Climate-Invariant, Causally-Consistent Neural Networks as Robust Emulators of Subgrid Processes across Climates

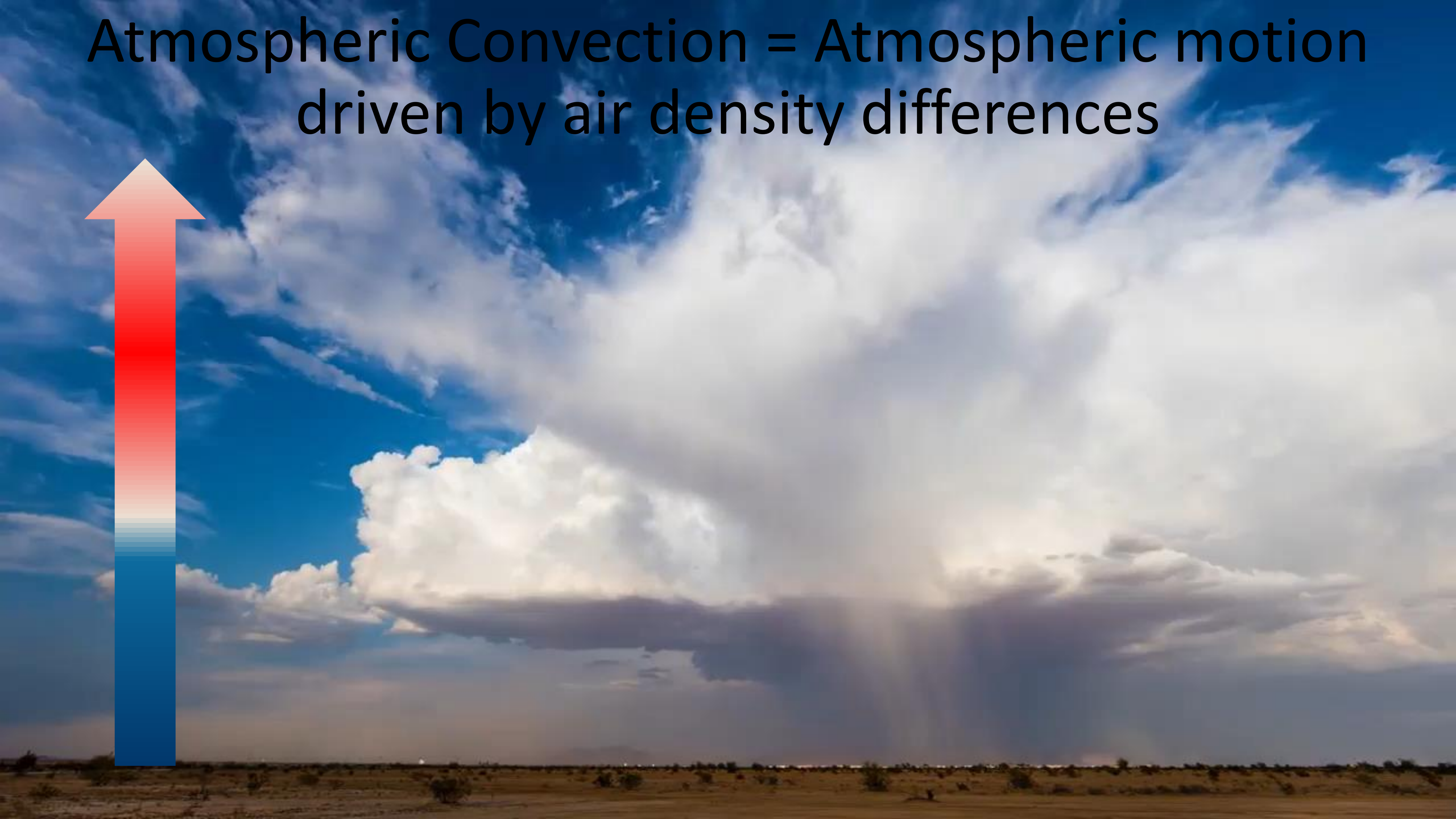
UNIL | Université de Lausanne

J. Yuval, P. O'Gorman (MIT)
A. Gupta, L. Peng (UCI)
S. Rasp (Climate AI)
F. Ahmed, D. Neelin (UCLA)
N. Lutsko (UCSD)

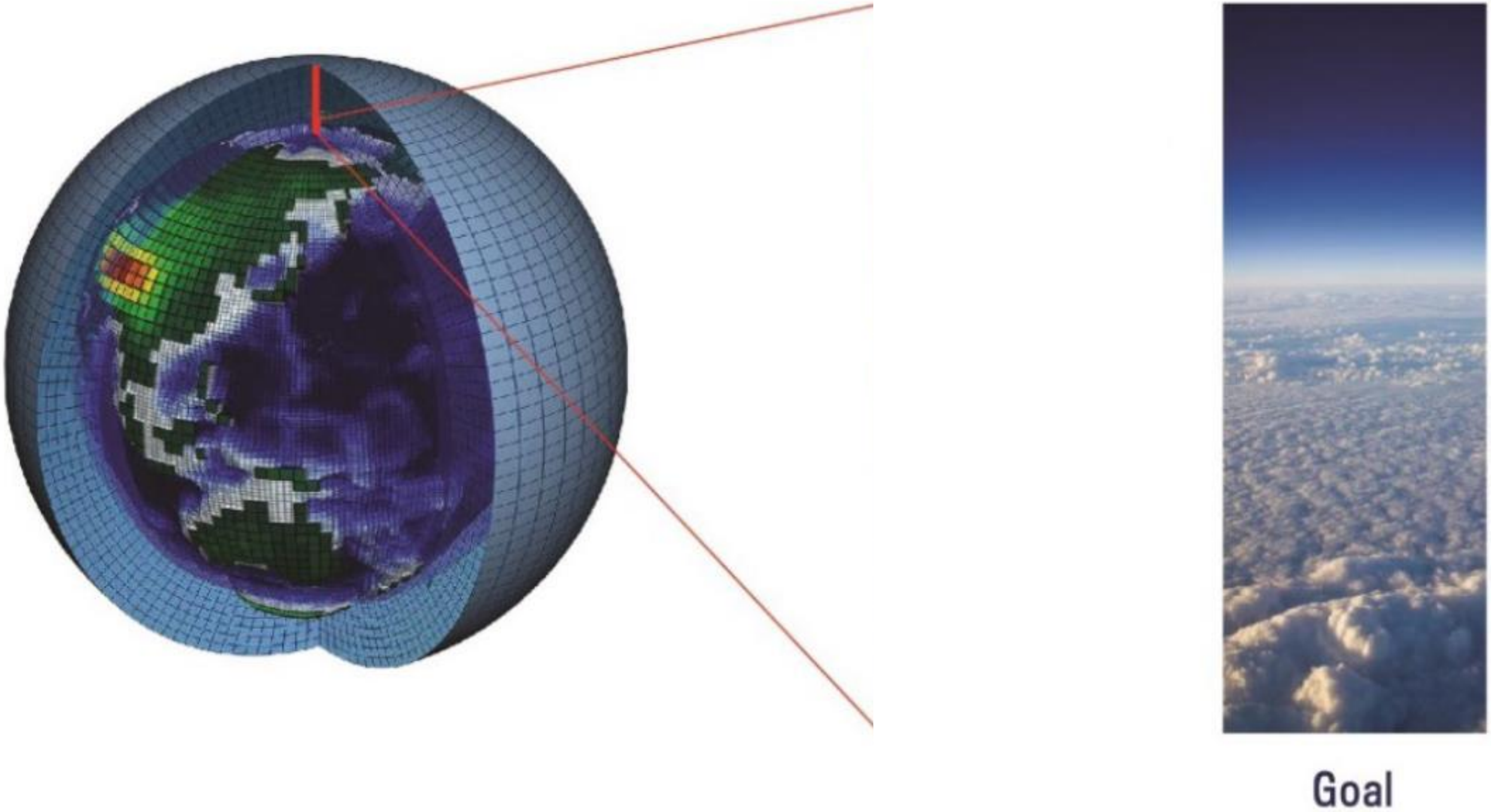
Tom Beucler (UNIL)
M. Pritchard (UCI),
P. Gentine (Columbia)

F. Iglesias-Suarez
V. Eyring **(DLR)**
J. Runge (DLR, TUB)

Atmospheric Convection = Atmospheric motion driven by air density differences



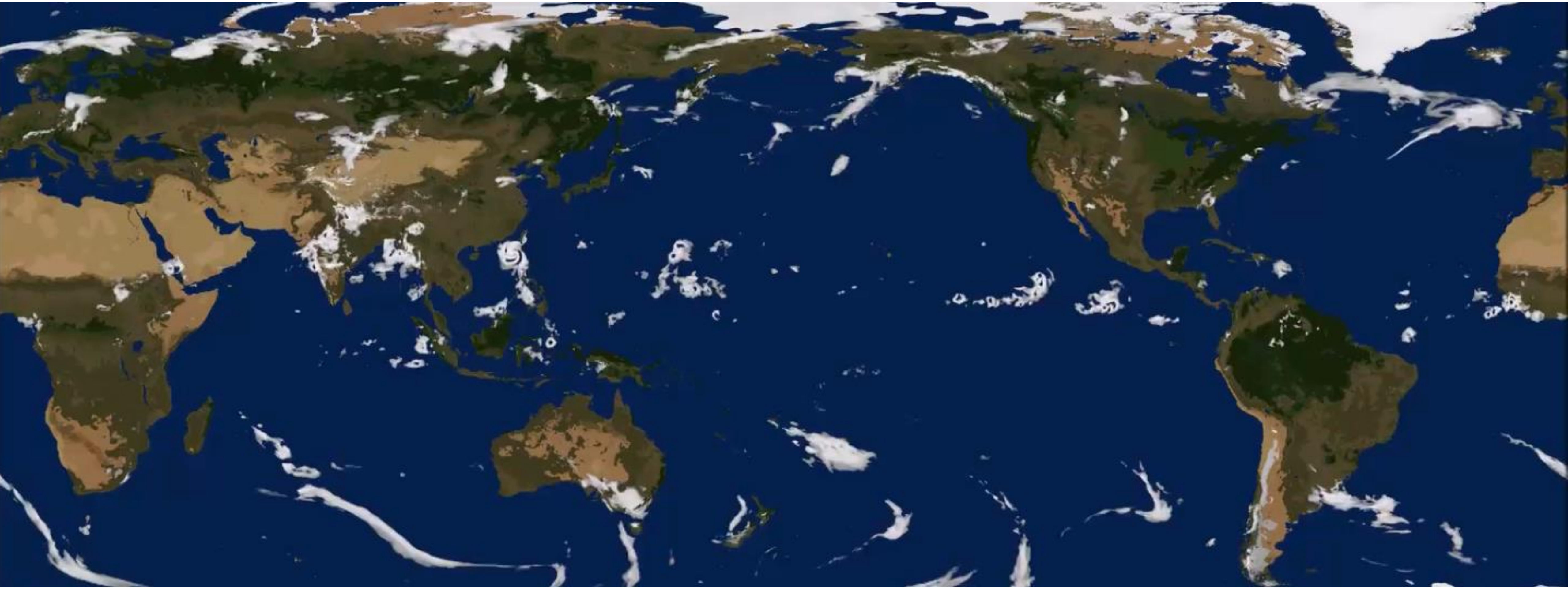
Motivation 1: Largest uncertainties in climate projections from clouds



Source: Zelinka et al. (2020), Meehl et al. (In Review), Gentine, Eyring & Beucler (2020)

__Motivation 1__: Largest uncertainties in climate projections from clouds

__Motivation 2__: Global cloud-resolving models can resolve convection & clouds at ~1km, but only for short period (1 year)



__Source:__ Stevens et al. (2019), Sato et al. (2009), SAM: Khairoutdinov and Randall (2003), Lee and Khairoutdinov (2015)

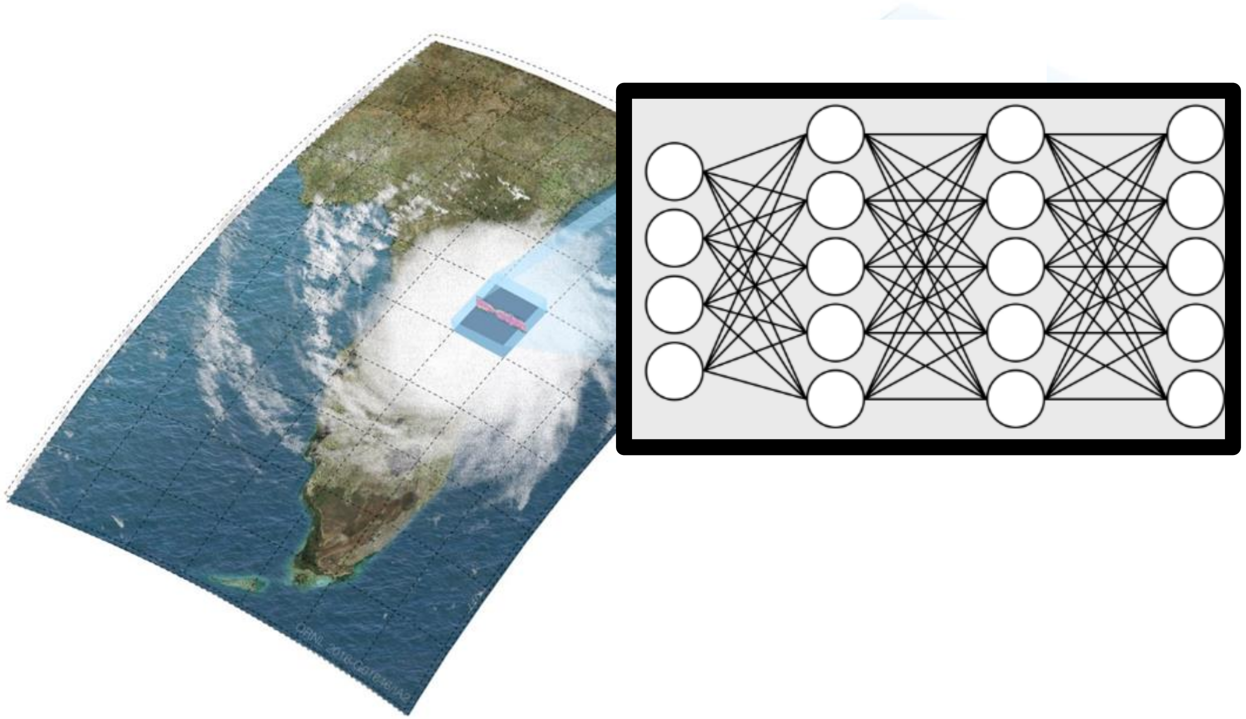
Motivation 1: Largest uncertainties in climate projections from clouds

Motivation 2: Global cloud-resolving models can resolve convection & clouds at ~1km, but only for short period (1 year)

Motivation 3: ML can accurately mimic ~1km convective processes

See: Rasp et al. (2018), Brenowitz et al. (2018,2019), Gentine et al. (2018), Yuval et al. (2020), Krasnopolsky et al. (2013)

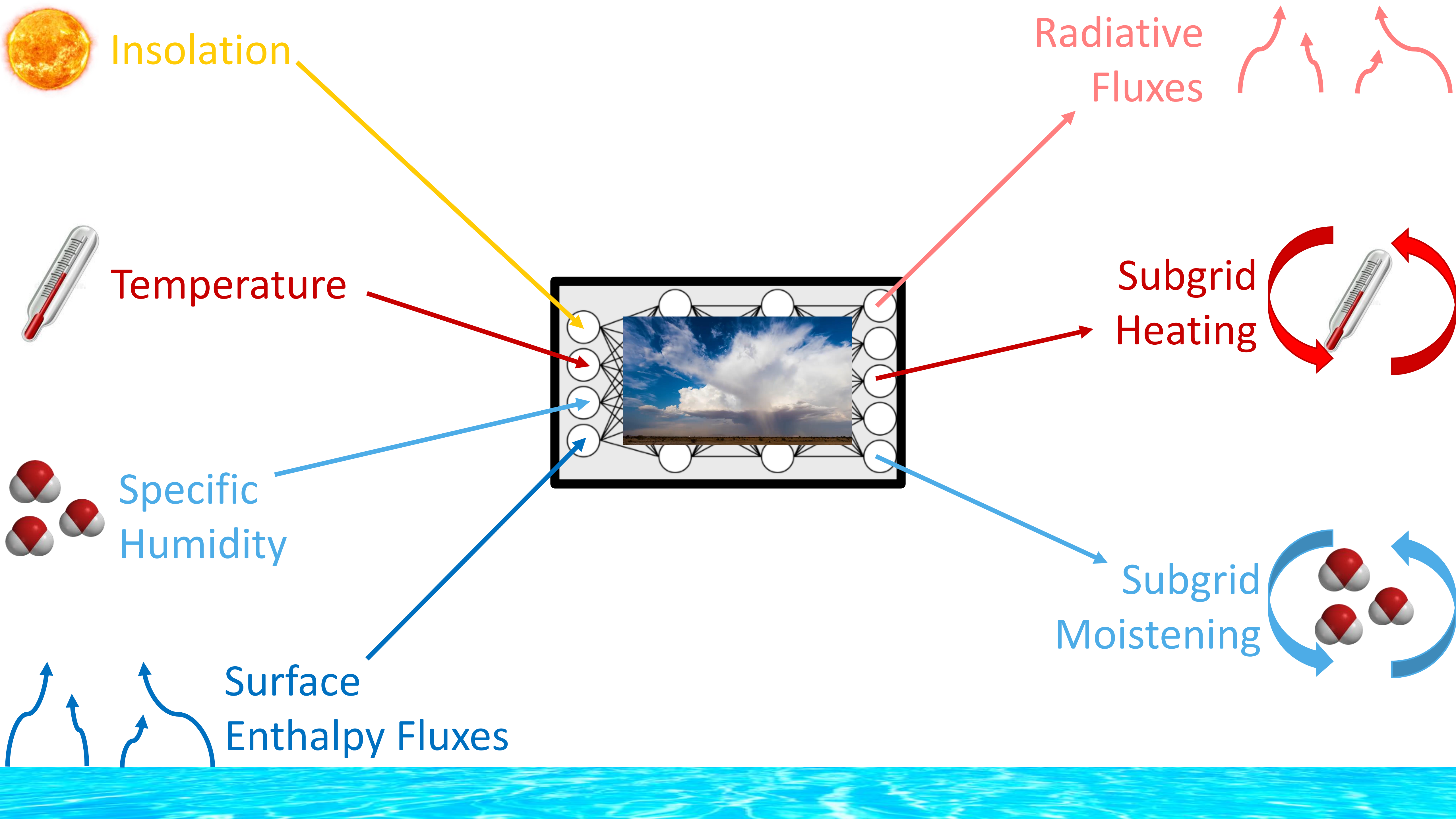
ML of Subgrid-Scale Thermodynamics



Neural Network:
20 times faster

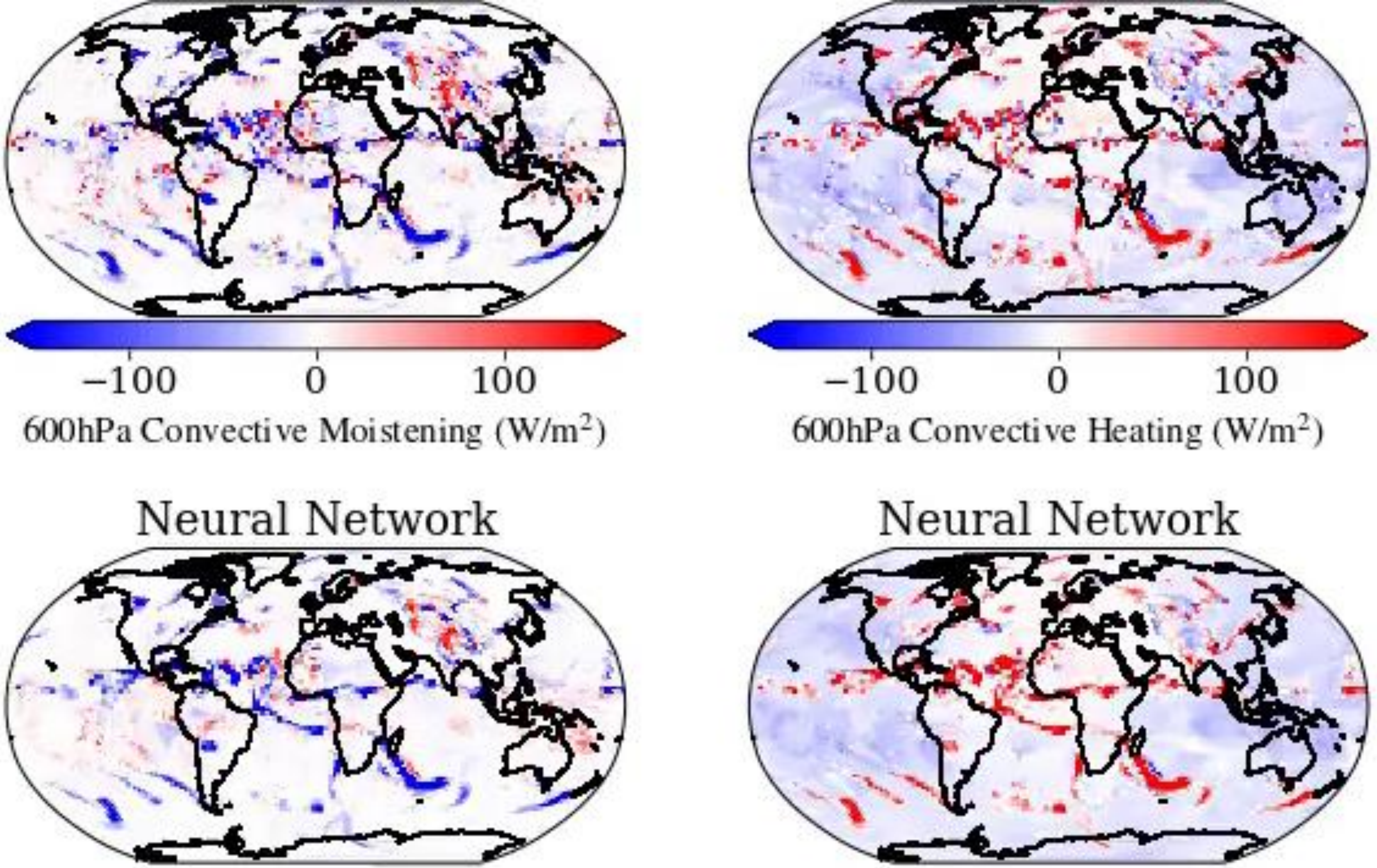
Setup : Super-Parameterized climate model, Aquaplanet & Earth-like
Year 1 for training (≈50M samples), Year 2 for validation/test

Image source: e3sm.org, Model source: Khairoutdinov et al. (2004)


Insolation
Temperature
Specific Humidity
Surface Enthalpy Fluxes
Radiative Fluxes
Subgrid Heating
Subgrid Moistening

Truth
Super-param. simulation

Prediction
NN
(offline)

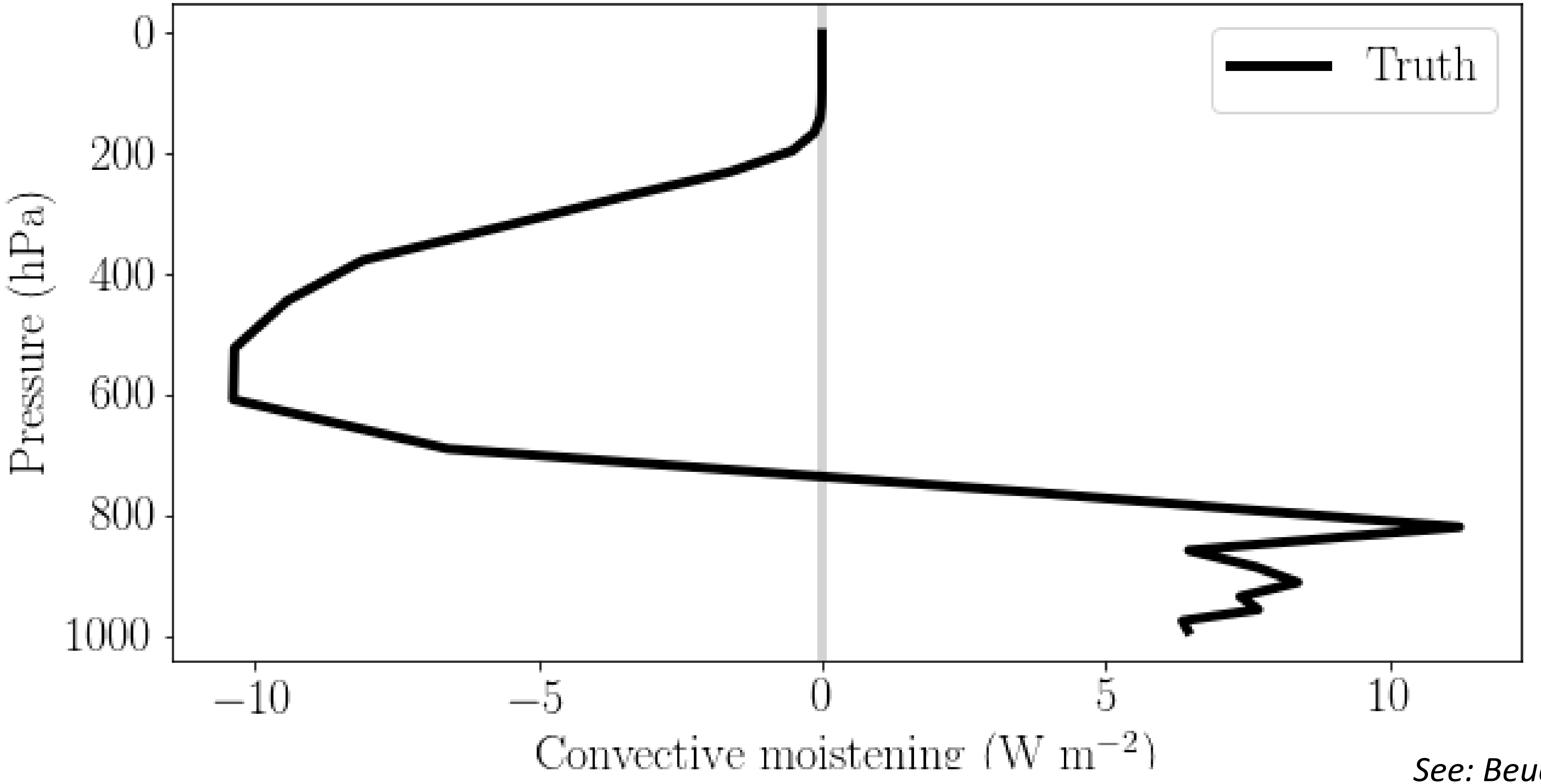


Source: Mooers, Pritchard, Beucler et al. (2021)

See: Rasp et al. (2018), Brenowitz et al. (2018,2019), Gentine et al. (2018), Yuval et al. (2020), Krasnopolsky et al. (2013)

Problem 1: ML algorithms fail to generalize

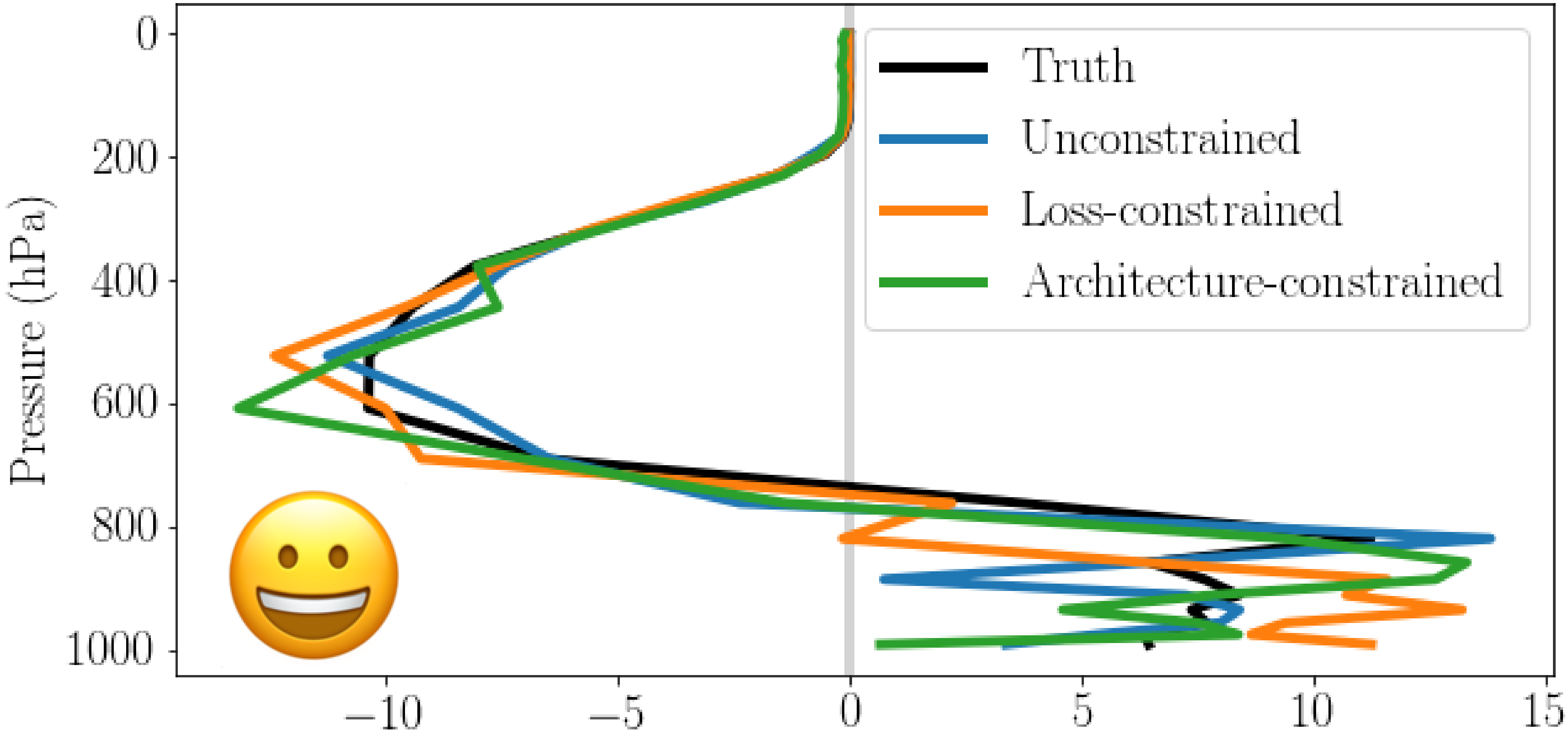
Daily-mean Tropical prediction in reference climate



See: Beucler et al. (2019)

Problem 1: ML algorithms fail to generalize

Daily-mean Tropical prediction in reference climate

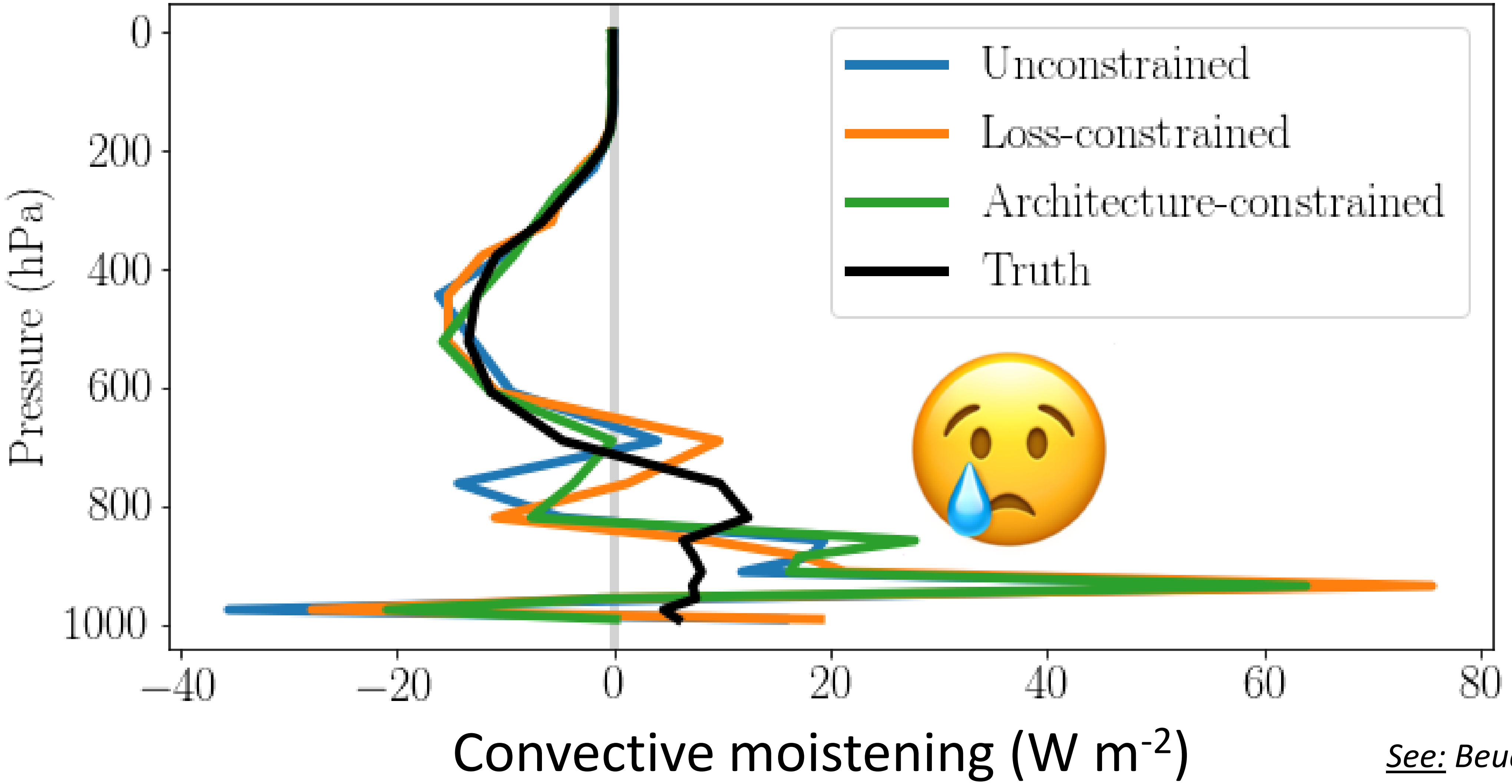


Convective moistening (W m^{-2})

