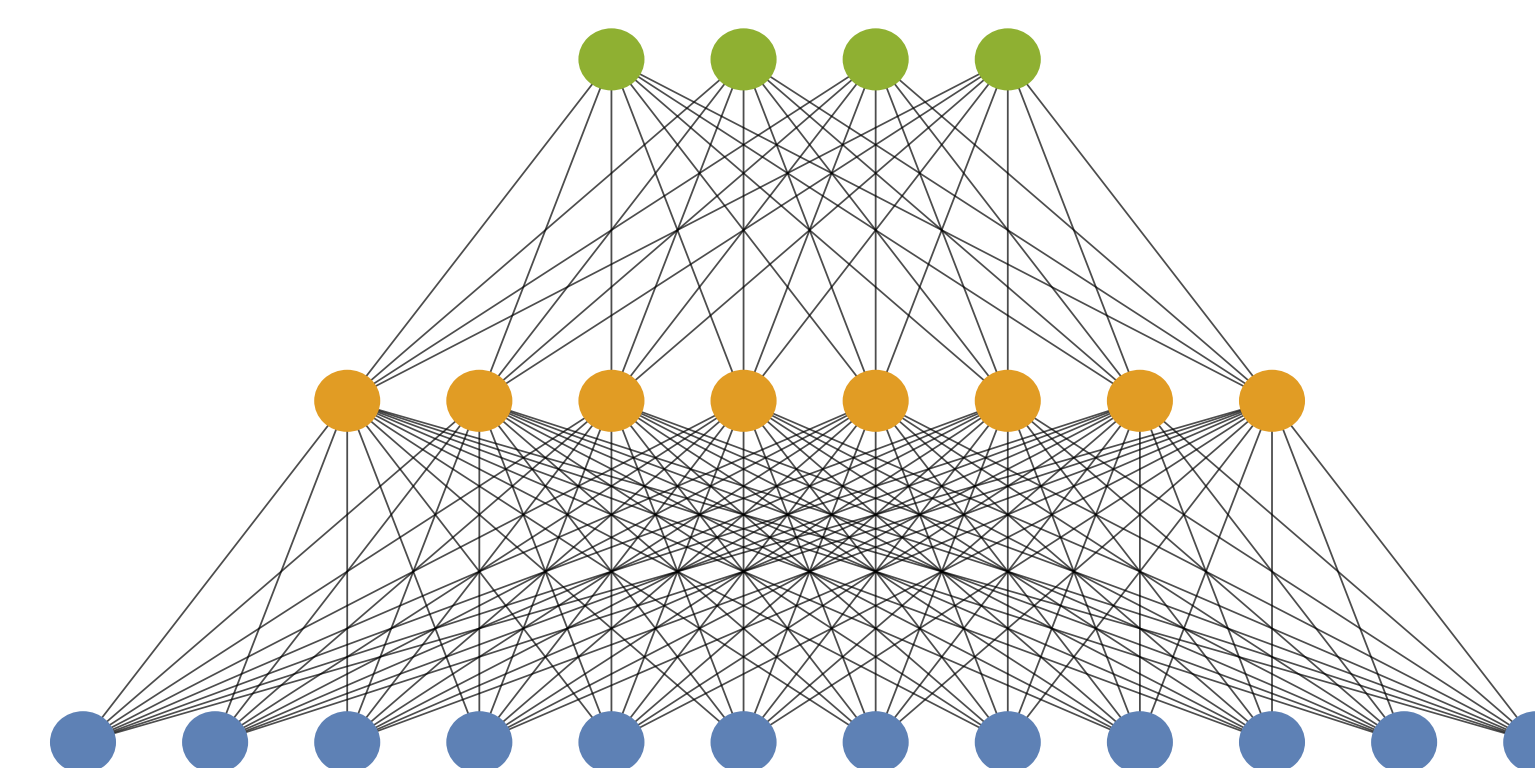
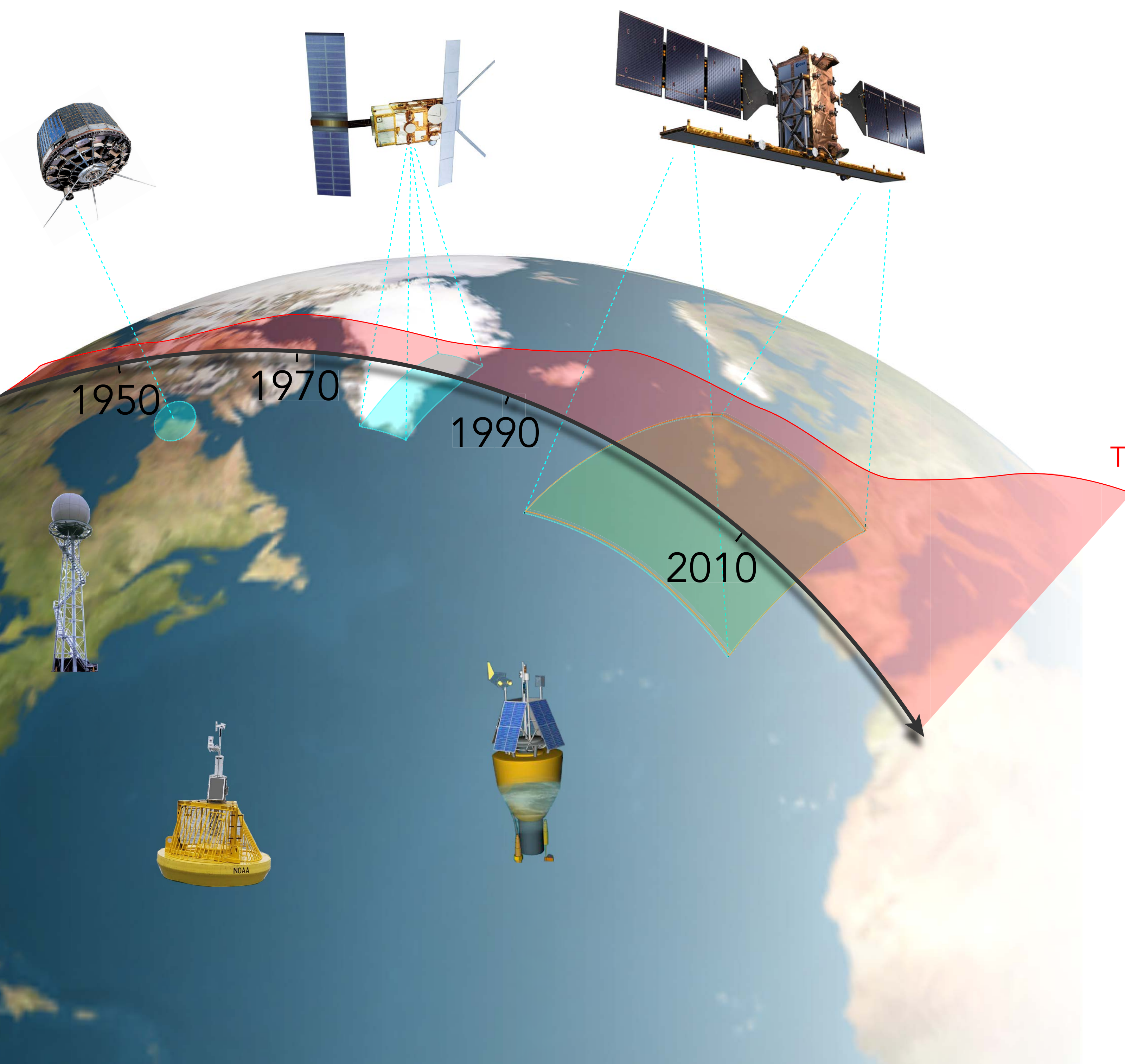


AtmoRep

Large scale representation learning of
atmospheric dynamics

Christian Lessig, Ilaria Luise, Martin Schultz, et al.



large scale representation learning

applications

scientific insight

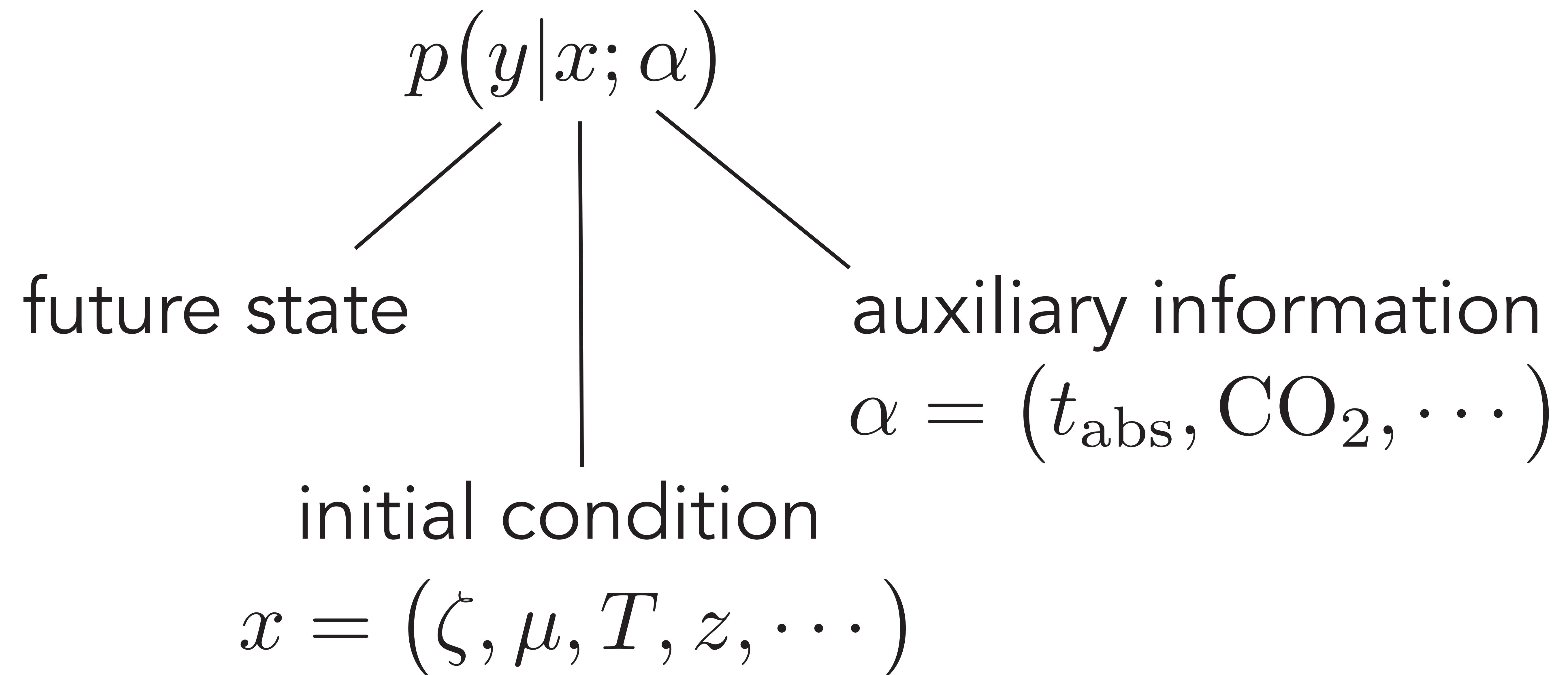
Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

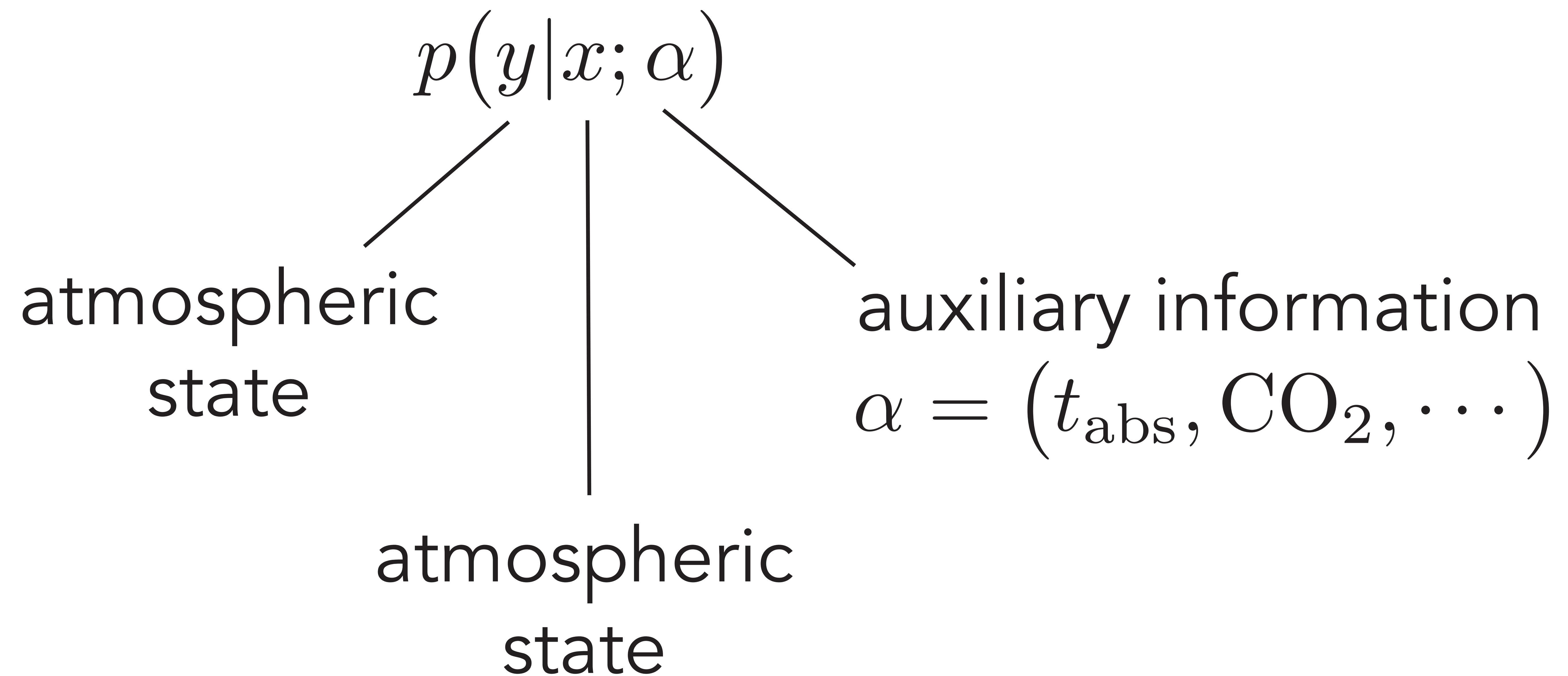
Theoretical formulation

- Atmosphere as stochastic dynamical system:



Theoretical formulation

- Atmosphere as stochastic dynamical system:



Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

- Forecasting, downscaling, interpolation: $p(y|x; \alpha)$

Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

- › Forecasting, downscaling, interpolation: $p(y|x; \alpha)$
- › Counterfactuals: $p(y|x; \hat{\alpha})$

Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

- › Forecasting, downscaling, interpolation: $p(y|x; \alpha)$

- › Counterfactuals: $p(y|x; \hat{\alpha})$

- › Climate: $p_{\alpha}(y) = \int p(y|x; \alpha) p(x) dx$

Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

- › Medium/extended range forecasting:

$$p(x_T|x_0; \alpha) = \prod_{i=1}^T p_{\theta}(x_{i+1}|x_i; \alpha)$$

Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha) \quad \text{highly complex, in-} \\ \text{stationary distribution}$$

- › Medium/extended range forecasting:

$$p(x_T|x_0; \alpha) = \prod_{i=1}^T p_{\theta}(x_{i+1}|x_i; \alpha)$$

Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

- Numerical statistical atmospheric model:

$$\tilde{p}(\tilde{y}|\tilde{x}; \tilde{\alpha})$$

Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

- Numerical statistical atmospheric model:

$$\tilde{p}(\tilde{y}|\tilde{x}; \tilde{\alpha}) \approx p(y|x; \alpha)$$

Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

- Numerical statistical atmospheric model:

$$\tilde{p}(\tilde{y}|\tilde{x}; \tilde{\alpha}) \approx p(y|x; \alpha)$$

approx. future state

approx. initial condition

Theoretical formulation

- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

- Numerical statistical atmospheric model:

$$\tilde{p}(\tilde{y}|\tilde{x}; \tilde{\alpha}) \approx p(y|x; \alpha)$$

highly complex,
instationary
distribution

Theoretical formulation

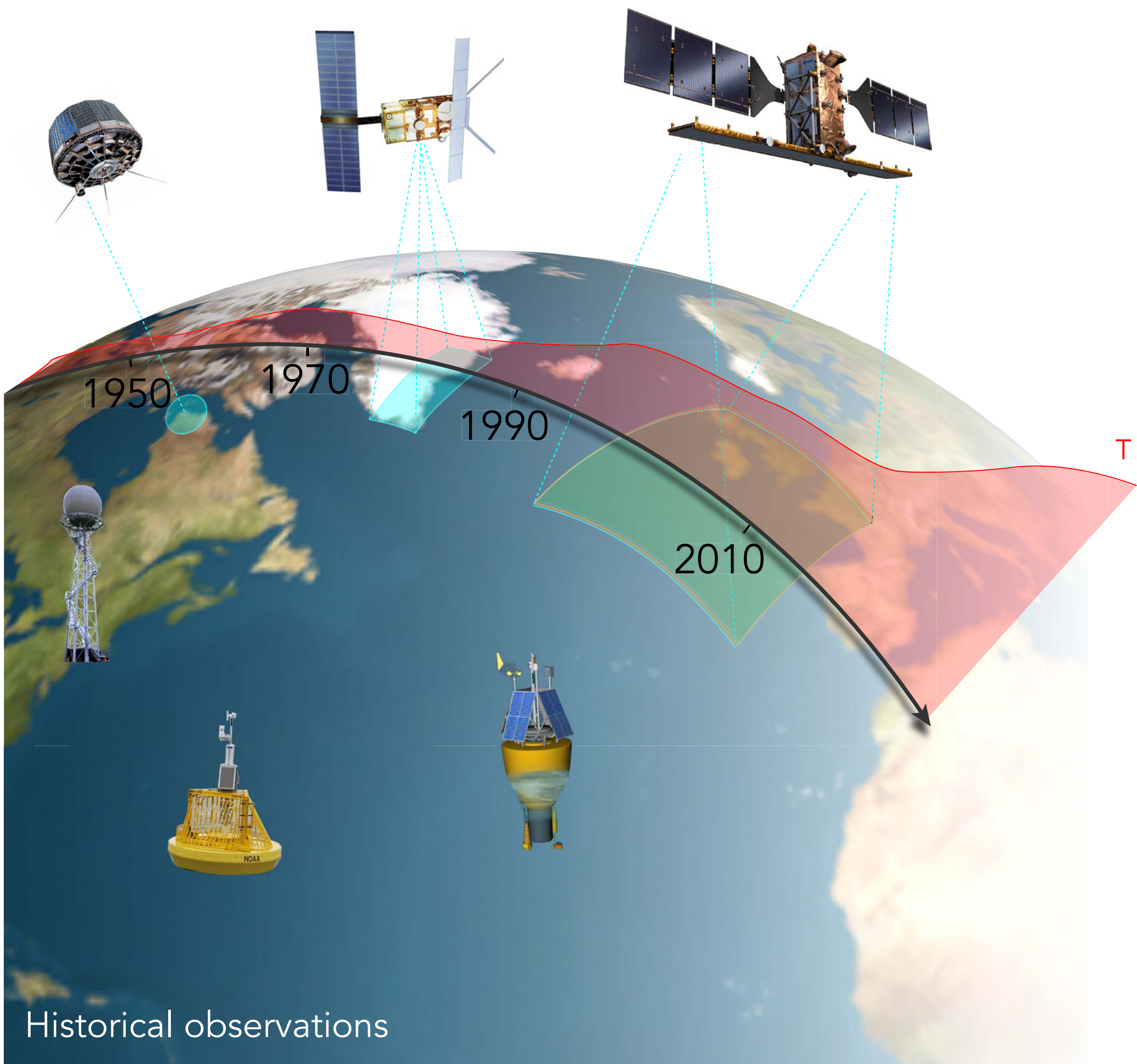
- Atmosphere as stochastic dynamical system:

$$p(y|x; \alpha)$$

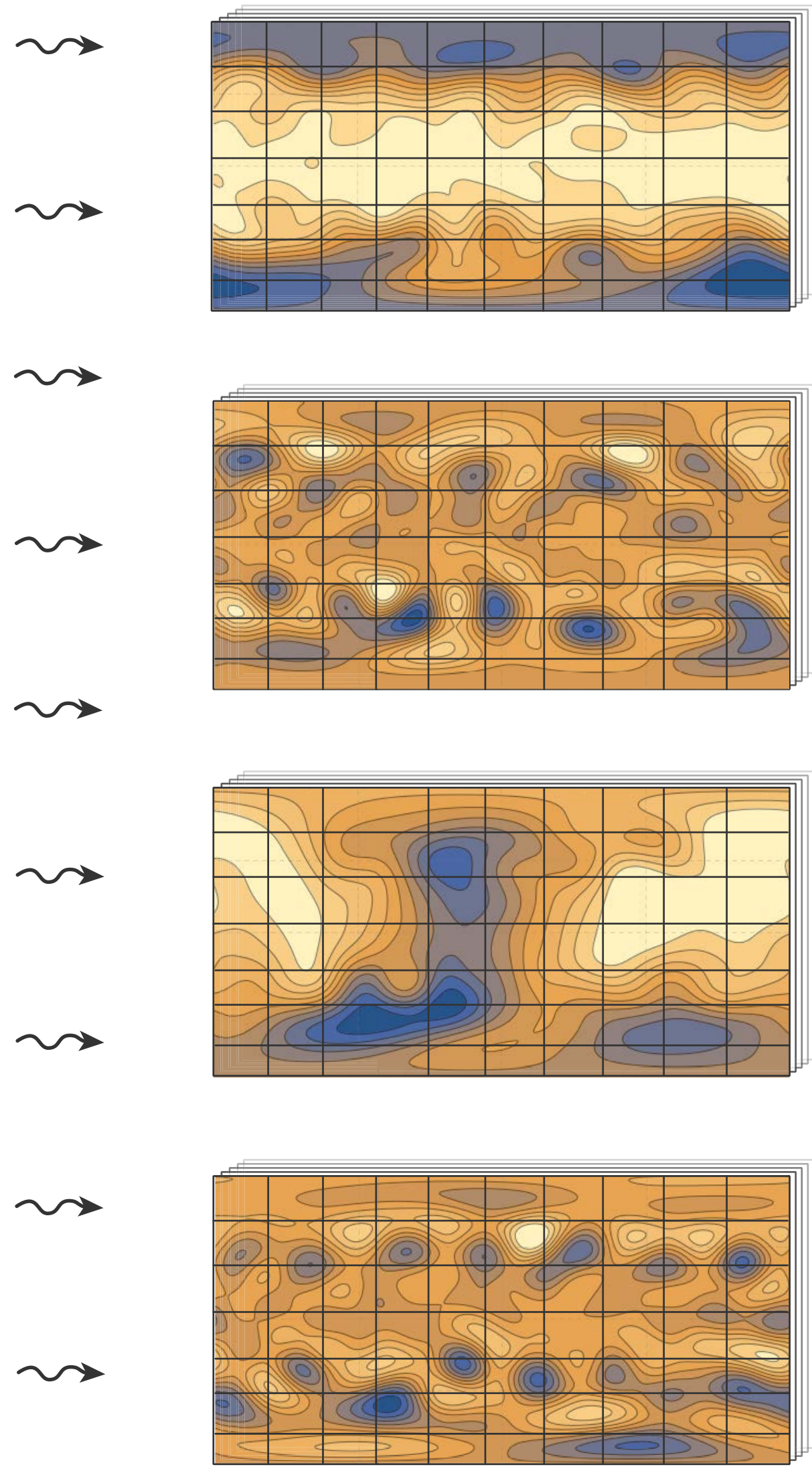
- Numerical statistical atmospheric model:

very large neural network $\tilde{p}_\theta(\tilde{y}|\tilde{x}; \tilde{\alpha}) \approx p(y|x; \alpha)$ highly complex, instationary distribution

