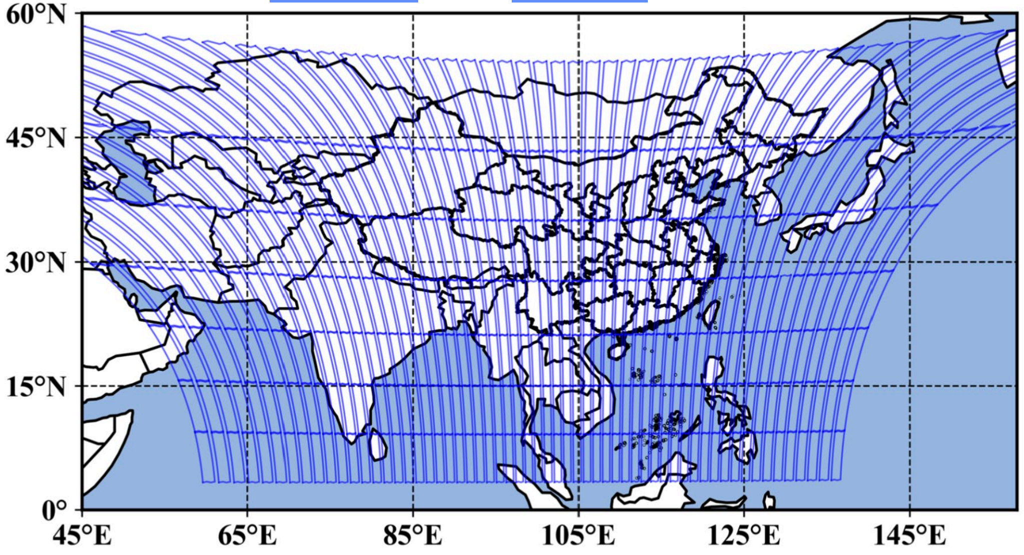
Han,2014,ITSC19; Han and Bormann,2016

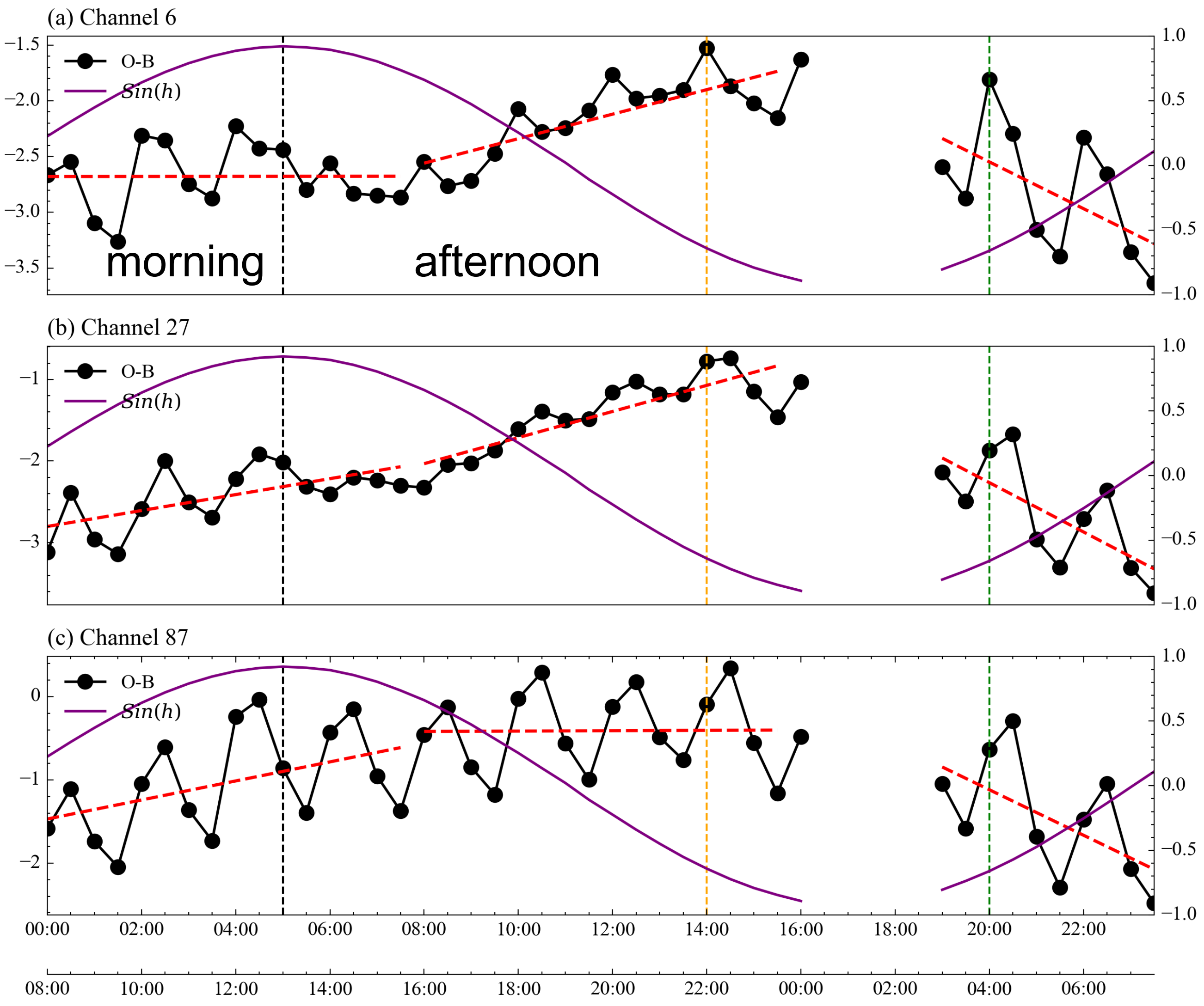
FY-4A GIIRS data bias: nonlinear dependence on time