See: *Beucler et al. (2019)*

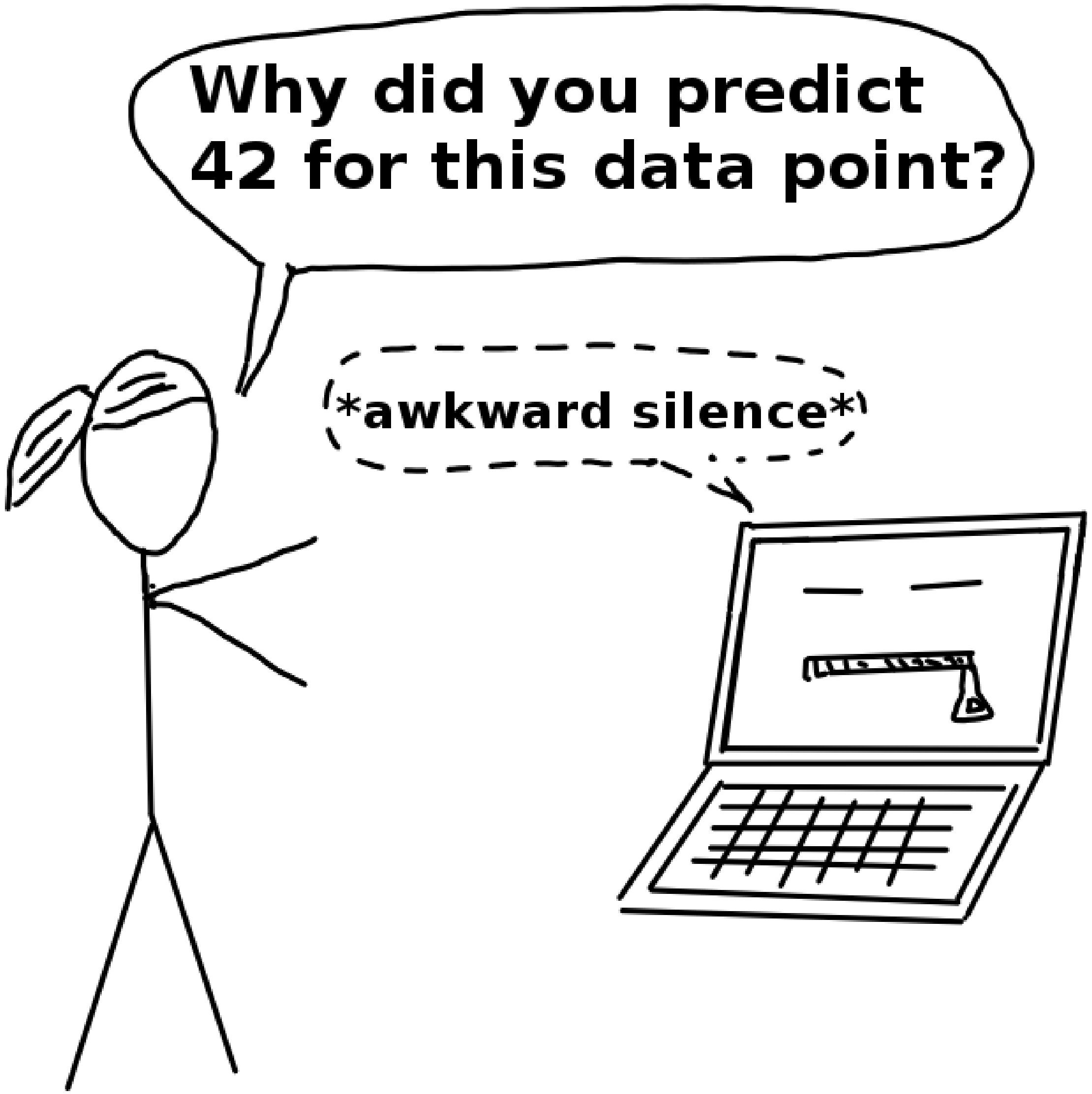
Problem 1: ML algorithms fail to generalize

Daily-mean Tropical prediction in (+4K) warming experiment



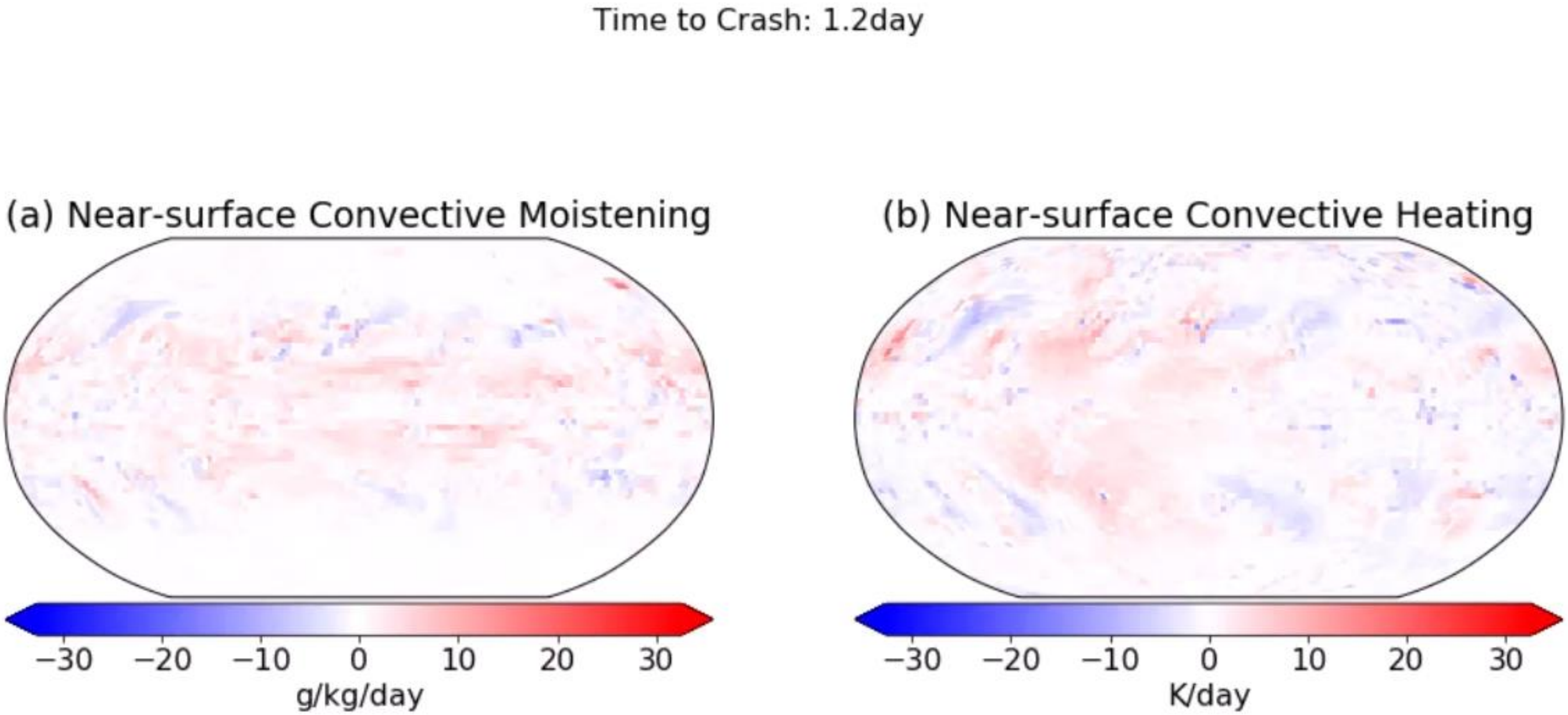
See: Beucler et al. (2019)

Problem 2: ML parametrizations are hard to interpret/trust



Source: *Interpretable Machine Learning, C. Molnar (2019)*

Problem 3: ML often unstable when coupled to dynamics



<u>See:</u> *Brenowitz, Beucler et al. (2020)*

Problem 1: ML algorithms fail to generalize

Problem 2: ML parametrization hard to interpret/trust

Problem 3: ML often unstable when coupled to dynam.

How can we design
data-driven, interpretable models of convection
generalize well & exhibit stable behavior?

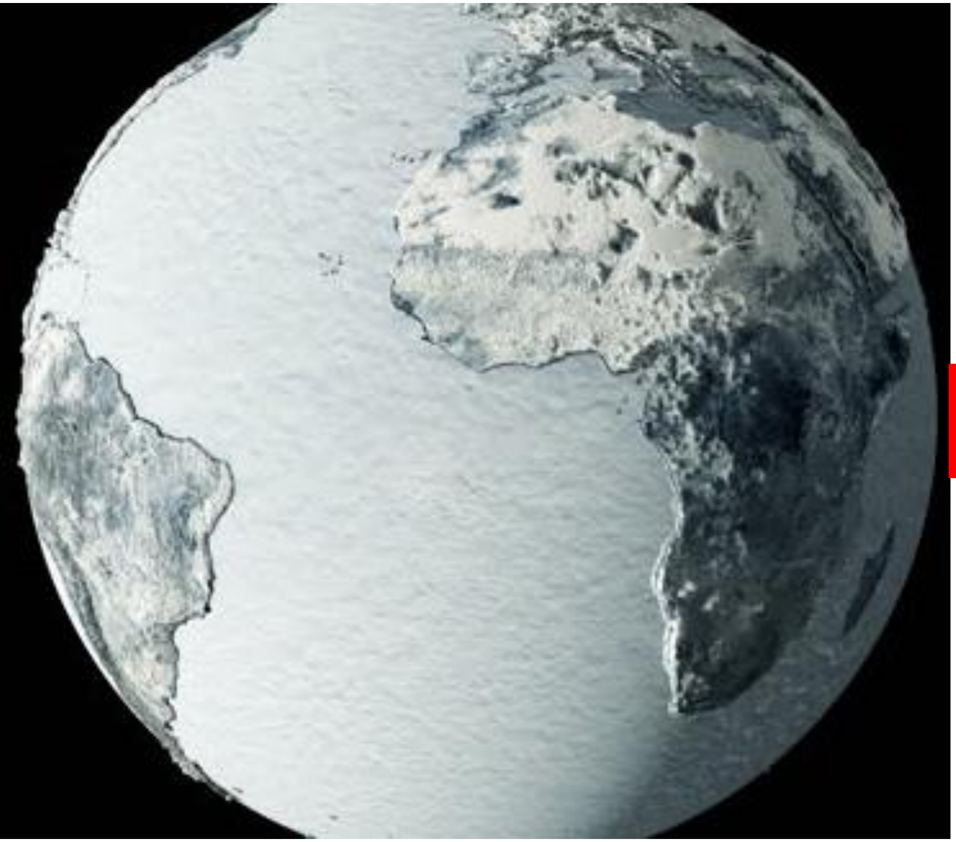
1) How to combine ML & physical knowledge?
2) How to build causally-consistent models?
3) Physically-consistent + Causally-consistent?

To test generalization: Break the model even more!

<u>Image source</u>: IT Biz Advisor

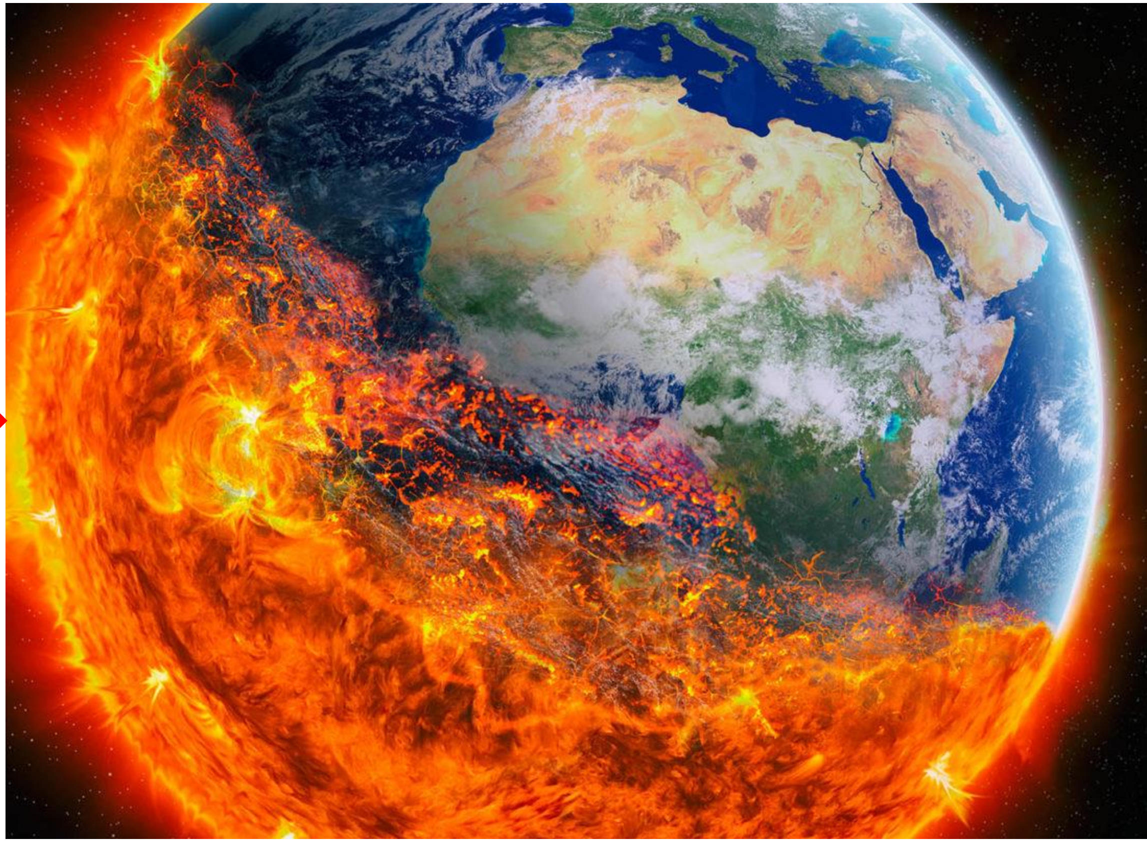
Generalization Experiment: Uniform +8K warming

Training and Validation on
cold aquaplanet simulation



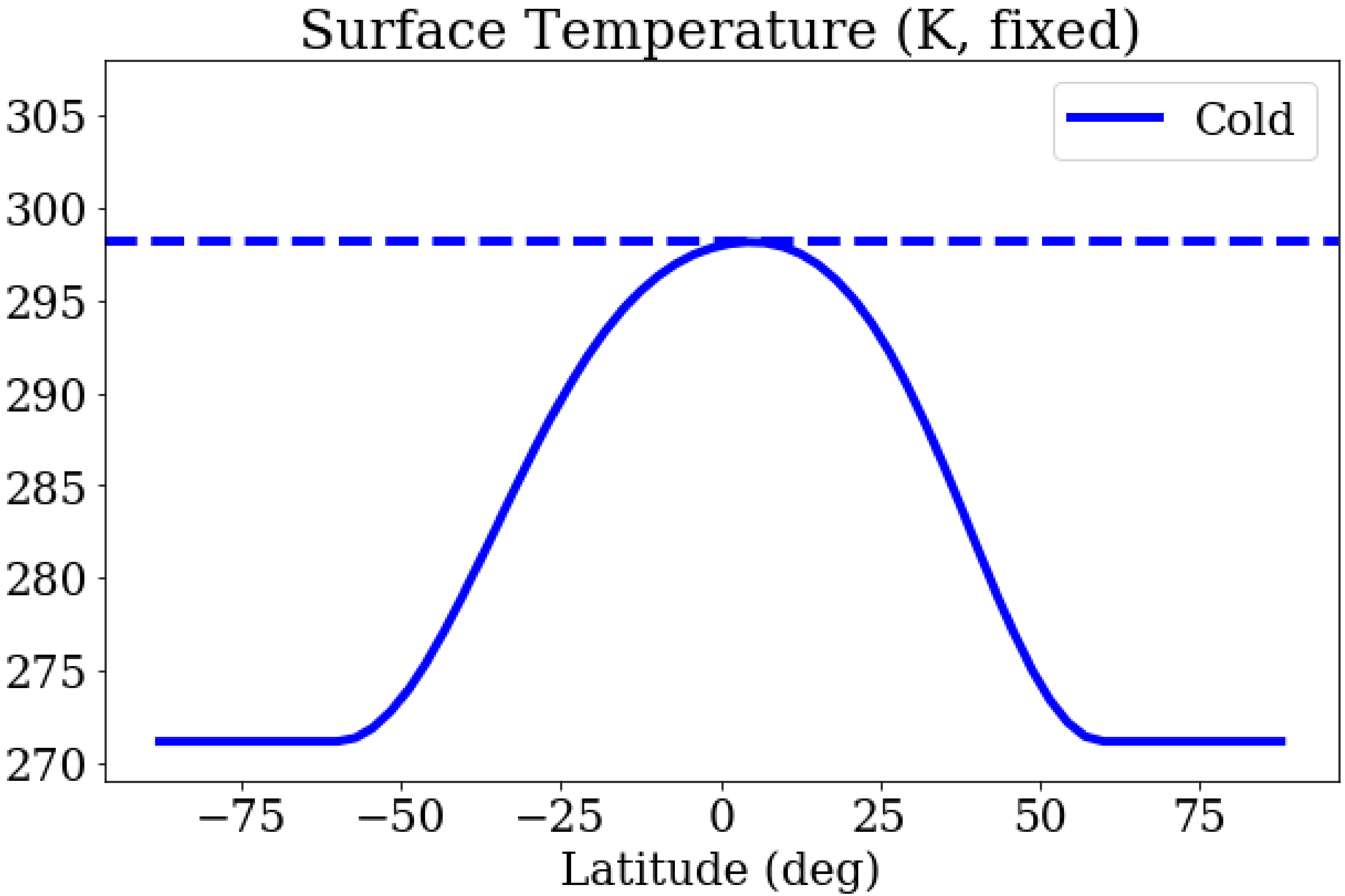
+8K

Test on warm aquaplanet simulation

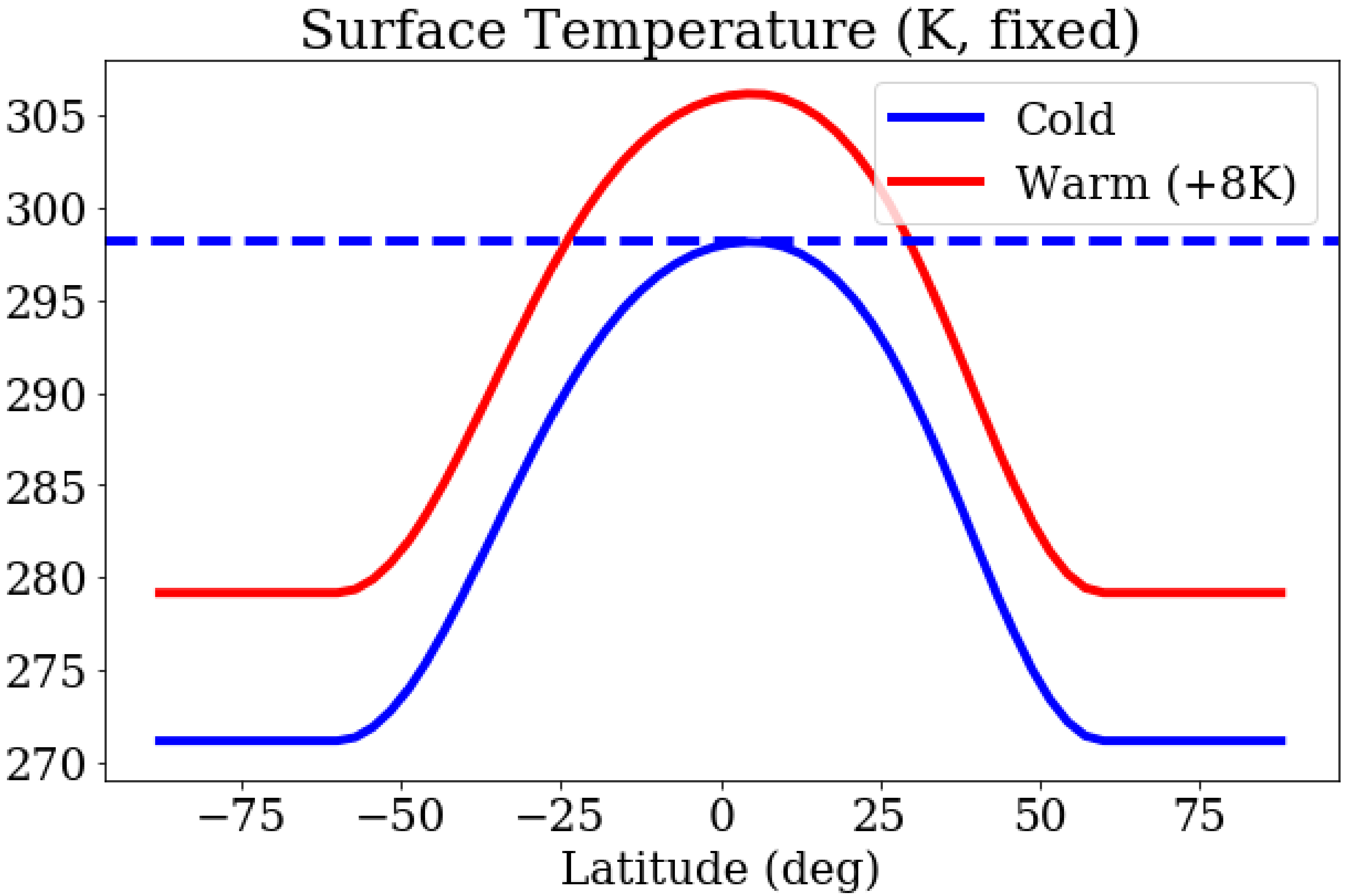


<u>Images</u>: Rashevskyi Viacheslav, Sebastien Decoret

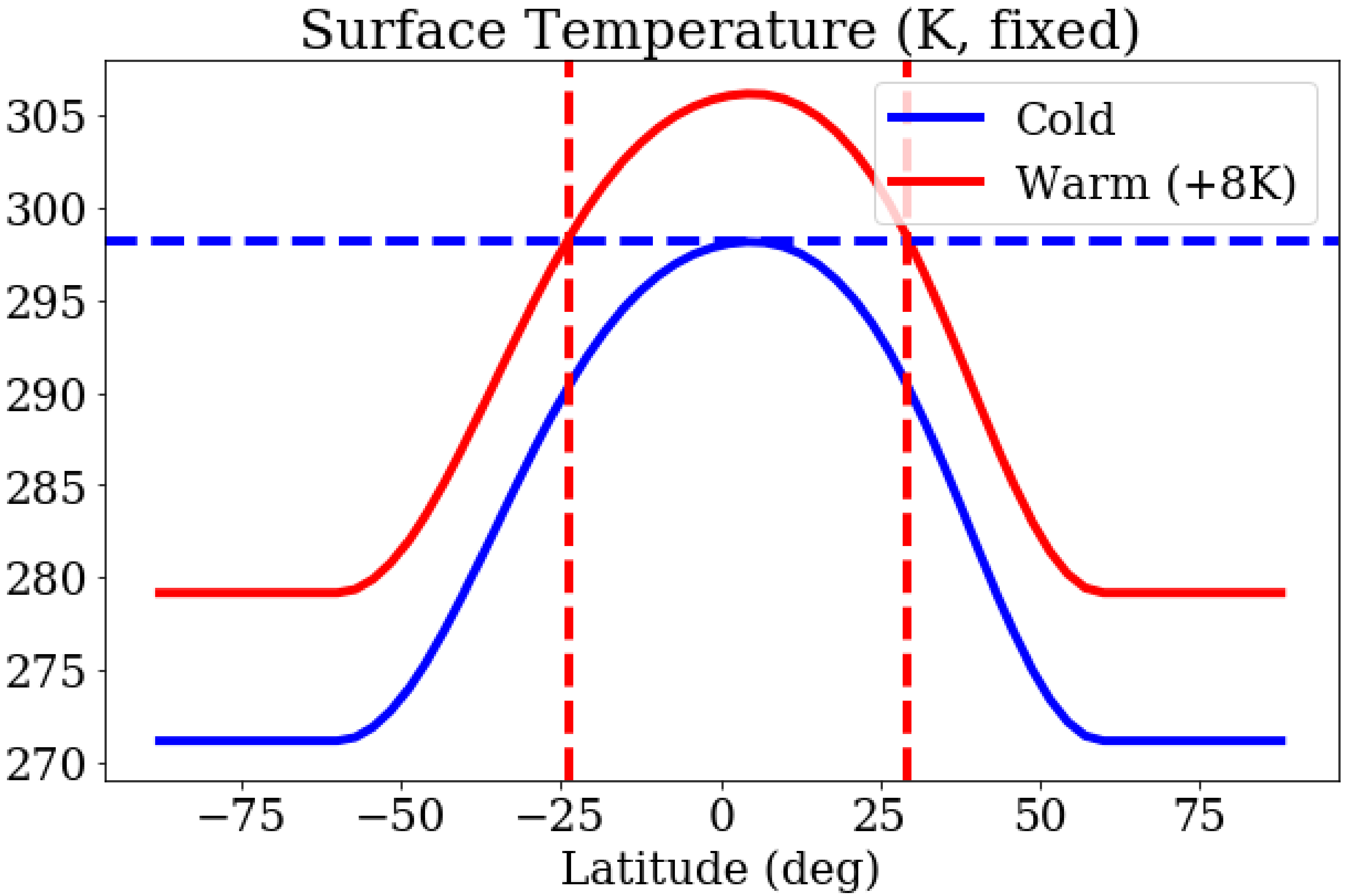
Generalization Experiment: Uniform +8K warming



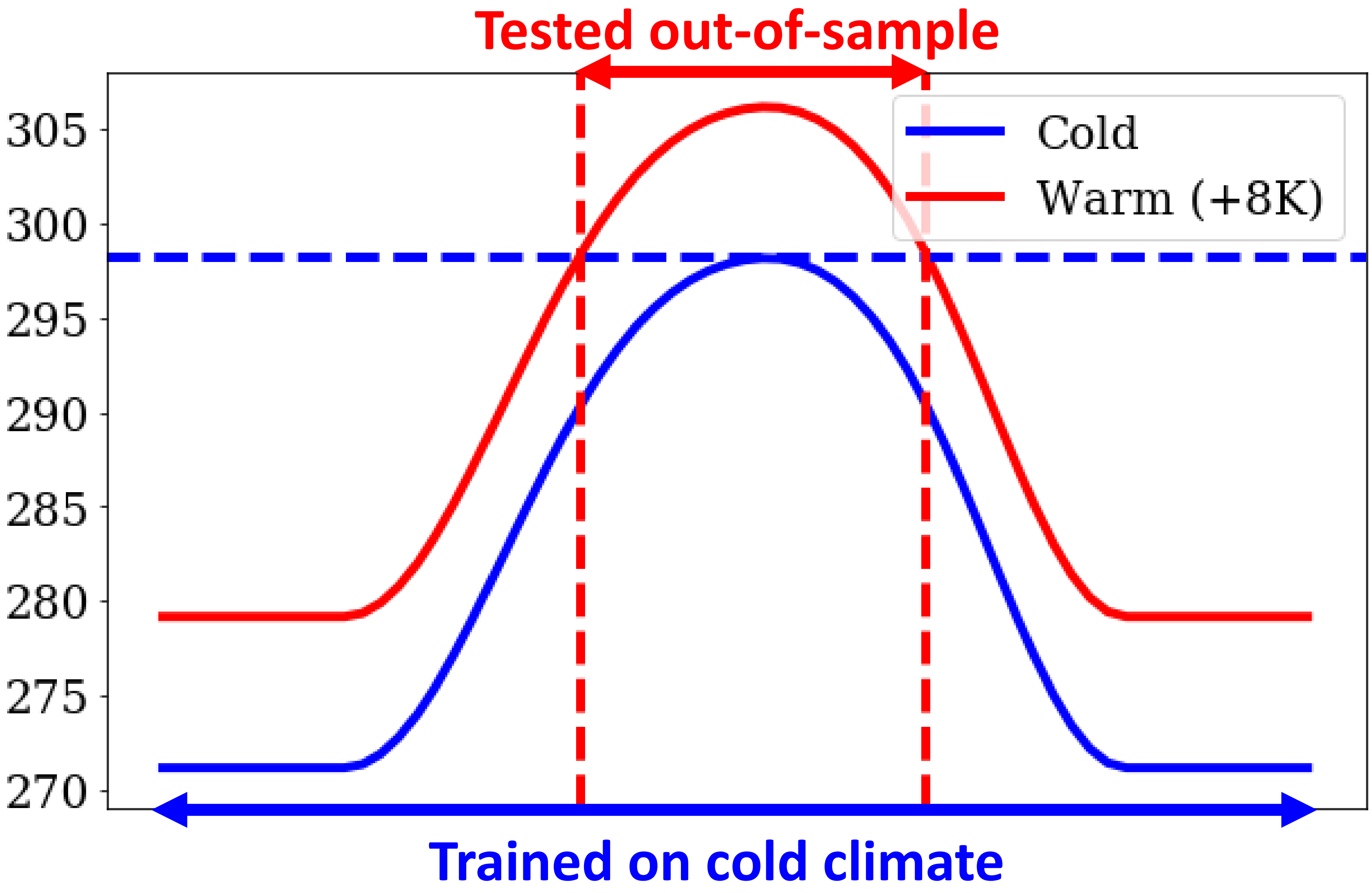
Generalization Experiment: Uniform +8K warming