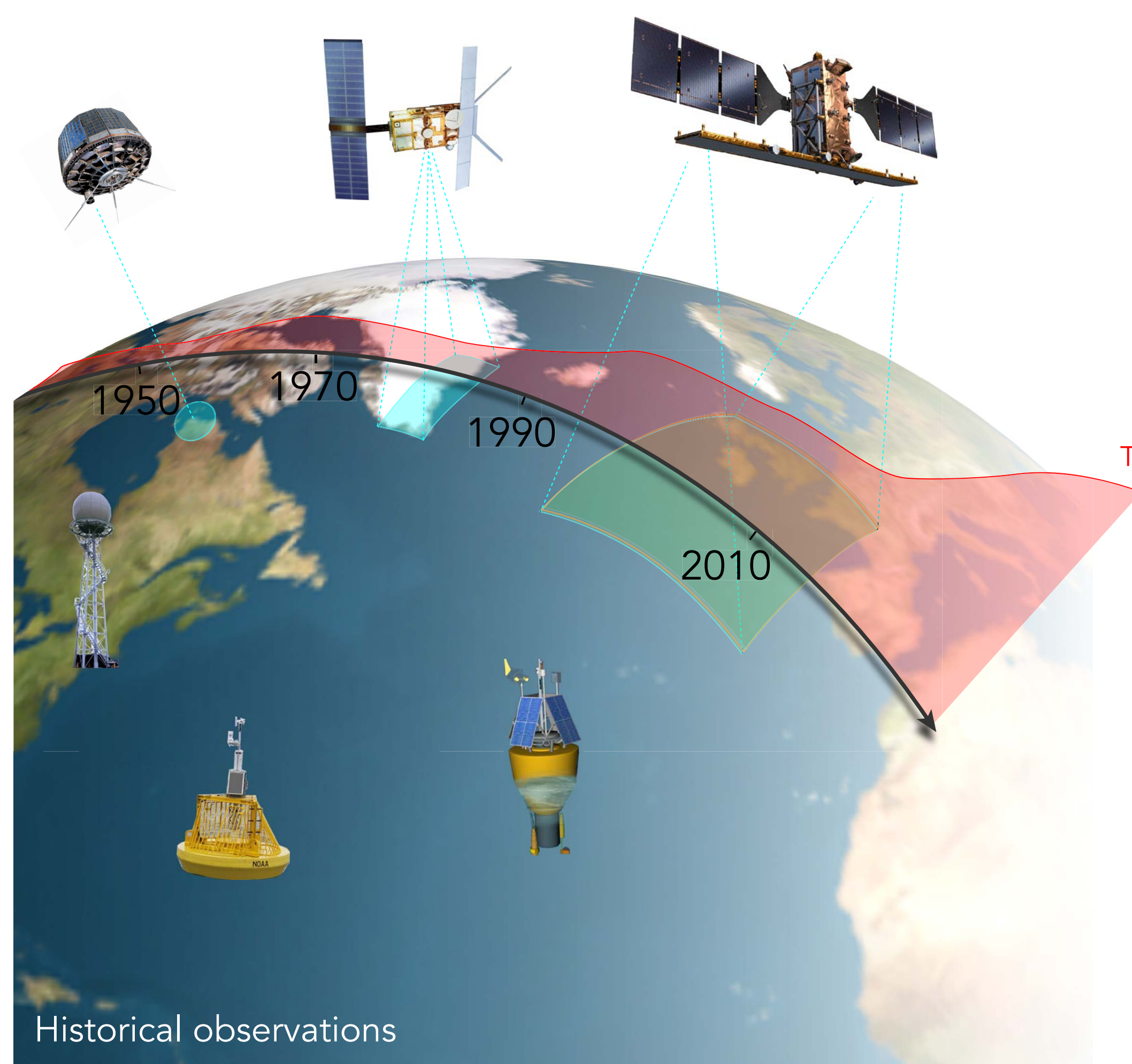
AtmoRep



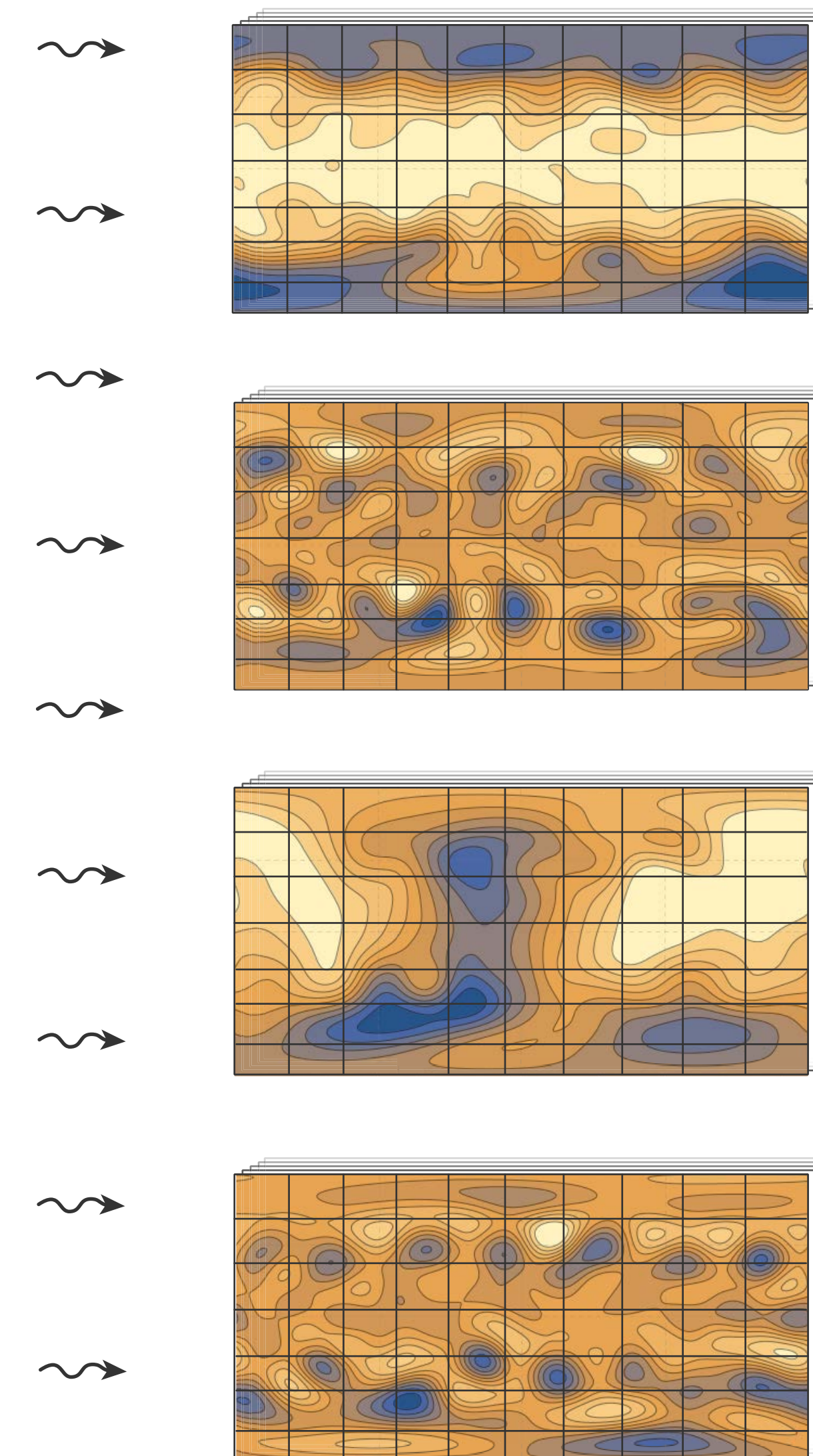
ERA5 reanalysis



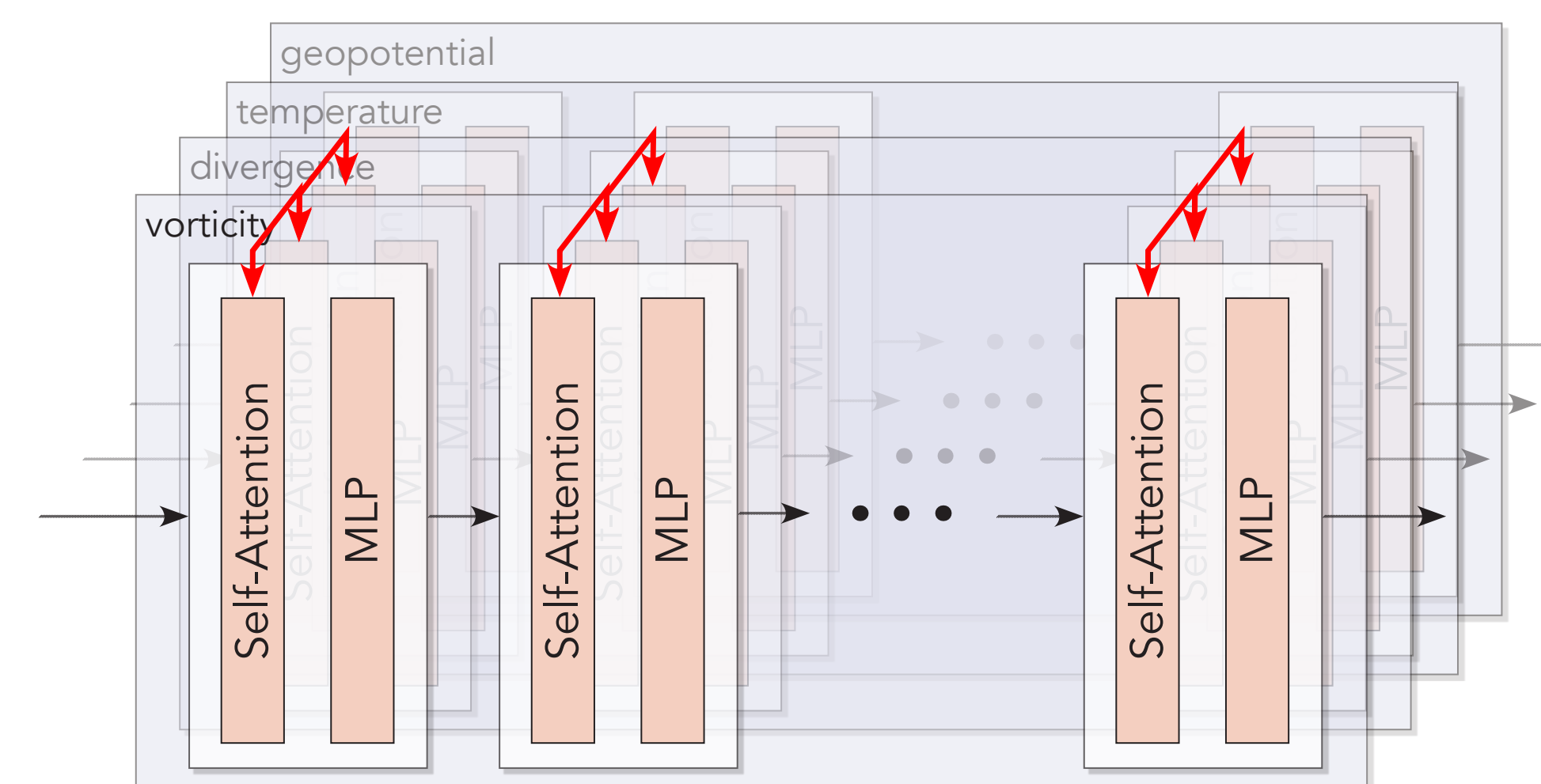
AtmoRep



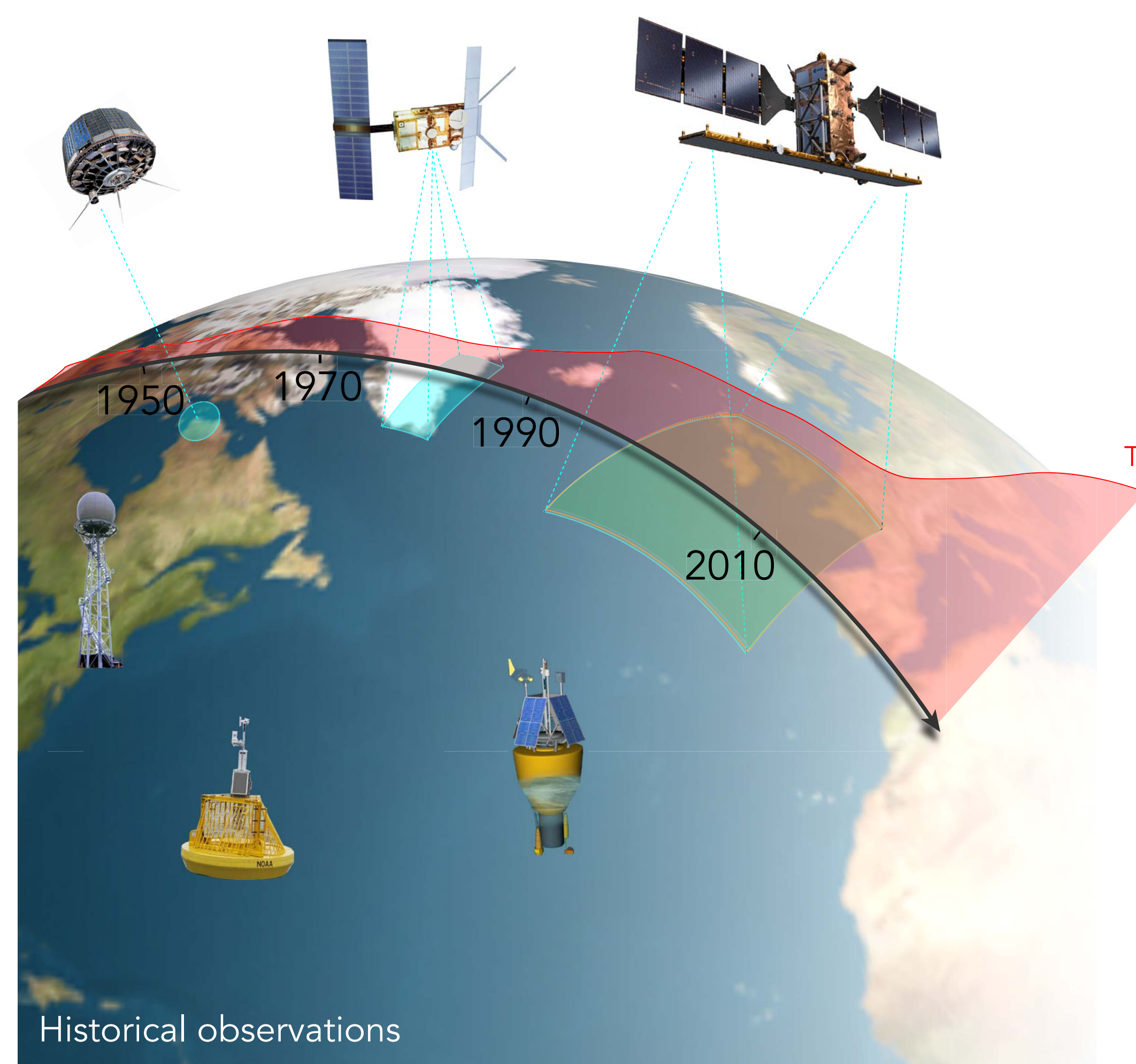
ERA5 reanalysis



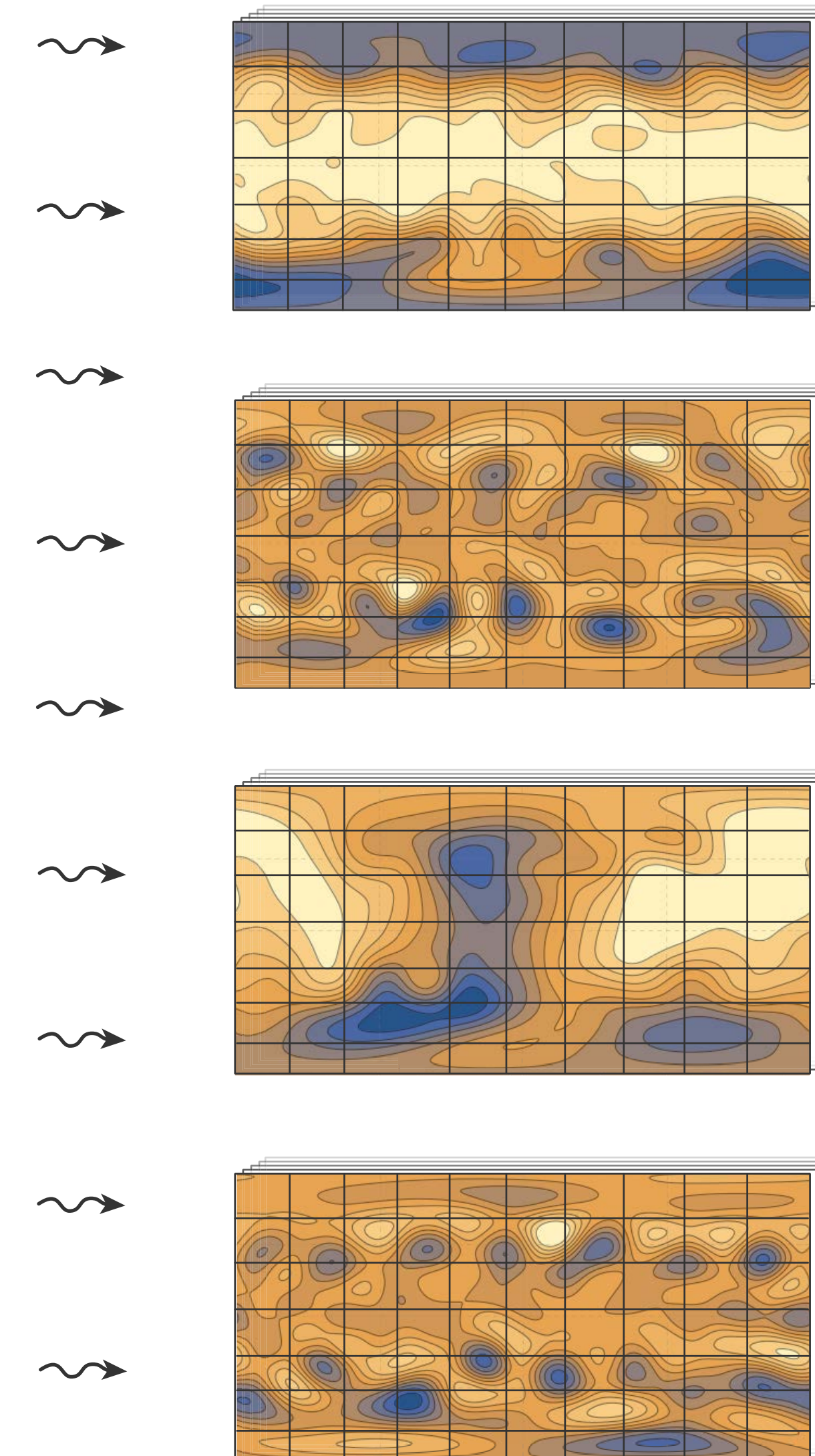
$$p_{\theta}(y|x)$$



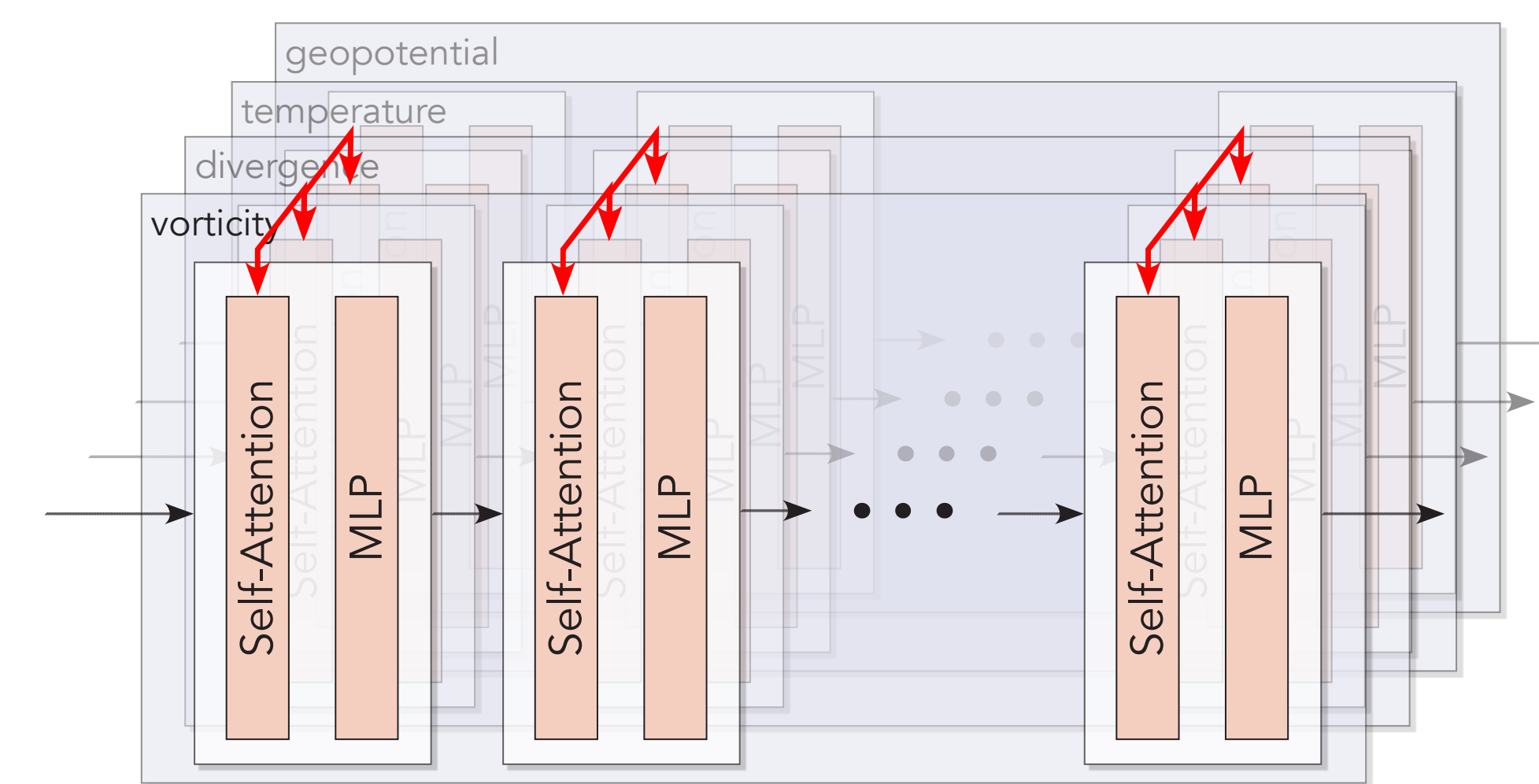