FY-4A GIIRS
FOV 44#
June 26-28,2021
Daily averaged

Or sensor temperature, …



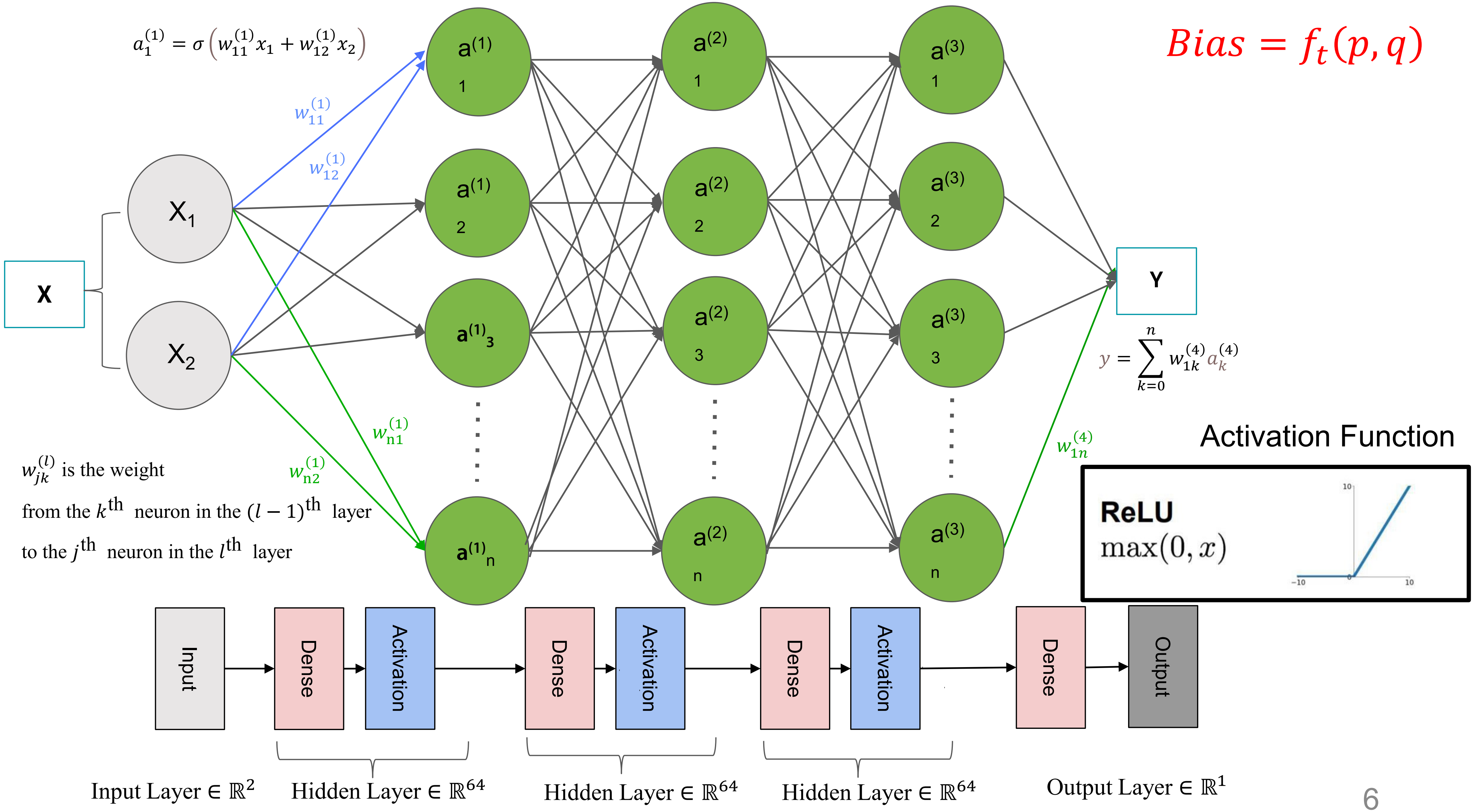




Deep Learning: Multi-Layer Perceptron

"Learn" Nonlinear Dependence

$Bias = f_t(p,q)$



Constraints using EEMD to get Max. Var Contribution as $C_0(t)$

Algorithm 1 Empirical Mode Decomposition(EMD)

Input: $s(t)$: original signal
Output: $\mathrm{IMF}_k(t)$, $r_L(t)$

initial $i=1, k=1, r(t)=s(t)$ and $x_1(t)=r(t)$
while $r(t) \neq 0$ or $r(t)$ is non-monotonic **do**
  while $x(t)$ has non-negligible local mean **do**
    Get the upper envelopes: $x_{upper}(t)$ and the lower envelopes: $x_{lower}(t)$, using cubic interpolation
    Compute the mean of envelopes: $Avg(t) = (x_{upper}(t) + x_{lower}(t))/2$
    Updates: $x_i(t) = x_i(t) - Avg(t)$
    $i = i+1$
  end while
  Extract mode: $\mathrm{IMF}_k(t) = x_i(t)$
  $k = k+1$
  Update the residual: $r(t) = r(t) - \mathrm{IMF}_k(t)$
end while
the original signal $s(t)$ can be reconstructed using the formulation:

$$s(t) = \left(\sum_{k=1}^{L} IMF_k(t)\right) + r_L(t)$$

where L is the number of IMFs and $r_L(t)$ is the residue term

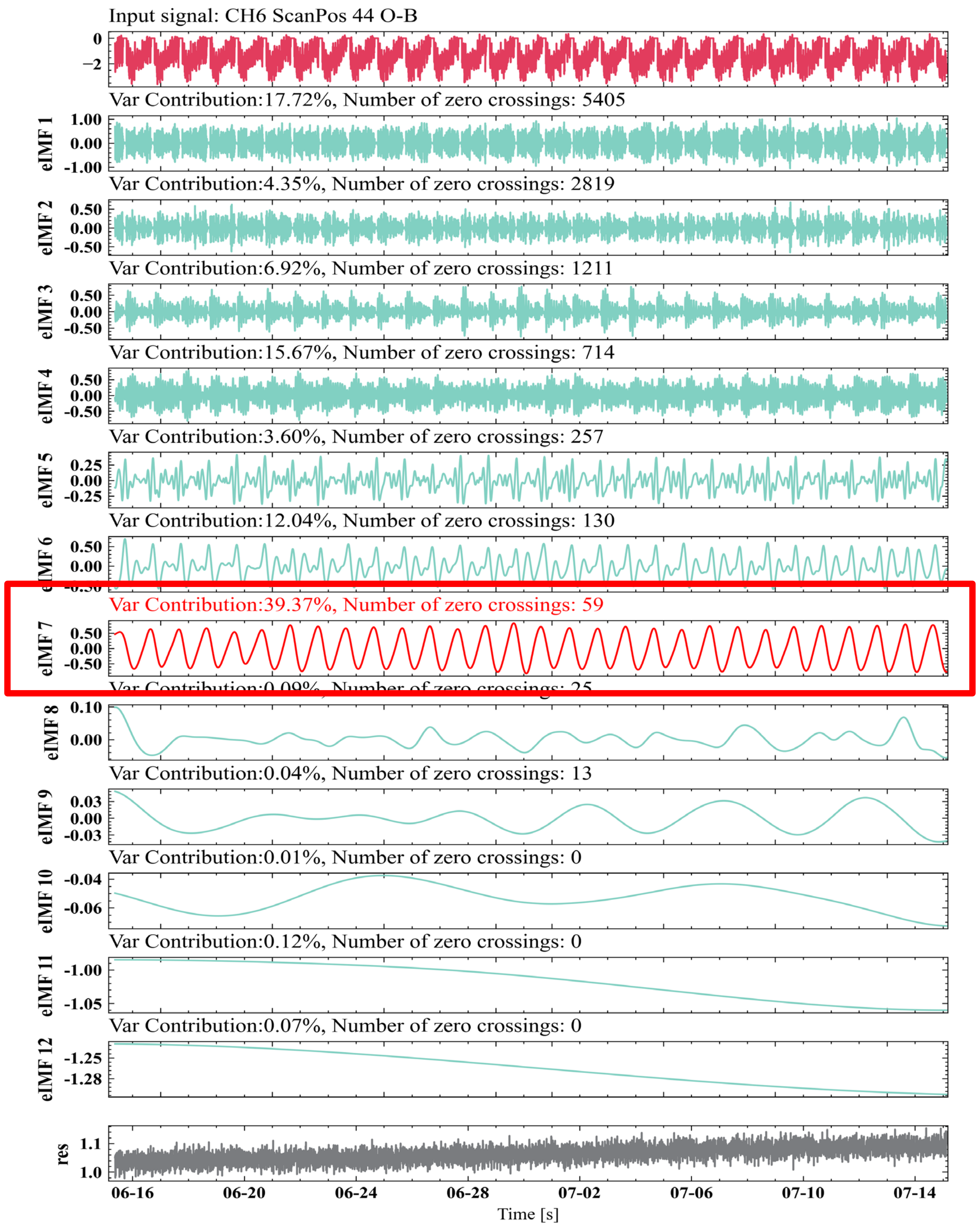
Algorithm 2 Ensemble Empirical Mode Decomposition(EEMD)

Input: $s(t)$: original signal; $n_i(t)$: white noise from standard normal distribution; N: number of ensemble members;
Output: $\mathrm{IMF}_k(t)$

initial $i=1$
while $i \leq N$ **do**
  Add white noise $n_i(t)$ to the original signal $s(t)$ to generate a new signal: $x_i(t) = s(t) + n_i(t)$
  Extracted IMFs by EMD: $C_{ij}(t)$, where j means different IMFs
  Updates: $i = i+1$
end while
Extract mode: $\mathrm{IMF}_j(t) = \sum_{i=1}^{N} Cij(t)$
the original signal $s(t)$ can be reconstructed using the formulation:

$$s(t) = \left(\sum_{j=1}^{L} IMF_j(t)\right) + r_L(t)$$

where L is the number of IMFs and $r_L(t)$ is the residue term

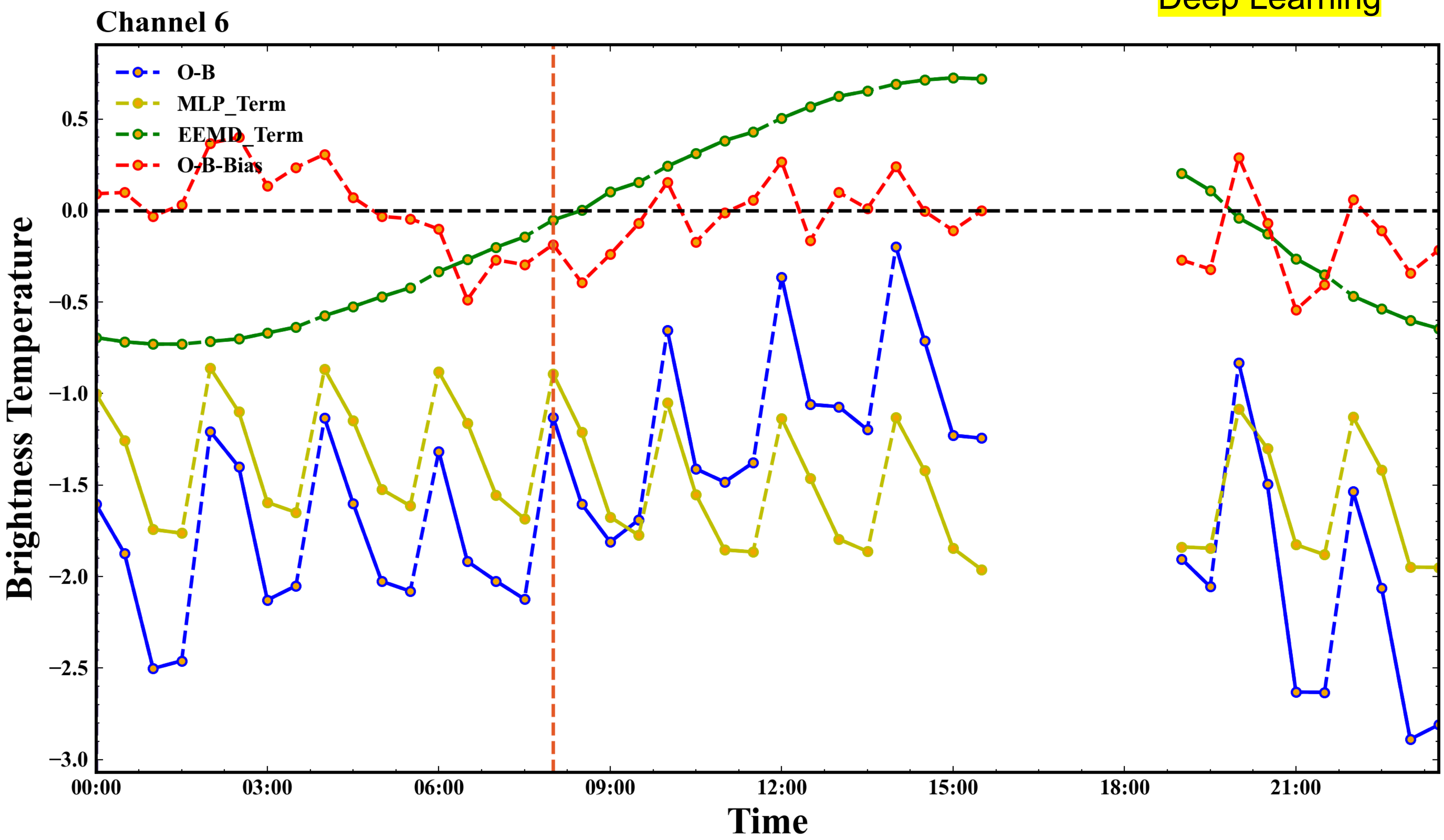


Constraints

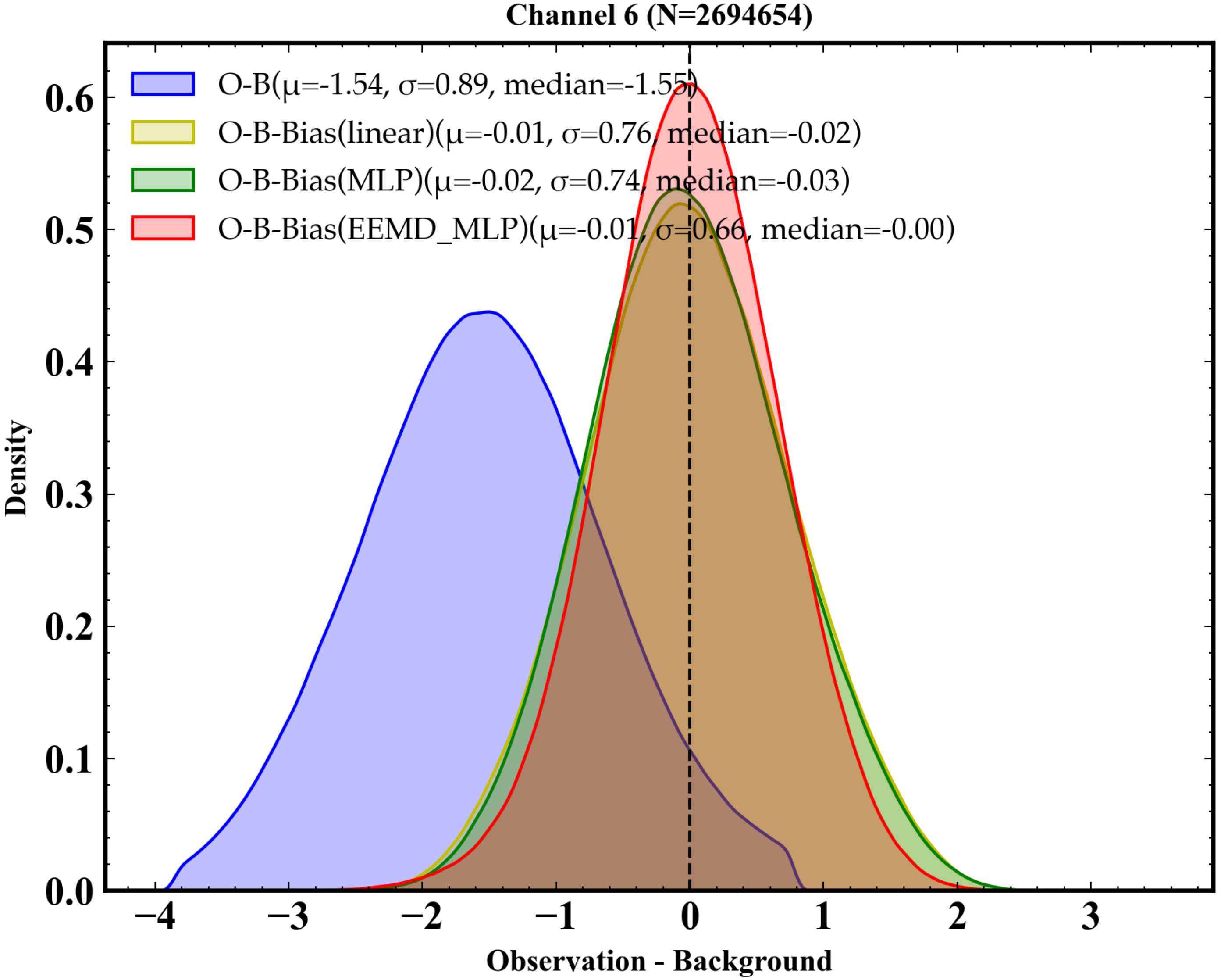
Constrained Deep learning for BC (CDBC)