Generalization Experiment: Uniform +8K warming

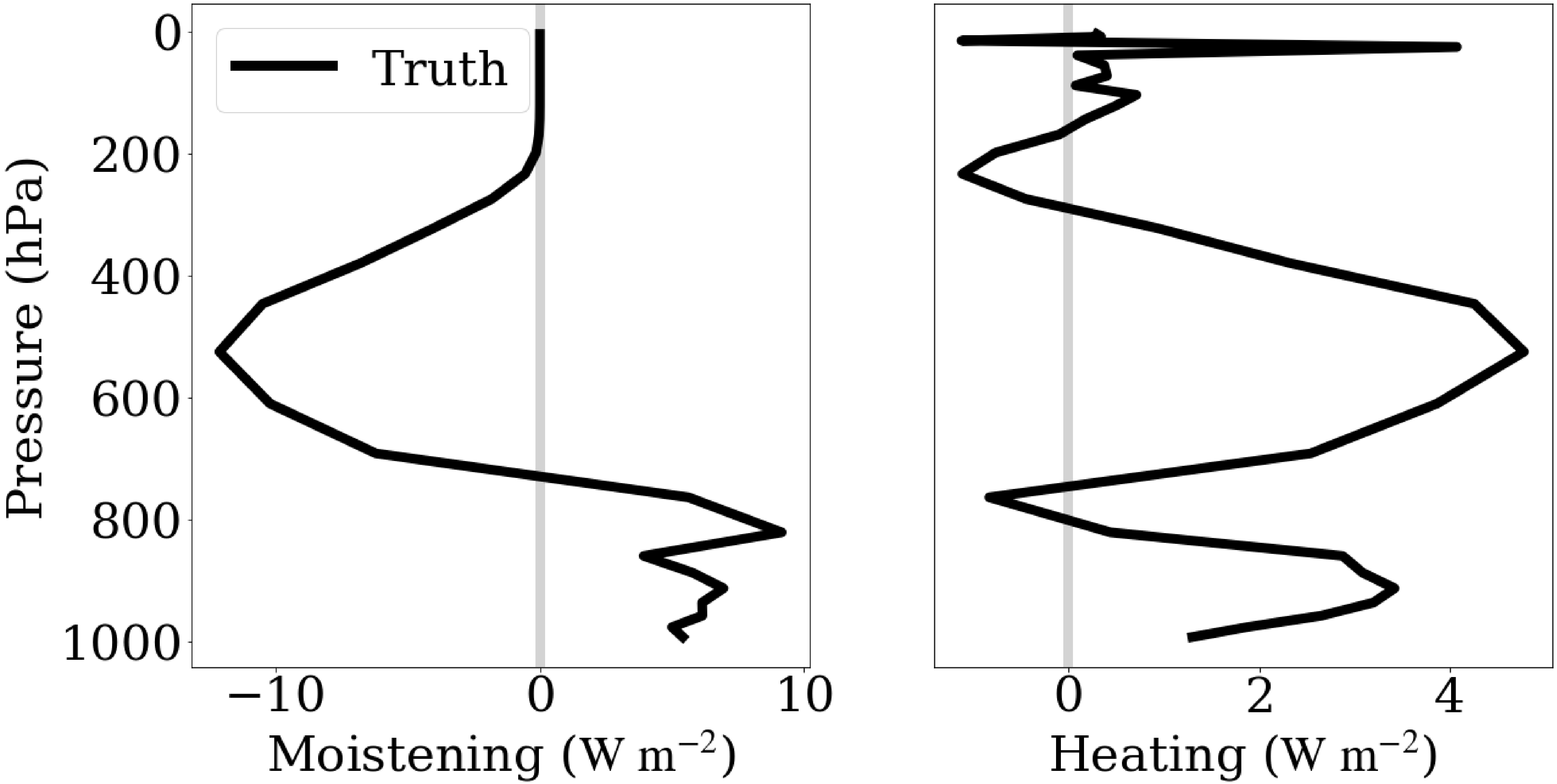


Generalization Experiment: Uniform +8K warming



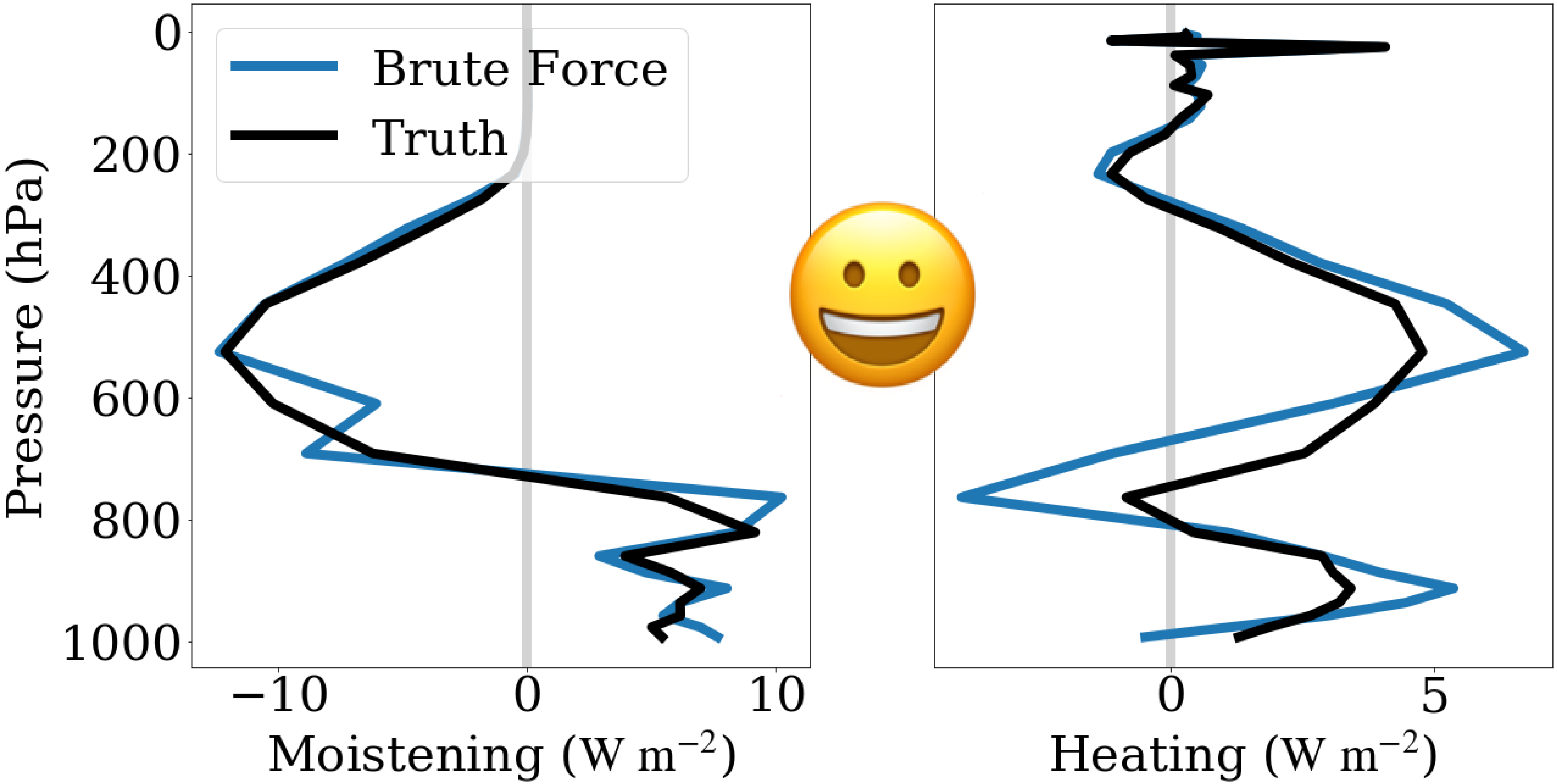
Problem 3: NNs fail to generalize to unseen climates

Daily-mean Tropical prediction in cold climate

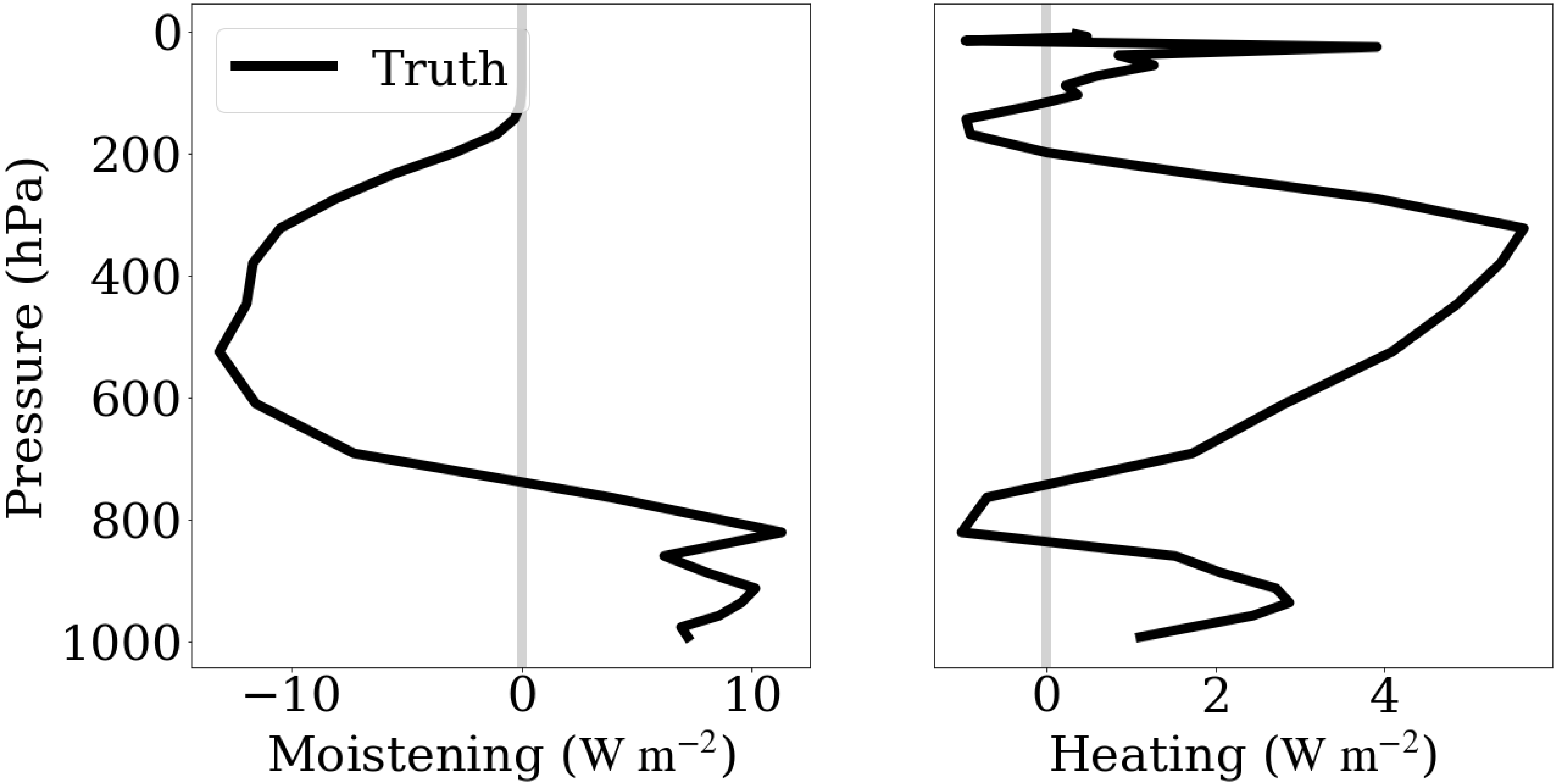


Problem 3: NNs fail to generalize to unseen climates

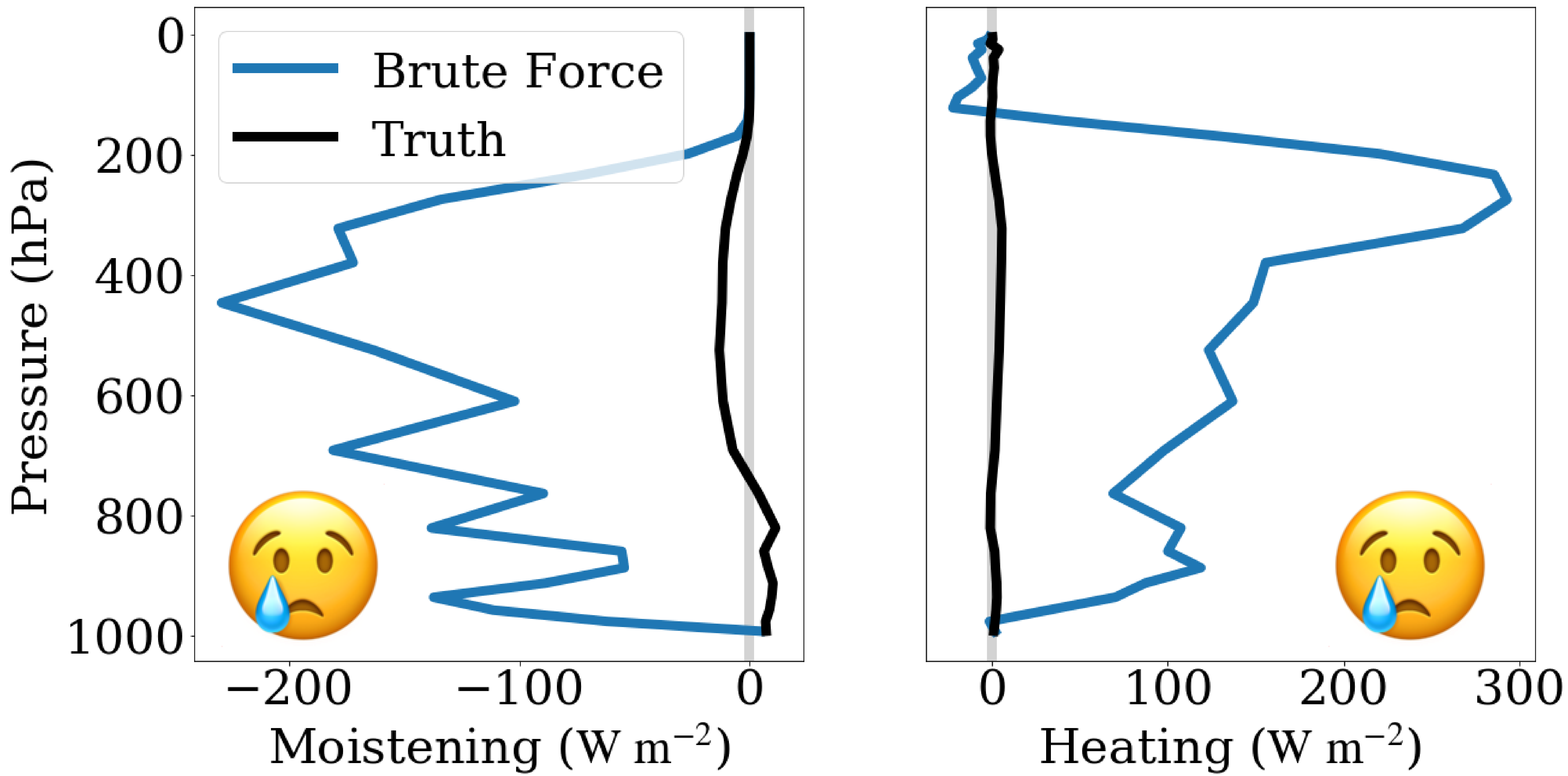
Daily-mean Tropical prediction in cold climate



Daily-mean Tropical prediction in warm climate



Daily-mean Tropical prediction in warm climate



Physically rescale the data to convert extrapolation into interpolation

$$\begin{bmatrix} \text{Specific humidity}\,(p) \\ \text{Temperature}\,(p) \\ \text{Surface Pressure} \\ \text{Solar Insolation} \\ \text{Latent Heat Flux} \\ \text{Sensible Heat Flux} \end{bmatrix} \overset{\text{NN}}{\longmapsto} \begin{bmatrix} \text{Subgrid moistening}\,(p) \\ \text{Subgrid heating}\,(p) \\ \text{Radiative fluxes} \end{bmatrix}$$

Brute Force: Not Climate-Invariant

Physically rescale the data to convert extrapolation into interpolation

Goal: Uncover **climate-invariant** mapping from climate to convection

$$\begin{bmatrix} \widetilde{\text{Specific humidity}}\,(\tilde{p}) \\ \widetilde{\text{Temperature}}\,(\tilde{p}) \\ \widetilde{\text{Surface Pressure}} \\ \widetilde{\text{Solar Insolation}} \\ \widetilde{\text{Latent Heat Flux}} \\ \widetilde{\text{Sensible Heat Flux}} \end{bmatrix} \overset{\widetilde{\text{NN}}}{\longmapsto} \begin{bmatrix} \widetilde{\text{Subgrid moistening}}\,(\tilde{p}) \\ \widetilde{\text{Subgrid heating}}\,(\tilde{p}) \\ \widetilde{\text{Radiative fluxes}} \end{bmatrix}$$

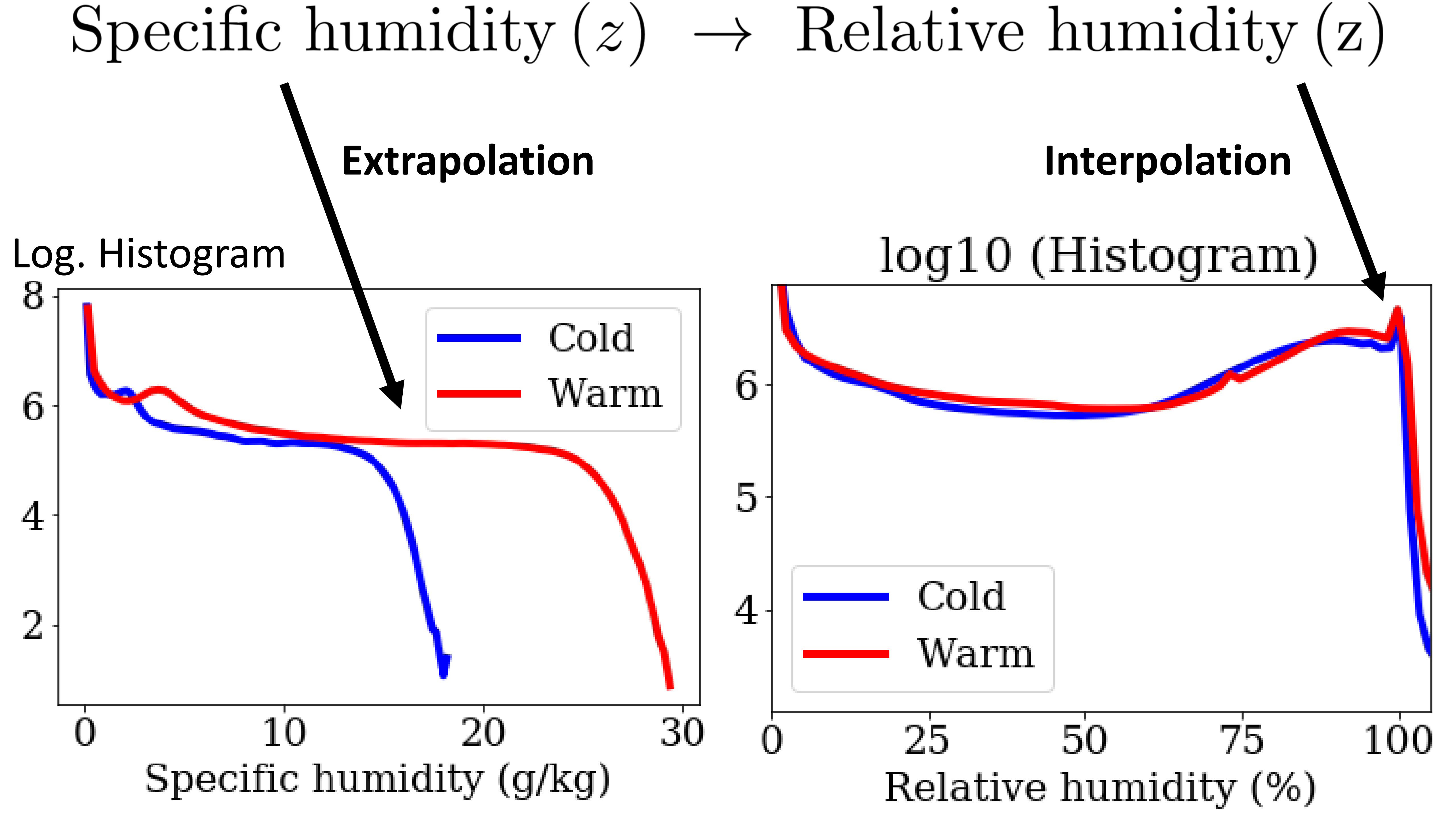
Goal: Climate-Invariant

Physically rescale the data to convert extrapolation into interpolation

<u>Goal</u>: Uncover **climate-invariant** mapping from climate to convection

$$\begin{bmatrix} \widetilde{\text{Specific humidity}}\,(\tilde{p}) \\ \widetilde{\text{Temperature}}\,(\tilde{p}) \\ \widetilde{\text{Surface Pressure}} \\ \widetilde{\text{Solar Insolation}} \\ \widetilde{\text{Latent Heat Flux}} \\ \widetilde{\text{Sensible Heat Flux}} \end{bmatrix} \overset{\widetilde{\text{NN}}}{\longmapsto} \begin{bmatrix} \widetilde{\text{Subgrid moistening}}\,(\tilde{p}) \\ \widetilde{\text{Subgrid heating}}\,(\tilde{p}) \\ \widetilde{\text{Radiative fluxes}} \end{bmatrix}$$

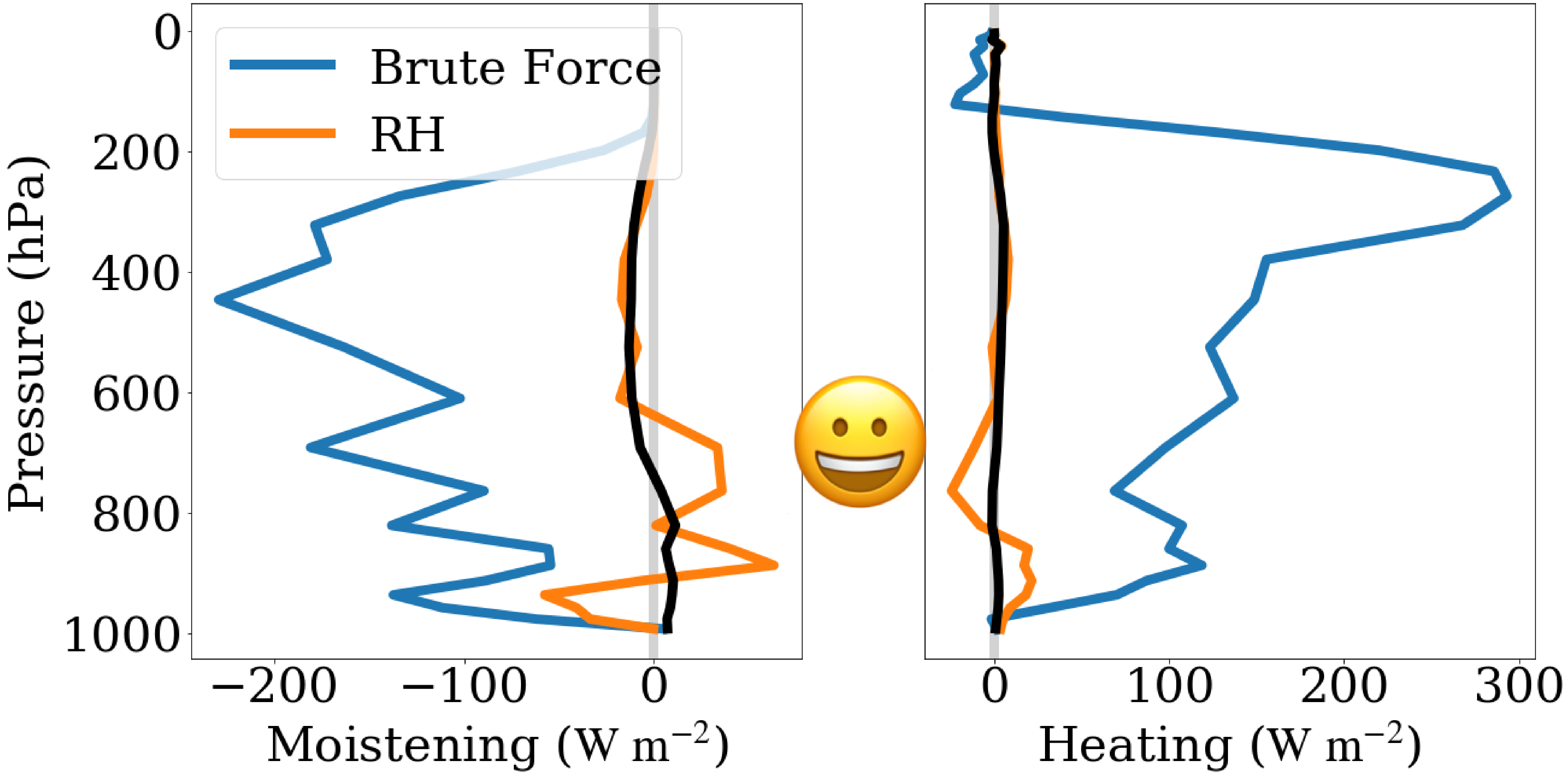
How to choose the physical rescaling?

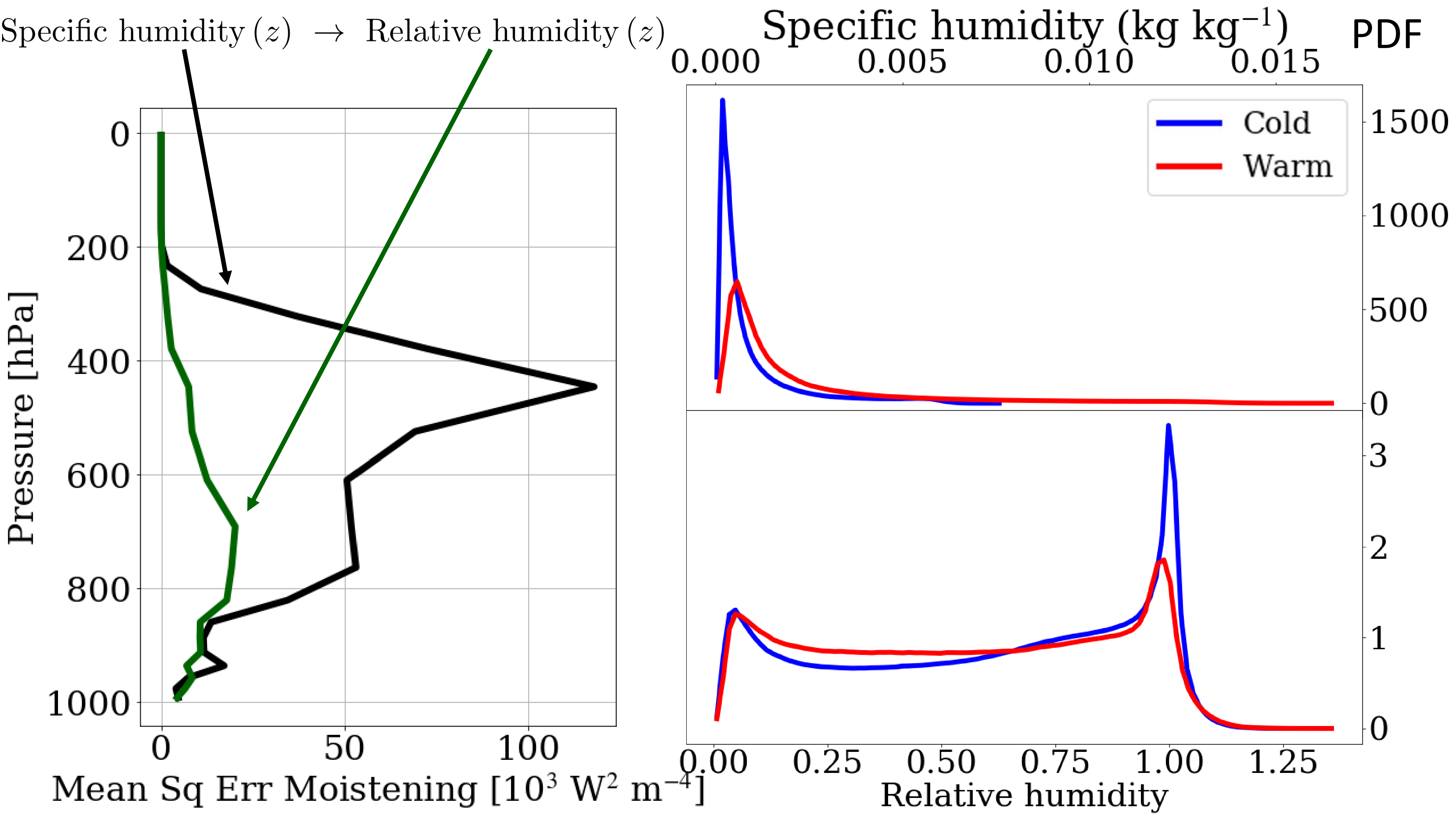


Specific humidity (z) → Relative humidity (z)
Extrapolation
Interpolation
Log. Histogram
8
6
4
2
0
10
20
30
Specific humidity (g/kg)
Cold
Warm
log10 (Histogram)
6
5
4
0
25
50
75
100
Relative humidity (%)
Cold
Warm

Specific humidity (z) $\rightarrow$ Relative humidity (z)

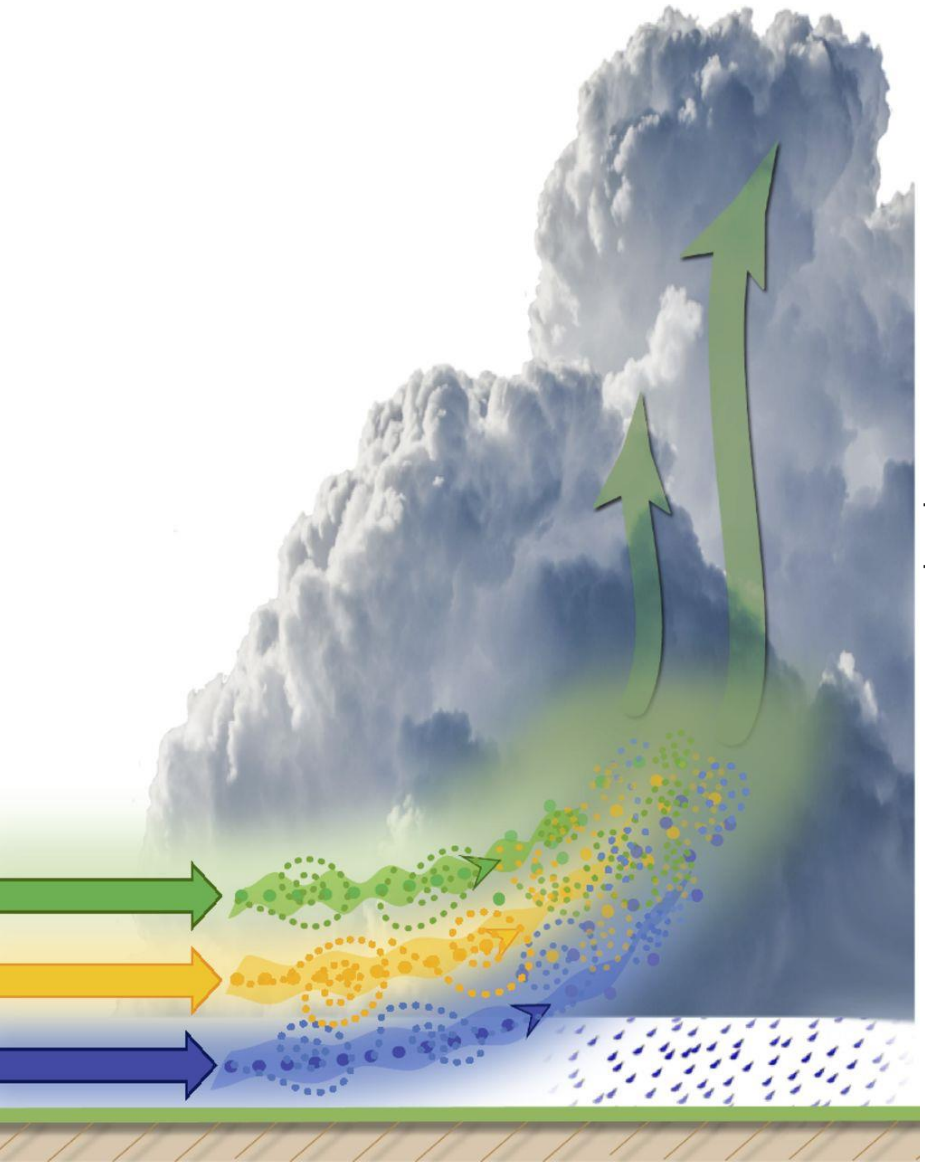
Generalization improves dramatically!