AtmoRep



ERA5 reanalysis

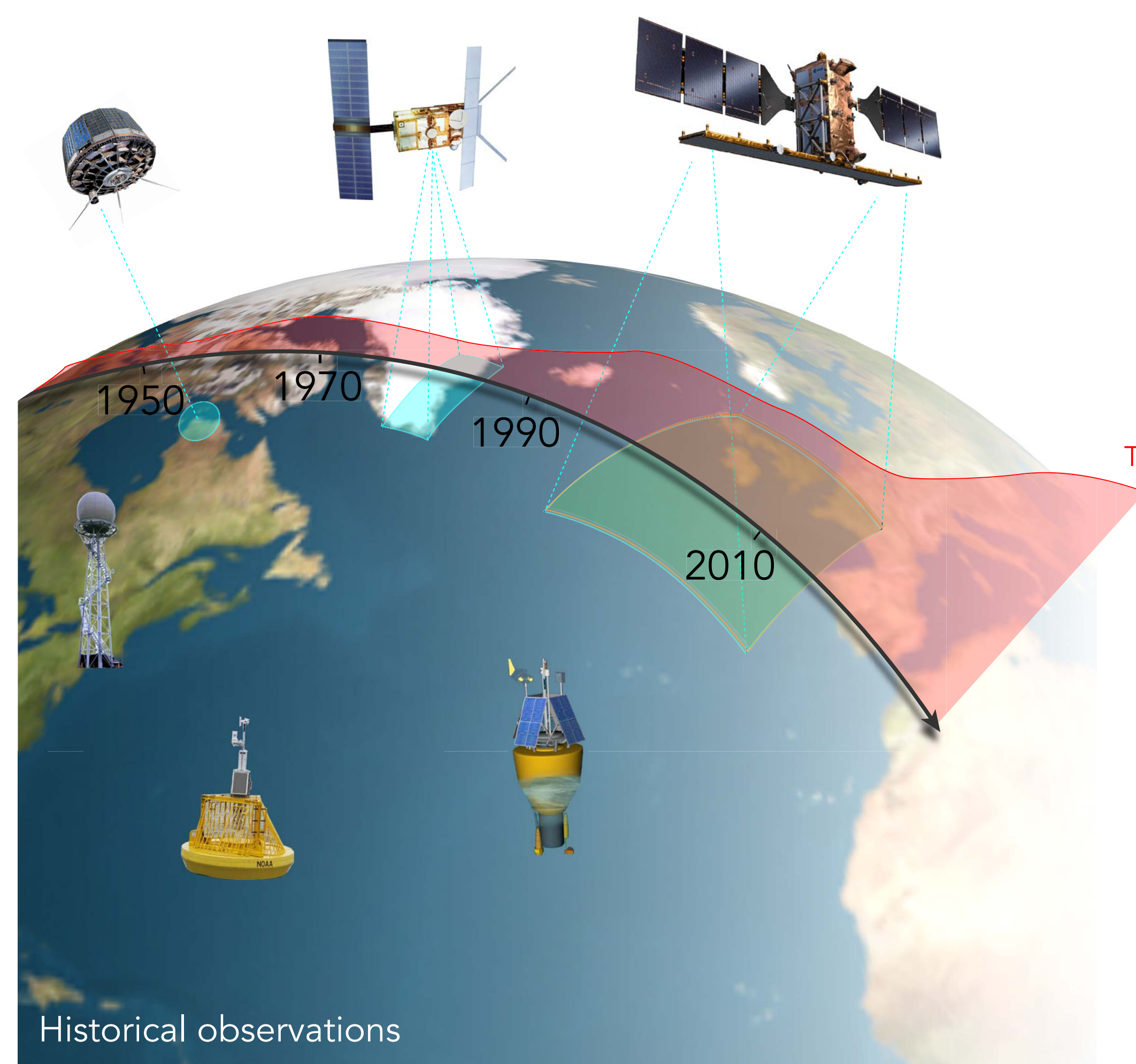


$$p_{\theta}(y|x)$$

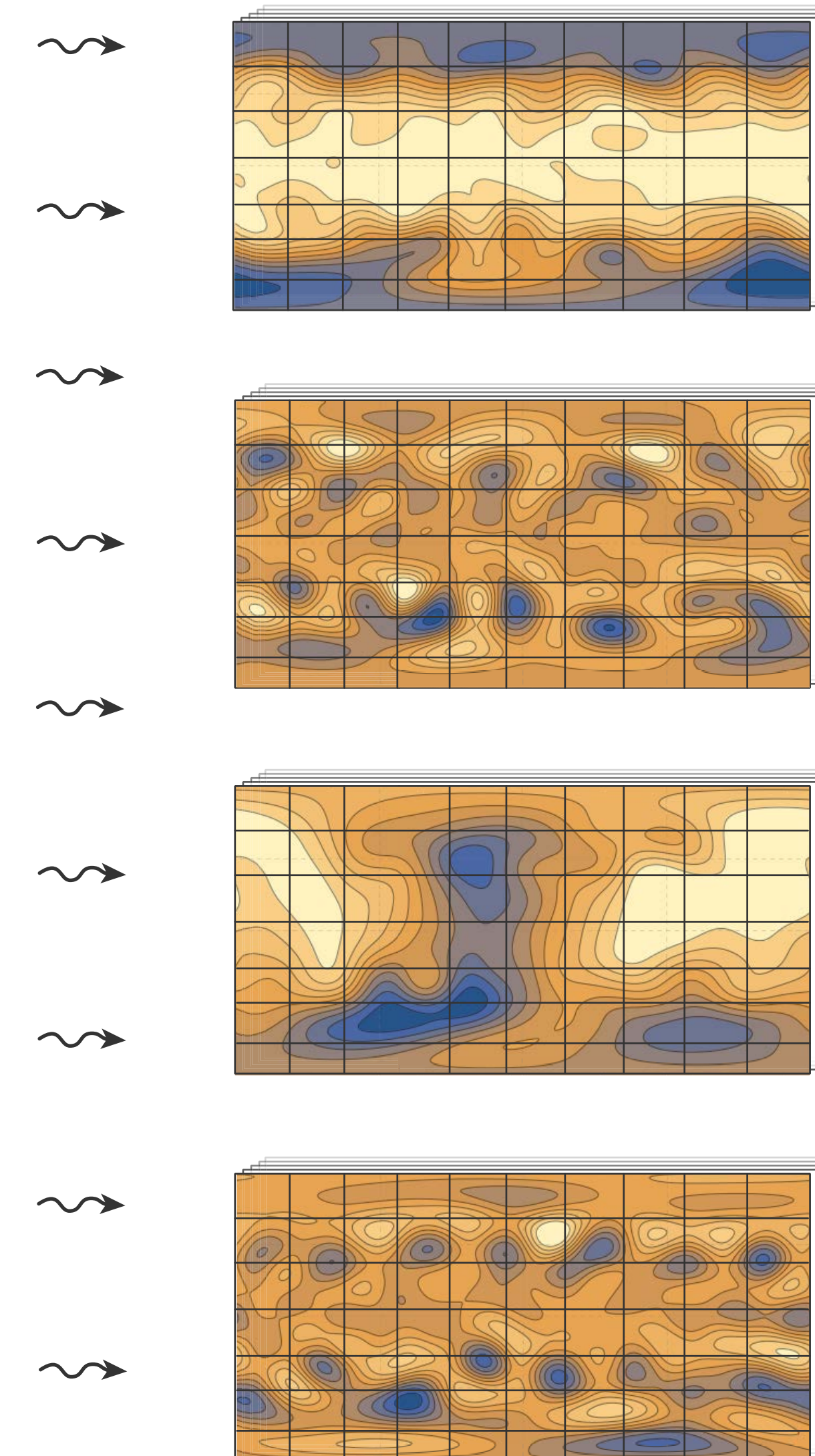


large transformer
with 3.5×10^9 parameters

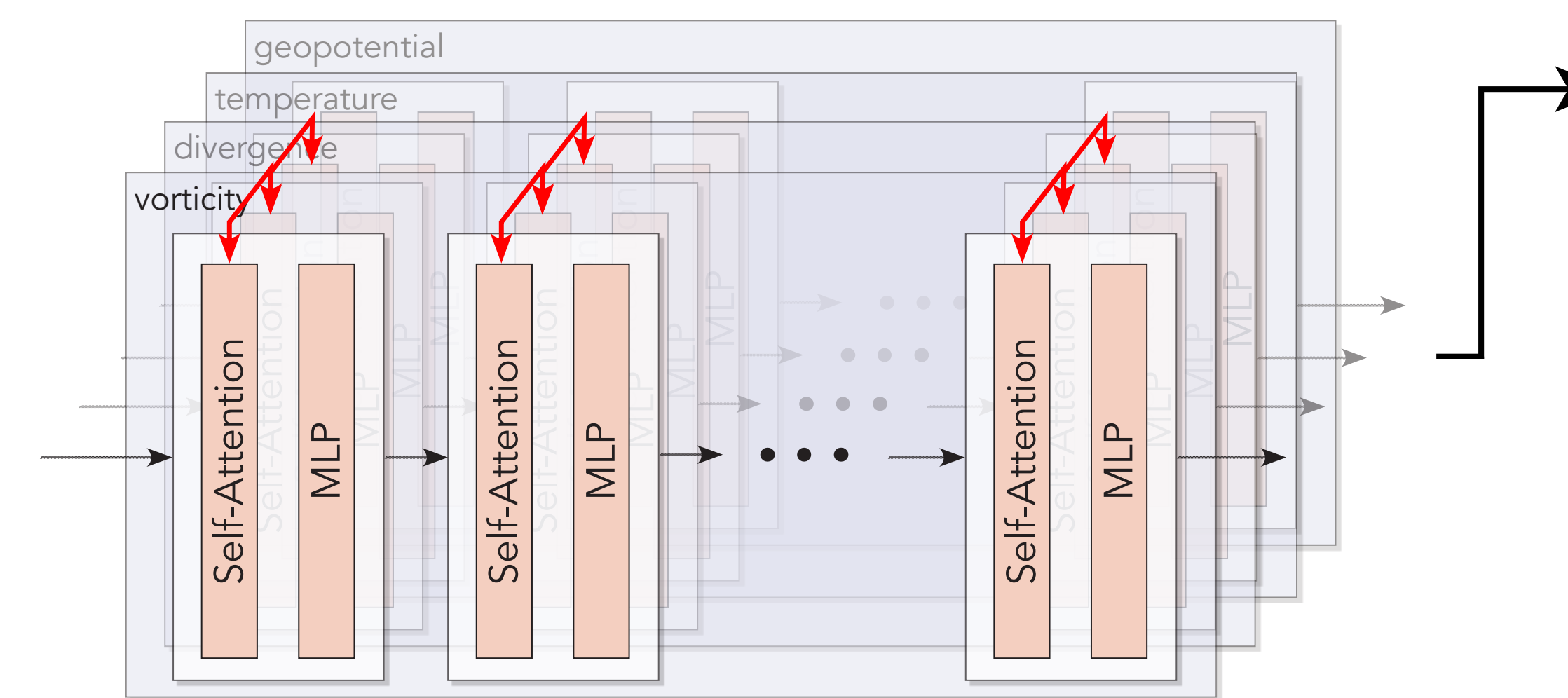
AtmoRep



ERA5 reanalysis

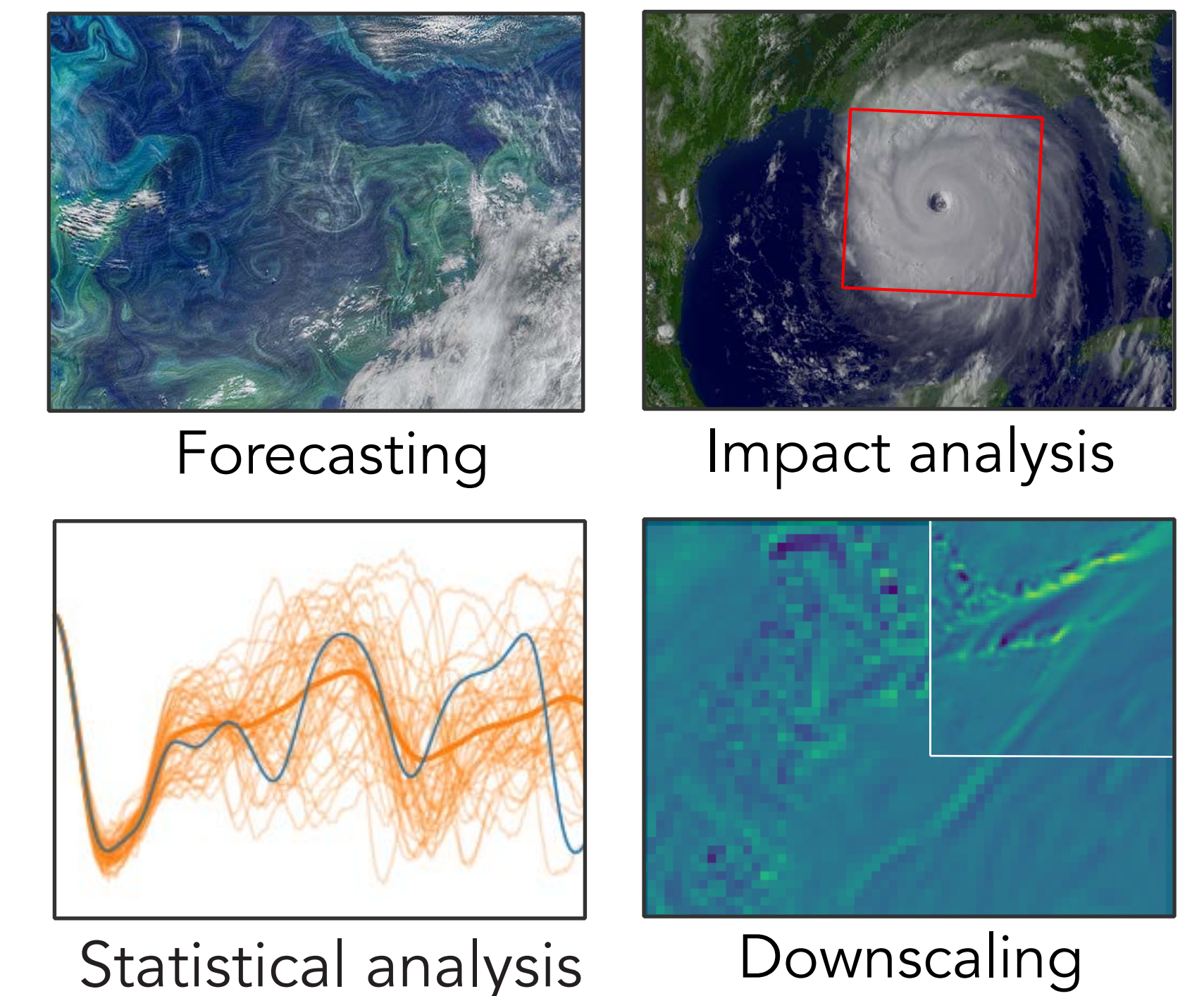