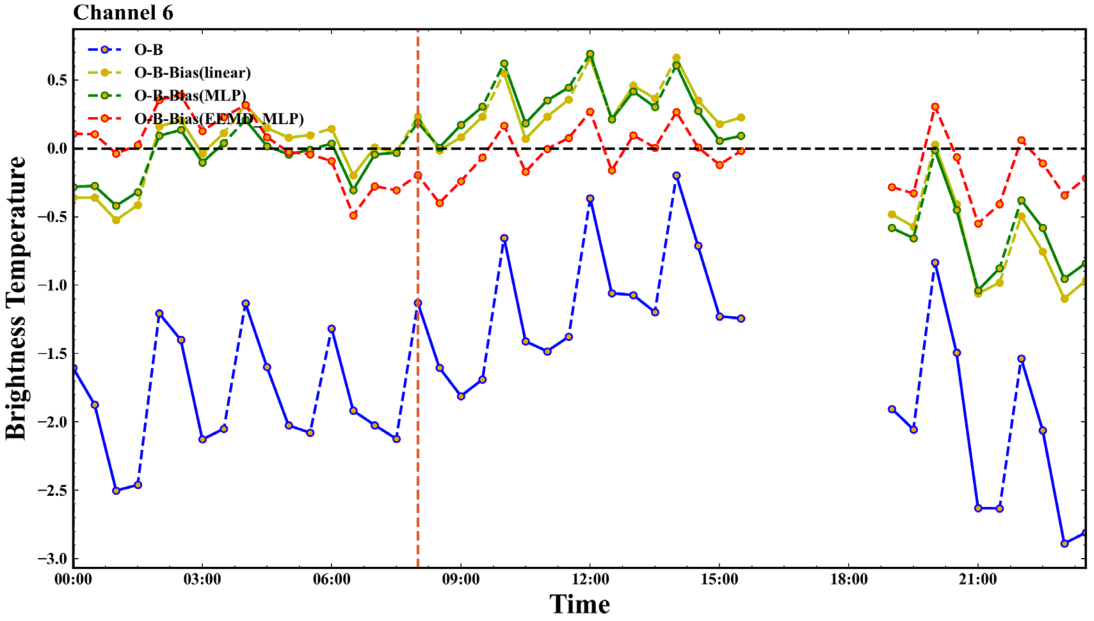
$$Bias = f_t(p,q) + C_0(t)$$

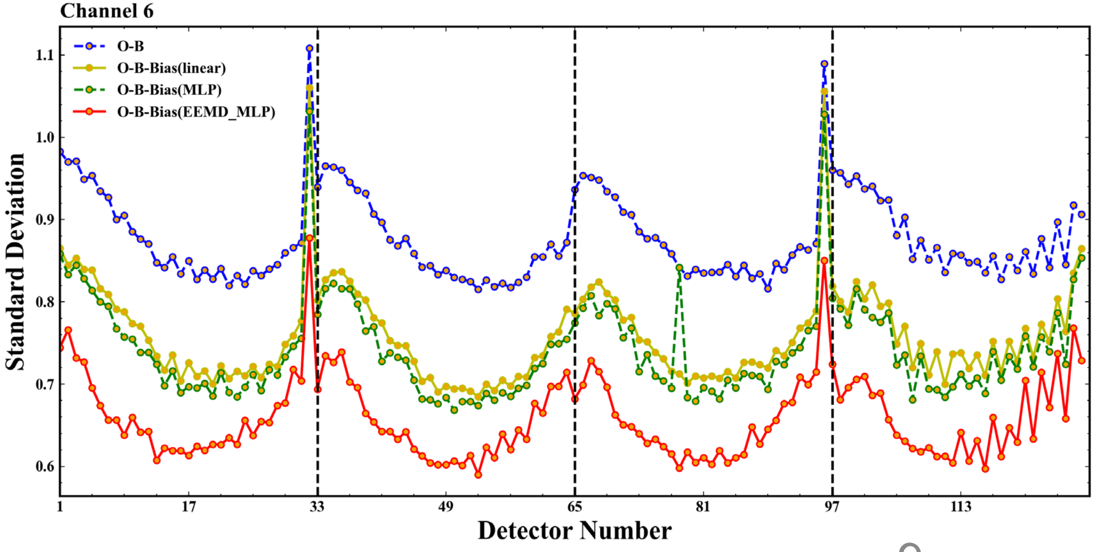
Deep Learning



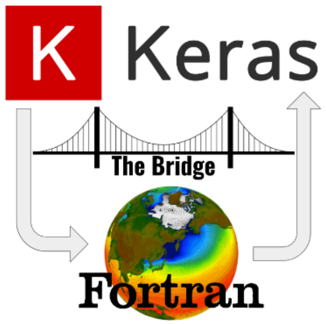
Linear Regression, MLP(DBC), EEMD-MLP(CDBC)