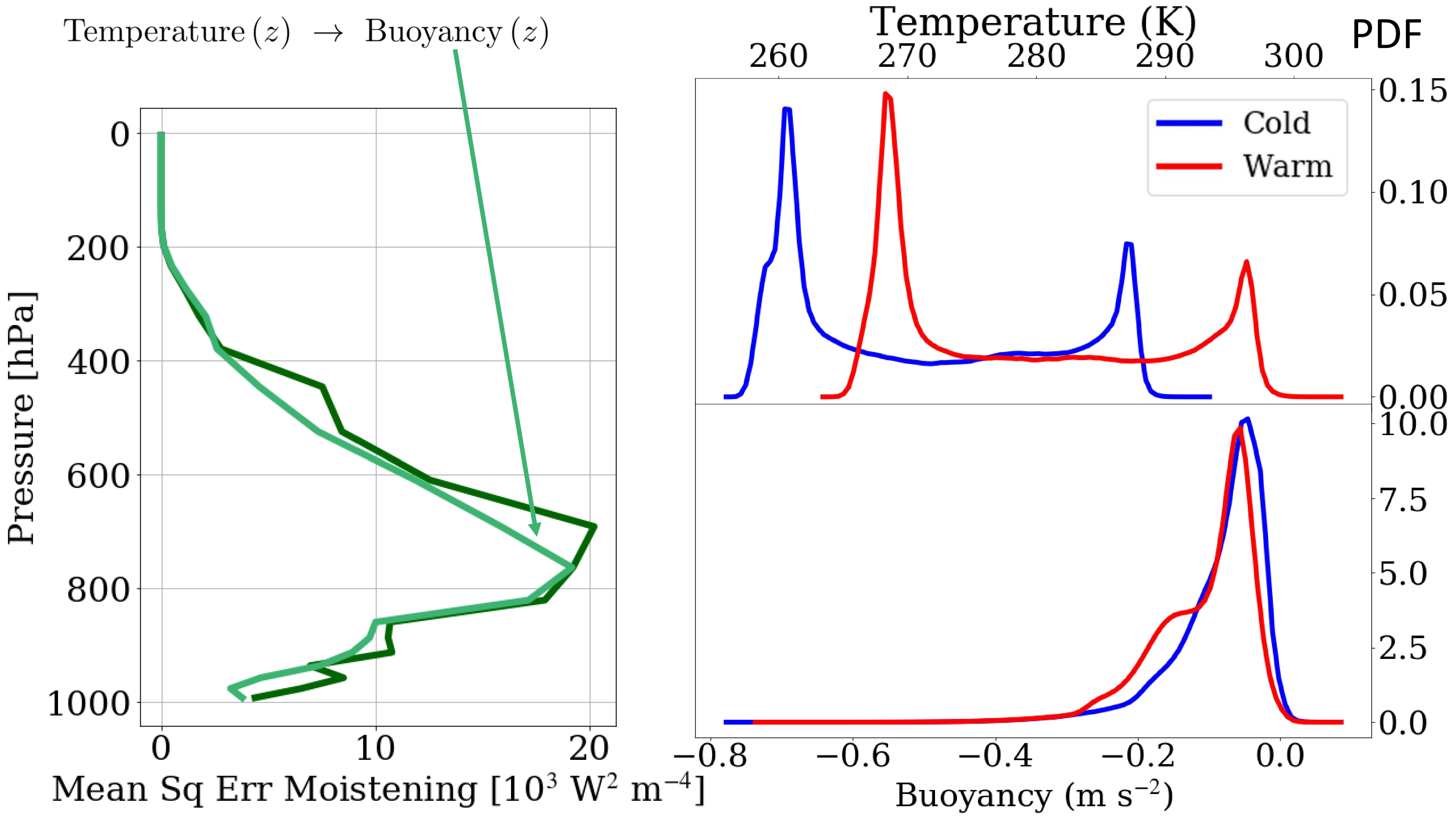
Specific humidity (z) → Relative humidity (z)
Pressure [hPa]
Mean Sq Err Moistening [10^3 W^2 m^{-4}]
Specific humidity (kg kg^{-1})
PDF
Cold
Warm
Relative humidity

Observations suggest a strong relationship between buoyancy & moist convection across scales

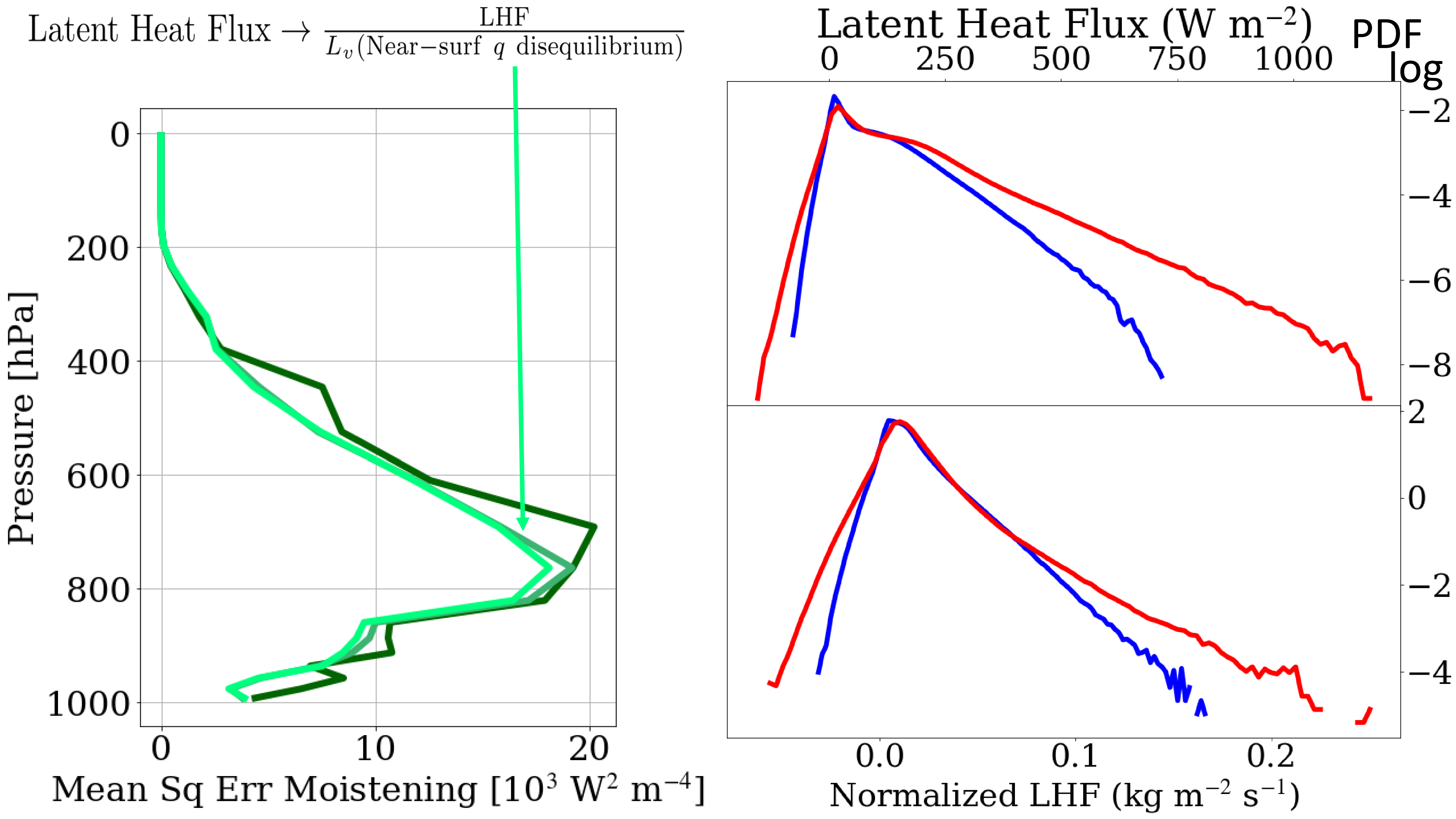


$$\text{Buoyancy}\,(z) \stackrel{\text{def}}{=} g \times \frac{\text{Temp parcel} - \text{Temp}(z)}{\text{Temp}(z)}$$

<u>See:</u> Schiro et al. (2018), Ahmed & Neelin (2018), Ahmed et al. (2020)

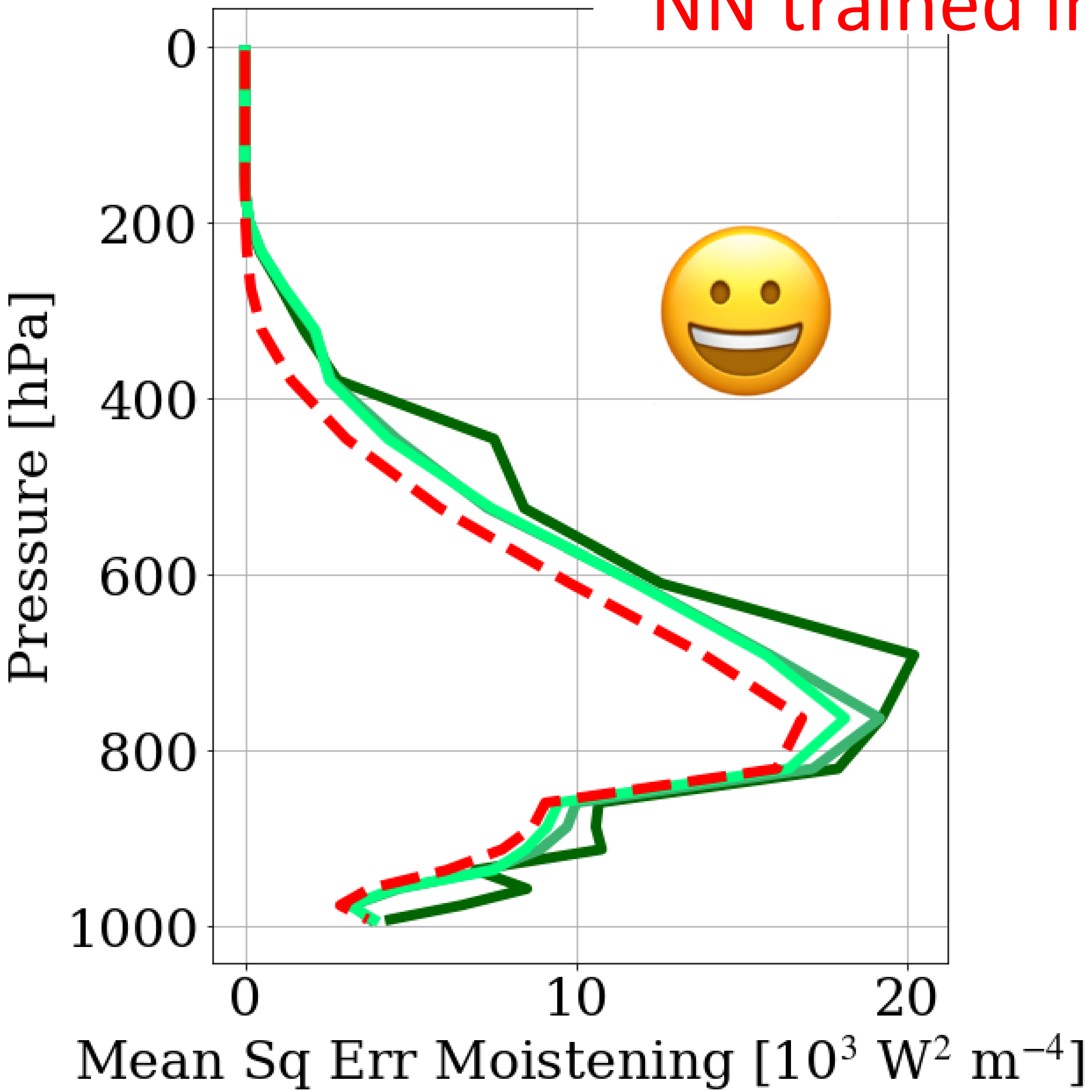


Temperature (z) → Buoyancy (z)
Pressure [hPa]
0
200
400
600
800
1000
0
10
20
Mean Sq Err Moistening [10^3 W^2 m^{-4}]
Temperature (K)
260
270
280
290
300
PDF
0.15
0.10
0.05
0.00
Cold
Warm
10.0
7.5
5.0
2.5
0.0
−0.8
−0.6
−0.4
−0.2
0.0
Buoyancy (m s^{-2})

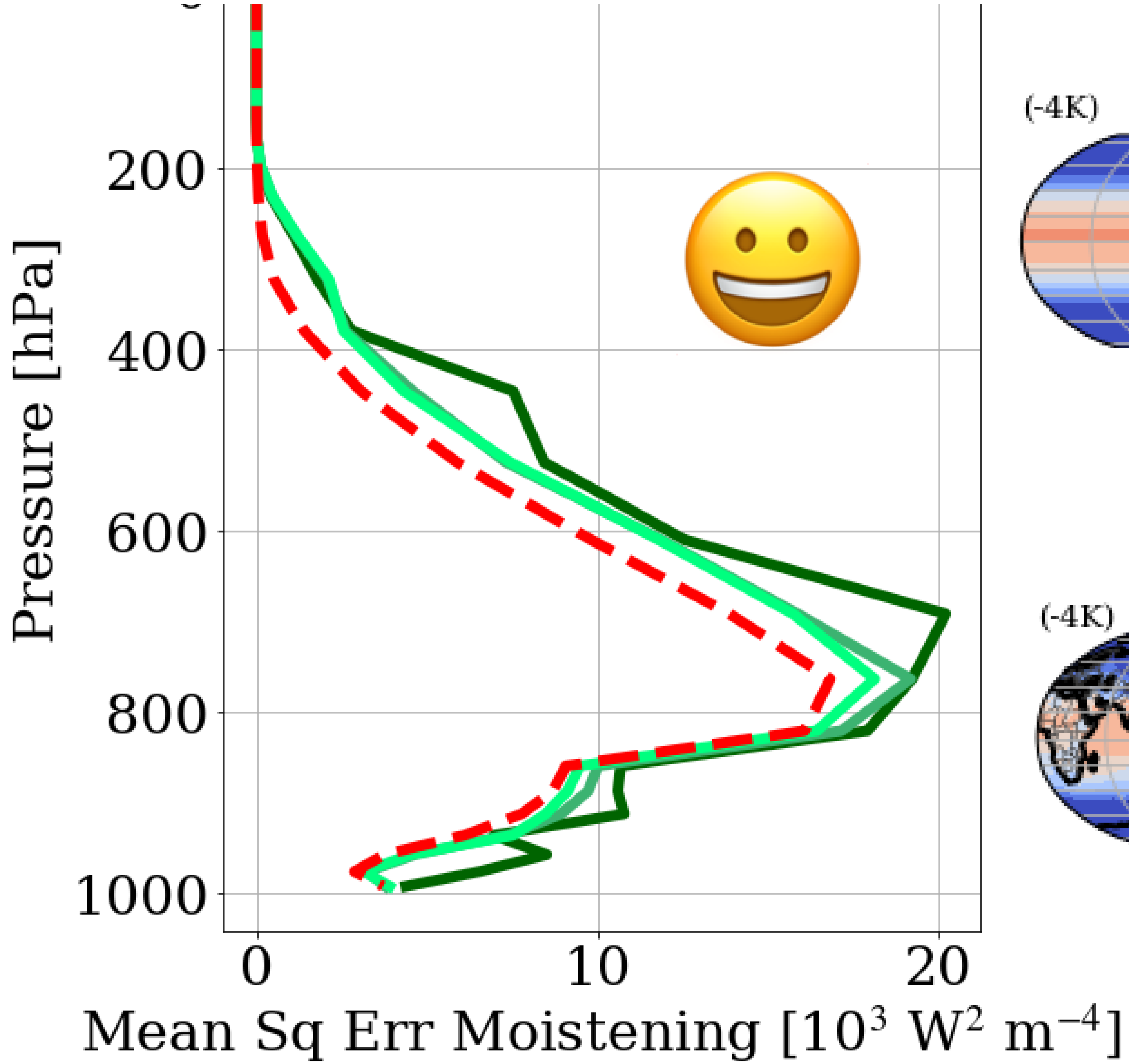


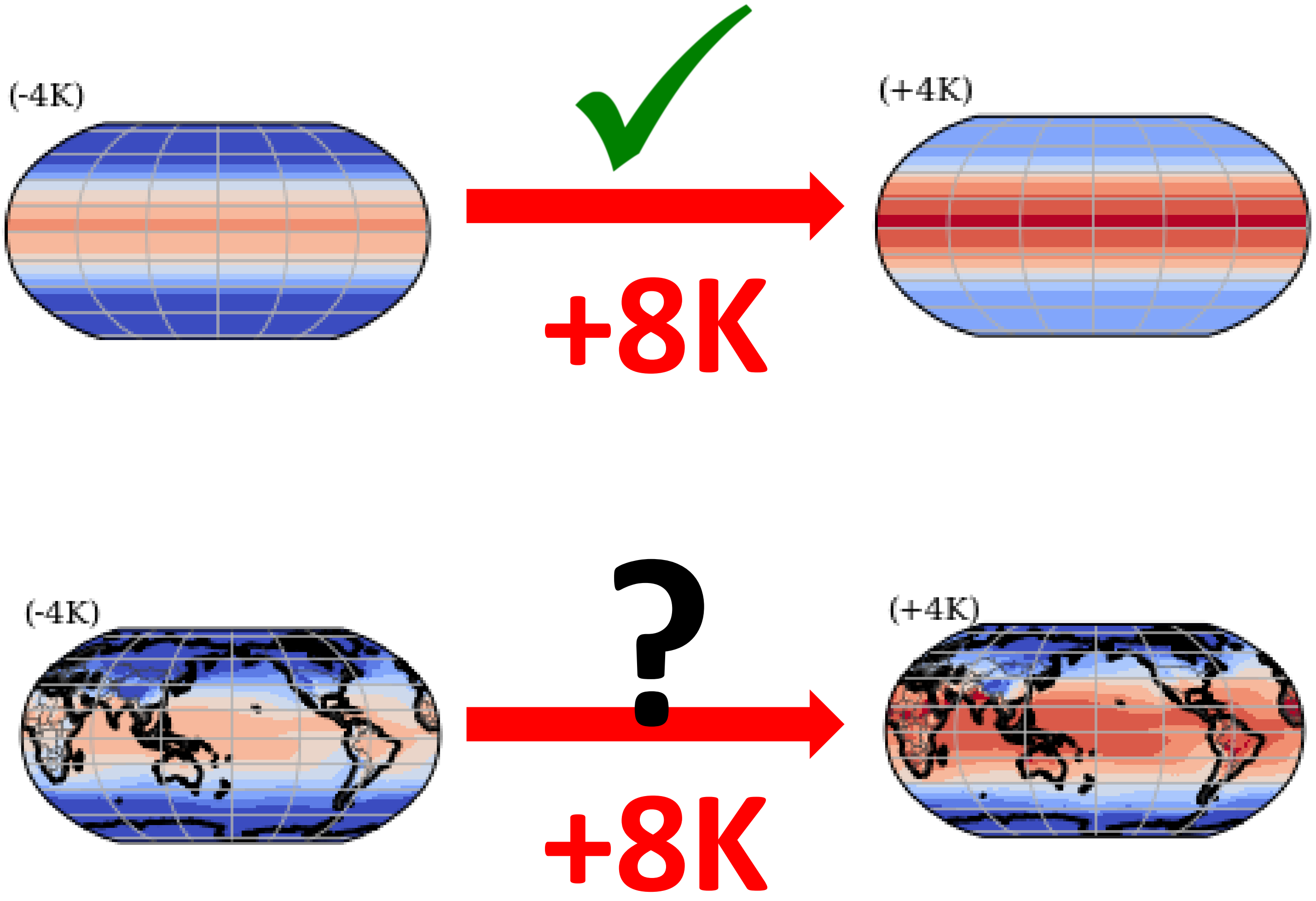
Latent Heat Flux → $\frac{\text{LHF}}{L_v(\text{Near-surf } q \text{ disequilibrium})}$
Pressure [hPa]
0
200
400
600
800
1000
0
10
20
Mean Sq Err Moistening [10^3 W^2 m^{-4}]
Latent Heat Flux (W m^{-2})
0
250
500
750
1000
PDF
log
−2
−4
−6
−8
2
0
−2
−4
0.0
0.1
0.2
Normalized LHF (kg m^{-2} s^{-1})

Climate-Invariant NNs generalization error close to NN trained in warm climate



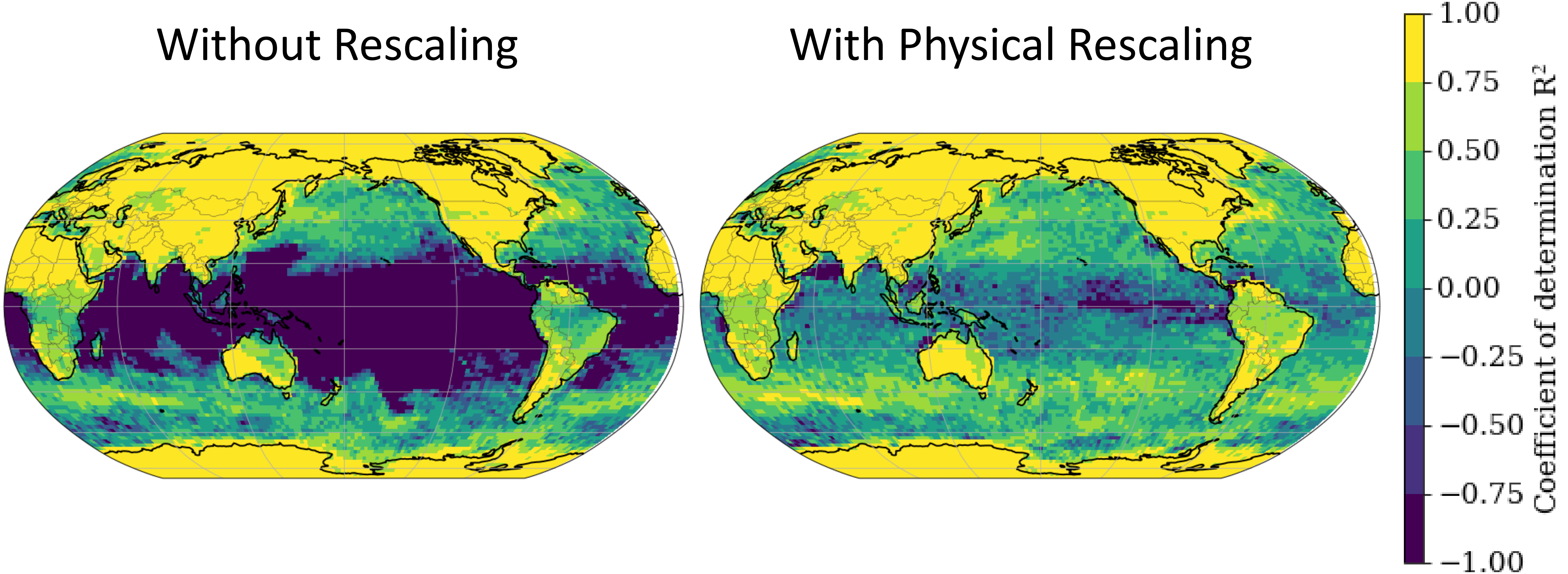
Problem 3: Physically Rescaling Inputs allows NNs to generalize from cold to warm climate





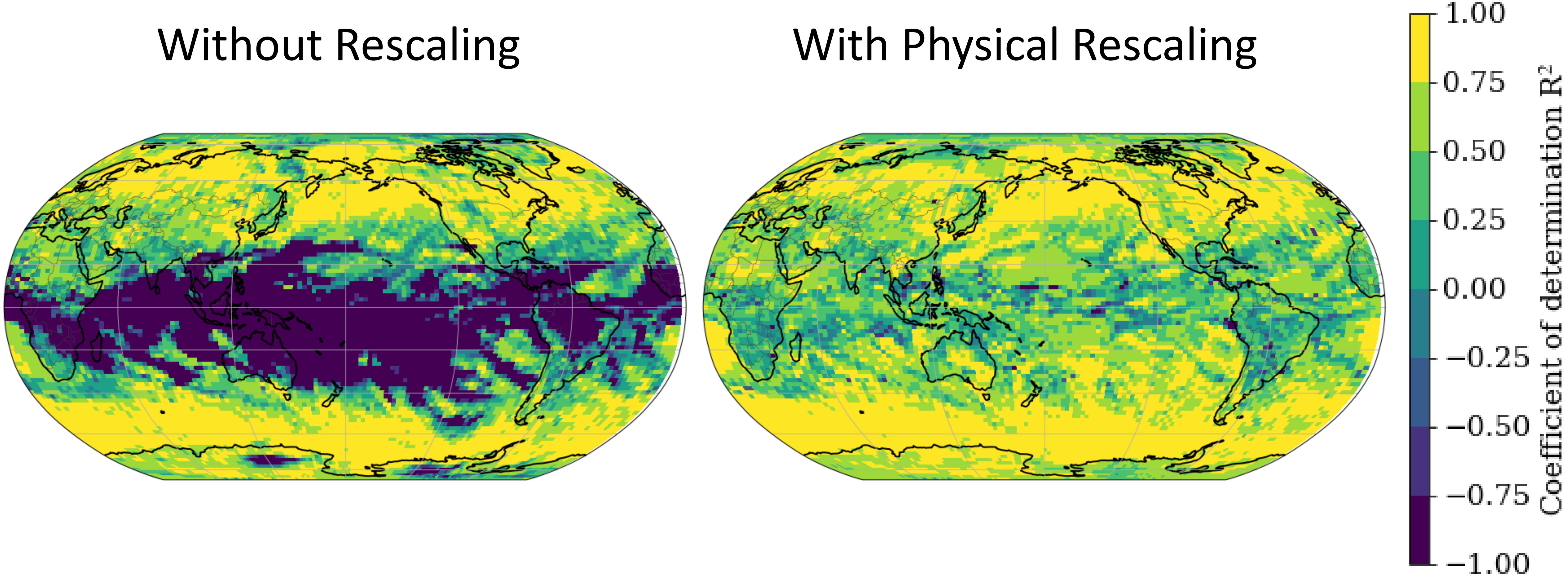
<u>See:</u> Beucler et al. (Under review)

Physically-Rescaled Neural Networks Generalize Better Across Climates **in Earth-like configurations**



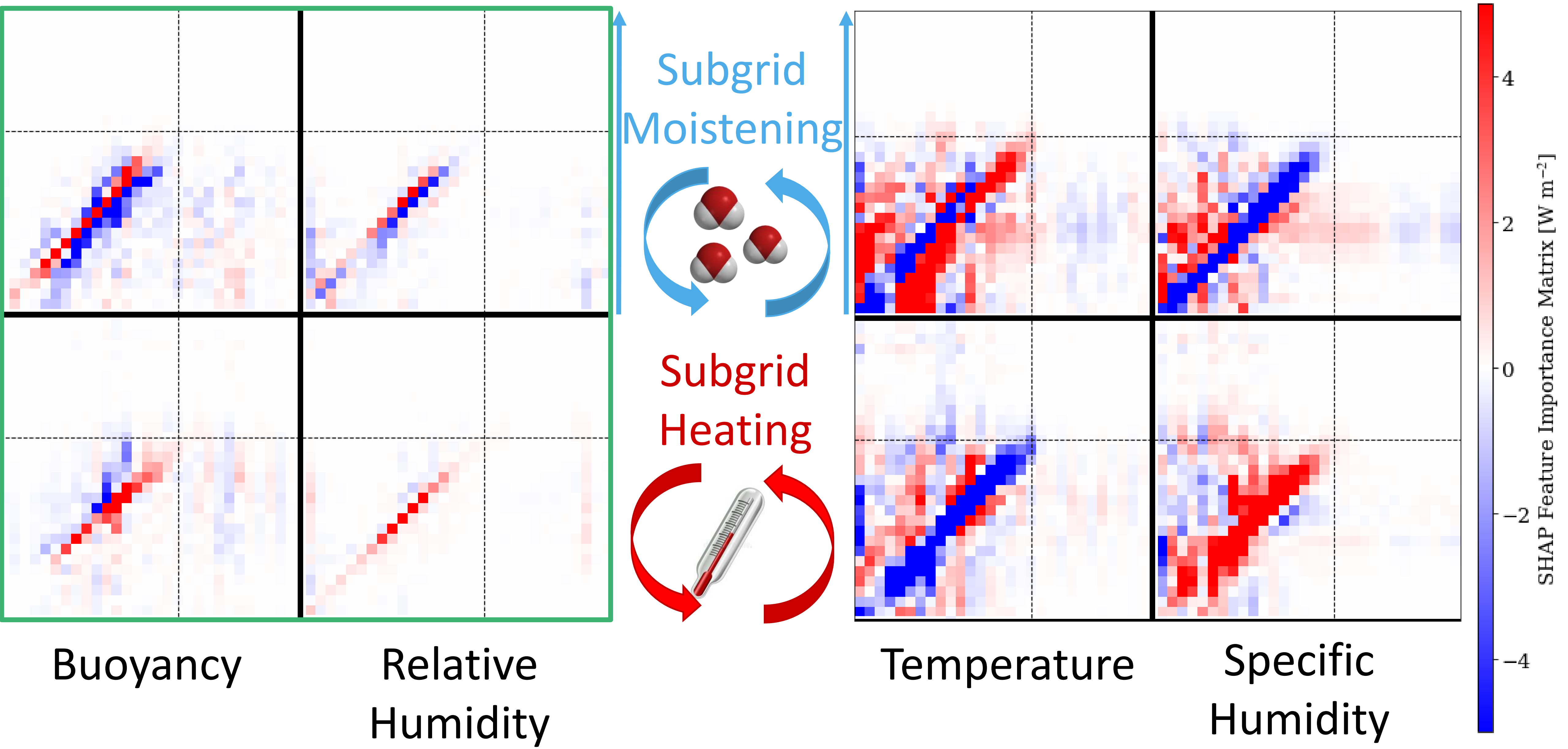
Near-Surface Subgrid Heating

Physically-Rescaled Neural Networks Generalize Better Across Climates **in Earth-like configurations**

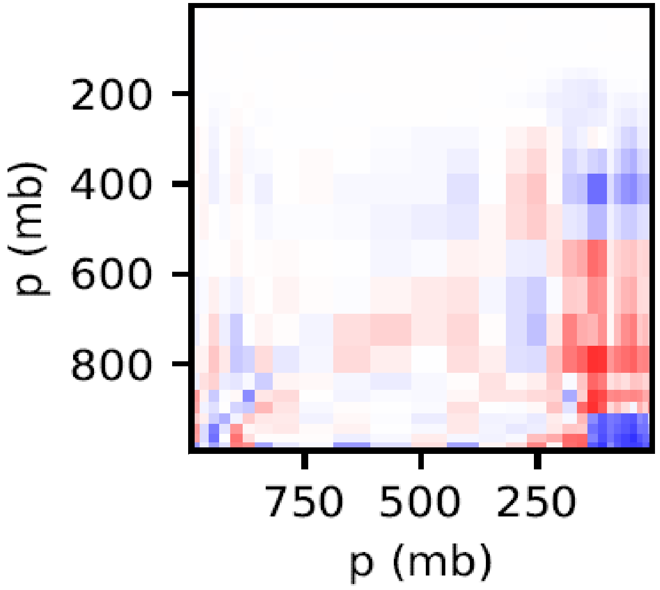


Mid-Tropospheric Subgrid Heating

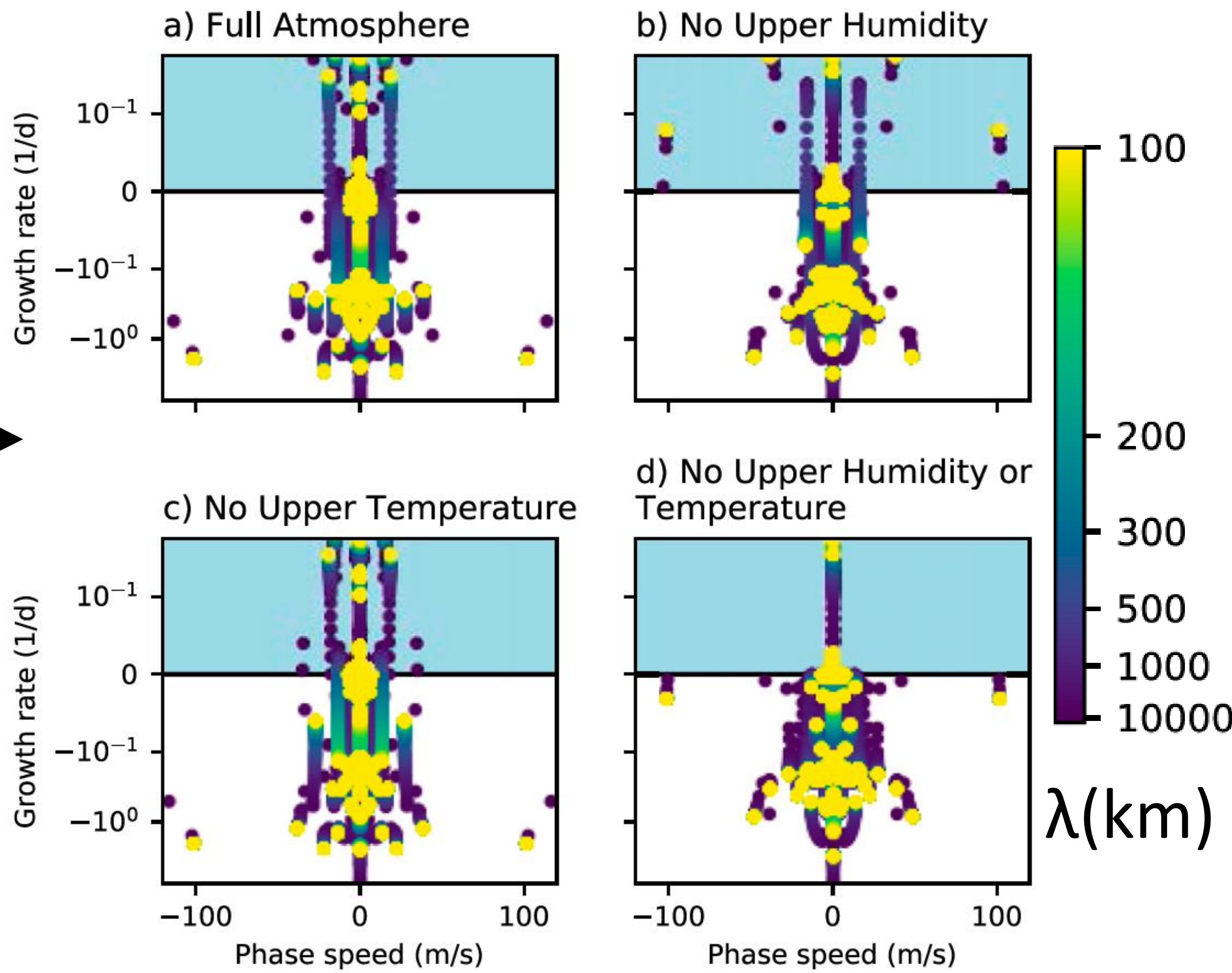
Climate-invariant NNs more local than Brute-Force NNs



Motivation for causal ML: Eliminating spurious link with stratospheric q & T eliminates instability