$$p_{\theta}(y|x)$$

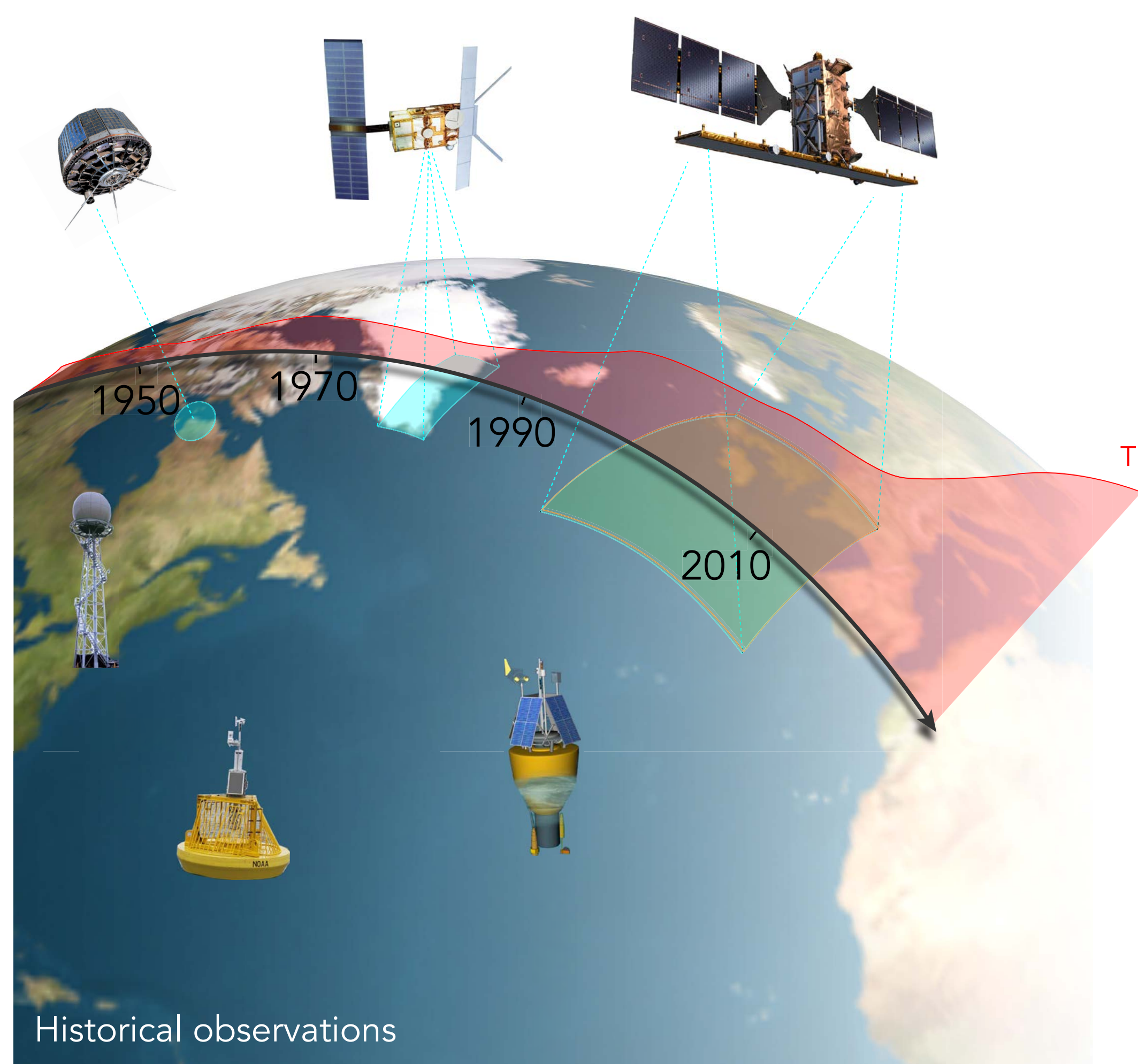


large transformer
with 3.5×10^9 parameters

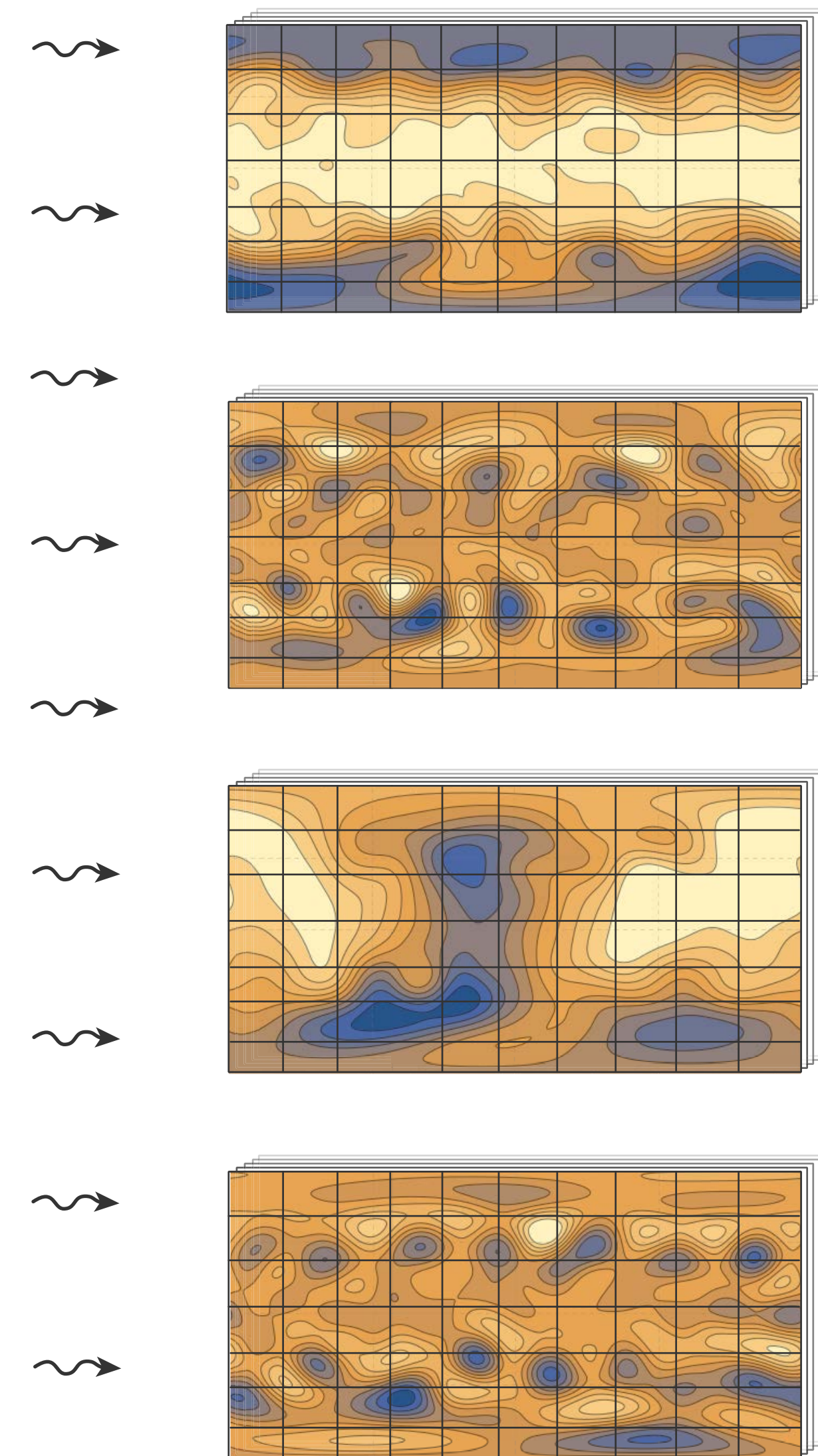
applications



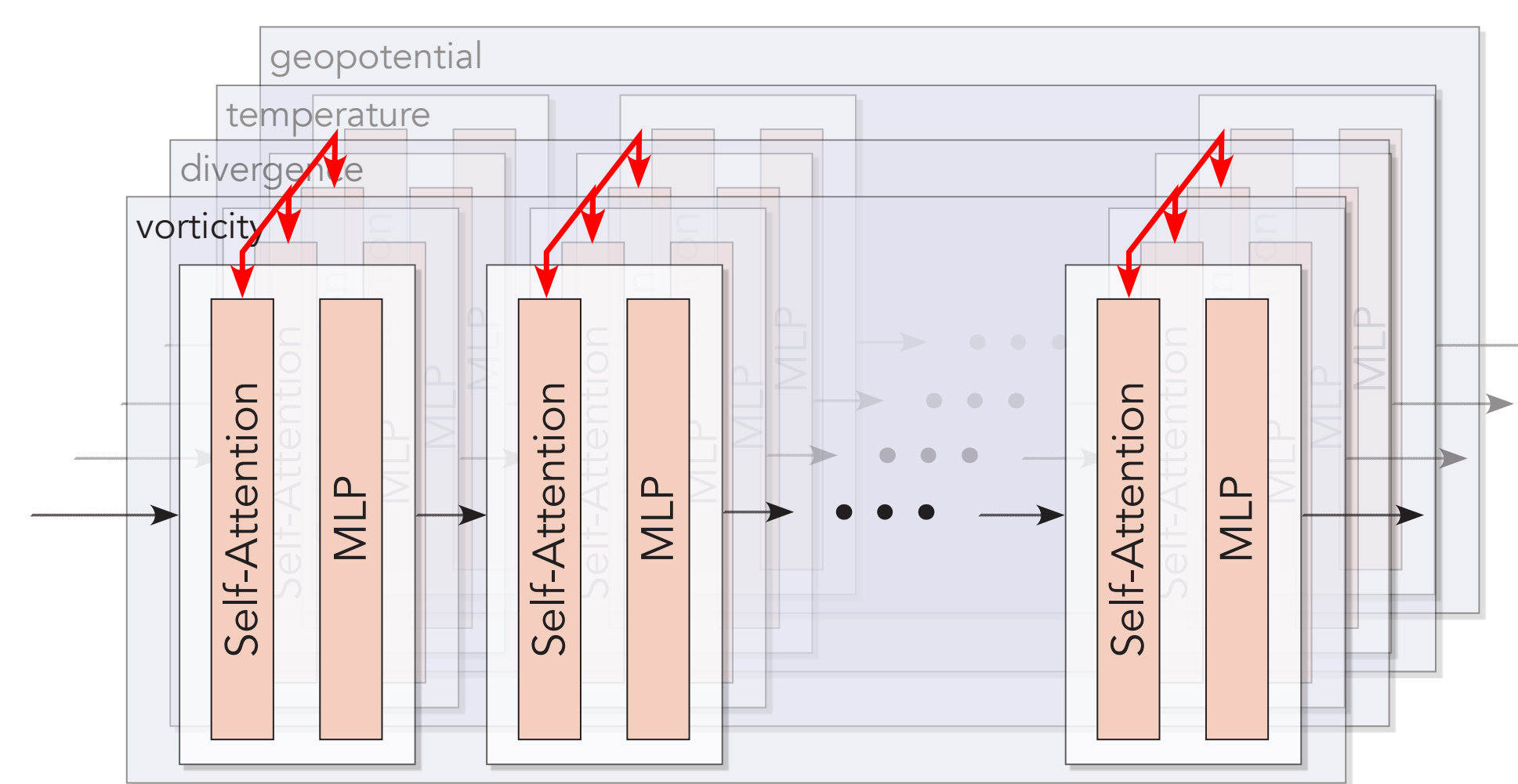
AtmoRep



ERA5 reanalysis

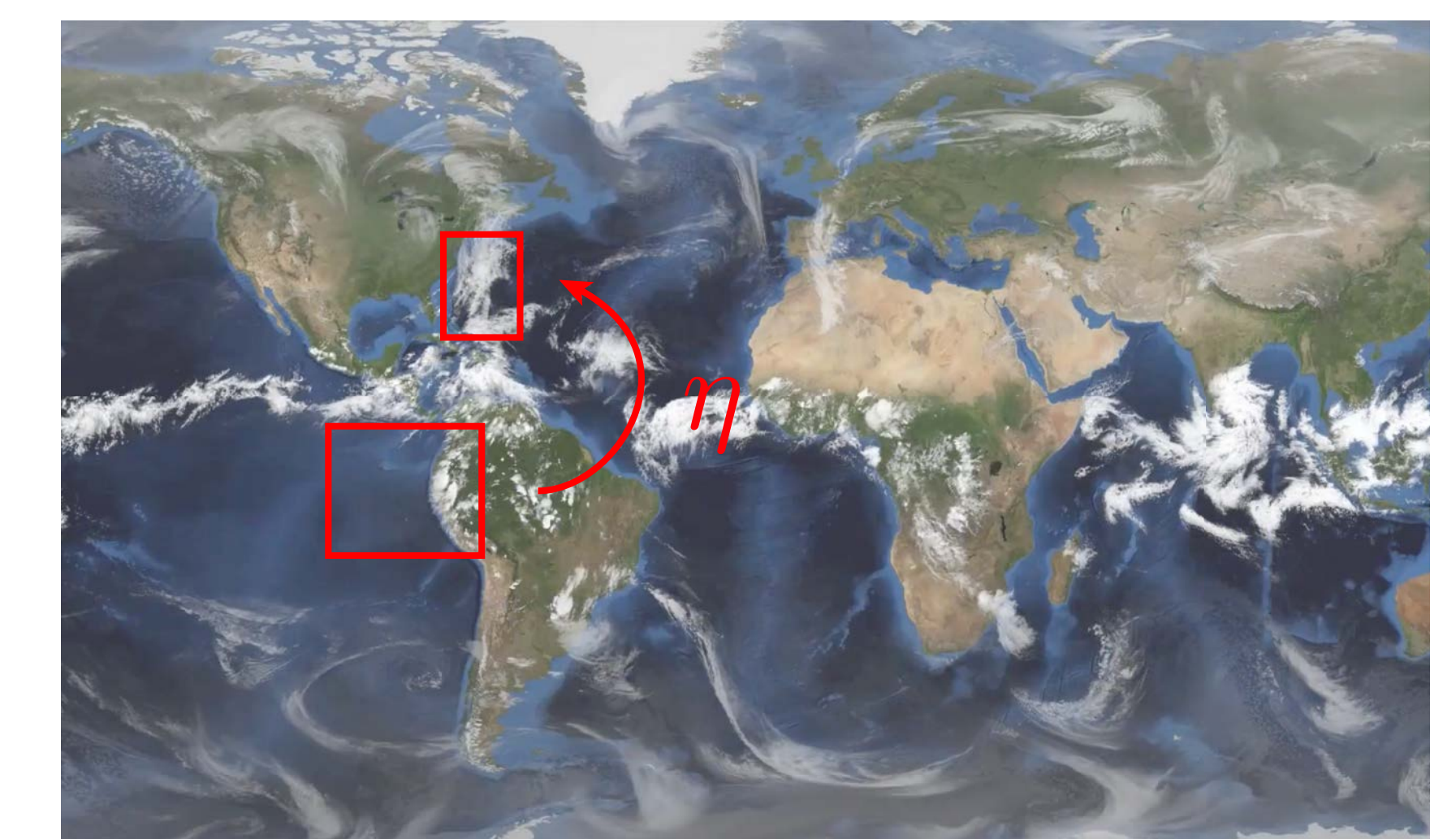
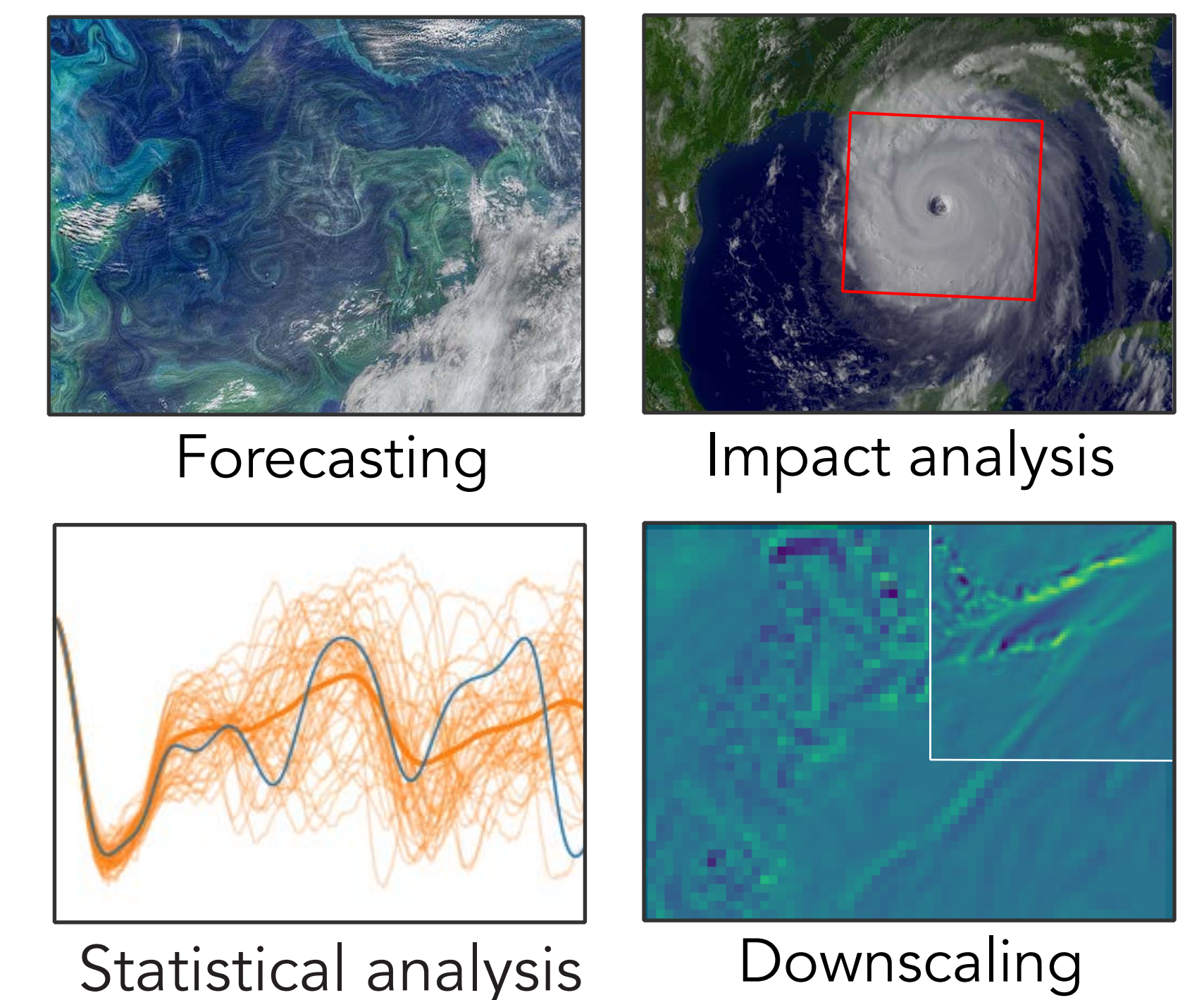