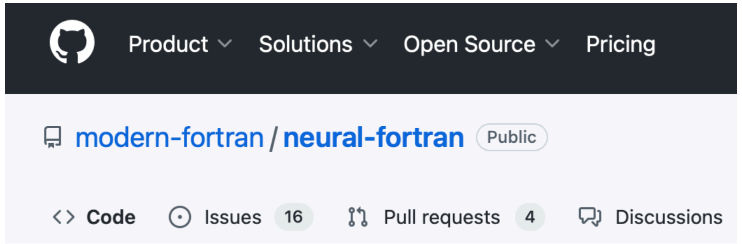






Application in Data Assimilation Cycle: DL models in Fortran

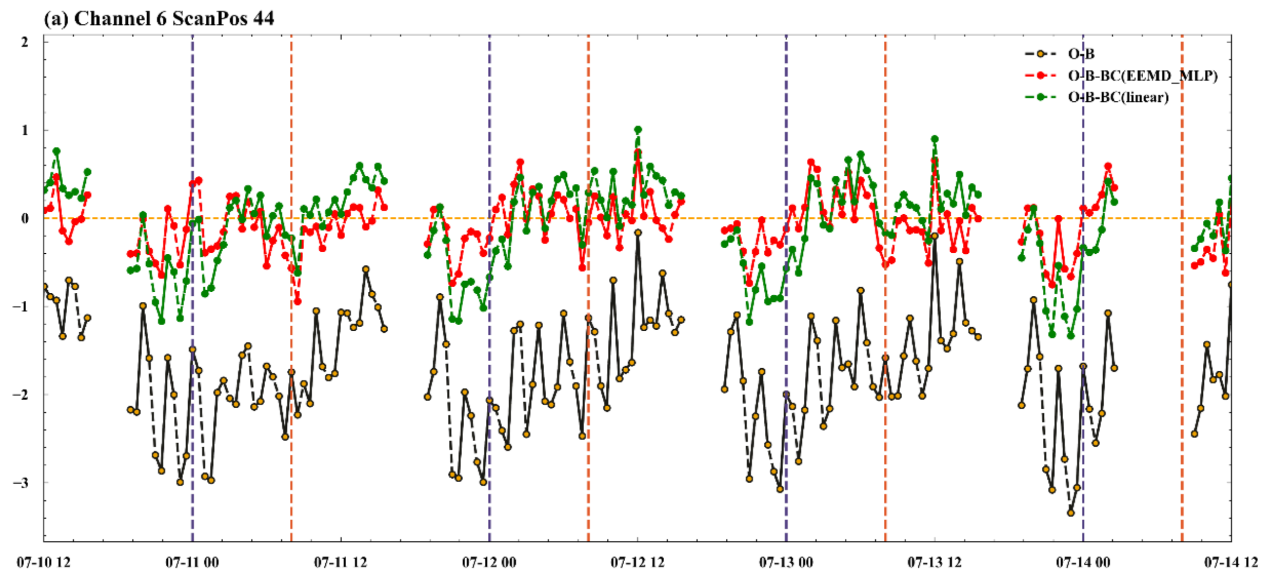


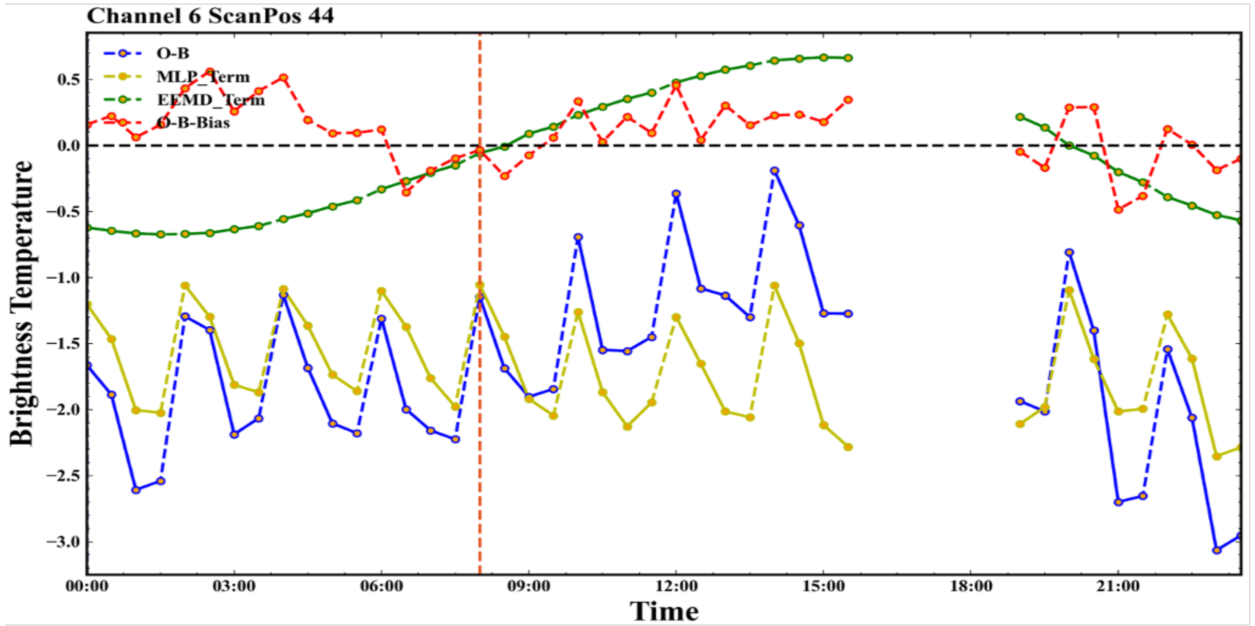


These libraries allow users to convert DL models to usable in Fortran

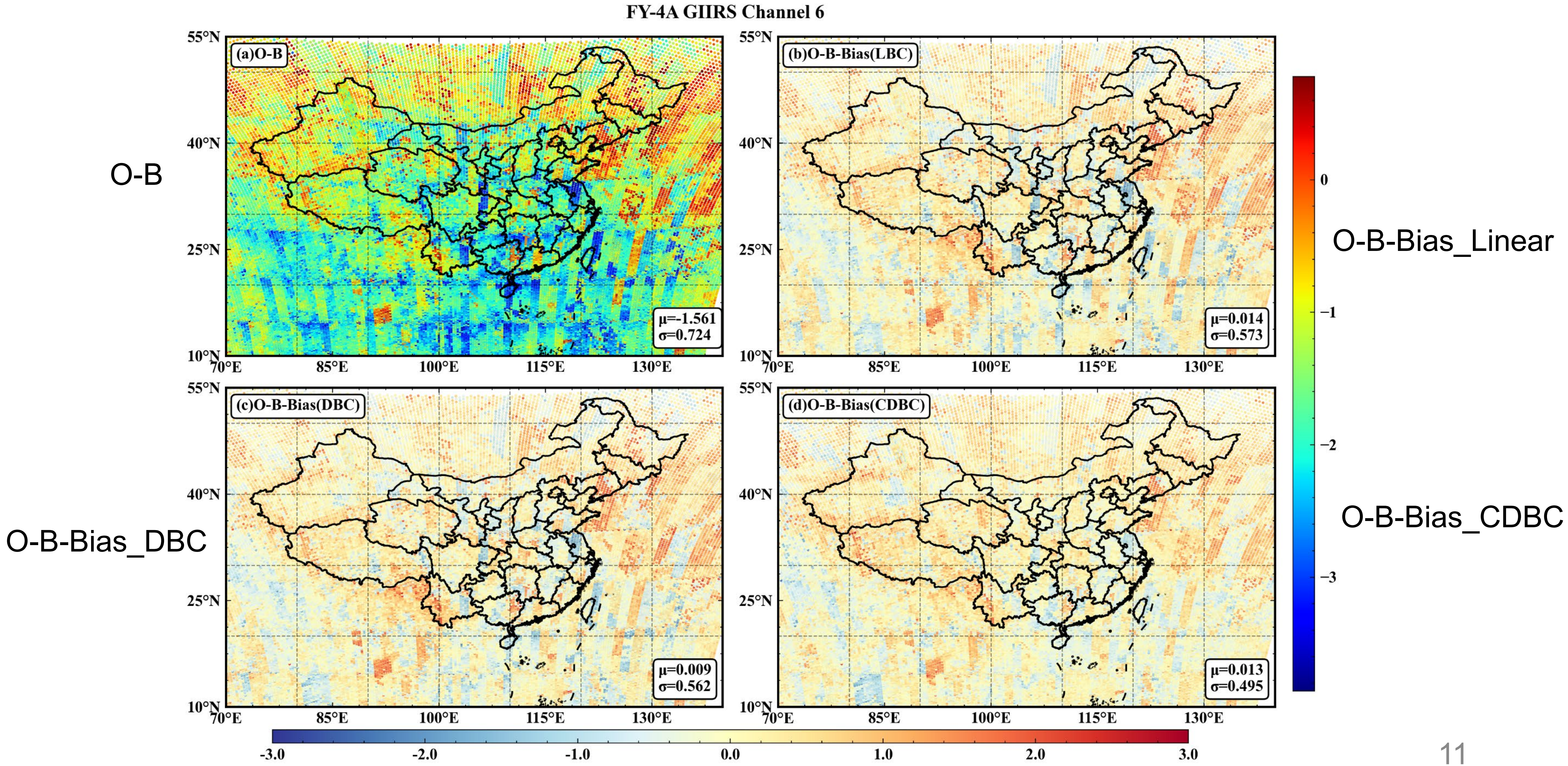
```
# add by sunhaofei for DeepLearn Model
mod_kinds.o: ./mod_kinds.F90
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_kinds.F90

mod_random.o: ./mod_random.F90 mod_kinds.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_random.F90
mod_parallel.o: ./mod_parallel.F90 mod_kinds.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_parallel.F90
mod_io.o: ./mod_io.F90 mod_kinds.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_io.F90

mod_activation.o: ./mod_activation.F90 mod_kinds.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_activation.F90

mod_layer.o: ./mod_layer.F90 mod_activation.o mod_kinds.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_layer.F90

mod_batchnorm_layer.o: ./mod_batchnorm_layer.F90 mod_layer.o mod_kinds.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_batchnorm_layer.F90
mod_dropout_layer.o: ./mod_dropout_layer.F90 mod_layer.o mod_kinds.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_dropout_layer.F90

mod_dense_layer.o: ./mod_dense_layer.F90 mod_layer.o mod_activation.o mod_kinds.o mod_random.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_dense_layer.F90
mod_network.o: ./mod_network.F90 mod_dense_layer.o mod_batchnorm_layer.o mod_dropout_layer.o
mod_layer.o mod_parallel.o mod_activation.o mod_kinds.o
	$(FC) $(CPPDEFS) $(CPPFLAGS) $(FFLAGS) -c	./mod_network.F90
```