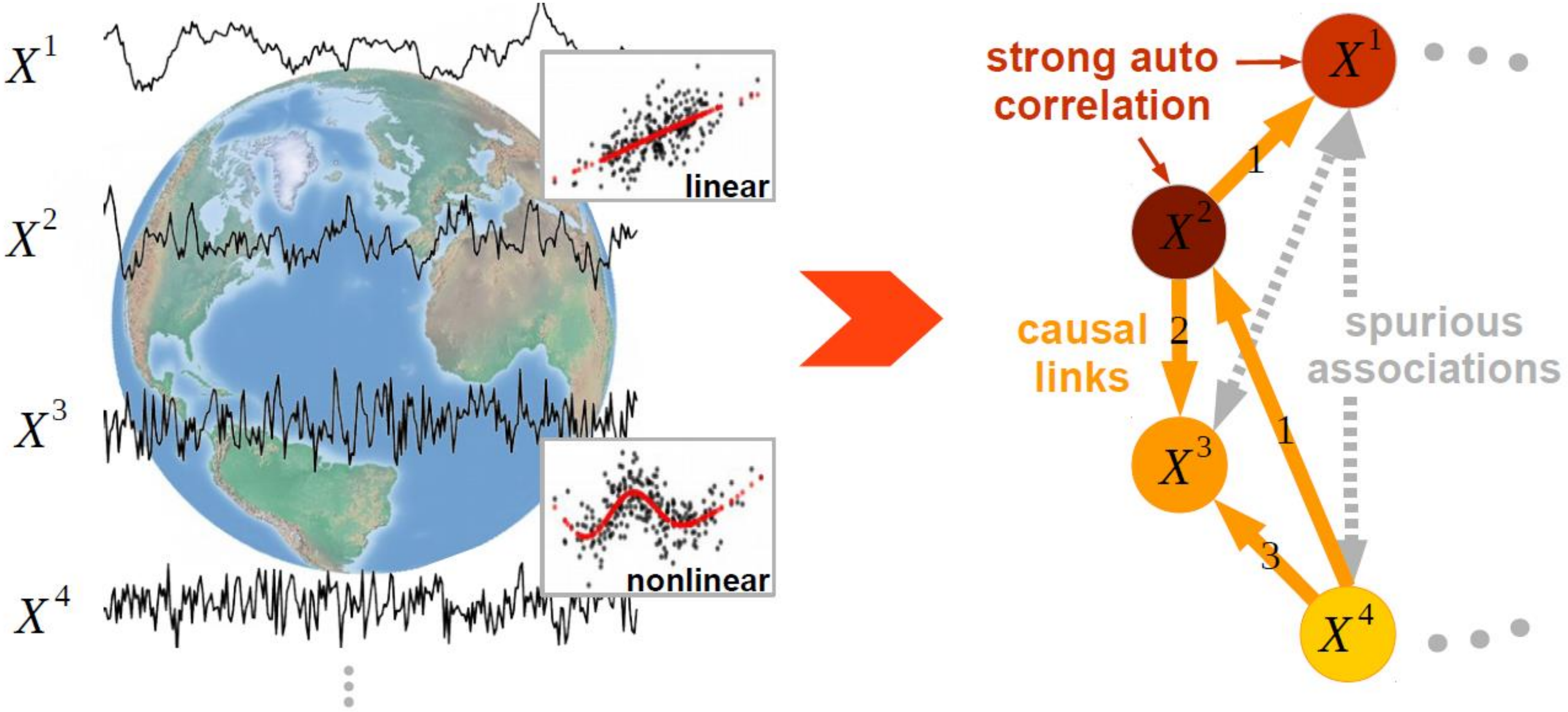


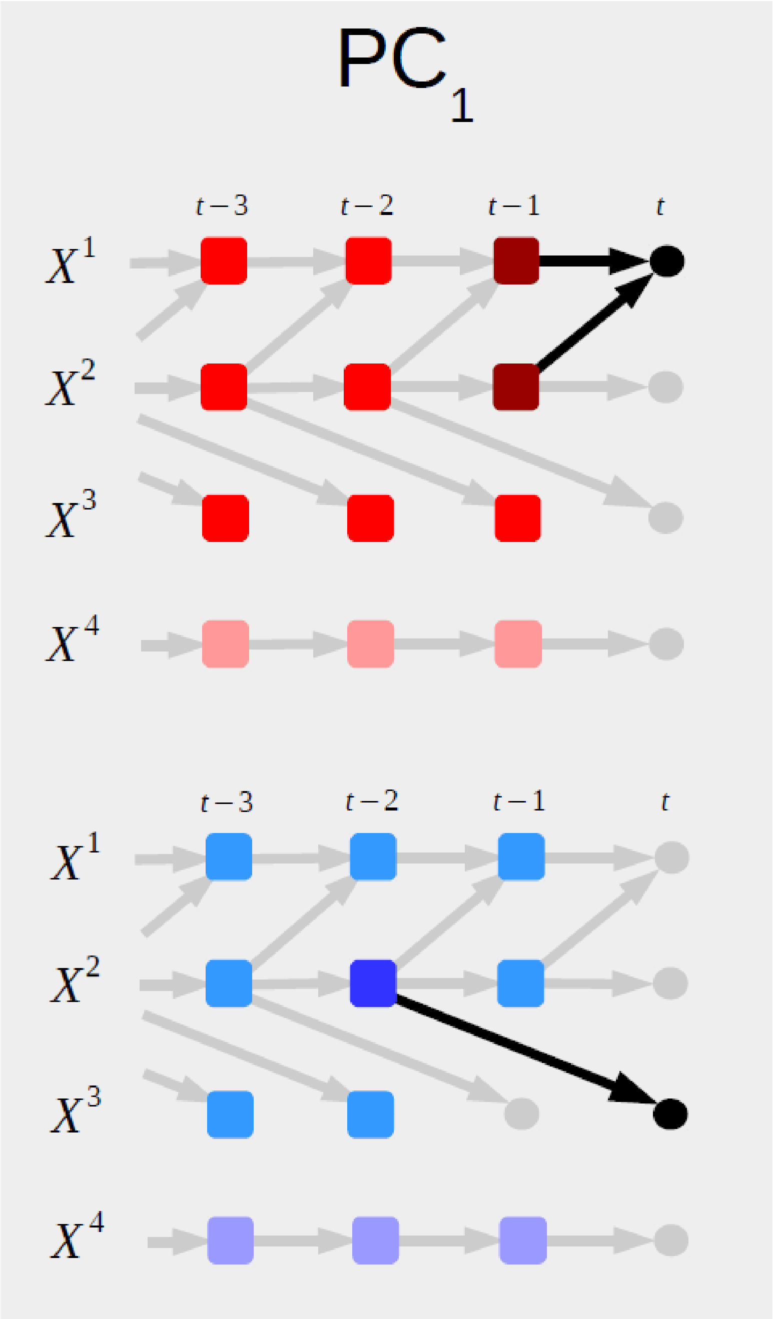
Couple to linearized gravity wave dynamics



<u>See</u>: ***Brenowitz, Beucler et al. (2020),*** *Kuang (2018, 2007), Herman and Kuang (2013)*

How can we **a priori** select relevant inputs for each output?

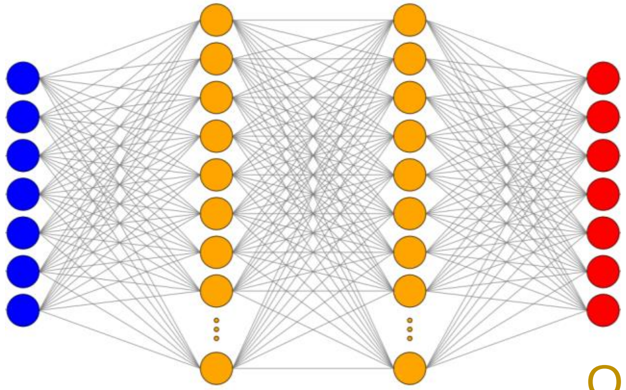




<u>Source</u>: Runge et al. (2019), <u>See</u>: Kretschmer et al. (2016), Runge et al. (2019), Spirtes & Glymour (1991)

Causally-Informed Neural Networks

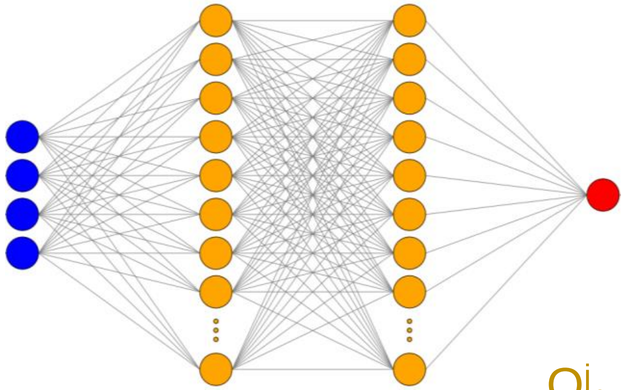
Multi-Output NN



$I_{N=94}$ $O_{N=65}$

Hidden layers
Number = 9
…

Causal-NNs

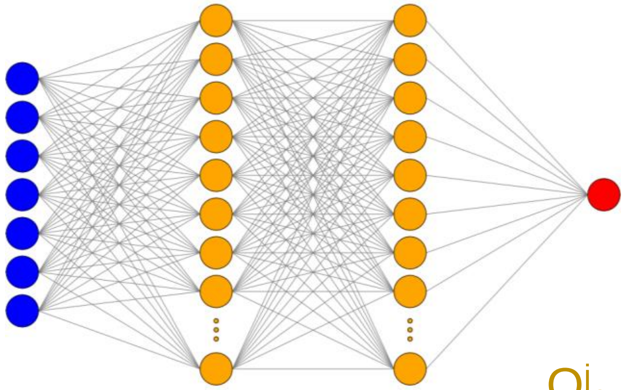


$I_{N=causal}$ $O^j_{(N=65)}$

Hidden layers
Number = 9
…

$$I^i_{causal} \rightarrow O^j \in P(O^j)$$

Single-Output NN

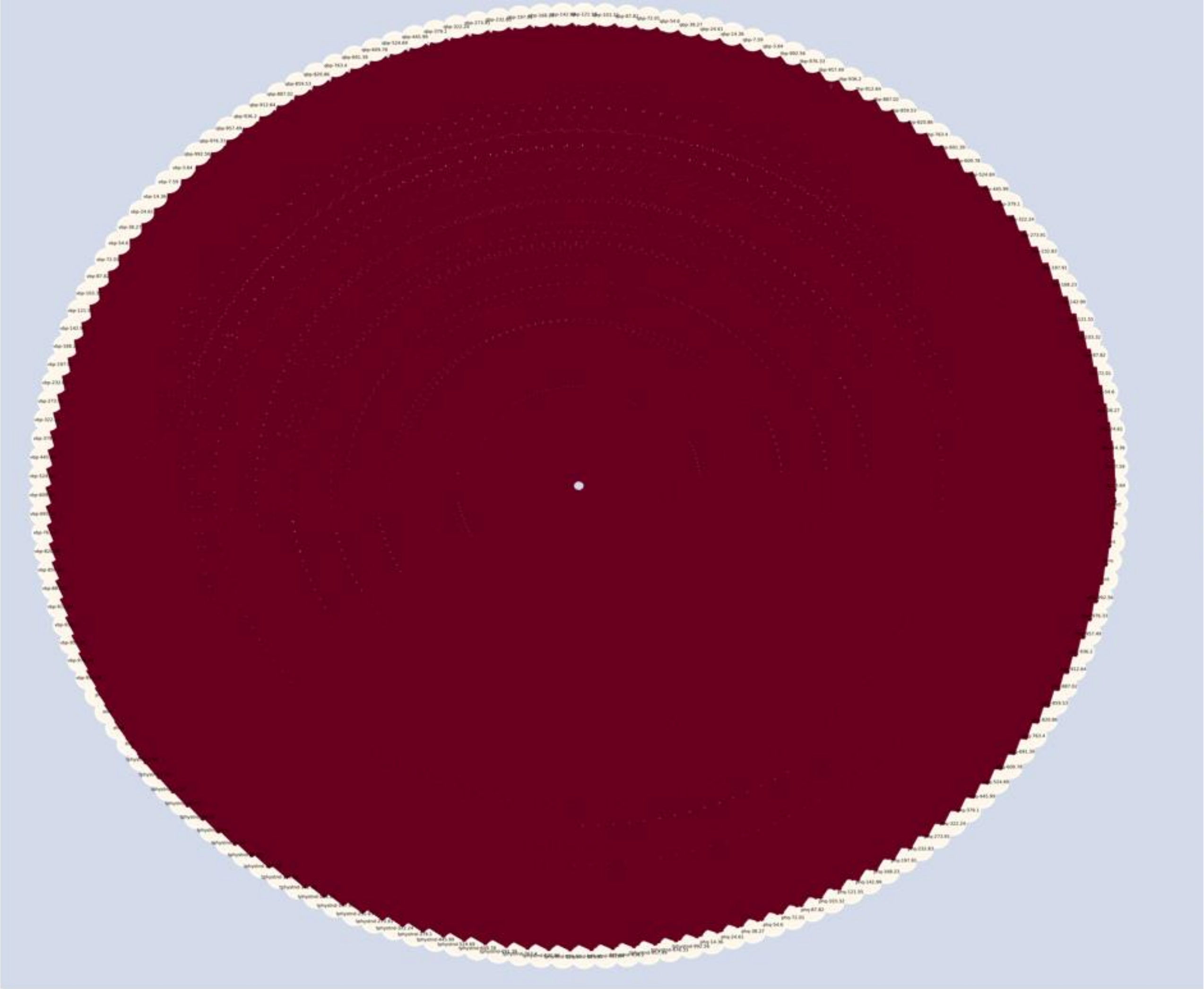


$I_{N=94}$ $O^j_{(N=65)}$

Hidden layers
Number = 9
…

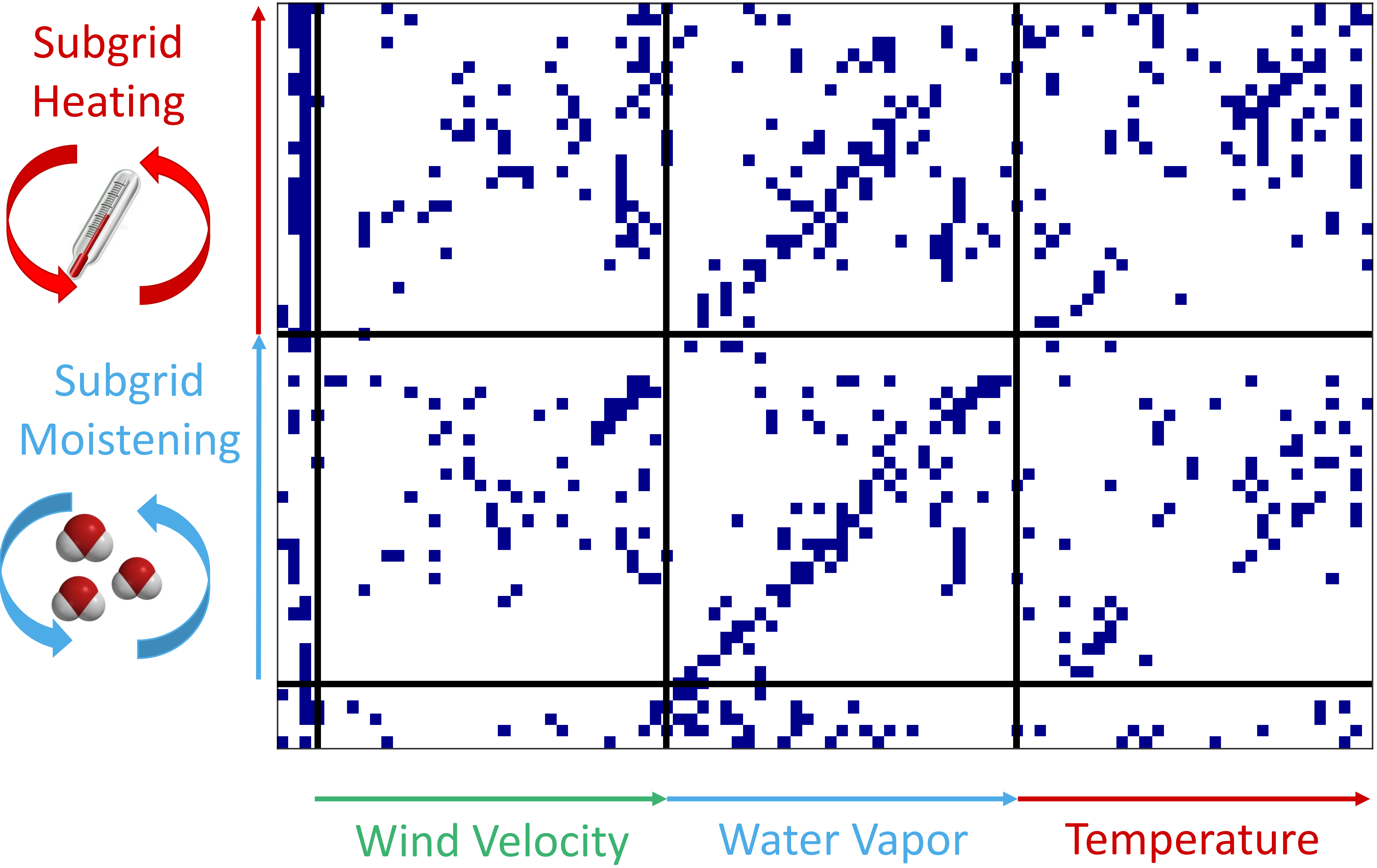
Source: Fernando Iglesias-Suarez (DLR)

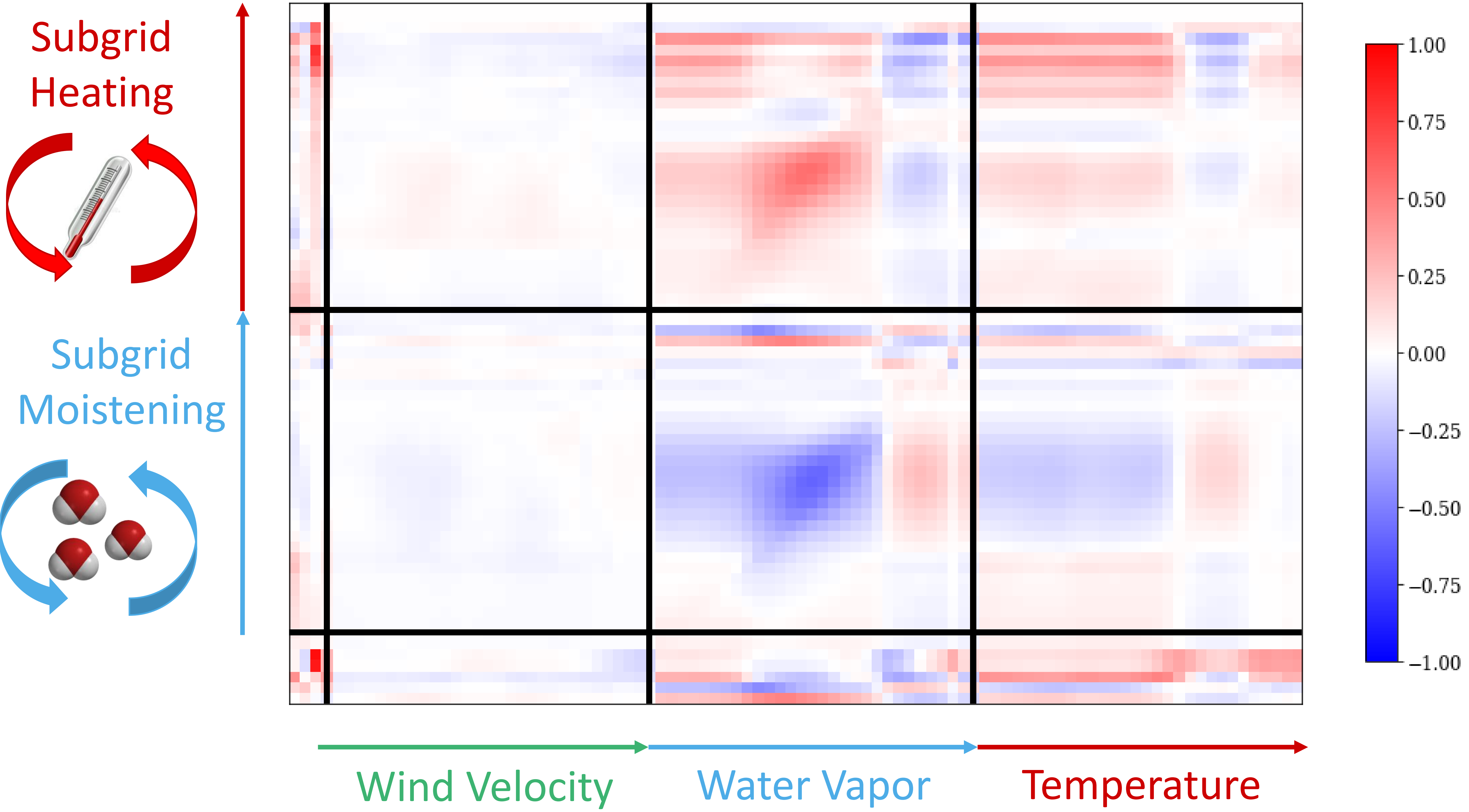
Before PC1: Fully-connected



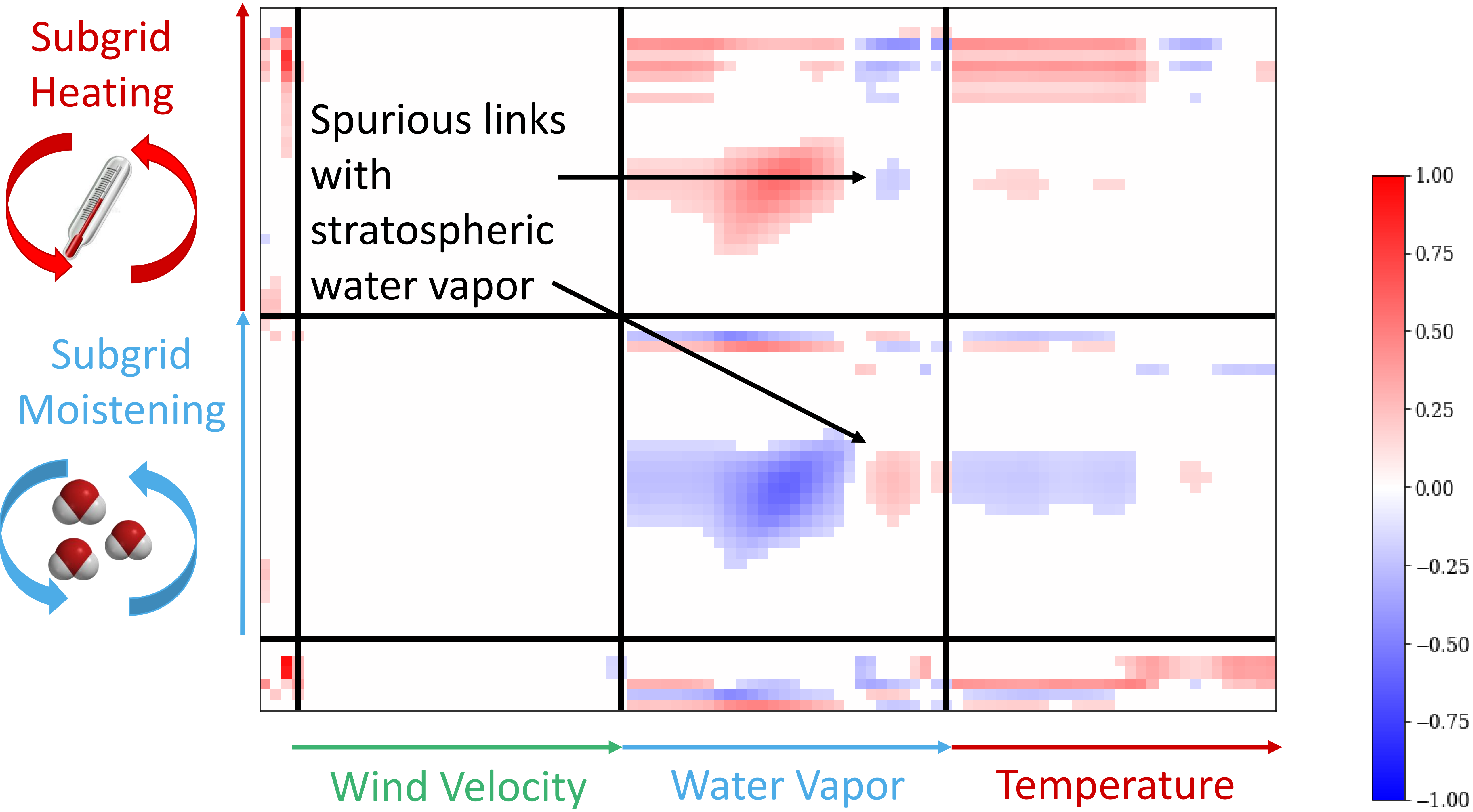
Fully-connected Inputs-to-Outputs

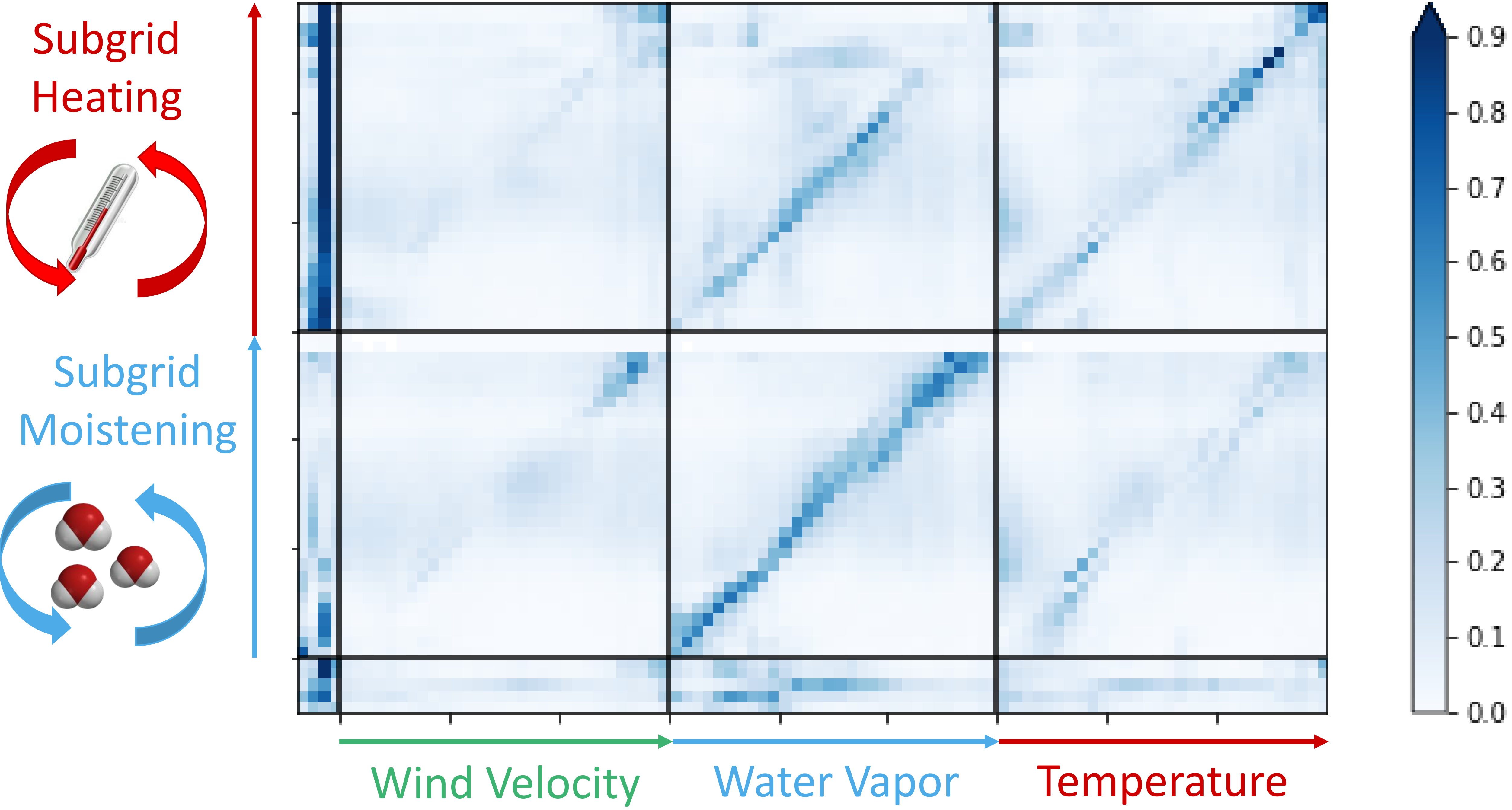
<u>Source</u>: Fernando Iglesias-Suarez (DLR), <u>See</u>: Spirtes and Glymour (1991)


Subgrid
Heating
Subgrid
Moistening
Wind Velocity
Water Vapor
Temperature

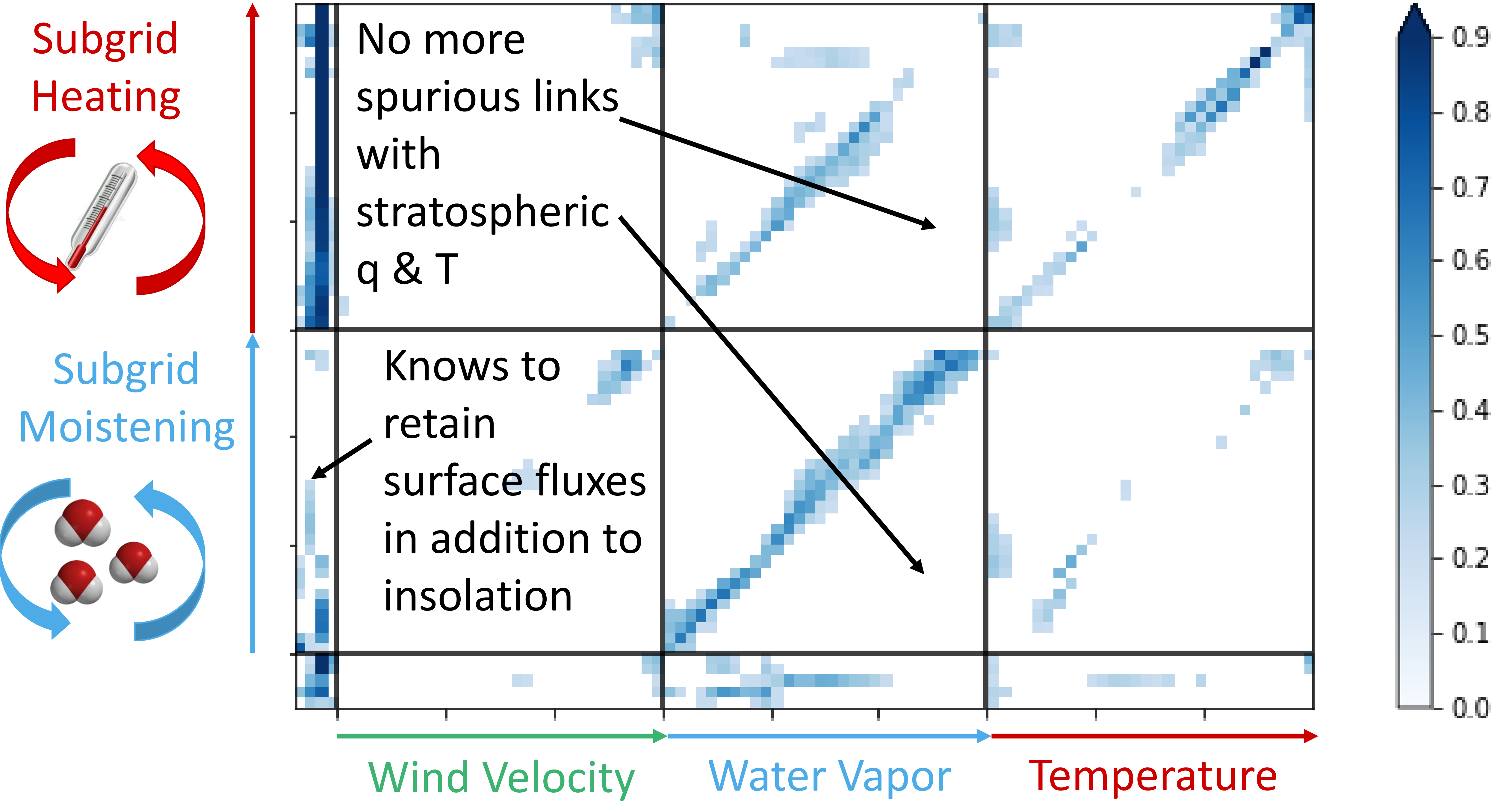


Subgrid Heating
Subgrid Moistening
Wind Velocity
Water Vapor
Temperature
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00


Subgrid Heating
Subgrid Moistening
Spurious links with stratospheric water vapor
Wind Velocity
Water Vapor
Temperature
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