$$p_{\theta}(y|x)$$



large transformer
with 3.5×10^9 parameters

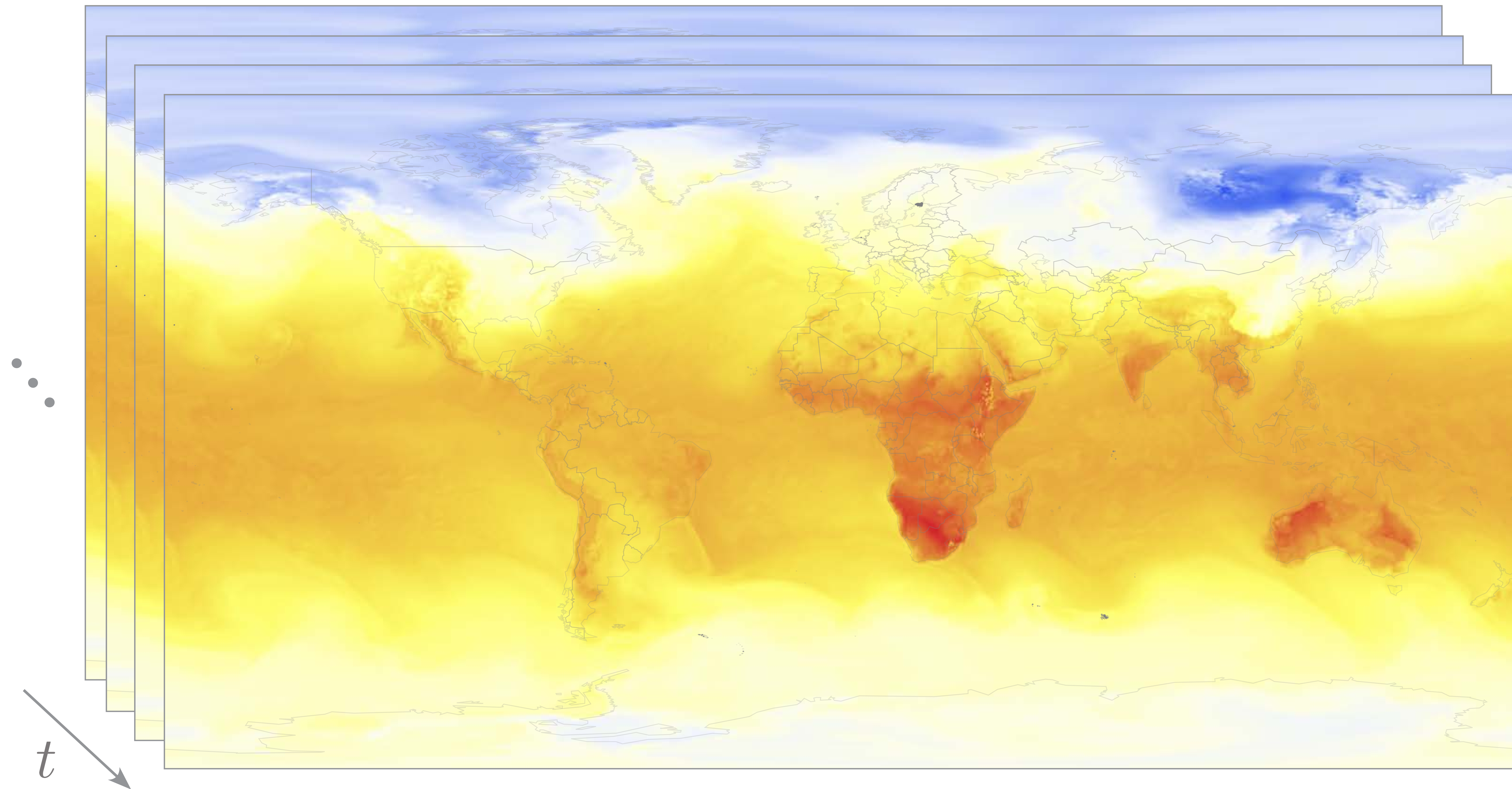
applications



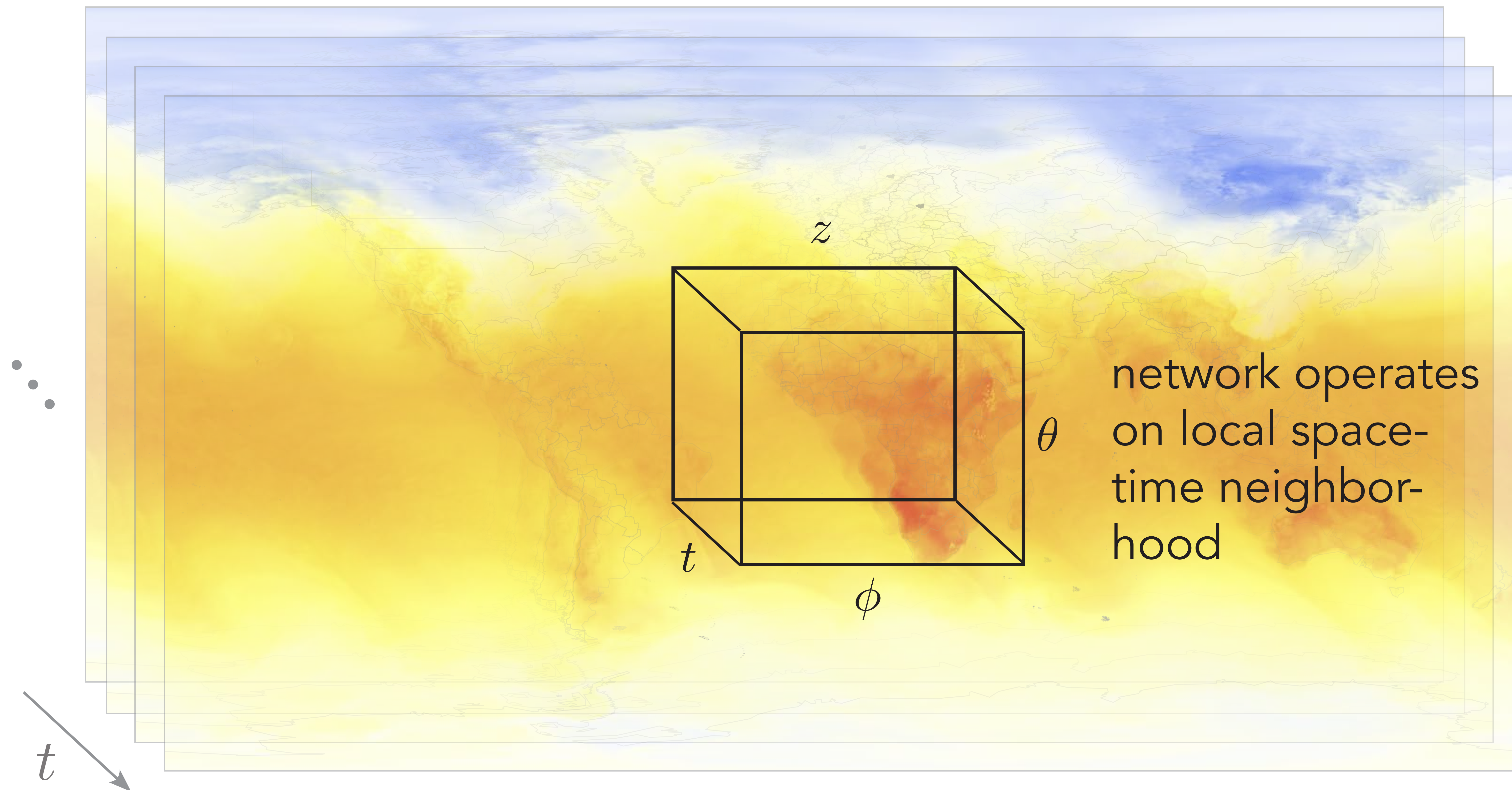
scientific insight

Network architecture

AtmoRep network architecture



AtmoRep network architecture



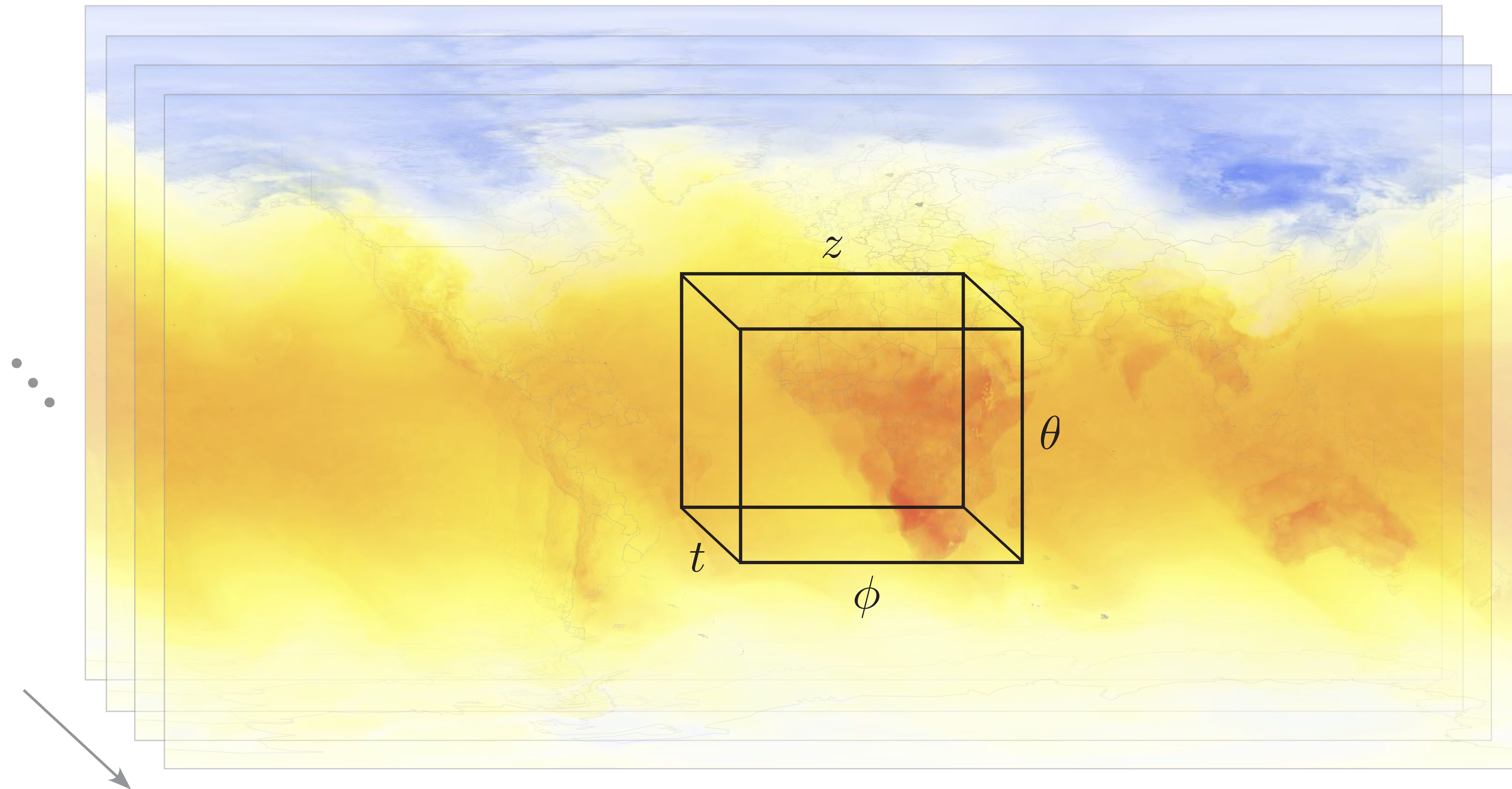
AtmoRep network architecture

- Network is local in space-time
 - › Physics are universally valid
 - › Machine learning model can be leaner and learn faster
 - › Local particularities can be learned by providing time + space position as network input

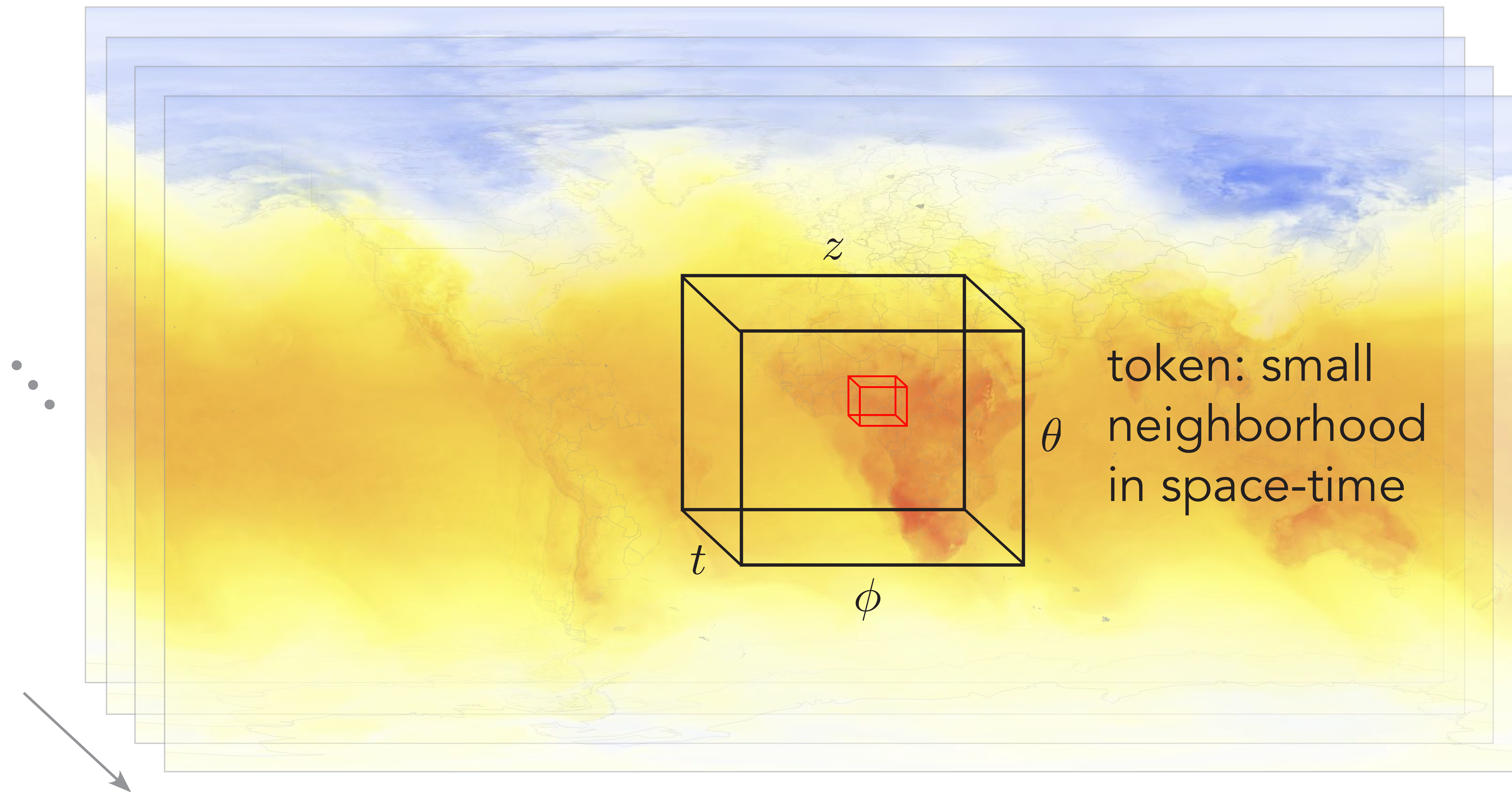
AtmoRep network architecture

- Network is local in space-time
 - › Physics are universally valid
 - › Machine learning model can be leaner and learn faster
 - › Local particularities can be learned by providing time + space position as network input
- Network is an extended transformer architecture

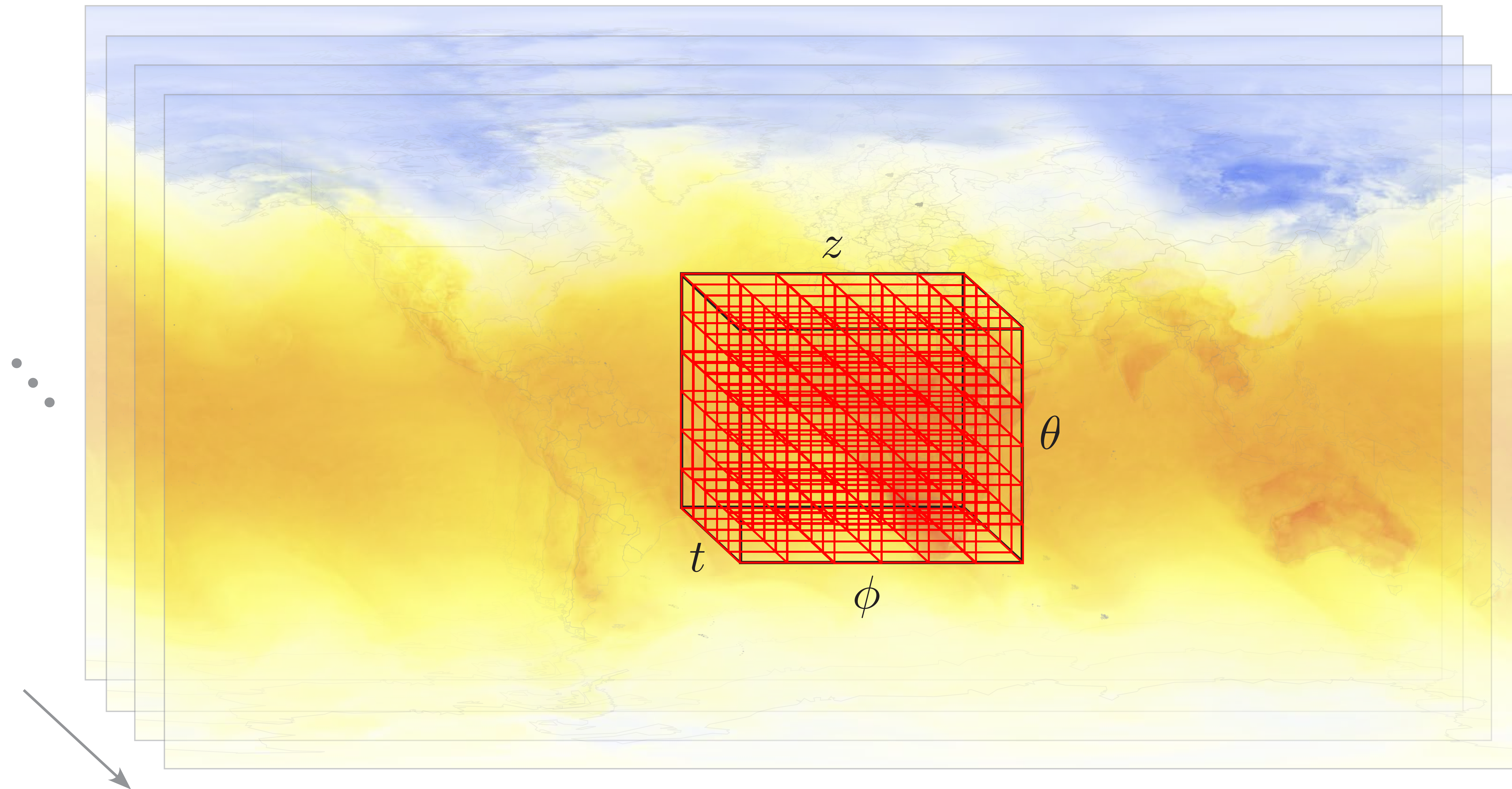
What is a token?



What is a token?

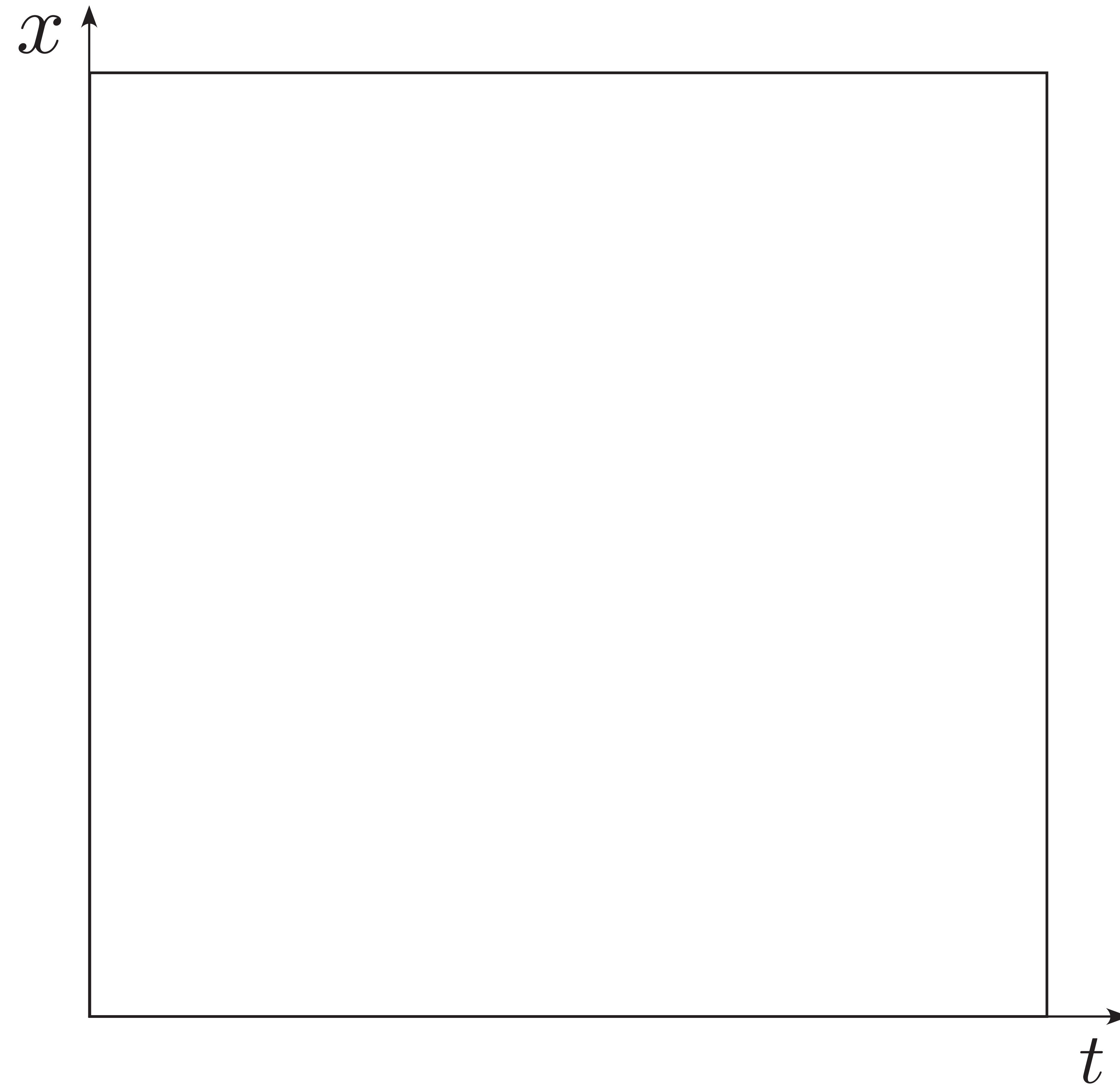


What is a token?



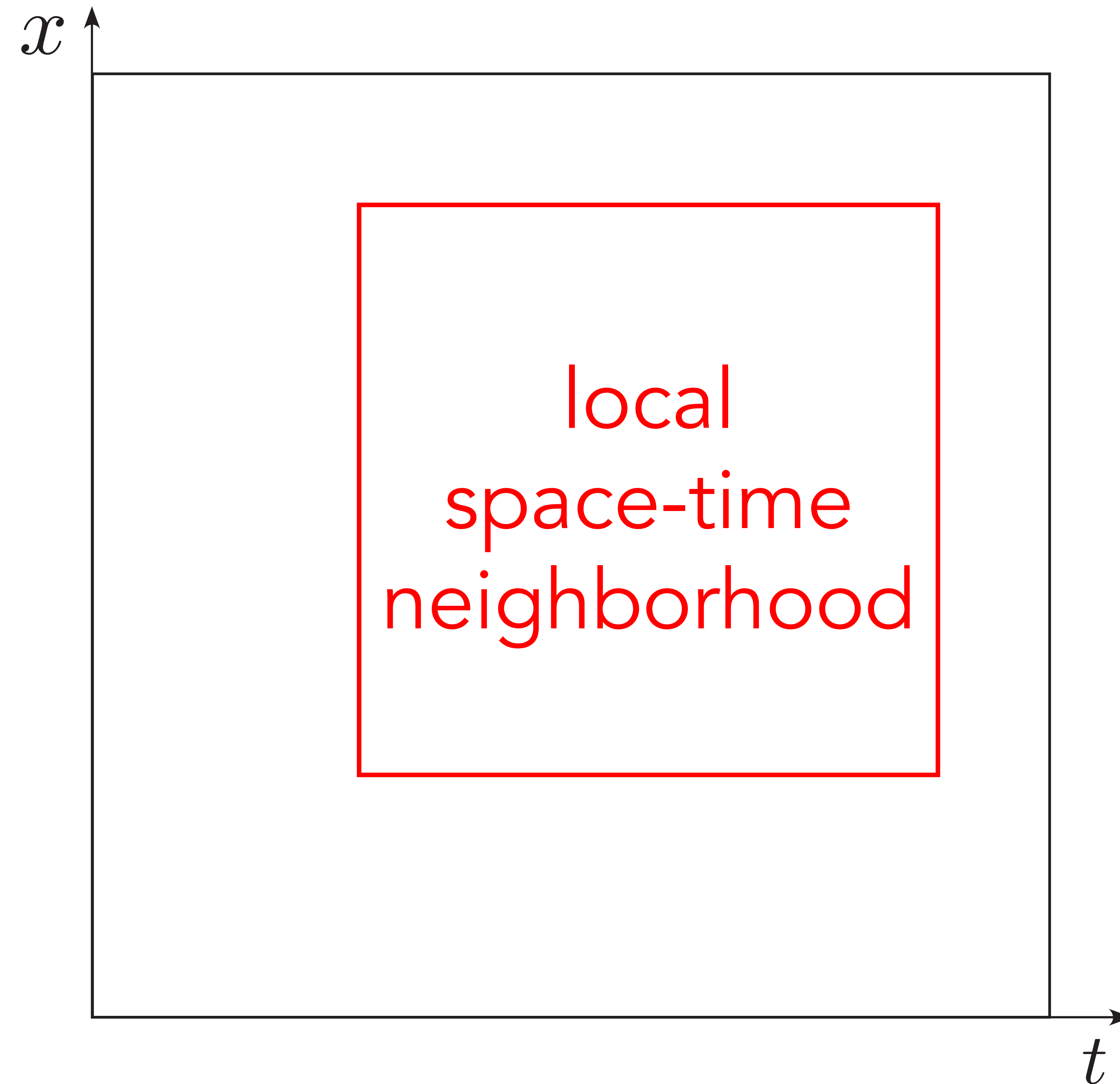
What is a token?