Ref: https://github.com/scientific-computing/FKB; https://github.com/modern-fortran/neural-fortran

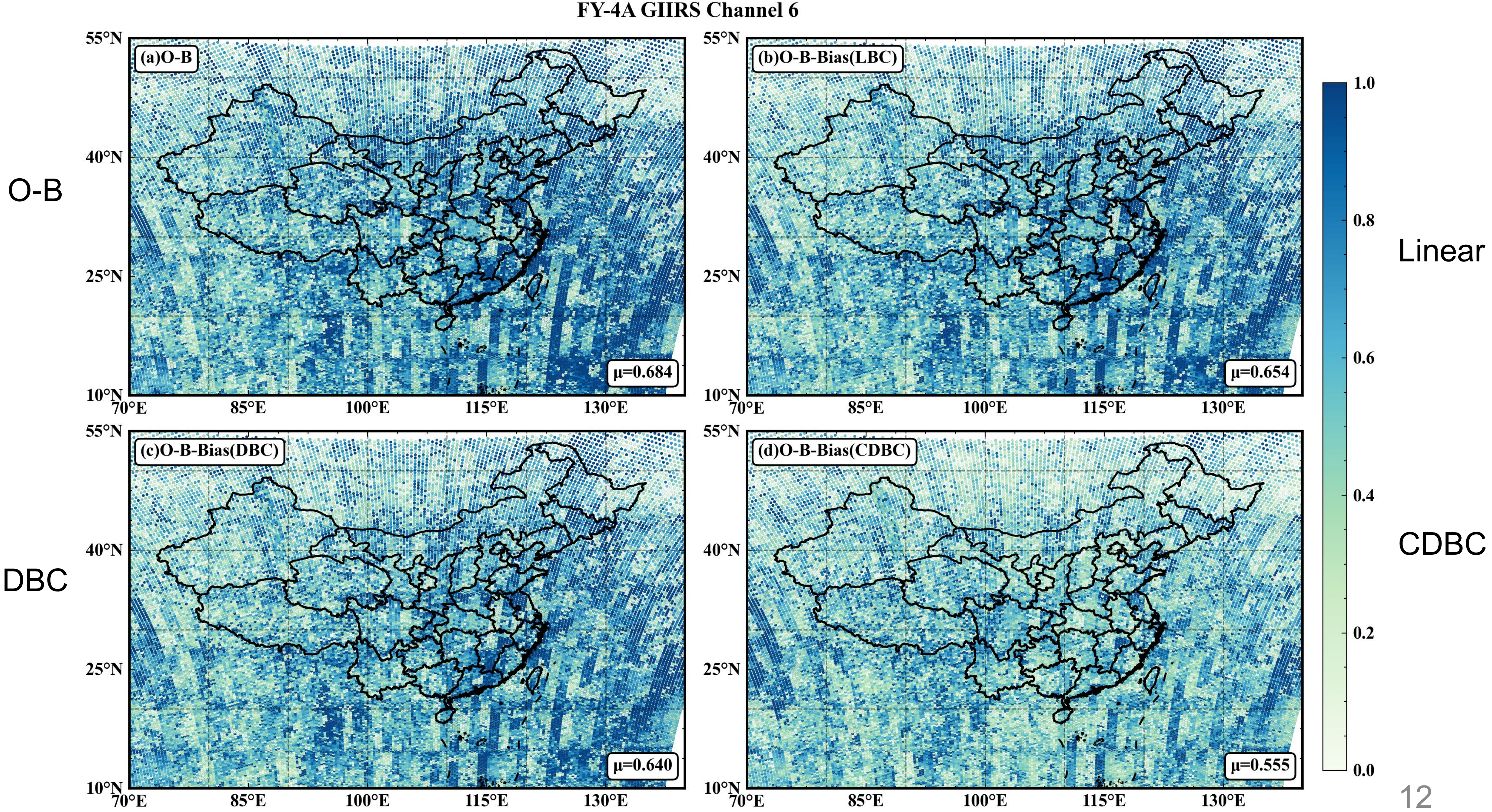




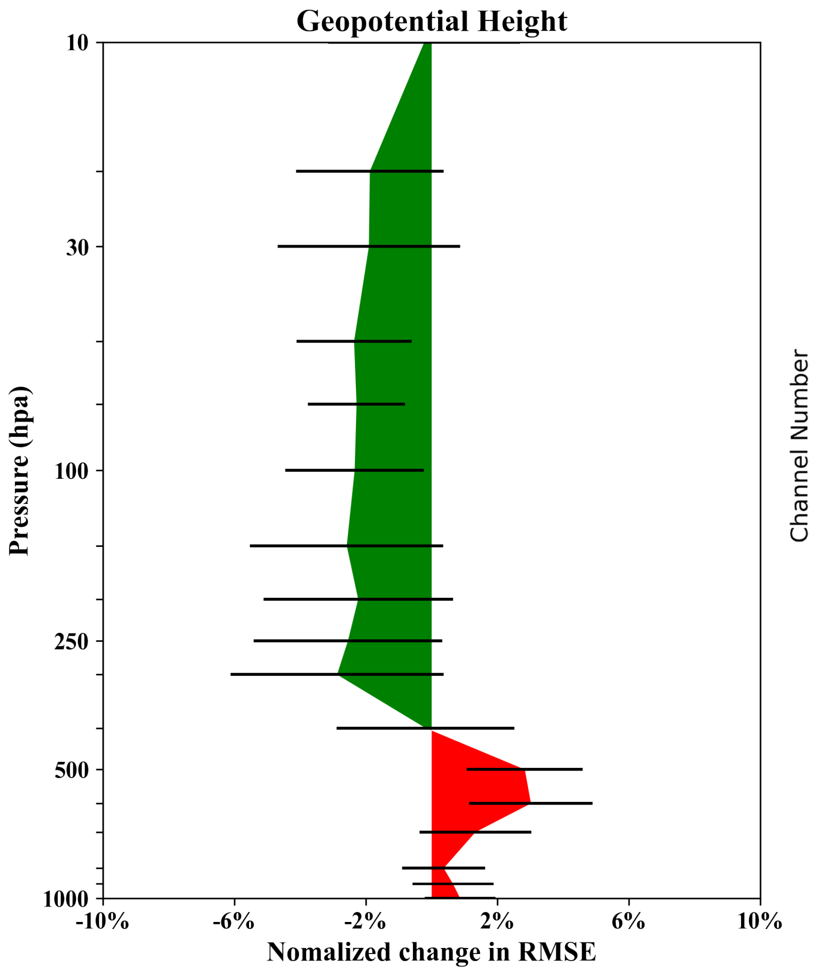
Mean(O-B): Linear Regression, DBC, CDBC

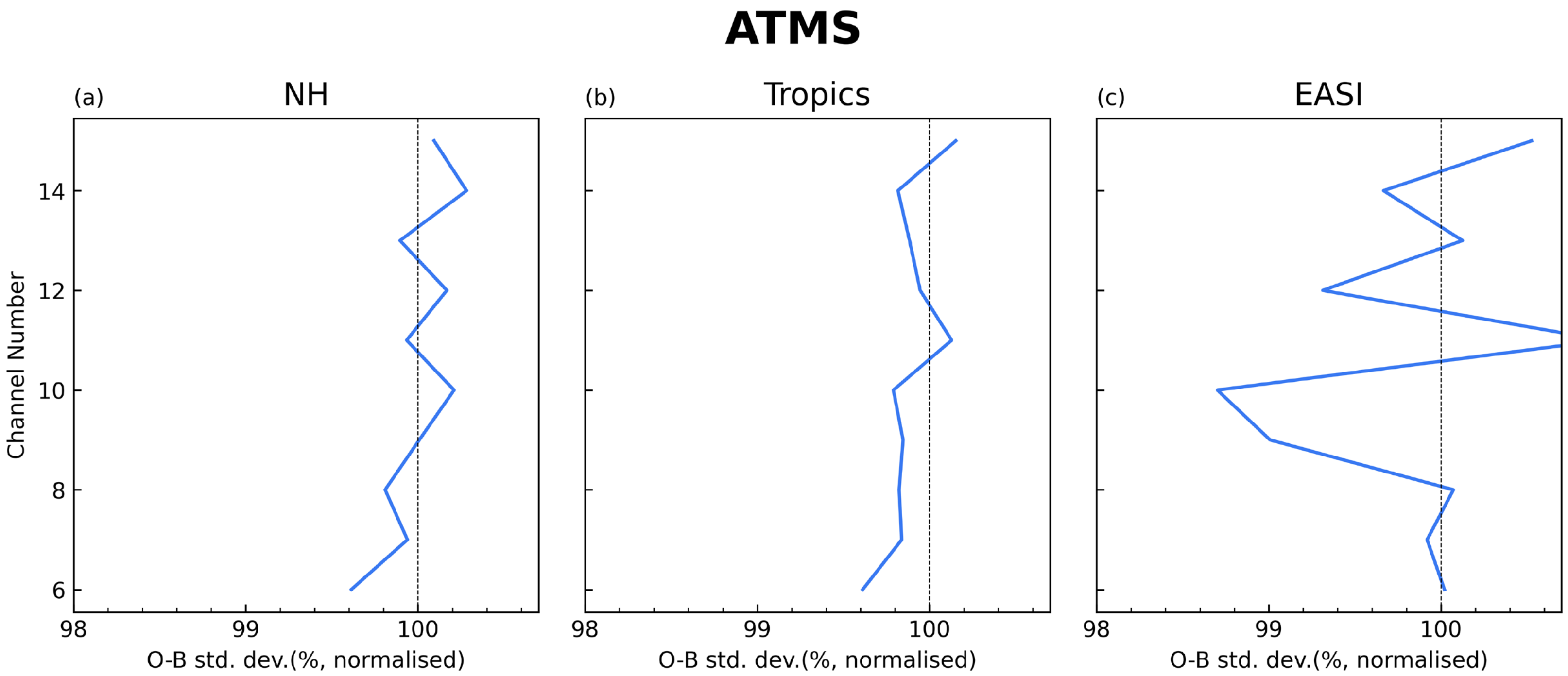


Std(O-B): Linear Regression, DBC, CDBC



Verification of Assimilation Cycle：CDBC against Linear Regression





Std (O-B) dev.(%,normalized)

Model diurnal bias in lower troposphere?

Summary and discussion

- **Could Deep learning “learn” nonlinear observation biases from O-B?**
  - Yes, But need physical constraints

- **Constrained Deep learning for Bias Correction(CDBC)**
  - Linear Regression, Deep Learning and Constrained Deep Learning
  - Application to FY-4A GIIRS Bias Correction
  - Impact on analyses and forecasts

- **Future work**
  - Observation bias **physical model + deep learning**
  - Online update of the CDBC in operational data assimilation
  - Automatic selection of predictors