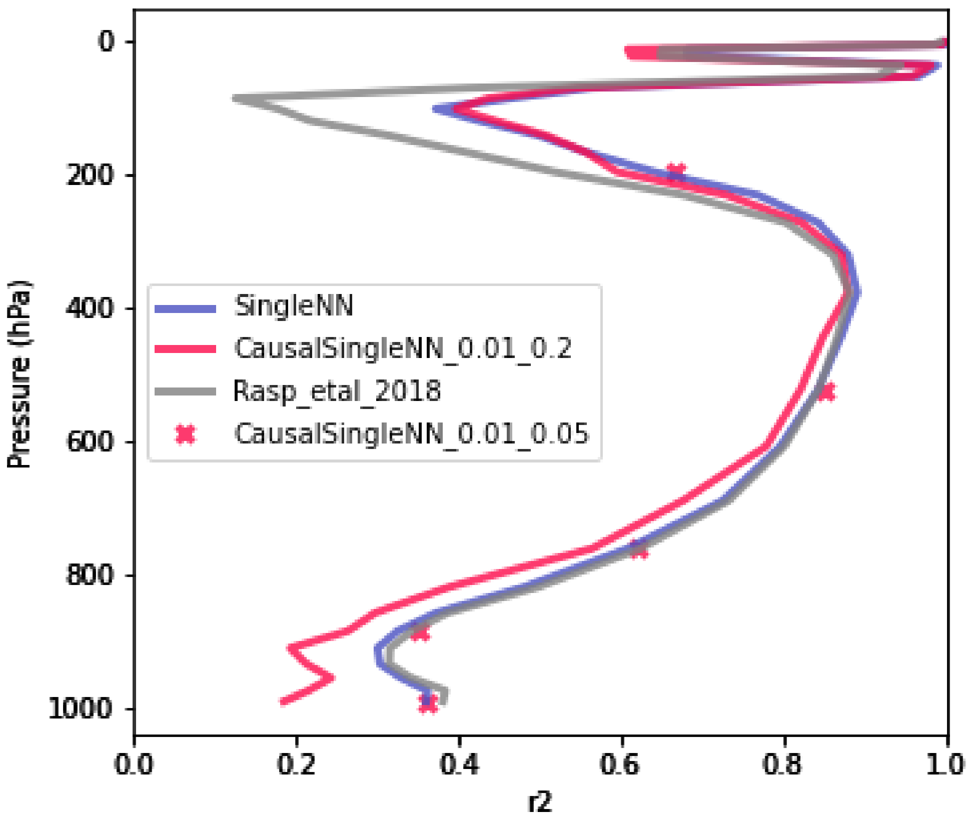


Source: Fernando Iglesias-Suarez (DLR)

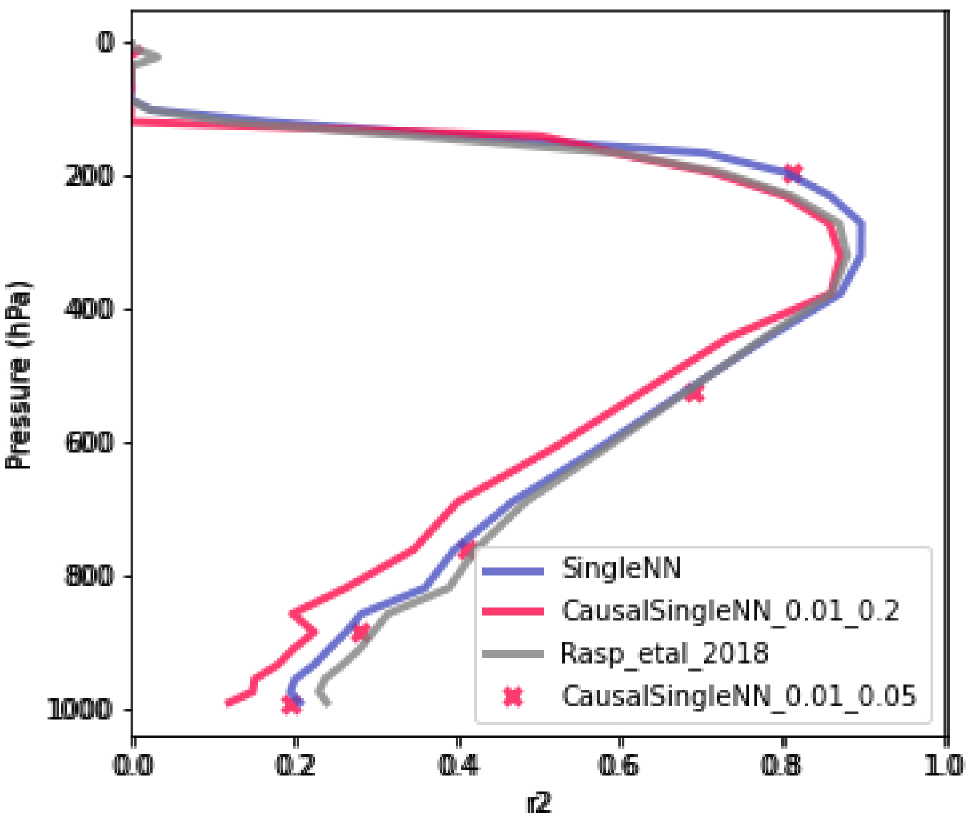


Source: Fernando Iglesias-Suarez (DLR)

Offline evaluation – global mean vertical profiles: R^2

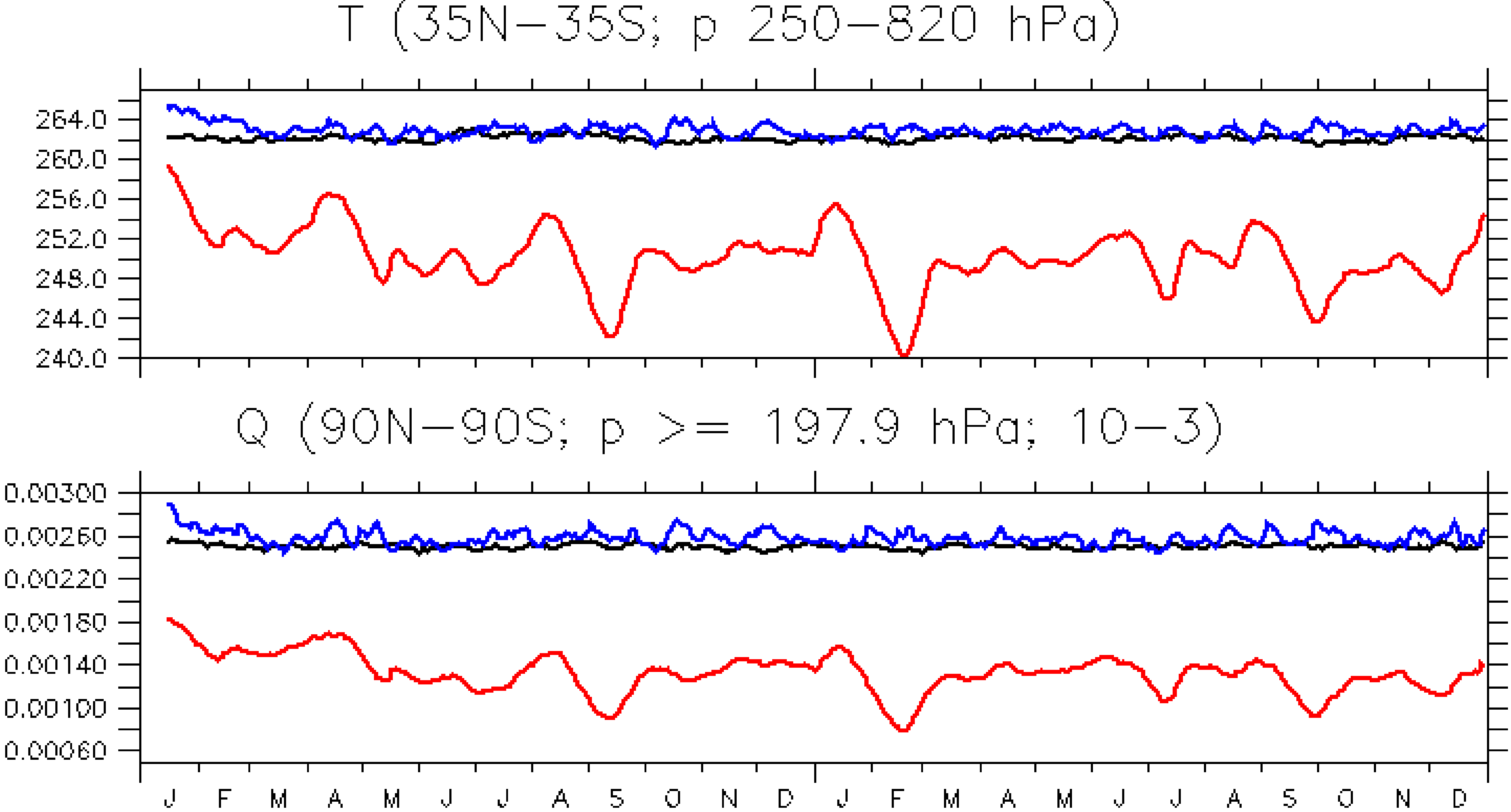


Subgrid Heating



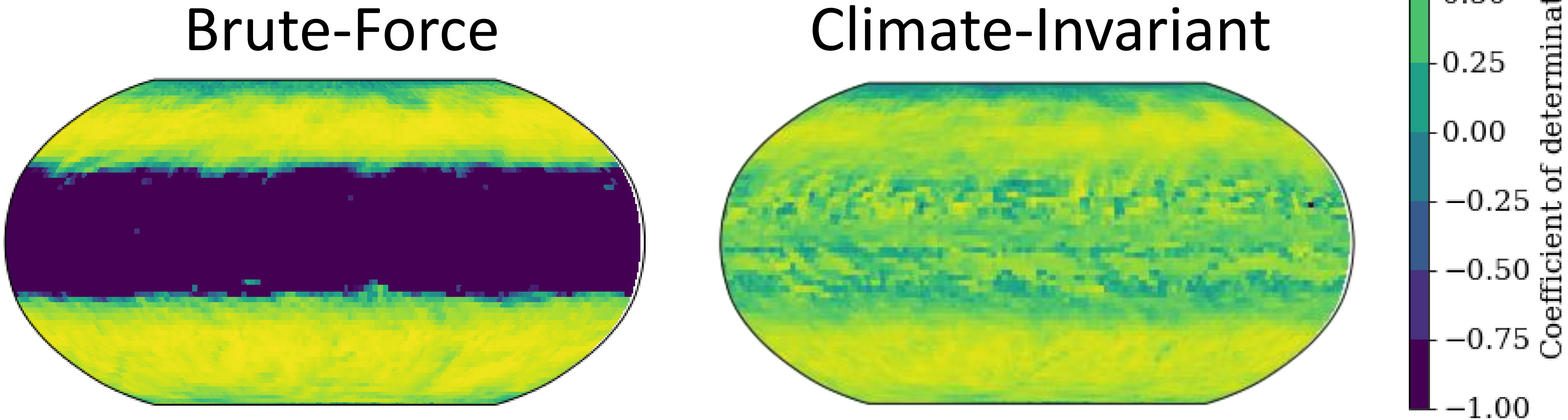
Subgrid Moistening

Source: Fernando Iglesias-Suarez (DLR)



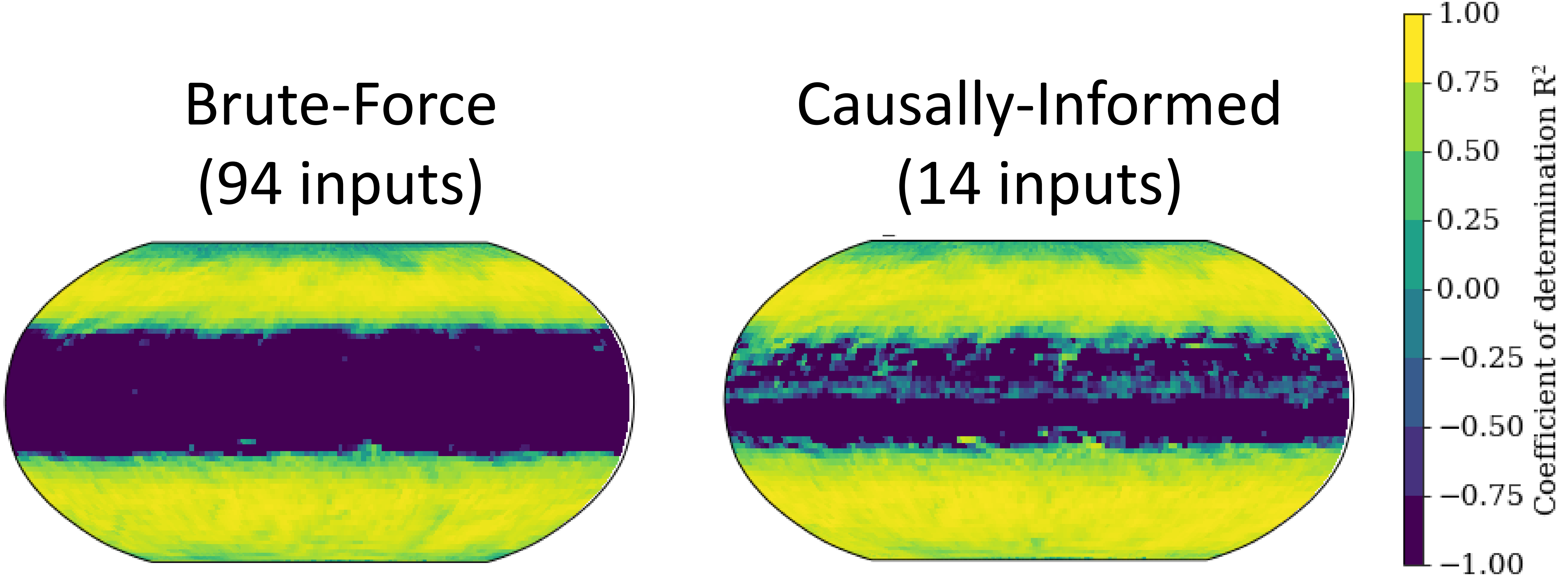
Source: Fernando Iglesias-Suarez (DLR)

Physical Rescalings helps ML algorithms generalize well to warmer climates (8K)



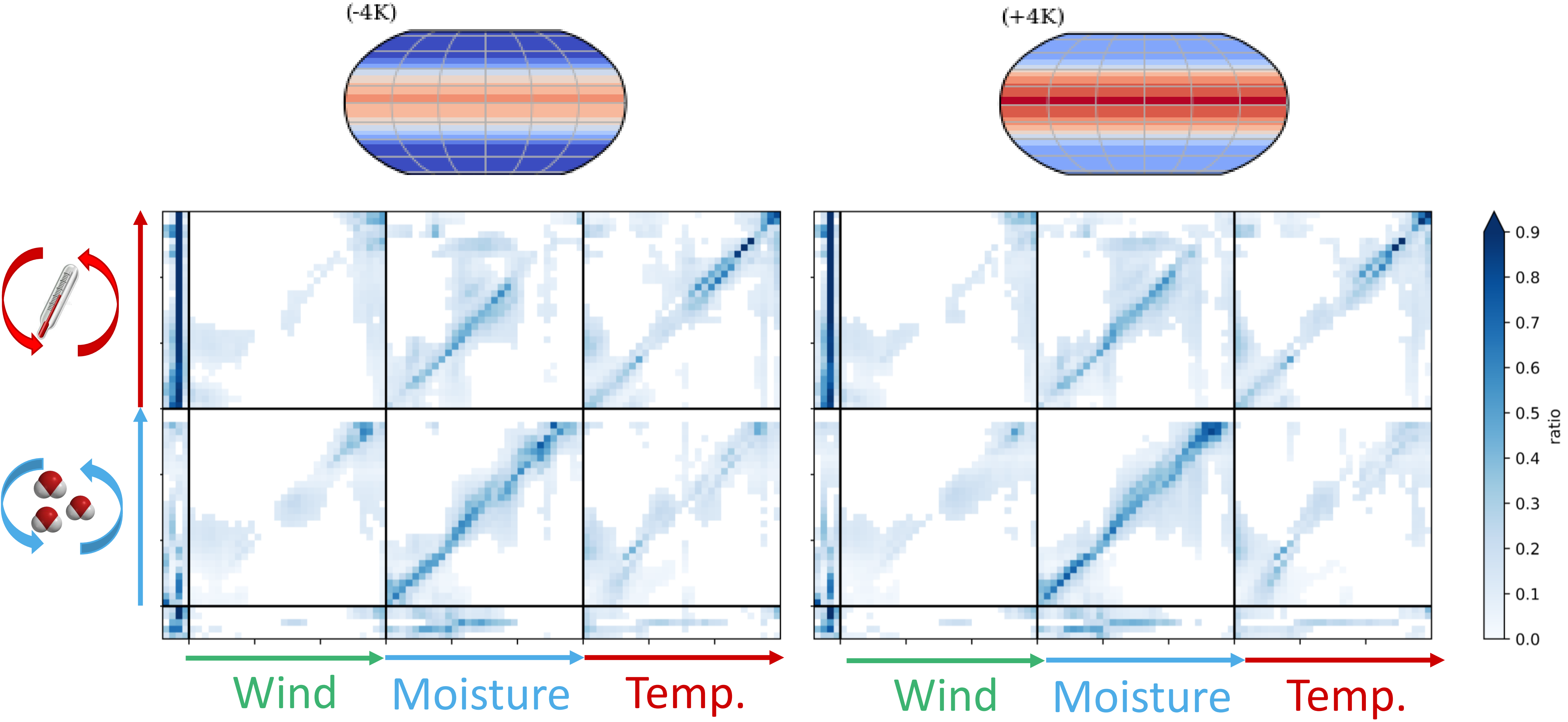
Mid-Tropospheric Subgrid Heating

Eliminating spurious links also helps ML algorithms generalize to warmer climates



Mid-Tropospheric Subgrid Heating

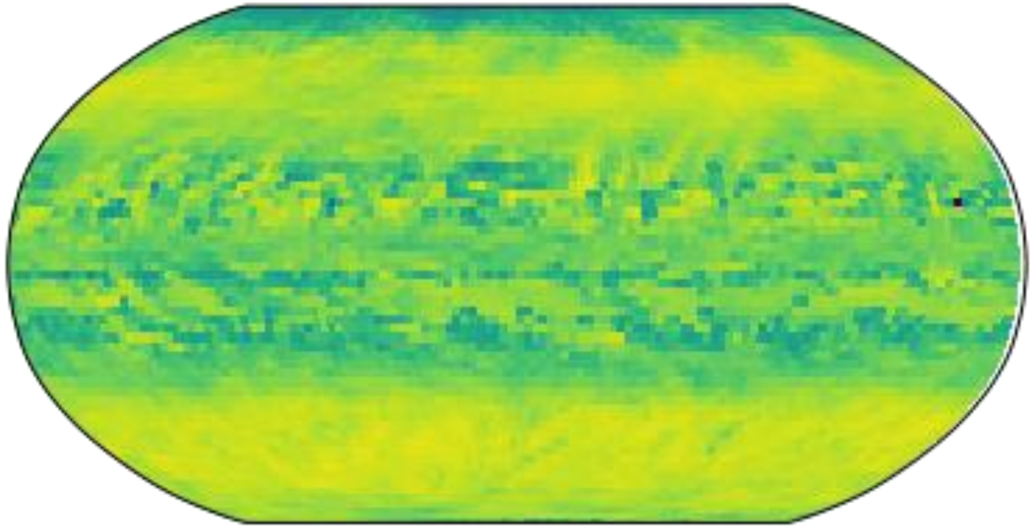
Possible Explanation: Causal Links are Climate-Invariant



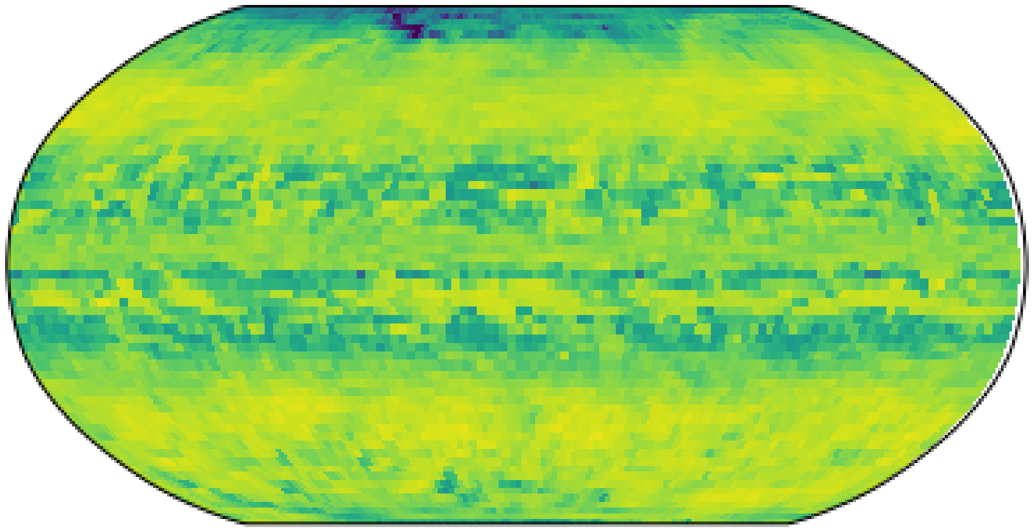
Source: Fernando Iglesias-Suarez (DLR)

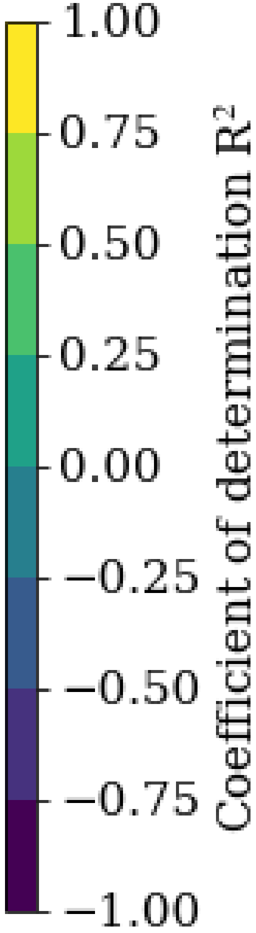
Physically and Causally-Informed Neural Nets generalize to warmer climates with fewer inputs

Climate-Invariant
(94 inputs)



Climate-Invariant
+ Causally-Informed
(11 inputs)





Mid-Tropospheric Subgrid Heating

Conclusion & Outlook

1. Generalization: Physically rescaling the inputs and outputs of neural networks helps them generalize to unseen climates and geographies

→ Test climate-invariant neural nets online, directly train on observations

2. - Interpretability framework can improve online stability

- Causal discovery helps objectively select inputs for parsimonious models

→ Thoroughly explore online stability and performance, tune HPs

3. Physics-informed + causally-informed ML helps create Physically-consistent + general + parsimonious models

→ Try in different settings (heatwave predictability), on more variables

When do we need physical knowledge and causal consistency?

Thank you

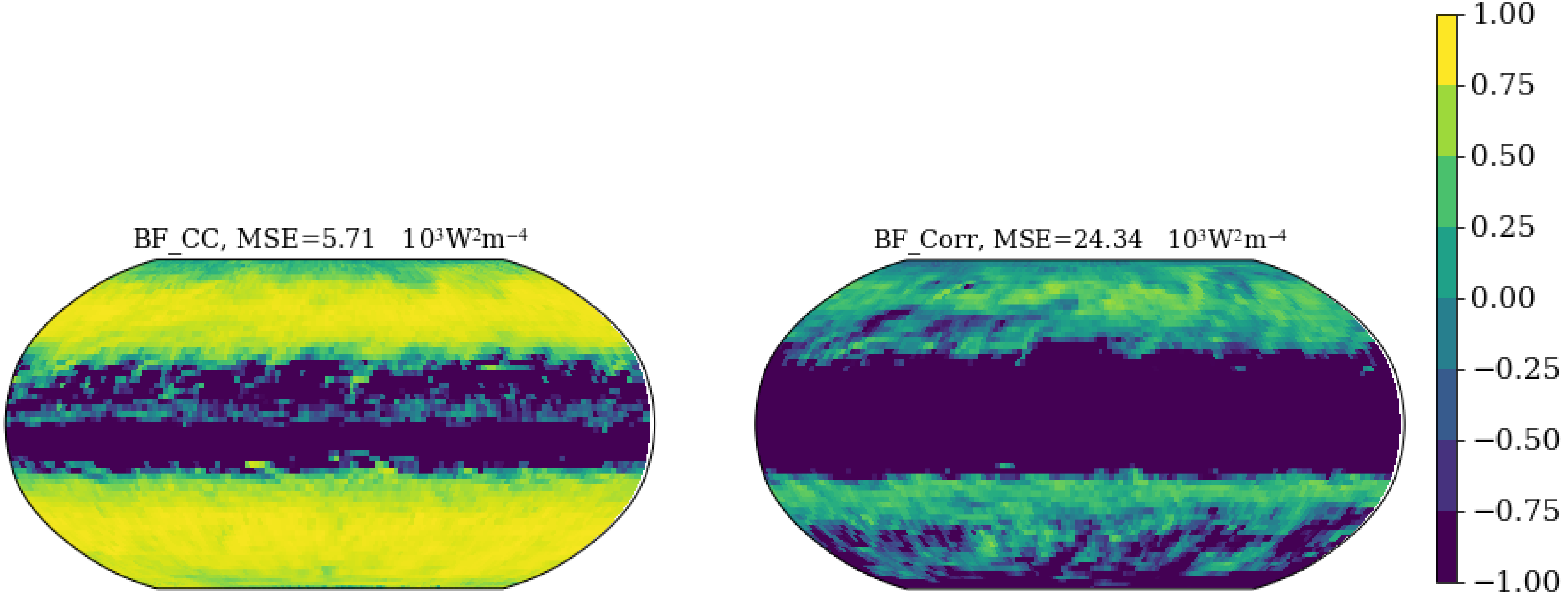
www.unil.ch/dawn

tom.beucler@unil.ch

Bonus slides

Causally-informed
Brute-force

Using abs value of correlations
Brute-force



Mid-Tropospheric Subgrid Heating

Links kept in BF vs CI

In [64]:
```
nam[:-1][link_locbyloc>0.2]
```
Out[64]:
```
array(['QBP274', 'QBP322', 'QBP379', 'QBP446', 'QBP525', 'QBP610',
       'TBP446', 'TBP525', 'TBP610', 'TBP993', 'PS', 'SOLIN', 'SHFLX',
       'LHFLX'], dtype=object)
```

In [68]:
```
nam[:-1][np.abs(corr_array)>0.15]
```
Out[68]:
```
array(['QBP198', 'QBP233', 'QBP274', 'QBP322', 'QBP379', 'QBP446',
       'QBP525', 'QBP610', 'QBP691', 'QBP763', 'QBP821', 'QBP860',
       'QBP887'], dtype=object)
```

In [105]:
```
nam[:-1][link_locbyloc>0.2]
```
Out[105]:
```
array(['RH233', 'RH274', 'RH322', 'RH379', 'RH446', 'RH525', 'RH610',
       'PS', 'SOLIN', 'SHFLX', 'LHF_nsDELQ'], dtype=object)
```

USMILE

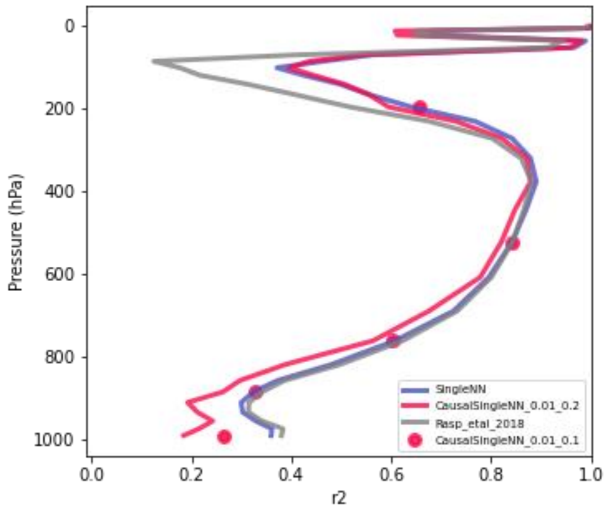
Causal discovery and deep learning to improve convection in climate models

Fernando Iglesias-Suarez[1], Veronika Eyring[1,2] , Pierre Gentine[3,4], Tom Beucler[5], Michael Pritchard[6], Jakob Runge[7,8], and Breixo Solino-Fernandez[1]

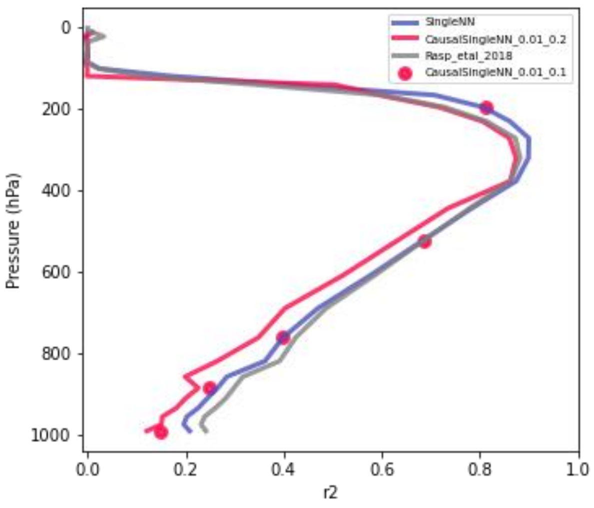
[1]Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR), Institute of Atmospheric Physics, Oberpfaffenhofen, Germany
[2]University of Bremen, Institute of Environmental Physics (IUP), Bremen, Germany
[3]Department of Earth and Environmental Engineering, Columbia University, New York, USA
[4]Earth Institute and Data Science Institute, Columbia University, New York, USA
[5]University of Lausanne, Institute of Earth Surface Dynamics, Lausanne, Switzerland
[6]University of California, Department of Earth System Science, Irvine, USA
[7]German Aerospace Center, Institute of Data Science, Jena, Germany
[8]TU Berlin, Berlin, Germany

DD MM. 2022

CausalNN offline performance (0K): spatially-based threshold

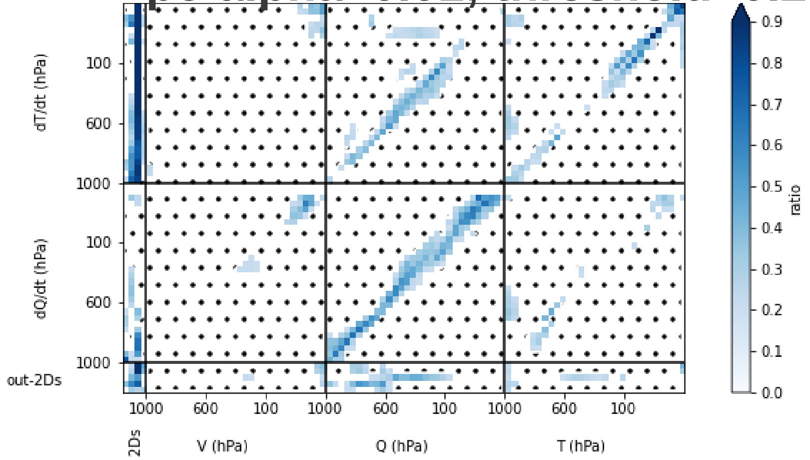


dT/dt

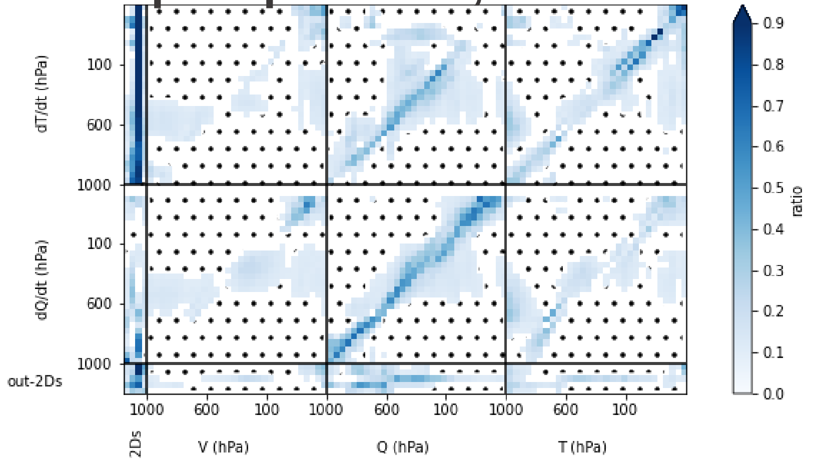


dQ/dt

pc-alpha=0.01; threshold=0.2

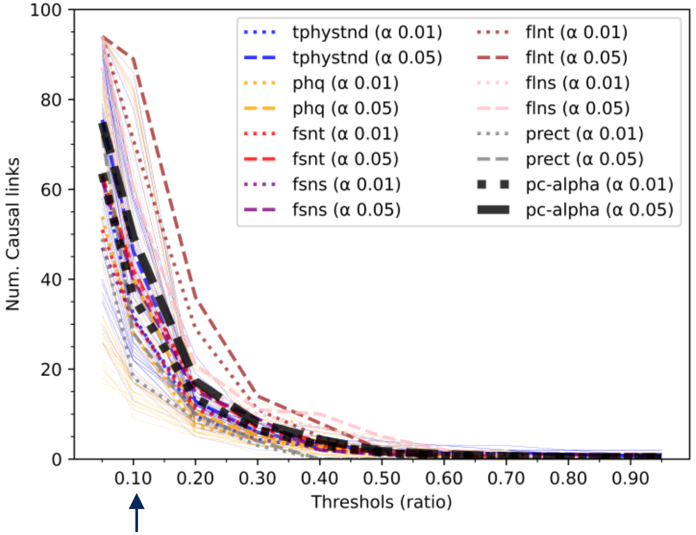


pc-alpha=0.01; threshold=0.1

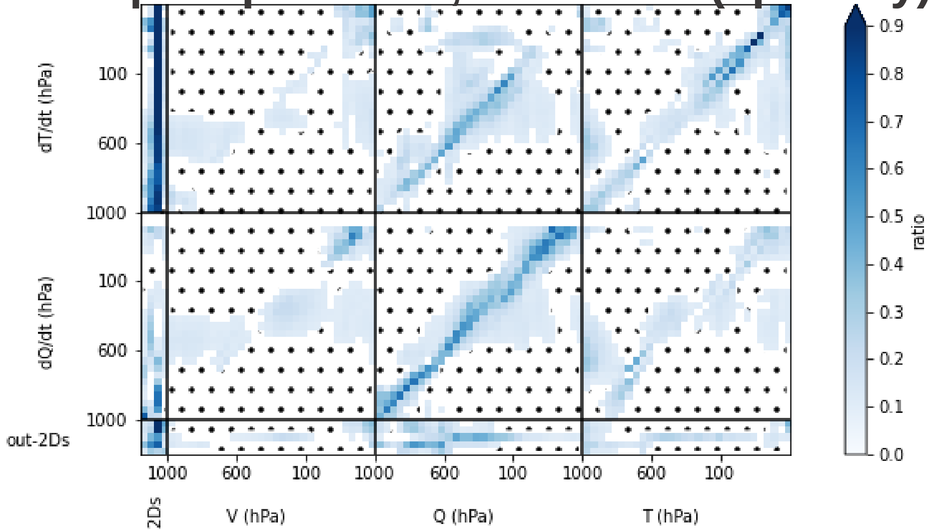


USMILE

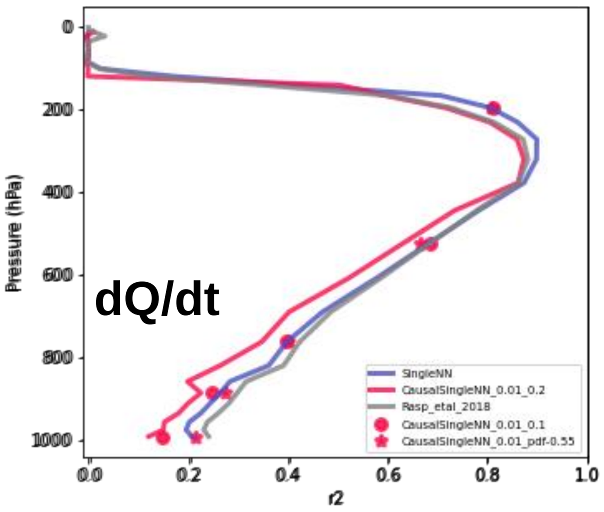
Causal discovery (thresholds): spatially-based vs quantile-based