- Flatland view:



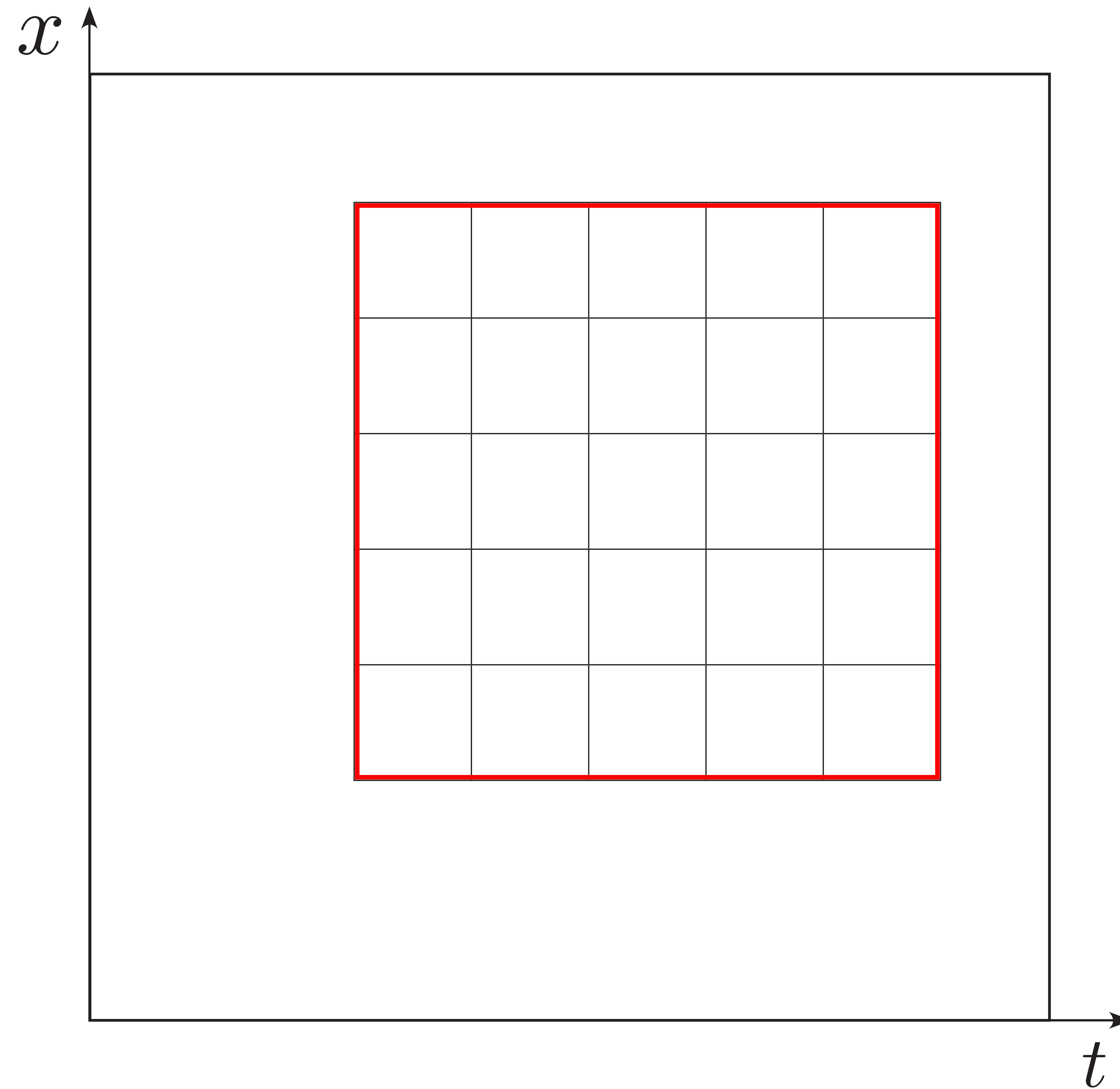
What is a token?

- Flatland view:



What is a token?

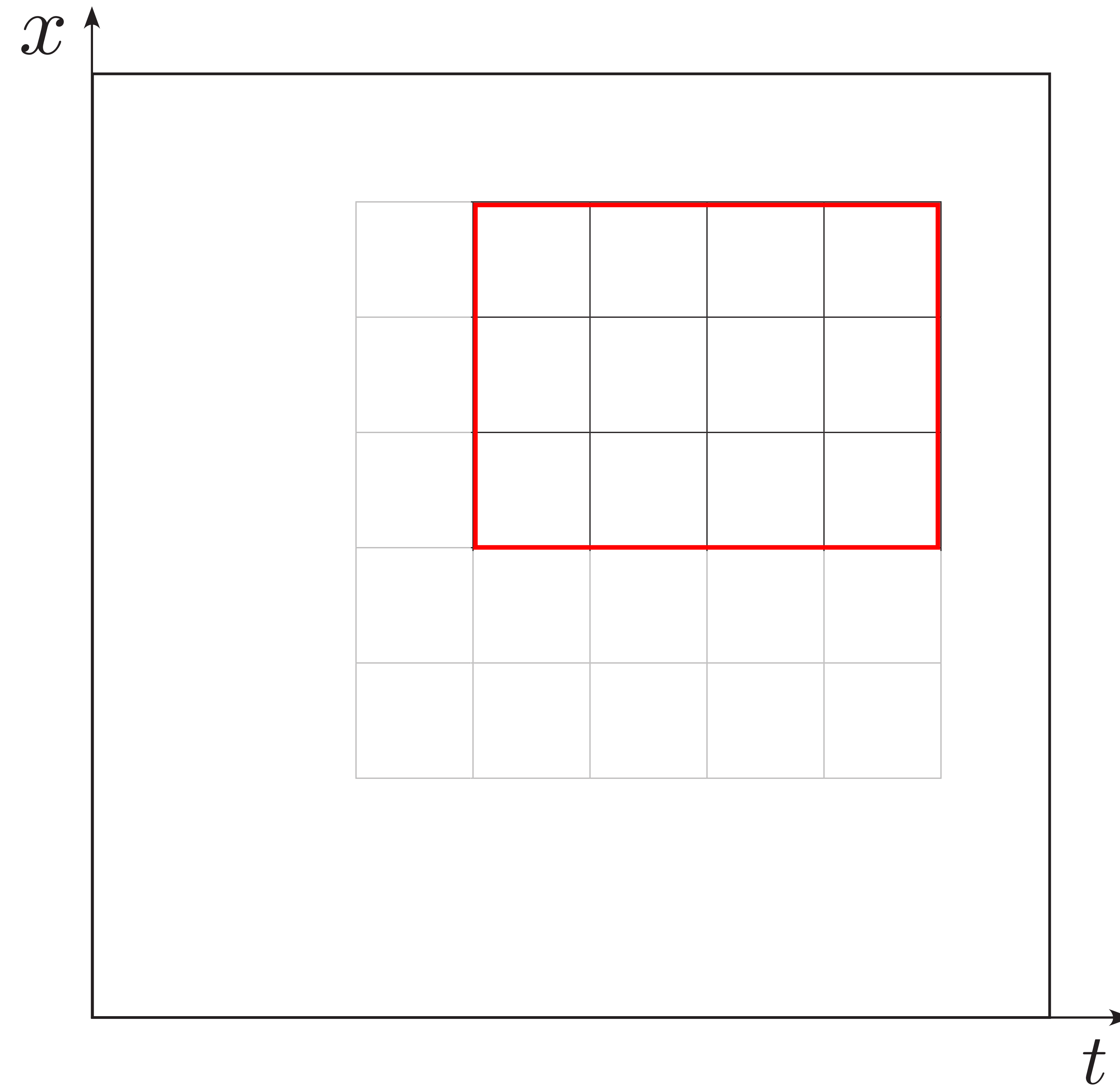
- Flatland view:



What is a token?

- Flatland view:

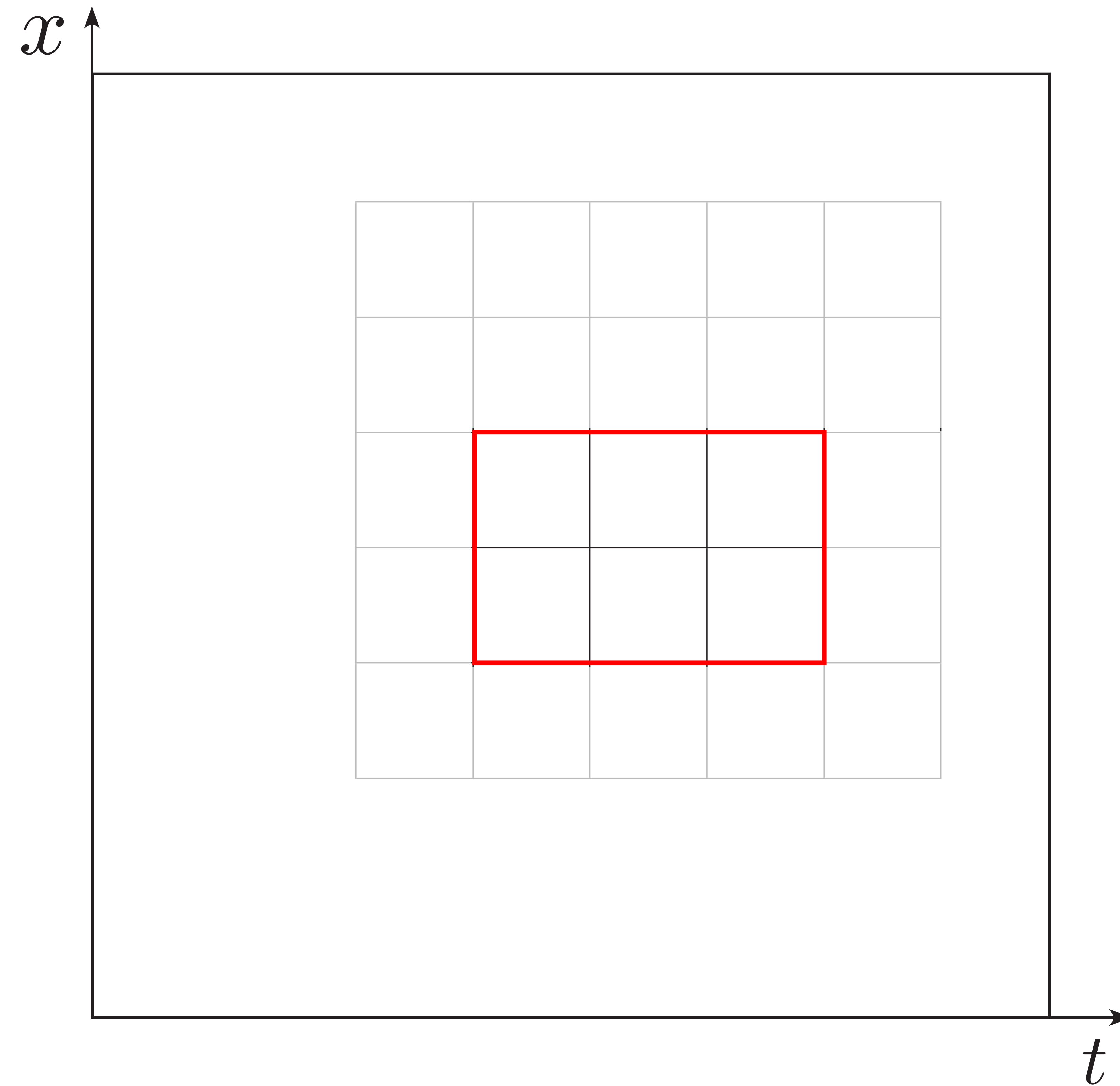
local window
size is flexible



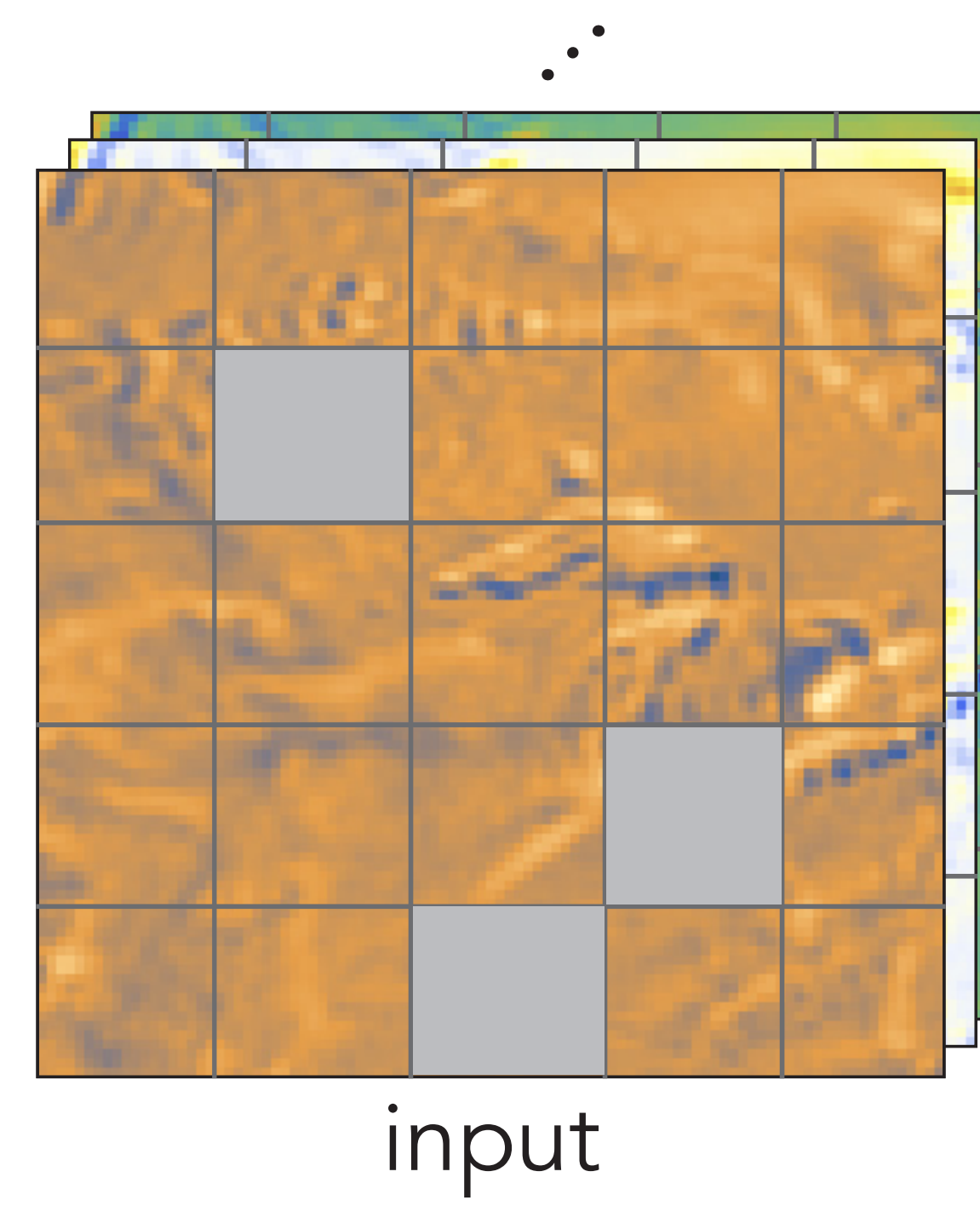
What is a token?

- Flatland view:

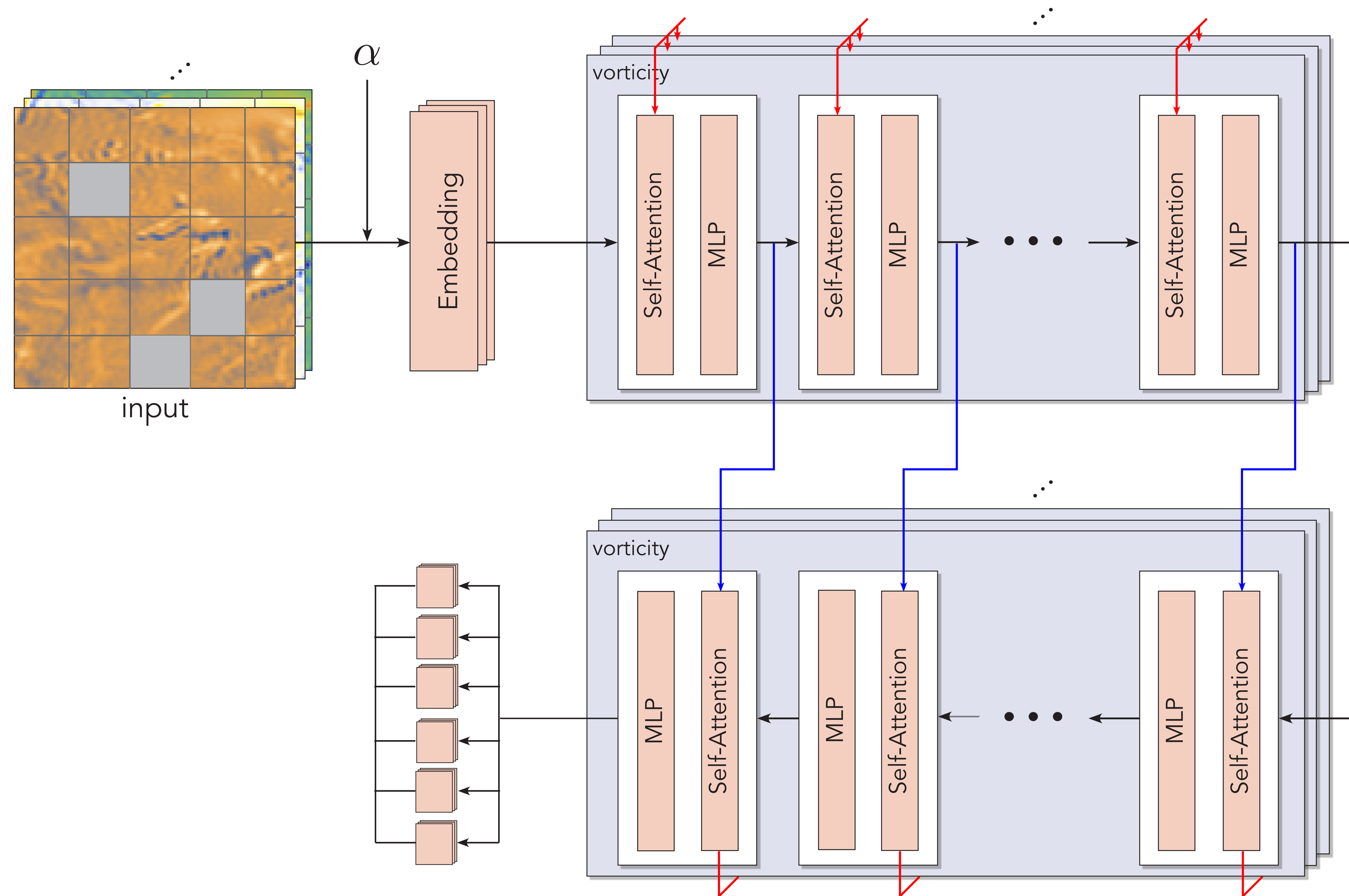
local window
size is flexible



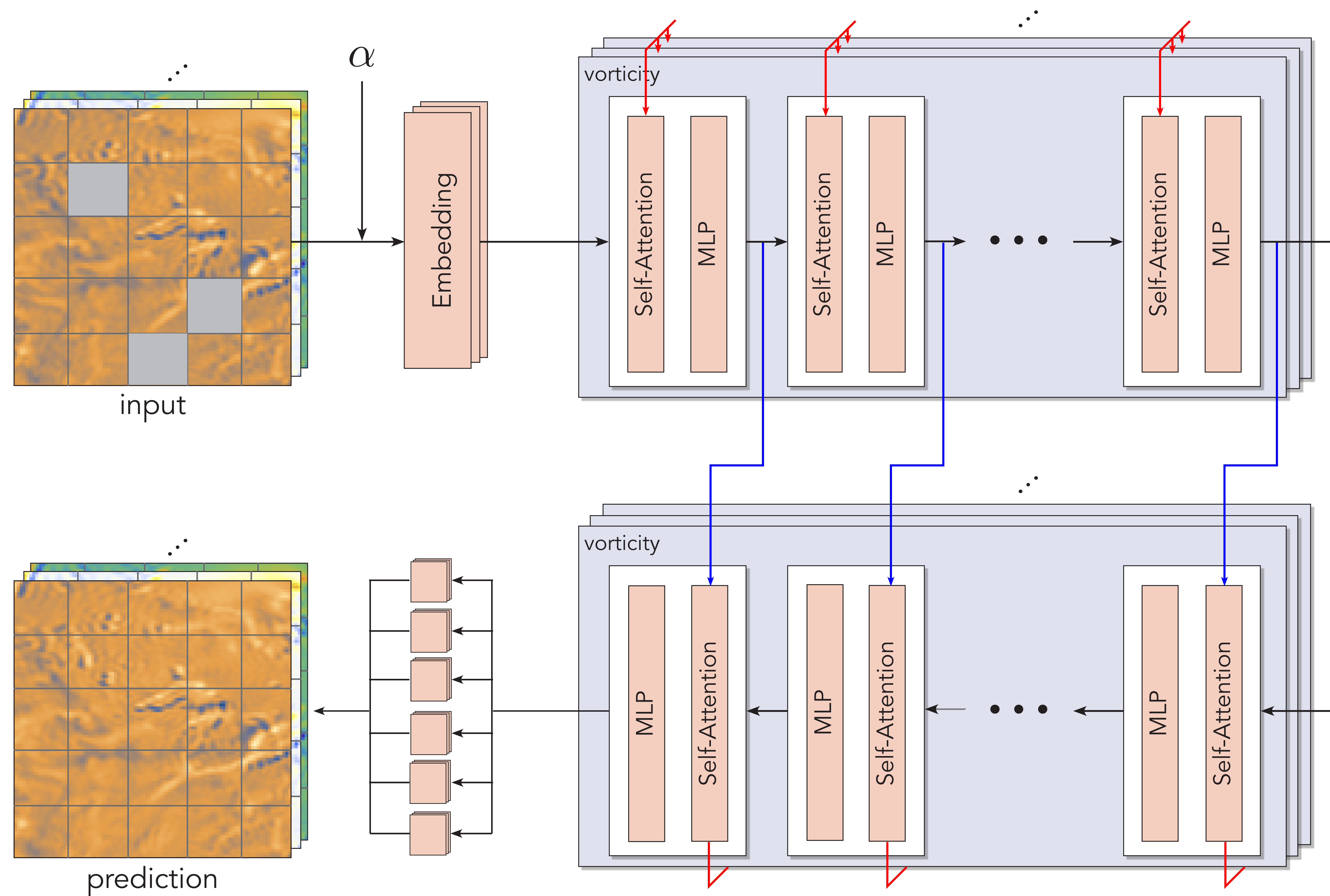
Multiformer



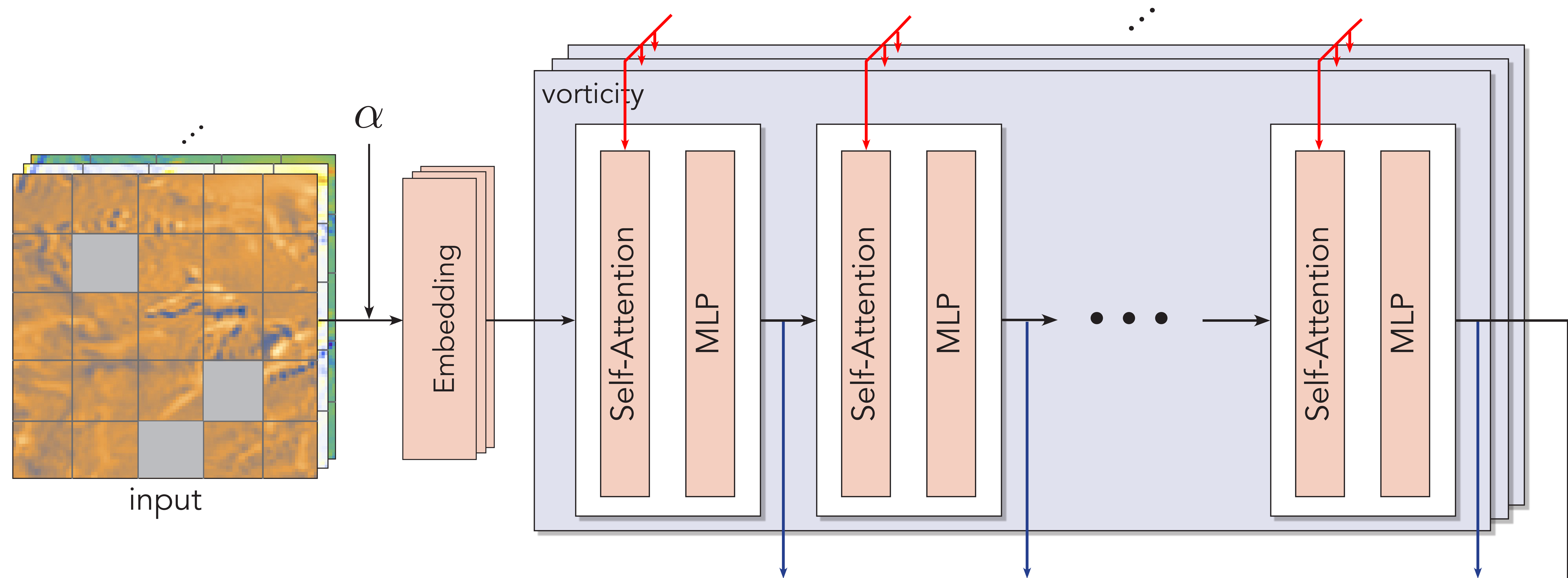
Multiformer



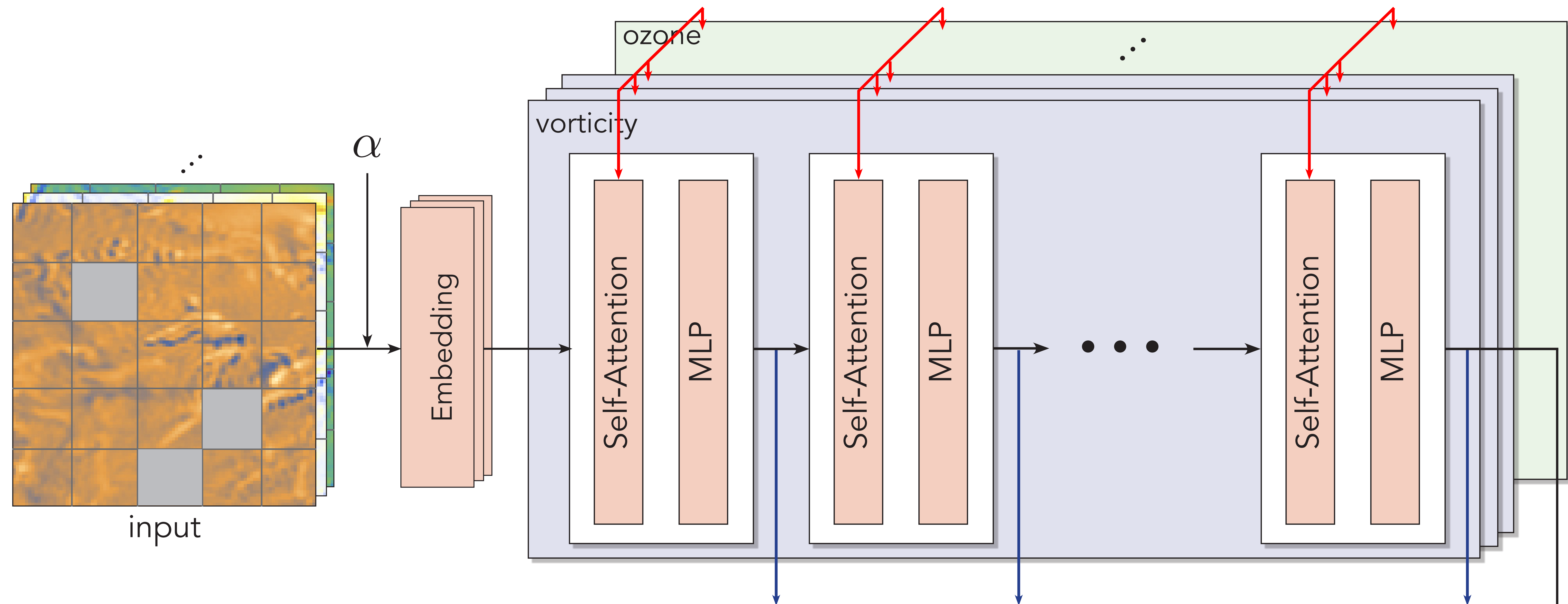
Multiformer



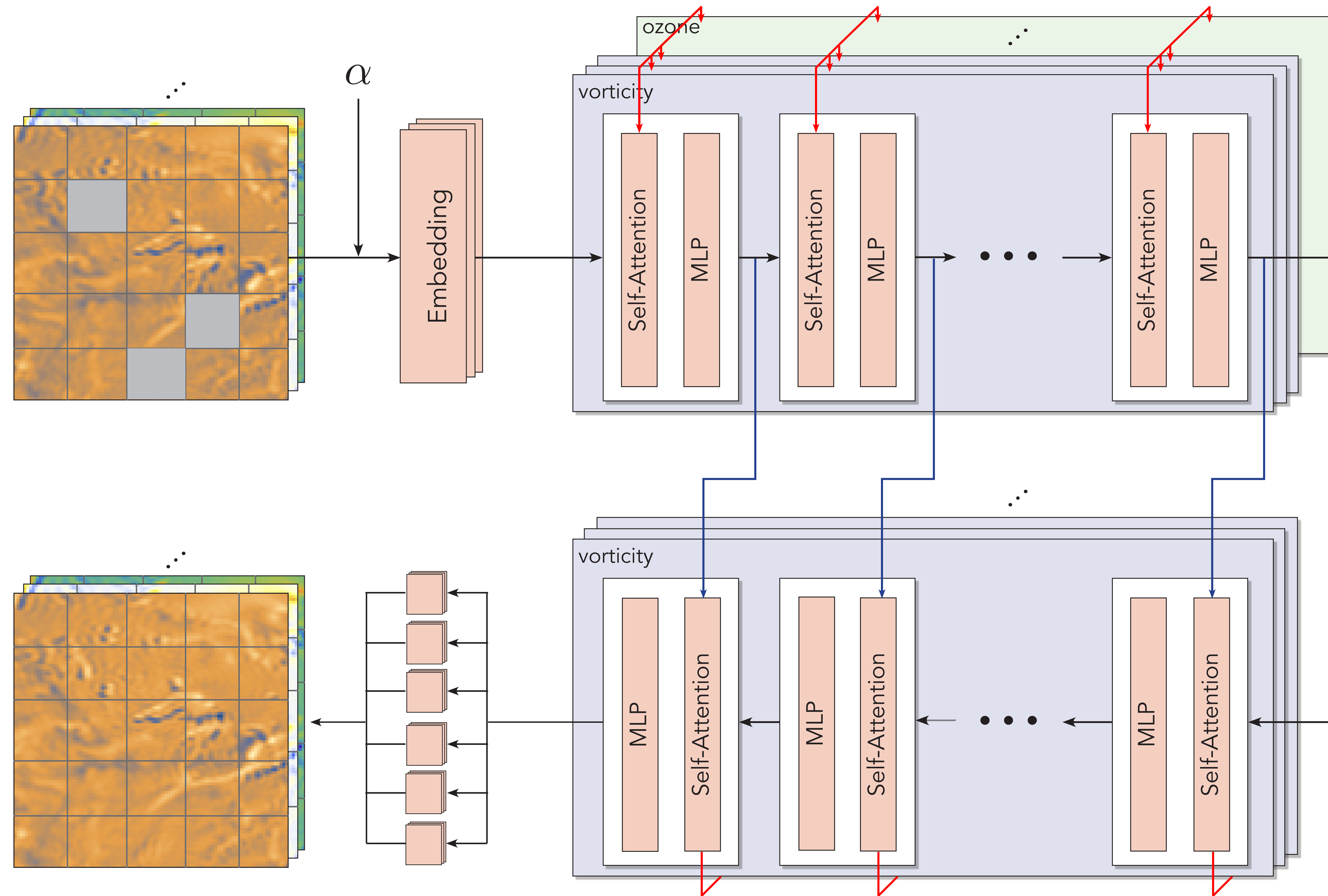
Multiformer



Multiformer



Multiformer



Loss and Pre-training

Training objective

- Numerical statistical atmospheric model

$$p_{\theta}(y|x, \alpha)$$

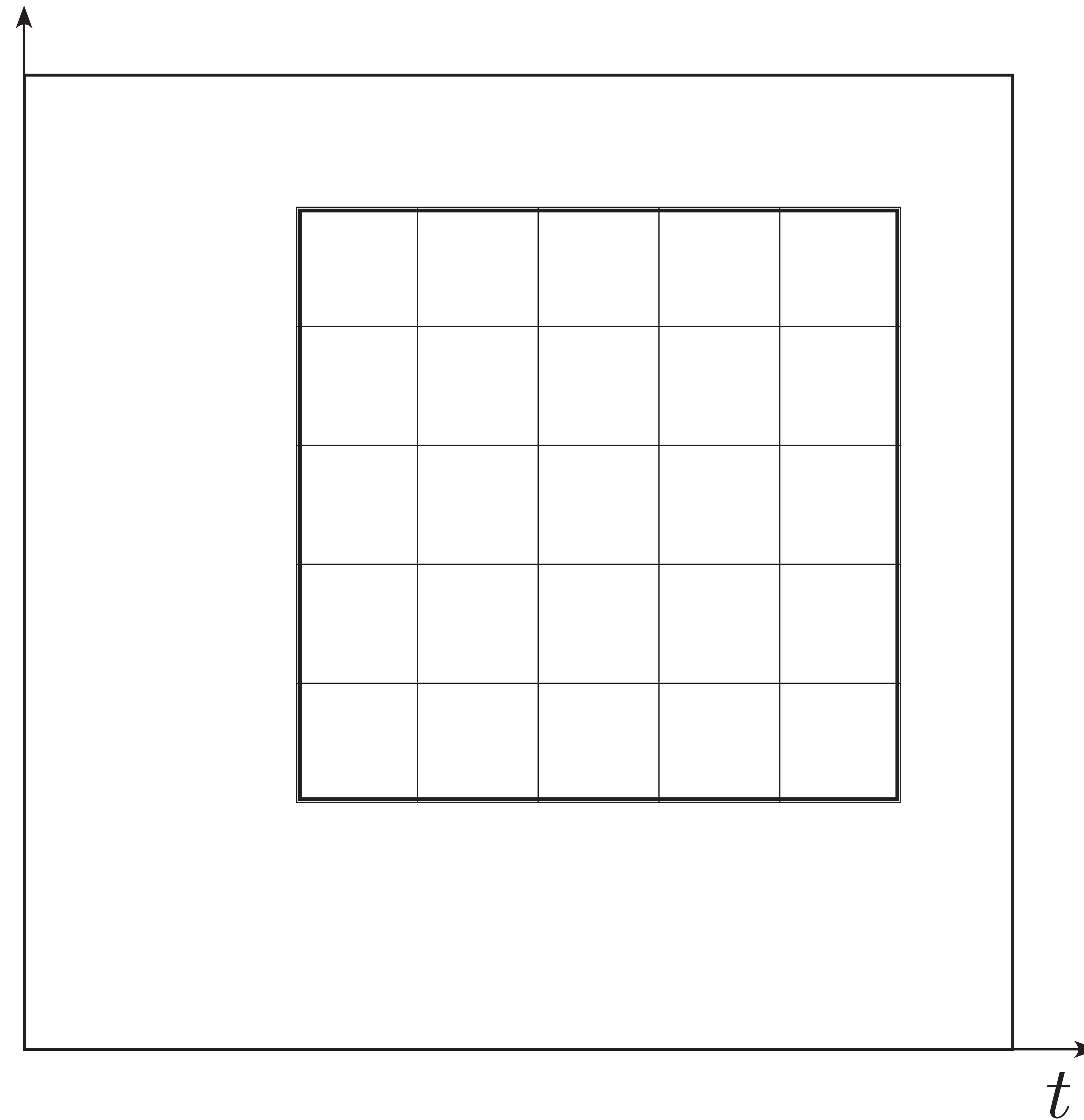
Training objective

- Numerical statistical atmospheric model

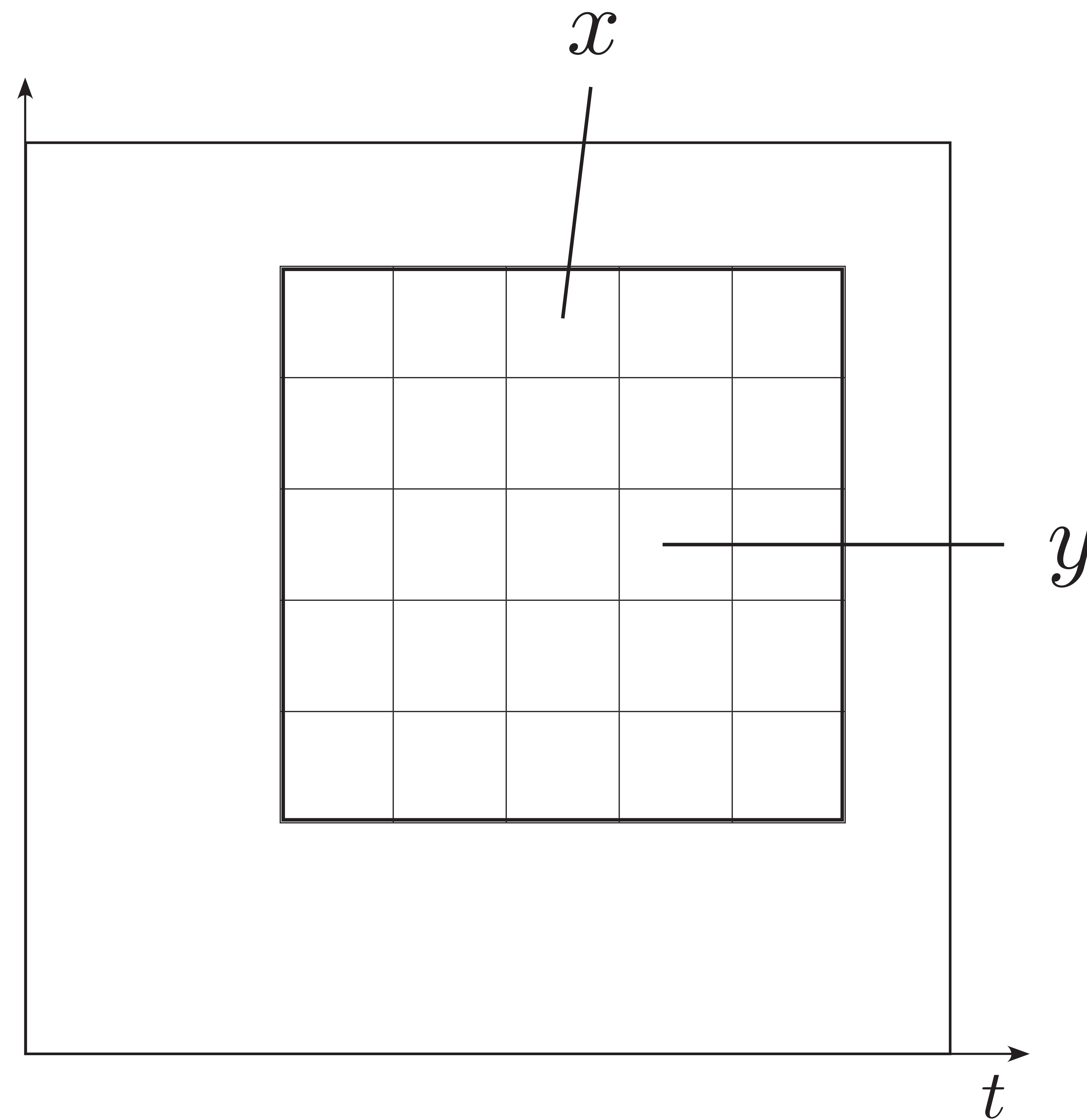
$$p_{\theta}(y|x, \alpha)$$

- › Training should model spatio-temporal relationship between arbitrary state x and y