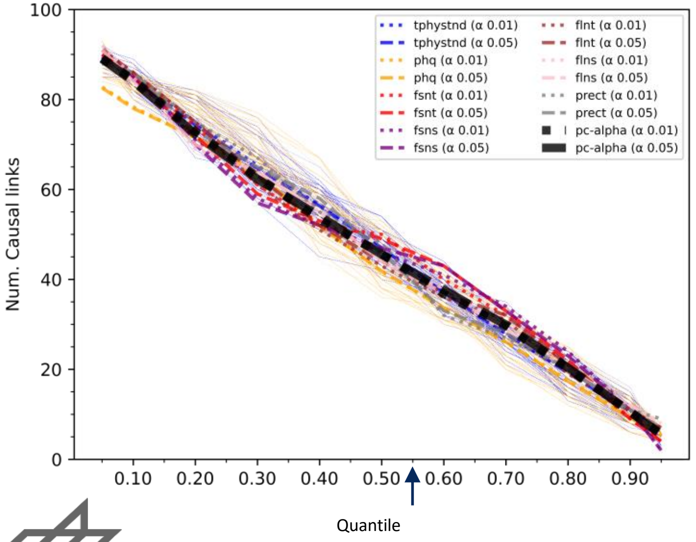


pc-alpha=0.01; threshold (spatially) = 0.1

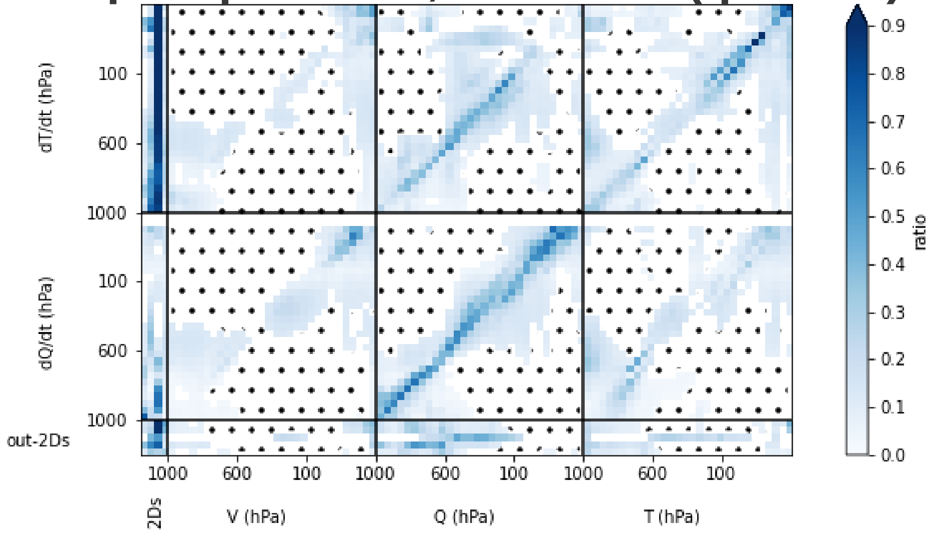


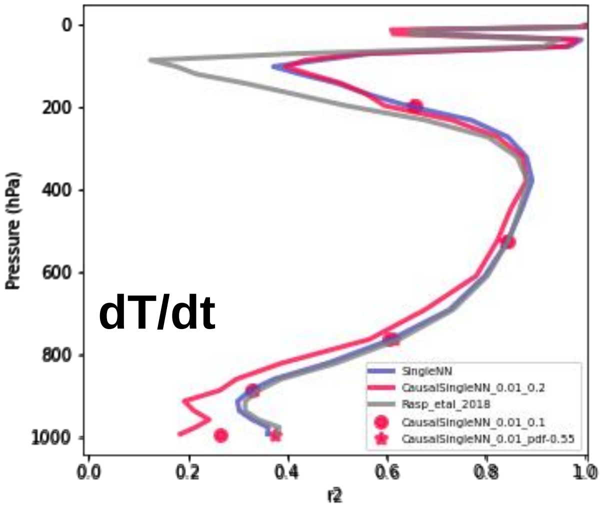
0K





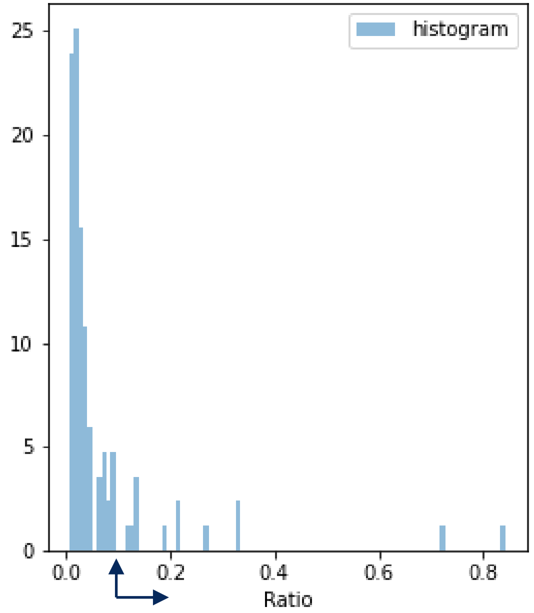
pc-alpha=0.01; threshold (quantile) = 0.55

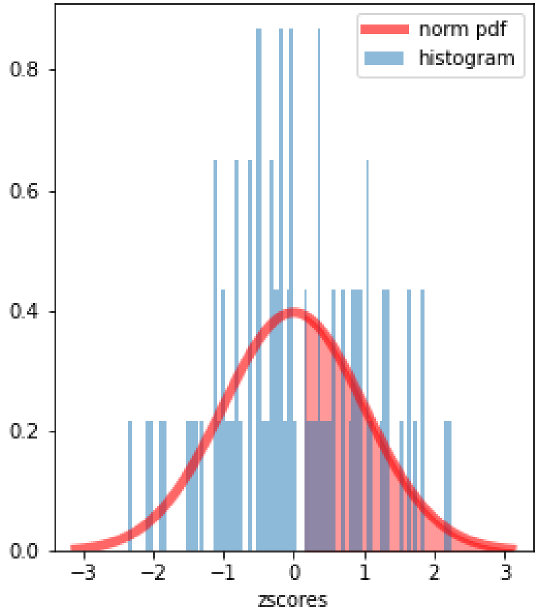




DLR

pa'(dT/dt$_{992hPa}$): spatially-based vs quantile-based thresholds





pa'(O_t^j)-ratio = { I_t^i : $\frac{\#(I_t^i \in pa_g(Otj))}{N_g}$ > ratio }

Total number of causal-inputs: 13 (13.8 %)

pa'-spatially [0.1] = {

T_{hPa}[820, 859, 957, 976, 992],

Q_{hPa}[912, 992],

V_{hPa}[936, 957, 992],

Sin, H, E

}

pa'(O_t^j)-pdf = { I_t^i : $P(I_t^i \in pa_g(Otj))$ > quantile }

Total number of causal-inputs: 40 (42.6 %)

pa'-quantile [0.55] = {

T_{hPa}[3, 14, 232, 691, 763, 820, 859, 887, 912, 936, 957, 976, 992],

Q_{hPa}[445, 524, 609, 691, 763, 820, 859, 887, 912, 936, 957, 976, 992],

V_{hPa}[3, 7, 14, 859, 887, 912, 936, 957, 976, 992],

PS, Sin, H, E

}

USMILE

Threshold optimization

Spatially-based: $pa'(O_t^j) = \{ I_t^i : \frac{\#(I_t^i \in pa_g(Otj))}{N_g} > ratio \}$

Quantile-based: $pa'(O_t^j)\text{-pdf} = \{ I_t^i : P(I_t^i \in pa_g(Otj)) > quantile \}$

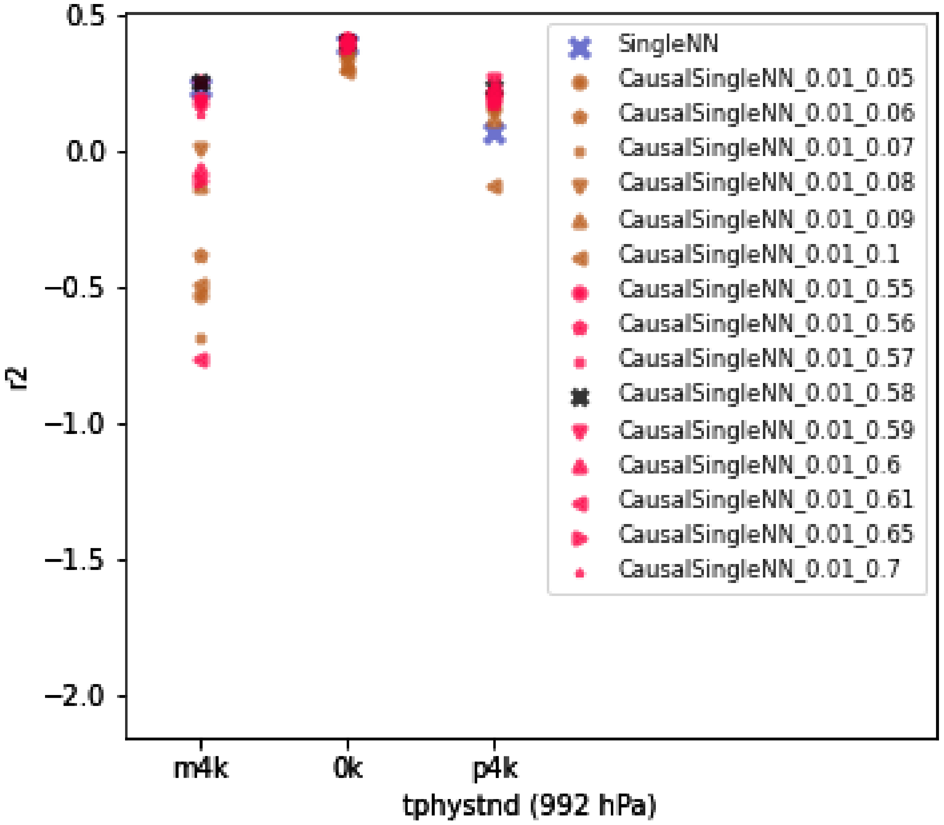
Threshold optimization: criteria

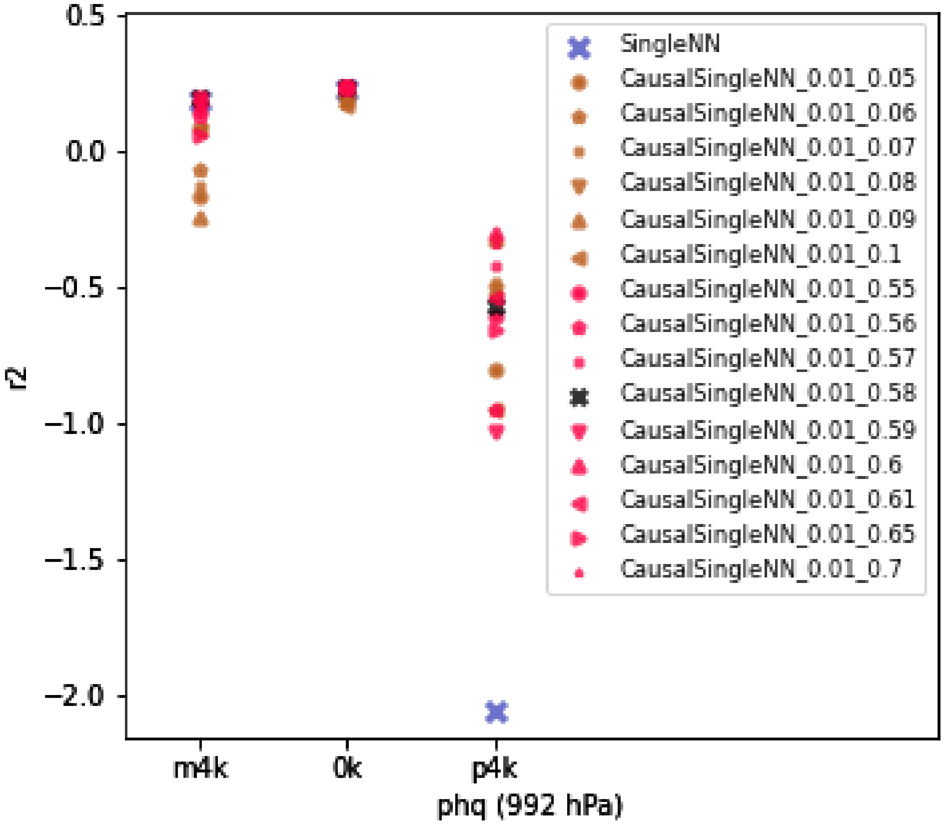
- **Threshold definitions**: spatially- & quantile-based approaches
- Optimization based on the 992 hPa level

Condition-1: $R^2_{CAUSALNN\text{-}thr} \geq R^2_{SINGLENN} \leftrightarrow R^2_{SINGLENN} > 0$

Condition-2: max(thr)

Threshold optimization: R^2_{992hPa}



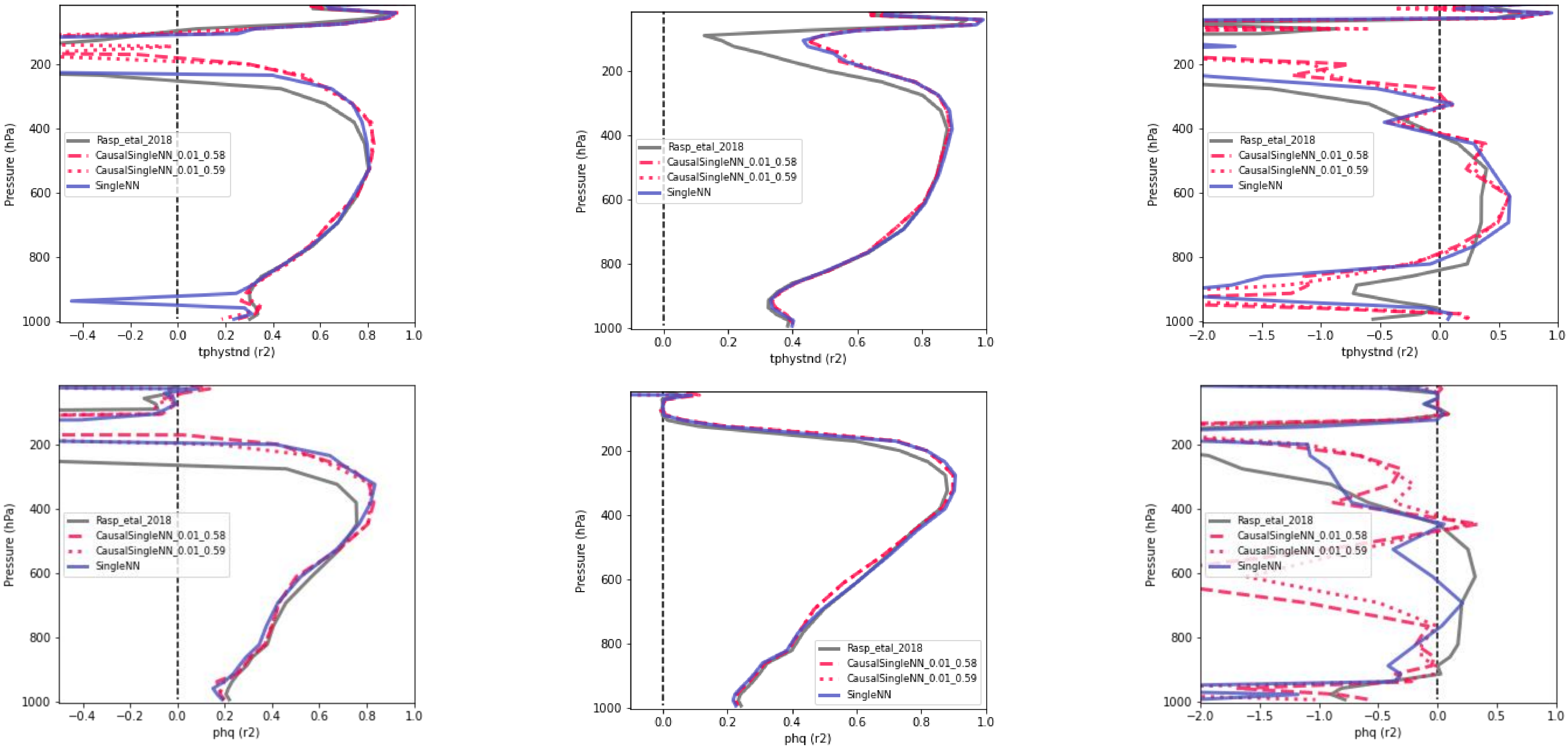


Condition-1: $R^2_{CAUSALNN\text{-}thr} \geq R^2_{SINGLENN} \leftrightarrow R^2_{SINGLENN} > 0$ # two dec. places precision

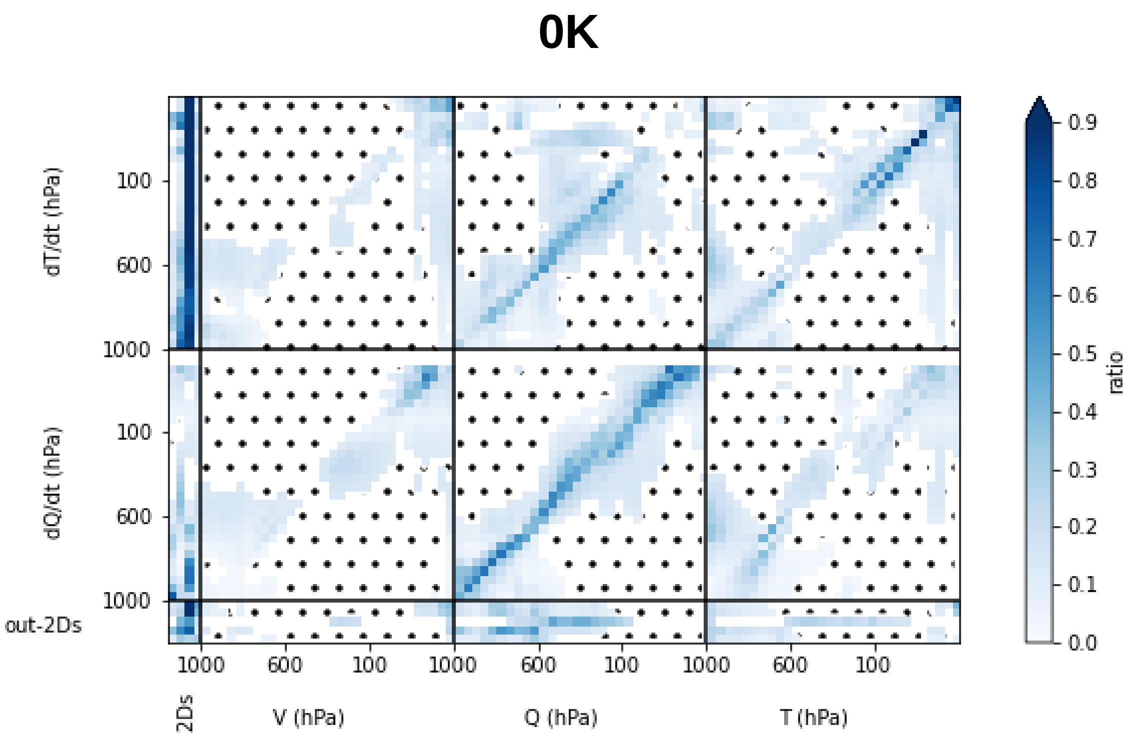
Condition-2: max(thr)

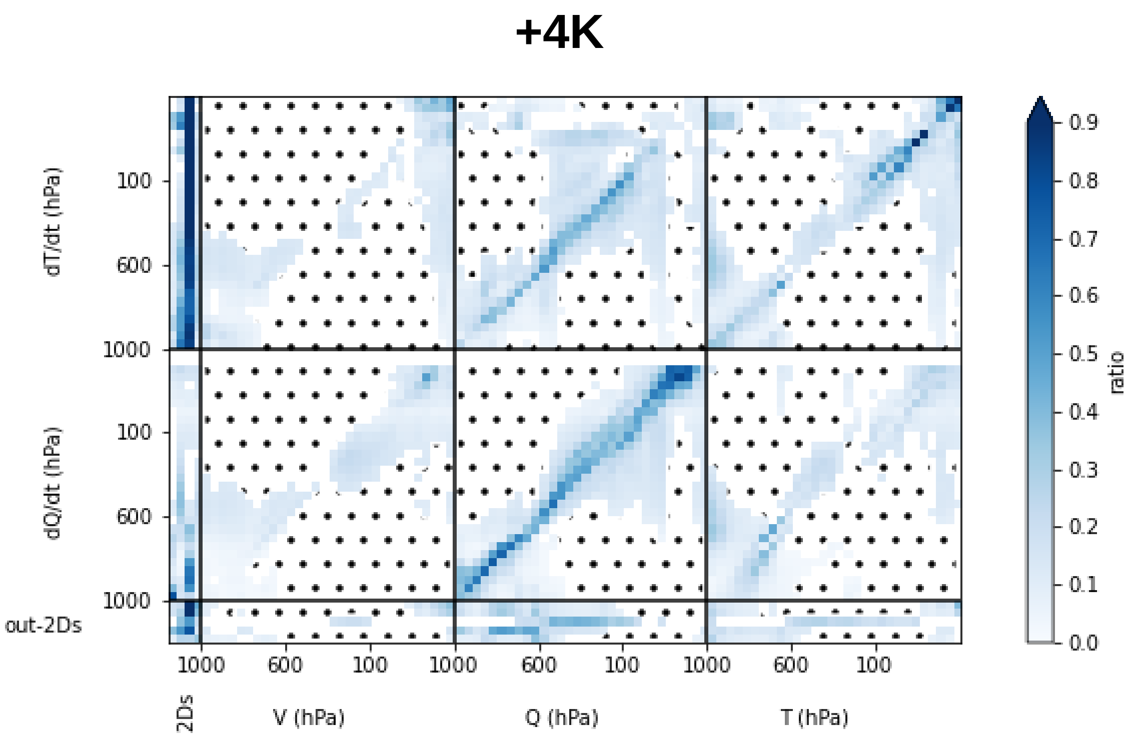
CausalNN offline performance (R^2): quantile-based [.58, .59]

–4K | 0K | +4K



Causal-inputs are consistent across 0K and +4K runs





pc-alpha=0.01; quantile-based[thr]=0.59

USMILE

Work in progress…

Hyperparameter tuning

- Aim: increase the CausalNN's performance while attaining sparser models (e.g., simpler models with increased interpretability); CausalNNCAM prognostic runs less dependable on the architecture.

- SHERPA. Hyperparameter space: batch normalization, activation function, optimizer, num. layers, num. nodes, and learning rates (e.g., Ott et al., 2020)

Hyperparameter tuning with SHERPA: strategy

Hyperparameter tuning is relatively computational costly and time consuming. **Potential approaches:**

- For each output (65)?
  - Each output would have its own best cases.
  - How do we construct, therefore, CausalNN cases?
    e.g., best case for each output, second best case, …,
    and random selection among top cases for each output.
  => JSC (juwels_booster) parallelization via distributed training (Horovod)

- Key-outputs?
  - e.g., 2-Ds and dT/dt & dQ/dt for key levels
  - CausalNN cases, e.g., train the rest of dT/dt & dQ/dt levels
    based following the hyperparameter tuning of the closest key level?
  => using single GPU

Hyperparameter tuning (SHERPA): hyperparameters & ranges

Taking advantage from Hertel et al. (2020), i.e., case study of hyperparameter tuning for NN learning SPCAM physics and that encompasses parameters specified in Rasp et al. (2018), we could focus on:

Goal (I): Increase CausalNNs' performance while attaining sparser models.
Goal (II): CausalNNCAM prognostic runs less dependable on the architecture.

Fixed parameters (~consistent between Hertel et al. & Rasp et al.):

- LeakyReLU coefficient: 0.3957 (Hertel et al.); or 0.3 (Rasp et al.)
- Learning rate: 0.001301 (Hertel et al.); or 0.001 (Rasp et al.)
- Learning rate decay: 0.843784 (Hertel et al.); or 0.58 (Rasp et al.)
- Epochs: 18 max with early-stopping (5 epochs patience?).

Hyperparameter optimization (parameters & ranges):

- Algorithm: random search; sampling hyperparameter settings uniformly from their ranges
- Number of layers: [1-10] # Type: discrete
- Nodes per layer: [32, 64, 128, 256, 512] # Type: choice
- Num. of trials: 50

Hertel et al. (2020)

Extra slides

In context of ML for atmospheric modeling

Three problems:

- Generalization, extrapolation outside of training set
- Physical consistency?
- Interpretability, Stability once coupled back online

Two frameworks: Physically-informed ML, causally-informed ML

Methods generally applicable to:

- All ML algorithms
- Spatiotemporal data ubiquitous in meteorological/climate applications

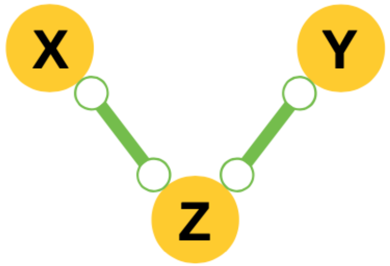
Testbed: Neural nets for subgrid-scale parameterization in climate model

Causal Markov Condition:

dependence ⇒ connectedness

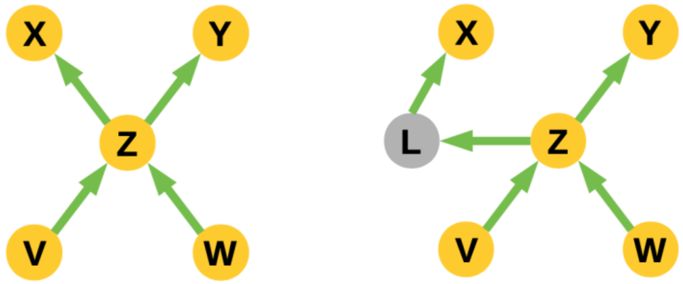
Faithfulness assumption:

independence ⇒ no causal link



Sufficiency

All common causes are “observed”



Methods: Overview

SPCAM → **Causally-informed**

Artificial Neural Network (NN)

emulating subgrid processes as represented by the SP component

→ NNCAM

CausalNNCAM

Methods: The causal discovery framework

Causally-informed

SPCAM → Artificial Neural Network (NN)
emulating subgrid processes as
represented by the SP component

Inputs $(I_t) = (I_t^1, \ldots, I_t^{N=94})$
[T(z), Q(z), V(z), PS, Sin, H, E]

Outputs $(O_t) = (O_t^1, \ldots, O_t^{N=65})$
[dT/dt(z), dQ/dt (z), Frad, P]

$$O_t^j = f(\mathcal{P}(O_t^j), \eta_t^j); \quad \mathcal{P}(O_t^j) \subset I_t^- = (I_{t-1}, I_{t-2}, \ldots)$$

$$I_{t-\tau}^i \rightarrow O_t^j \in \mathcal{P}(O_t^j)$$

Causally-linked Inputs

Methods: The causal discovery framework (PCMCI)

Causally-informed

SPCAM → **Artificial Neural Network (NN)**
emulating subgrid processes as
represented by the SP component

System

Inputs (I_t) = (I_t^1,…, $I_t^{N=94}$)

[T(z), Q(z), V(z), PS, Sin, H, E]

Outputs (O_t) = (O_t^1,…, $O_t^{N=65}$)

[dT/dt(z), dQ/dt (z), Frad, P]

PCMCI (setup)

- **Algorithm:** **PC-component (PC1)**

Methods: Causal discovery via PC1 algorithm (example)

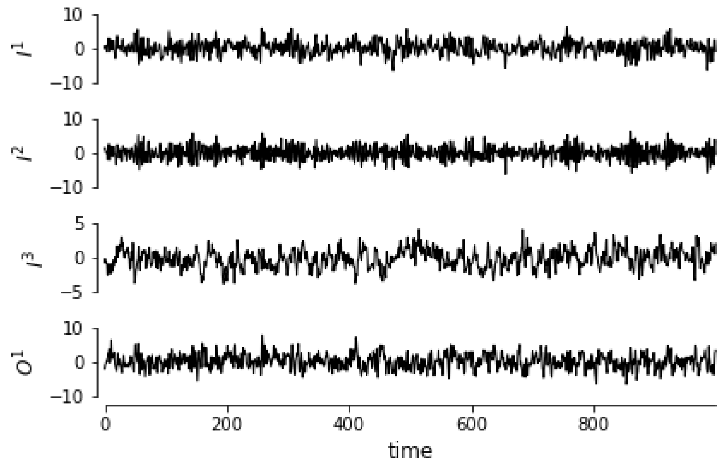
Causally-informed

SPCAM → **Artificial Neural Network (NN)** emulating subgrid processes as represented by the SP component

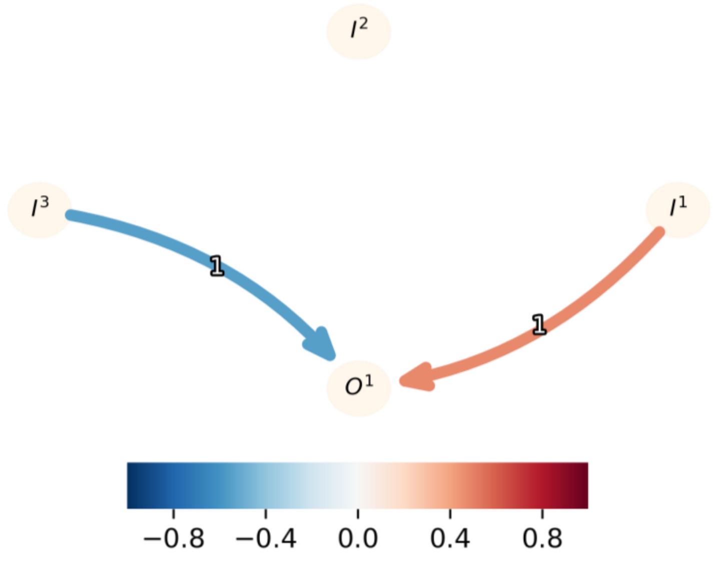
Made-up system

$$I_t^1 = 0.8 I_{t-1}^1 - 0.8 I_{t-1}^2 + \eta_t^1$$
$$I_t^2 = 0.5 I_{t-2}^2 + 0.5 I_{t-1}^1 + \eta_t^2$$
$$I_t^3 = 0.7 I_{t-1}^3 + \eta_t^3$$
$$O_t^1 = 0.7 I_{t-1}^1 - 0.8 I_{t-1}^3 + \eta_t^4$$

Time-series (1k samples)



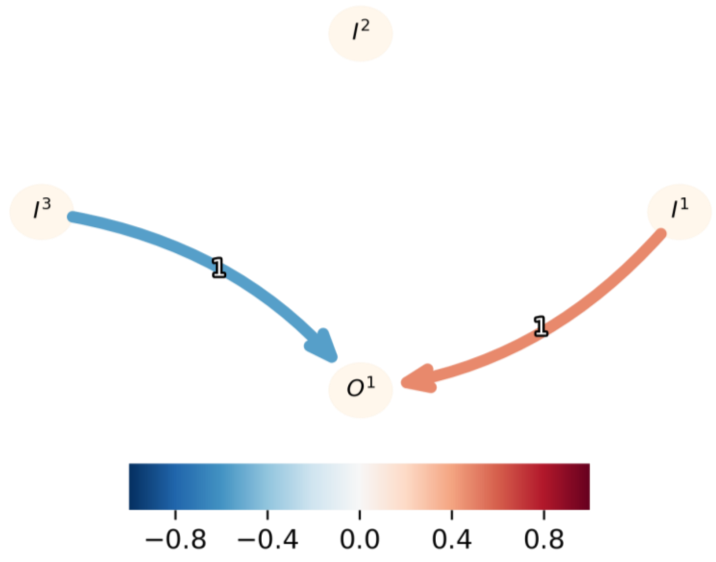
Causal discovery (PC1)



Methods: Causal discovery via PC1 algorithm (example)

Made-up system

$$I_t^1 = 0.8 I_{t-1}^1 - 0.8 I_{t-1}^2 + \eta_t^1$$
$$I_t^2 = 0.5 I_{t-2}^2 + 0.5 I_{t-1}^1 + \eta_t^2$$
$$I_t^3 = 0.7 I_{t-1}^3 + \eta_t^3$$
$$O_t^1 = 0.7 I_{t-1}^1 - 0.8 I_{t-1}^3 + \eta_t^4$$



PC1 algorithm

1. Significance level: **PC-alpha** (e.g., 0.01)

2. Pearson correlation $\boldsymbol{\rho}(\mathbf{I}_{t-\tau}{}^{i}, O_t{}^1)$ for $\boldsymbol{\tau}$ = 1,2,…

   $\mathcal{P}^0 = \{I_{t-1}{}^3, I_{t-1}{}^1, I_{t-1}{}^2\}$

3. **Partial correlations** (conditional independence)

   i. $\boldsymbol{\rho}(\mathbf{I}_{t-1}{}^{[1,2]}, O_t{}^1 \mid \mathbf{I}_{t-1}{}^3 \in \mathcal{P}^0)$; $\mathcal{P}^1 = \mathcal{P}^0$

   ii. $\boldsymbol{\rho}(\mathbf{I}_{t-1}{}^{[3,2]}, O_t{}^1 \mid \mathbf{I}_{t-1}{}^1 \in \mathcal{P}^1)$; $\mathcal{P}^2 = \{I_{t-1}{}^3, I_{t-1}{}^1\}$

   iii. **… continue or converges**

Methods: The causal discovery framework (PCMCI)

Causally-informed

SPCAM → Artificial Neural Network (NN)
emulating subgrid processes as
represented by the SP component

System

Inputs (I_t) = (I_t^1,…, $I_t^{N=94}$)

[T(z), Q(z), V(z), PS, S_{in}, H, E]

Outputs (O_t) = (O_t^1,…, $O_t^{N=65}$)

[dT/dt(z), dQ/dt (z), F_{rad}, P]

PCMCI (setup)

- **Algorithm:** PC-component (PC1)
- **Conditional test:** Partial correlation
- **Lag-time (τ):** 1
- **PC-alpha:** [0.01, 0.05]

N_c (SPCAM): 8192 (lat×lon)

$\sum \mathcal{P}(O_t^j)_{Nc} \approx I_t$

The removal of a true link: PC1 power detection

Detection power depends on:

- **Dimensionality** of the Conditional independence (CI) test.

  Conditioning on the past of other adjacent links, increases the dimensionality of the CI test (iteration + 2)

  *Note the large number of inputs (I_t = 94) in a highly correlated system.

- **Effect size**.

  As the conditioning set increases, the significance of the CI (i.e., partial correlation) can get very small, such that it falls below the significance threshold.

Imagine the PC1 algorithm misses a “true” link due to high dimensionality and a very small effect size, then this could lead to “false” positives.

false negatives (removal of true links) -> false positives (due to missing true links)

Runge et al. (2019, Sci. Adv.); Runge (2020)

Methods: The causal discovery framework (PCMCI)

Causally-informed

SPCAM → **Artificial Neural Network (NN)**
emulating subgrid processes as
represented by the SP component

System

Inputs (I_t) = (I_t^1,…, $I_t^{N=94}$)

[T(z), Q(z), V(z), PS, Sin, H, E]

Outputs (O_t) = (O_t^1,…, $O_t^{N=65}$)

[dT/dt(z), dQ/dt (z), Frad, P]

PCMCI (setup)

- **Algorithm:** PC-component (PC1)
- **Conditional test:** Partial correlation
- **Lag-time (τ):** 1
- **PC-alpha:** [0.01, 0.05]

$$\mathcal{P}'(O_t^j) = \left(\sum \mathcal{P}(O_t^j)_{Nc} > \text{threshold} \right)$$

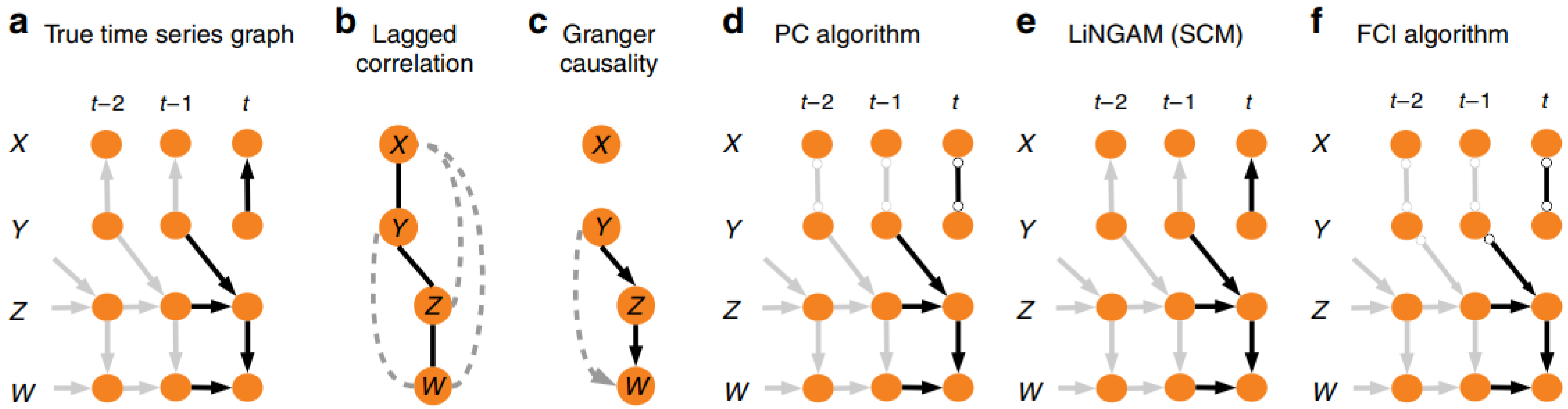
Outline

1. Generalization: Physically rescaling the inputs of neural networks helps them generalize to unseen climates and geographies

2. Stability:
- Interpretability framework can improve online stability
- Causal discovery helps objectively select inputs for parsimonious models

3. Physics-guided ML and causal discovery can be combined to create Physically-informed + parsimonious models, improving generalization

Box 1 | very short introduction to causal inference



Consider the time-dependent causal relations

$$
\begin{aligned}
X_t &= aY_t + E_t^X \\
Y_t &= E_t^Y \\
Z_t &= bZ_{t-1} + cY_{t-1} + E_t^Z \\
W_t &= dW_{t-1} + eZ_t + E_t^W,
\end{aligned}
\qquad (1)
$$

Methods: Causal discovery via PCMCI algorithm (example)

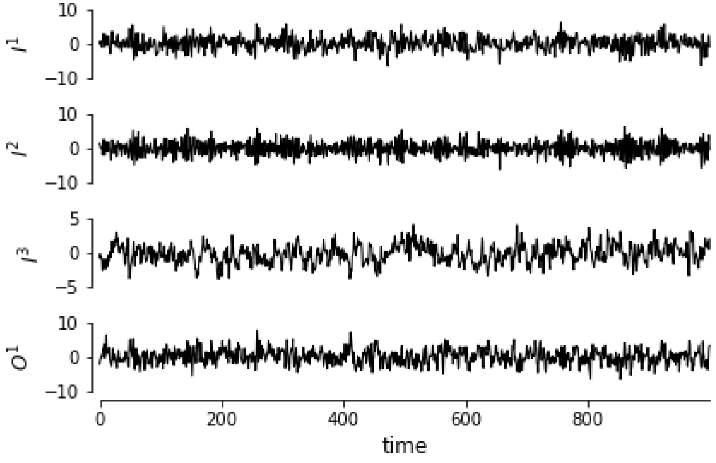
Causally-informed

SPCAM → Artificial Neural Network (NN)
emulating subgrid processes as
represented by the SP component

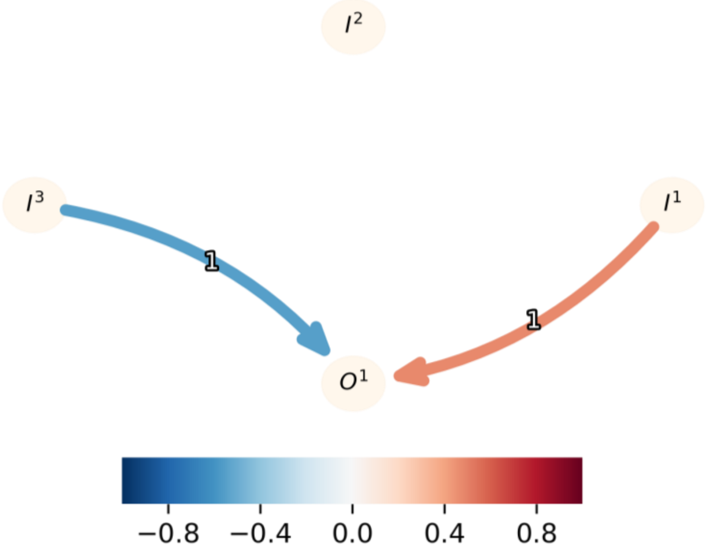
Made-up system

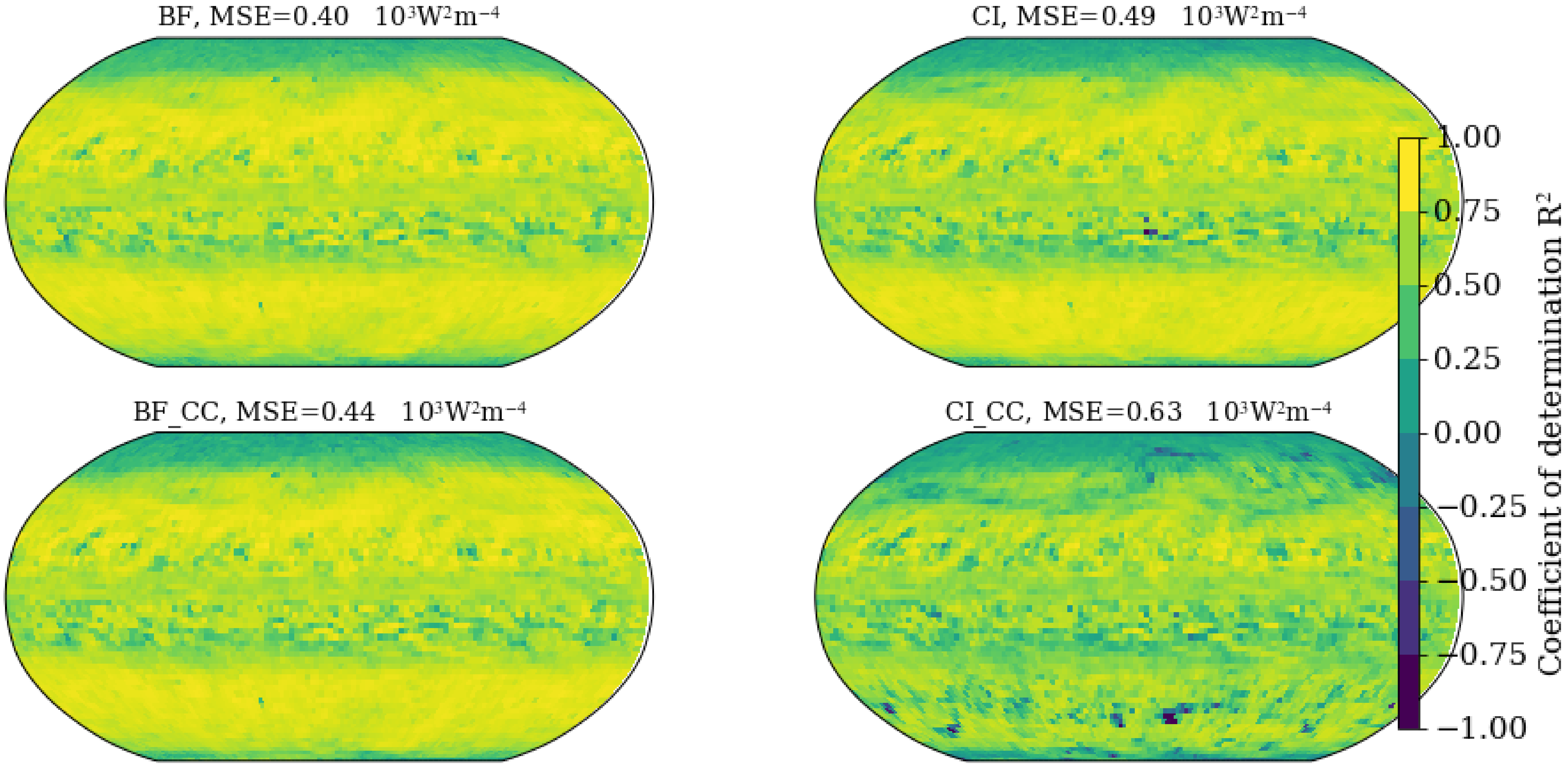
$$I_t^1 = 0.8 I_{t-1}^1 - 0.8 I_{t-1}^2 + \eta_t^1$$
$$I_t^2 = 0.5 I_{t-2}^2 + 0.5 I_{t-1}^1 + \eta_t^2$$
$$I_t^3 = 0.7 I_{t-1}^3 + \eta_t^3$$
$$O_t^1 = 0.7 I_{t-1}^1 - 0.8 I_{t-1}^3 + \eta_t^4$$

Time-series (1k samples)

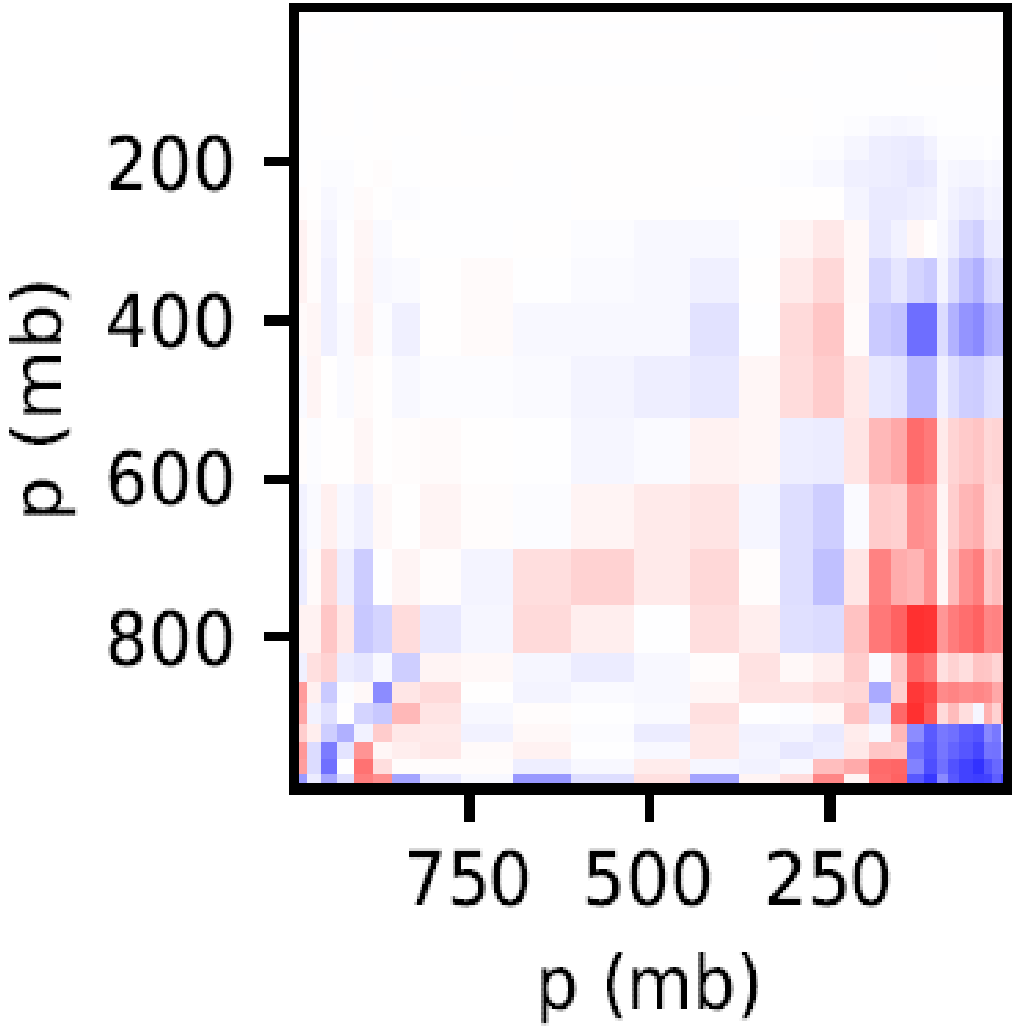


Causal discovery (PCMCI)



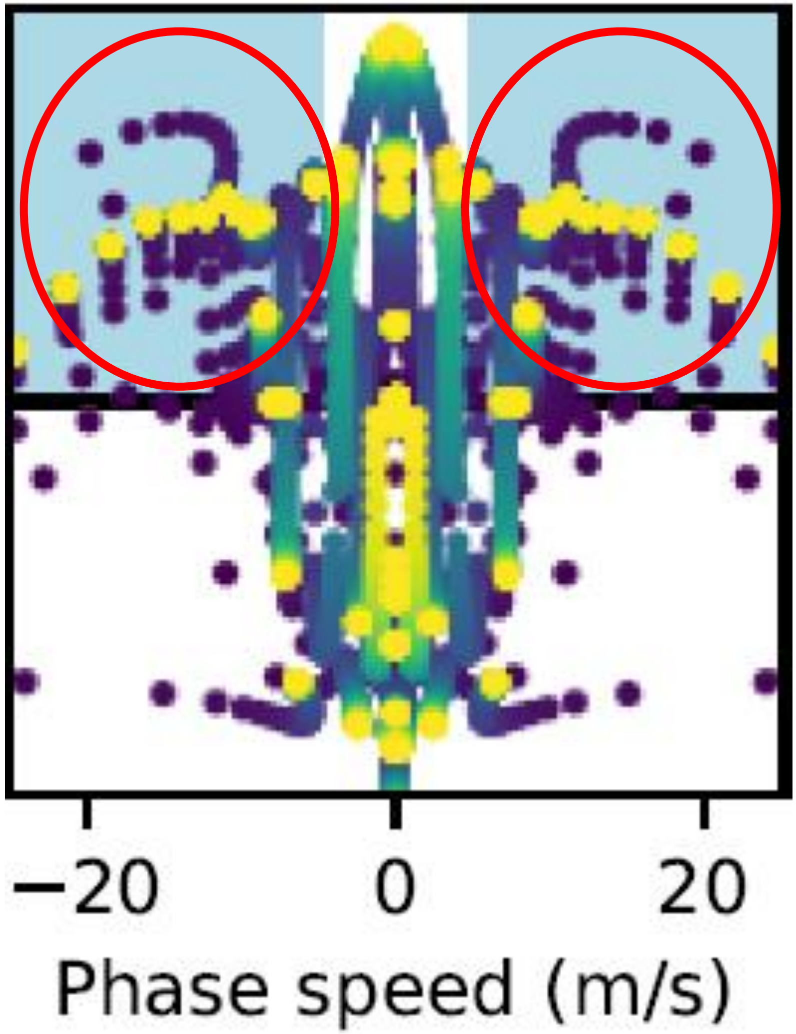

BF, MSE=0.40 10³W²m⁻⁴
CI, MSE=0.49 10³W²m⁻⁴
BF_CC, MSE=0.44 10³W²m⁻⁴
CI_CC, MSE=0.63 10³W²m⁻⁴
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
Coefficient of determination R²

(Delete) Interpretability-based methods



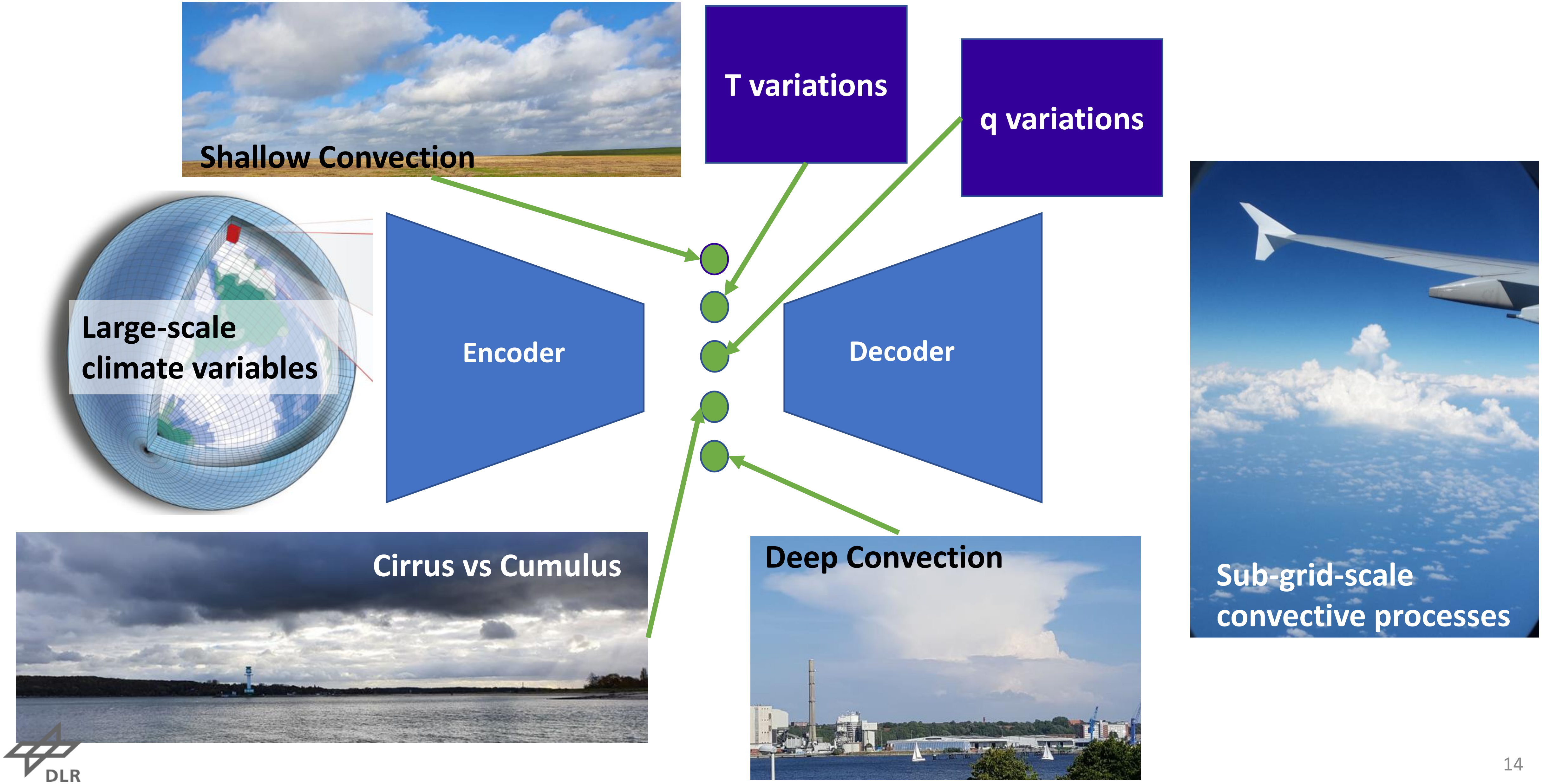
Spurious unstable propagating modes

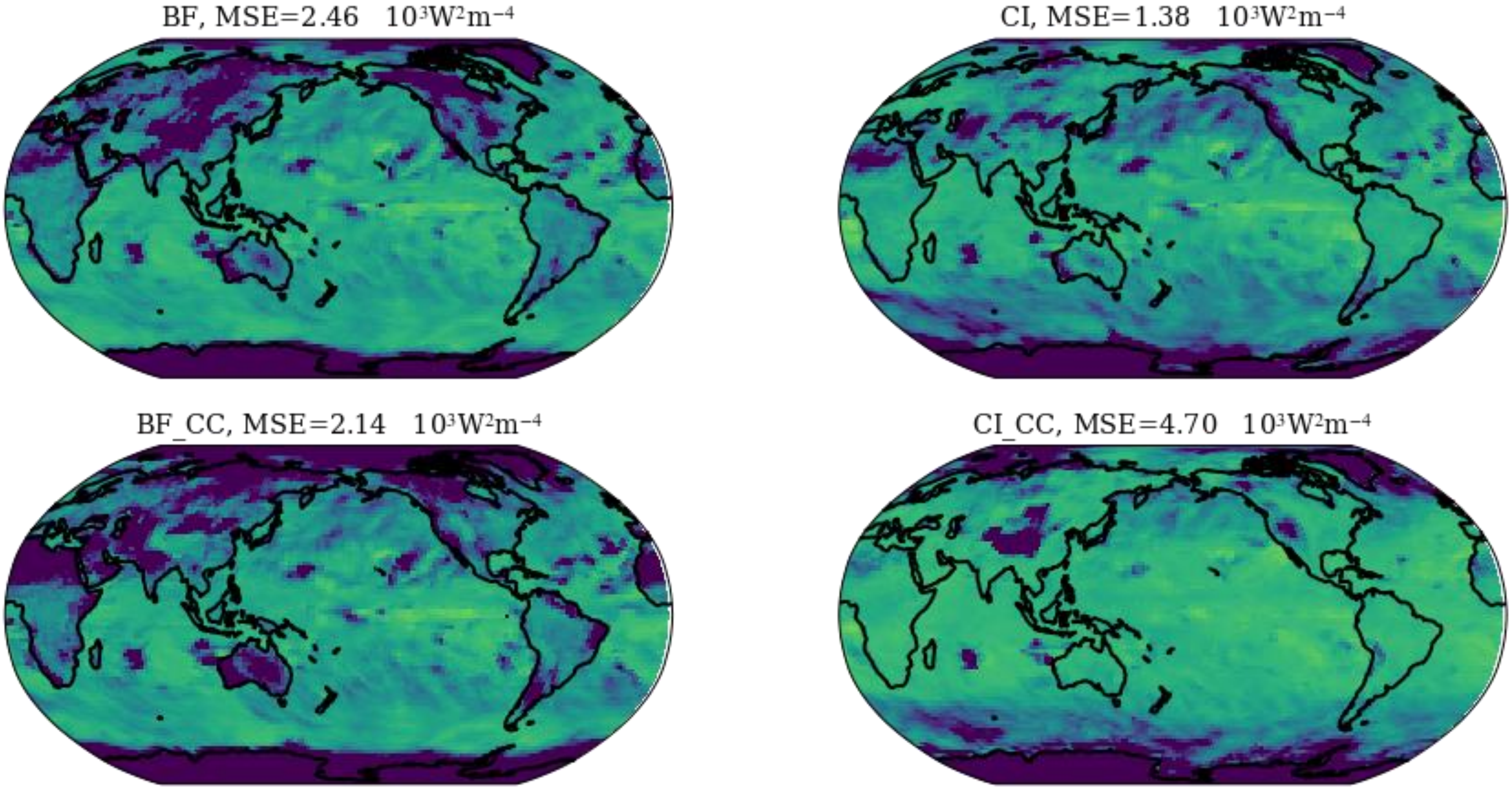
Couple to linearized gravity wave dynamics



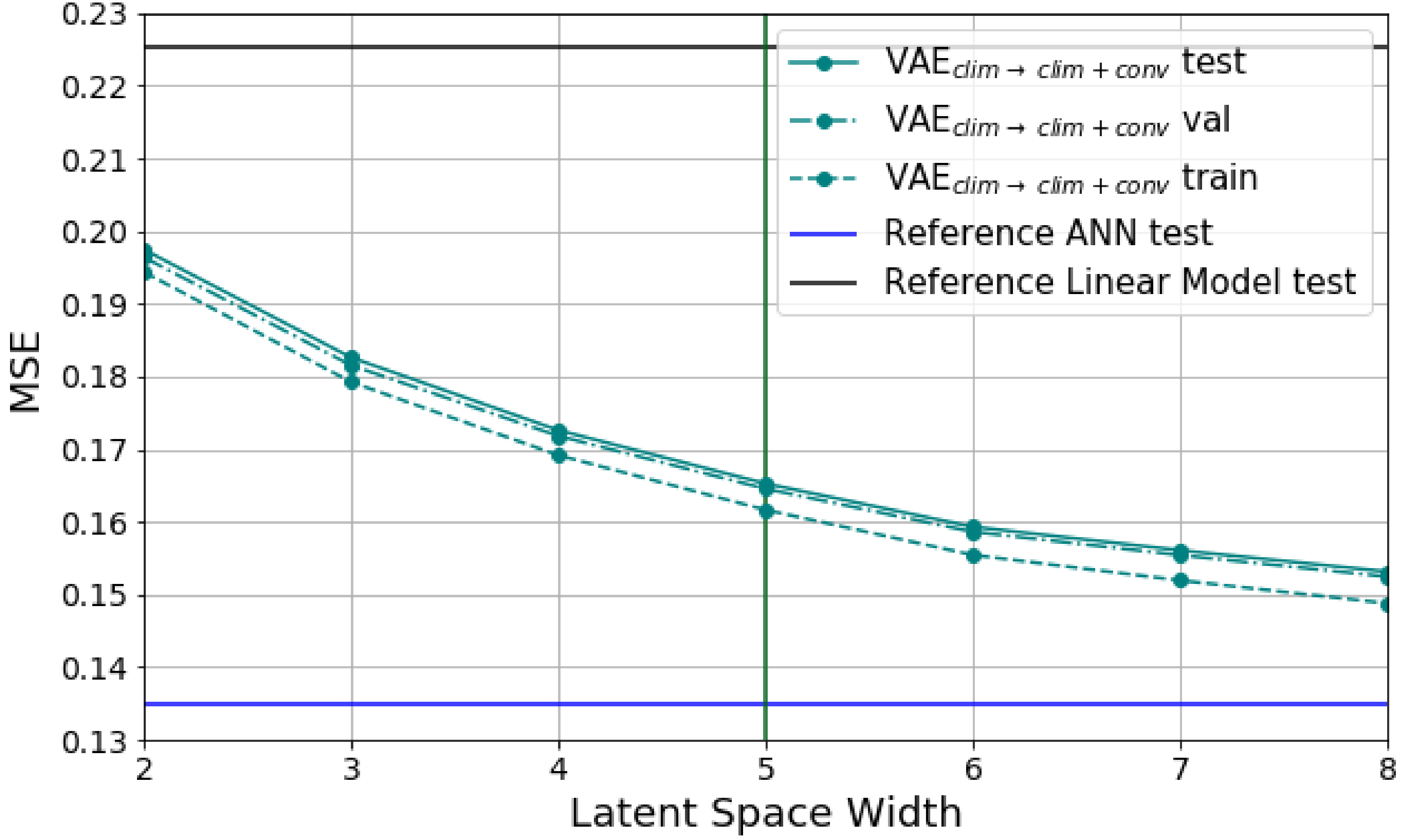
See: Kuang (2018, 2007), Herman and Kuang (2013), Beucler et al. (2018), ***Brenowitz, Beucler et al. (2020)***

Convective Processes are complex, yes we can encode them!




BF, MSE=2.46 $10^3 W^2 m^{-4}$
CI, MSE=1.38 $10^3 W^2 m^{-4}$
BF_CC, MSE=2.14 $10^3 W^2 m^{-4}$
CI_CC, MSE=4.70 $10^3 W^2 m^{-4}$

There has been too many inputs
BF Dim reduction ~= causal relevance



0.23
0.22
0.21
0.20
0.19
0.18
0.17
0.16
0.15
0.14
0.13
MSE
2
3
4
5
6
7
8
Latent Space Width
VAE$_{clim \rightarrow clim + conv}$ test
VAE$_{clim \rightarrow clim + conv}$ val
VAE$_{clim \rightarrow clim + conv}$ train
Reference ANN test
Reference Linear Model test