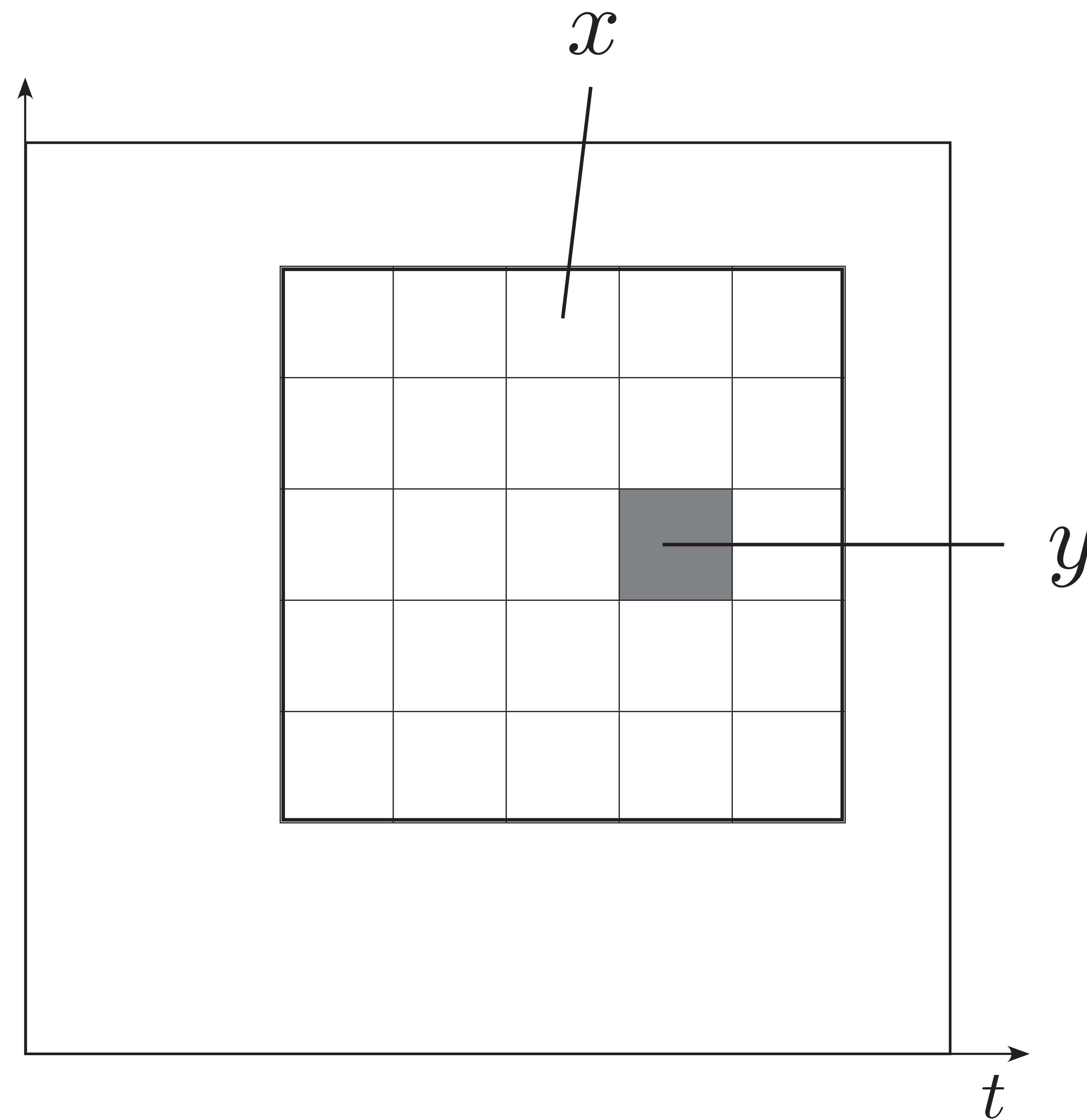
Training objective



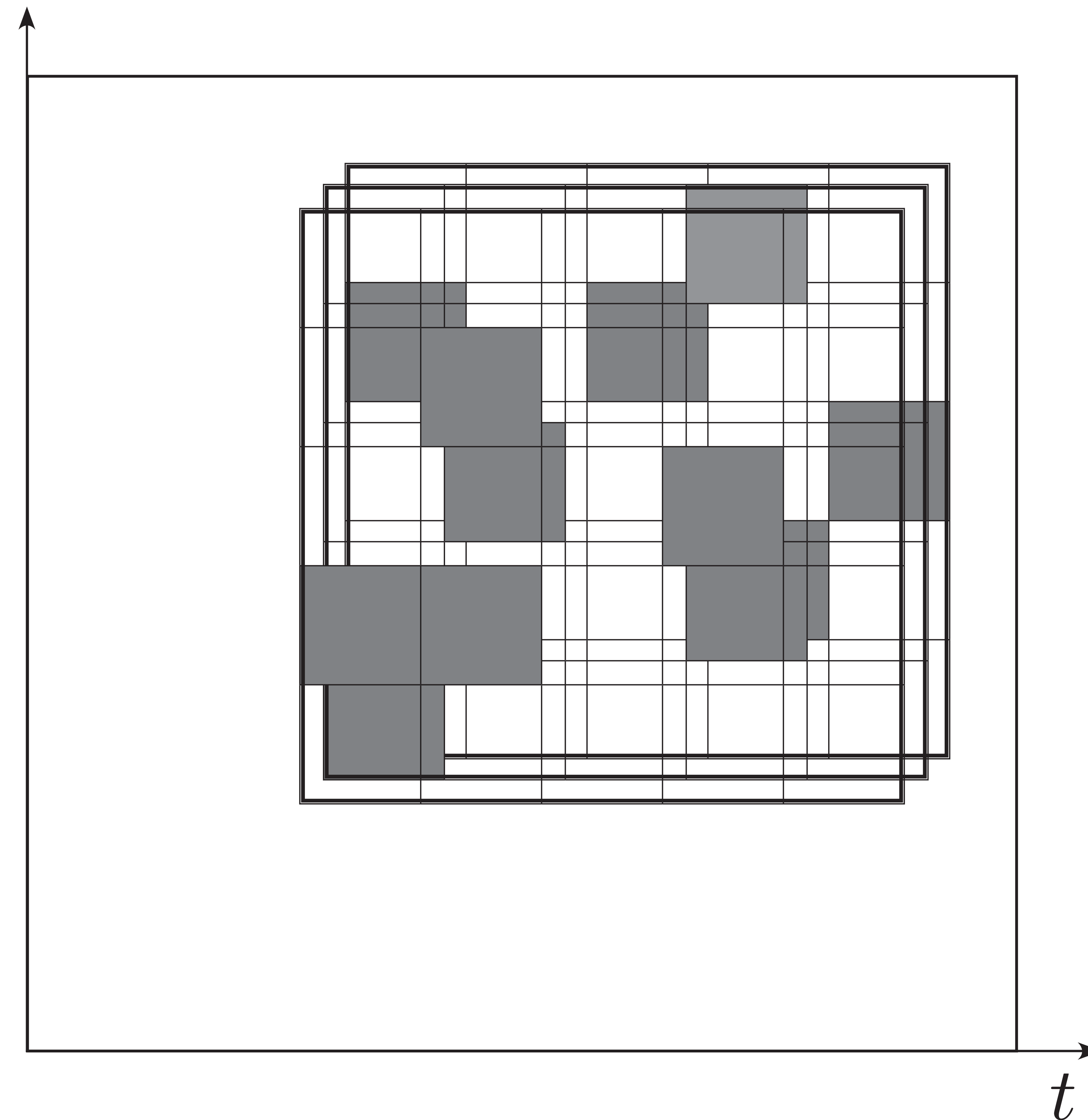
Training objective



Training objective

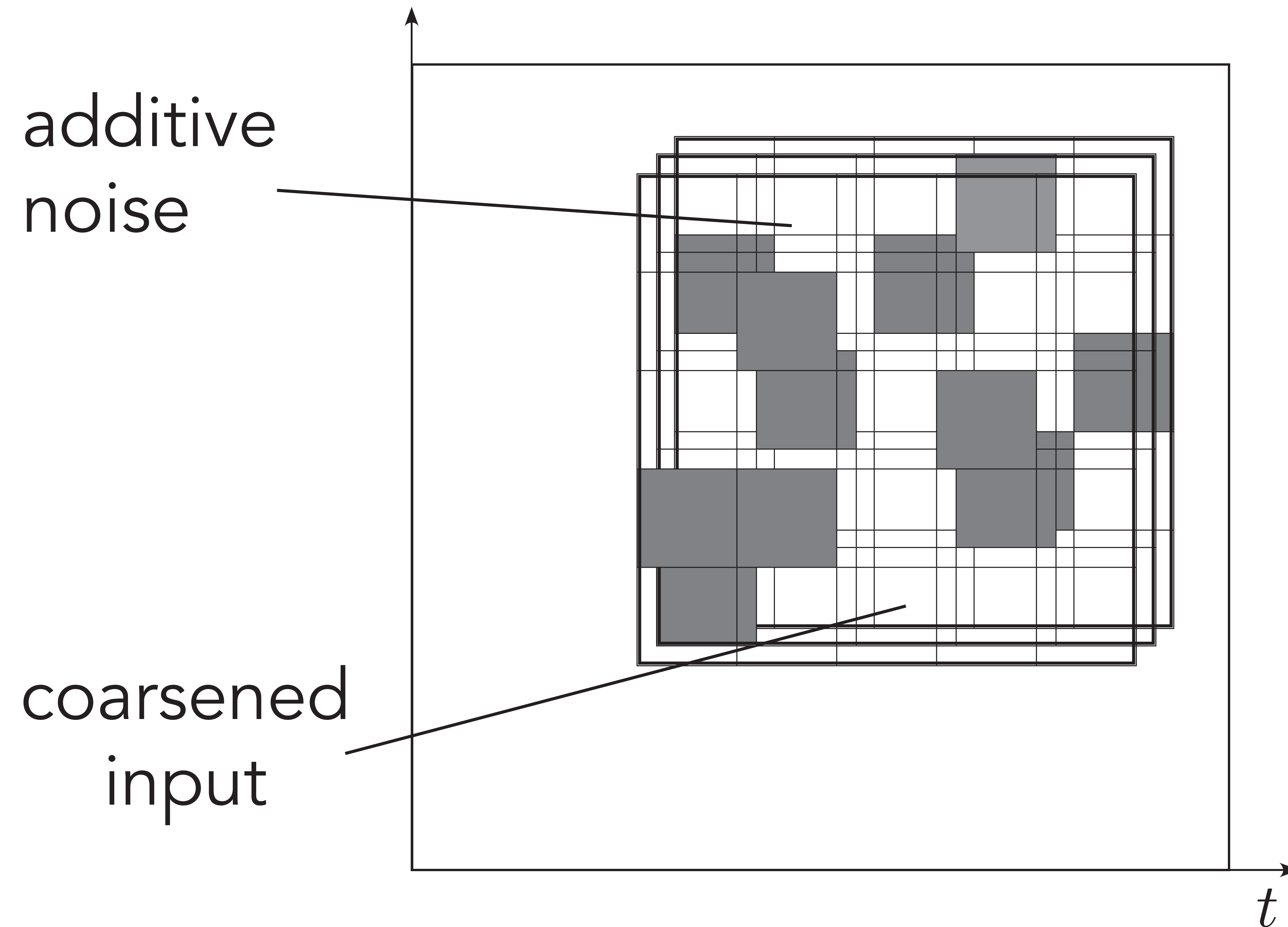


Training objective



masked
token model:
training to predict
randomly masked
information

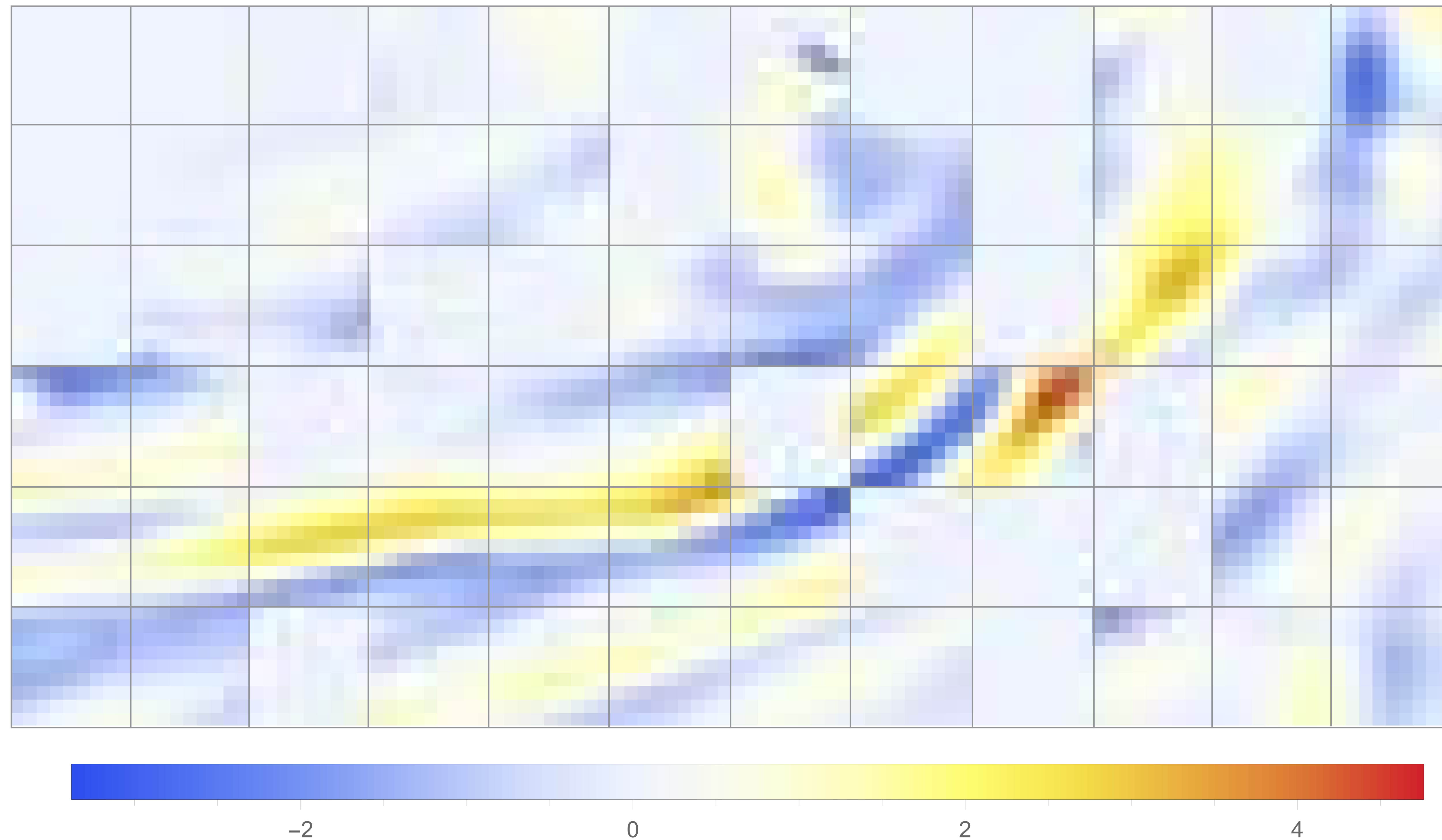
Training objective



masked
token model:
training to predict
randomly masked
information

Training objective

divergence, $ml=96$



Loss

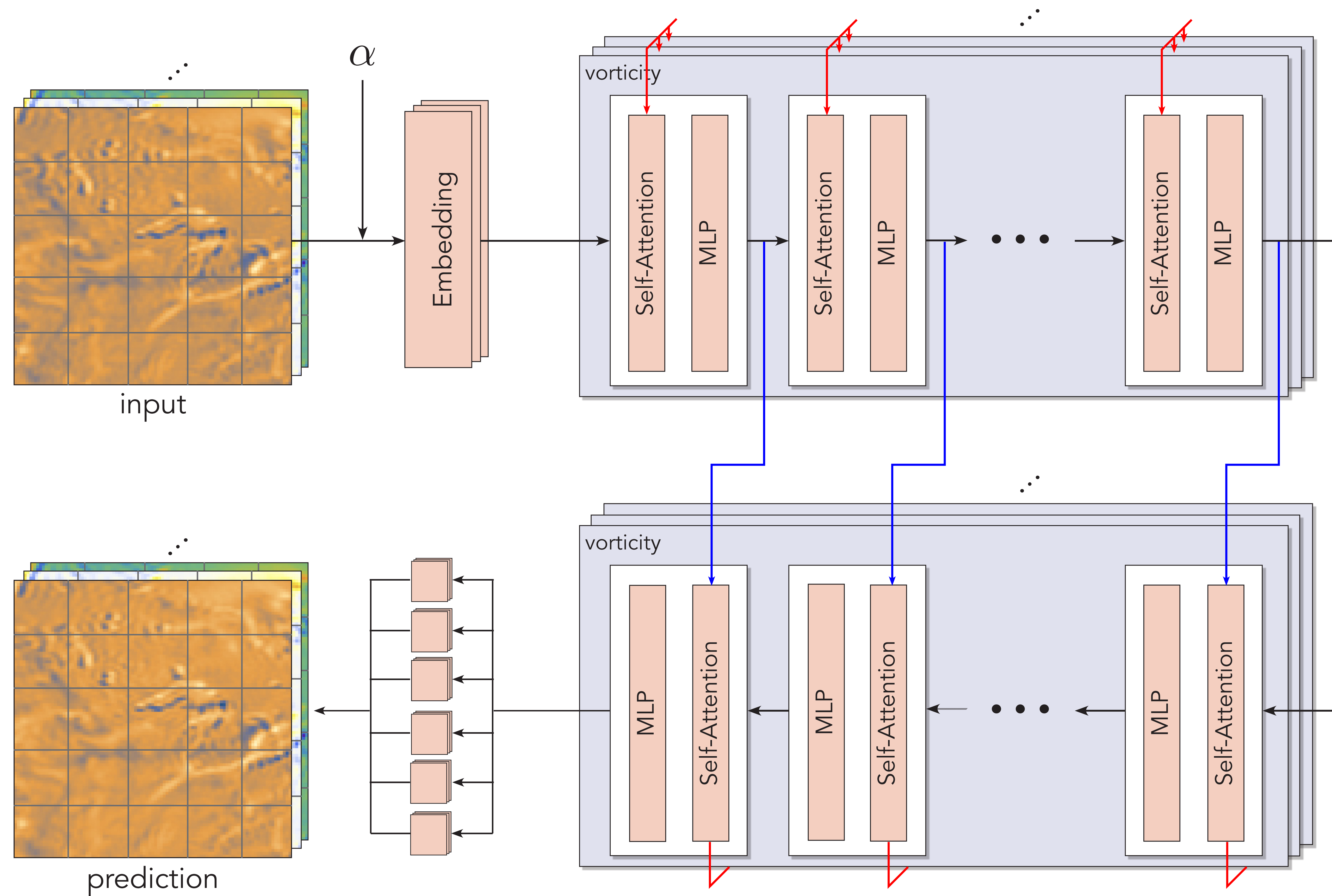
- MSE measures only very approximately what is of interest in applications
- To train a statistical model, deterministic training and loss is insufficient

Loss

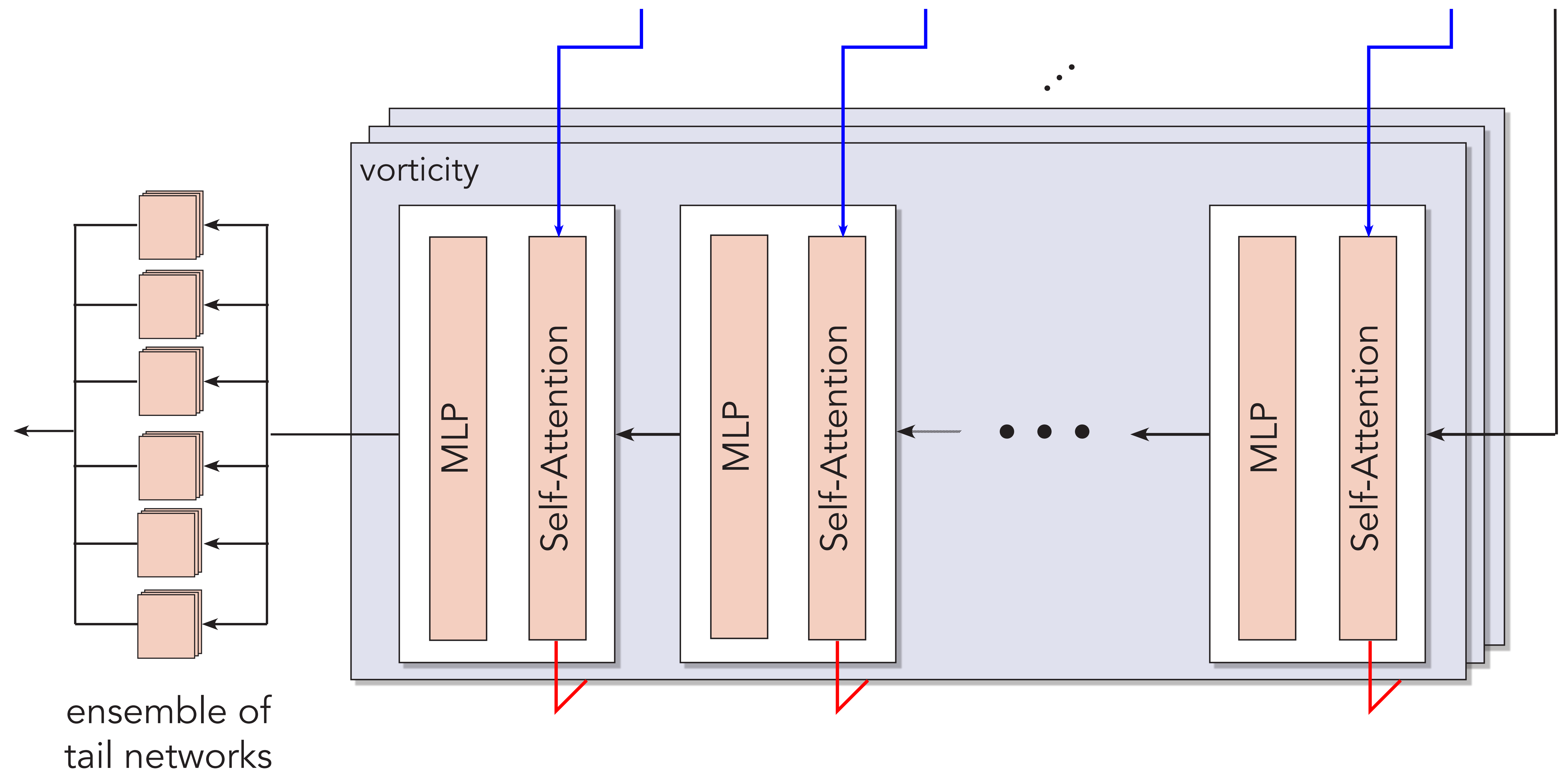
- MSE measures only very approximately what is of interest in applications
- To train a statistical model, deterministic training and loss is insufficient

=> Analog of cross-entropy loss for regression problems?

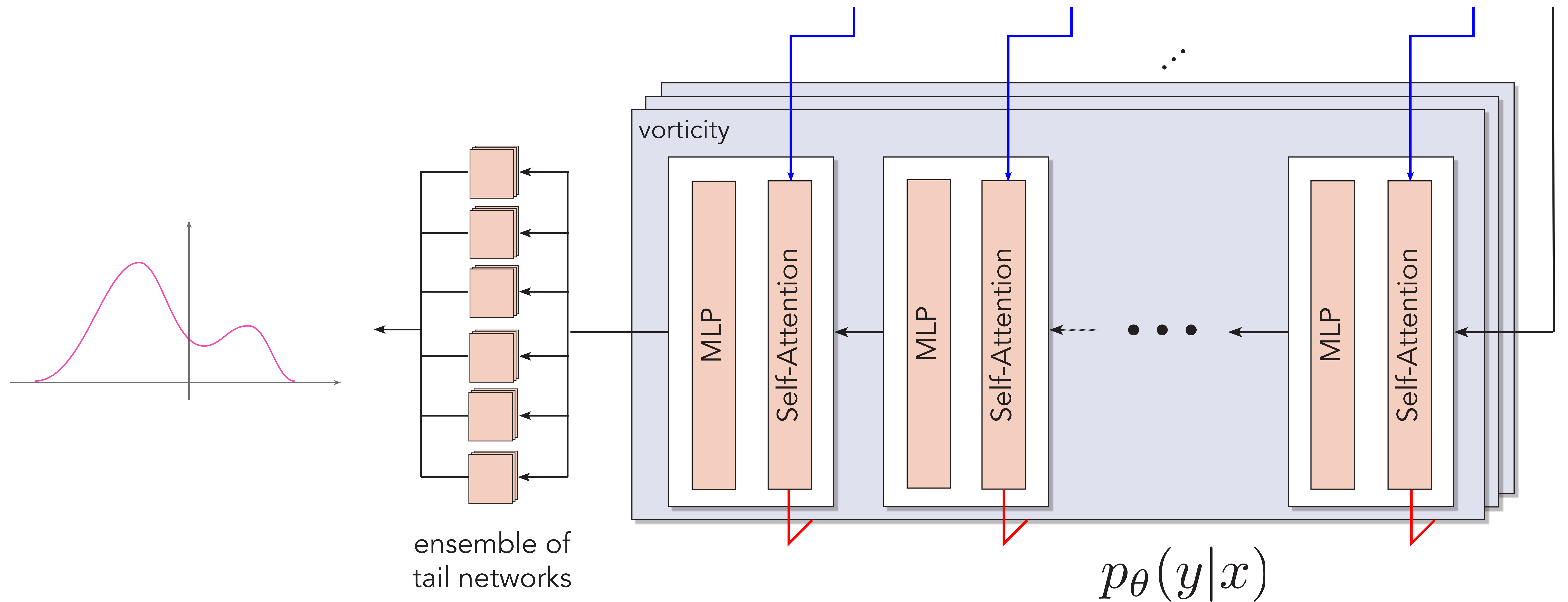
Statistical loss



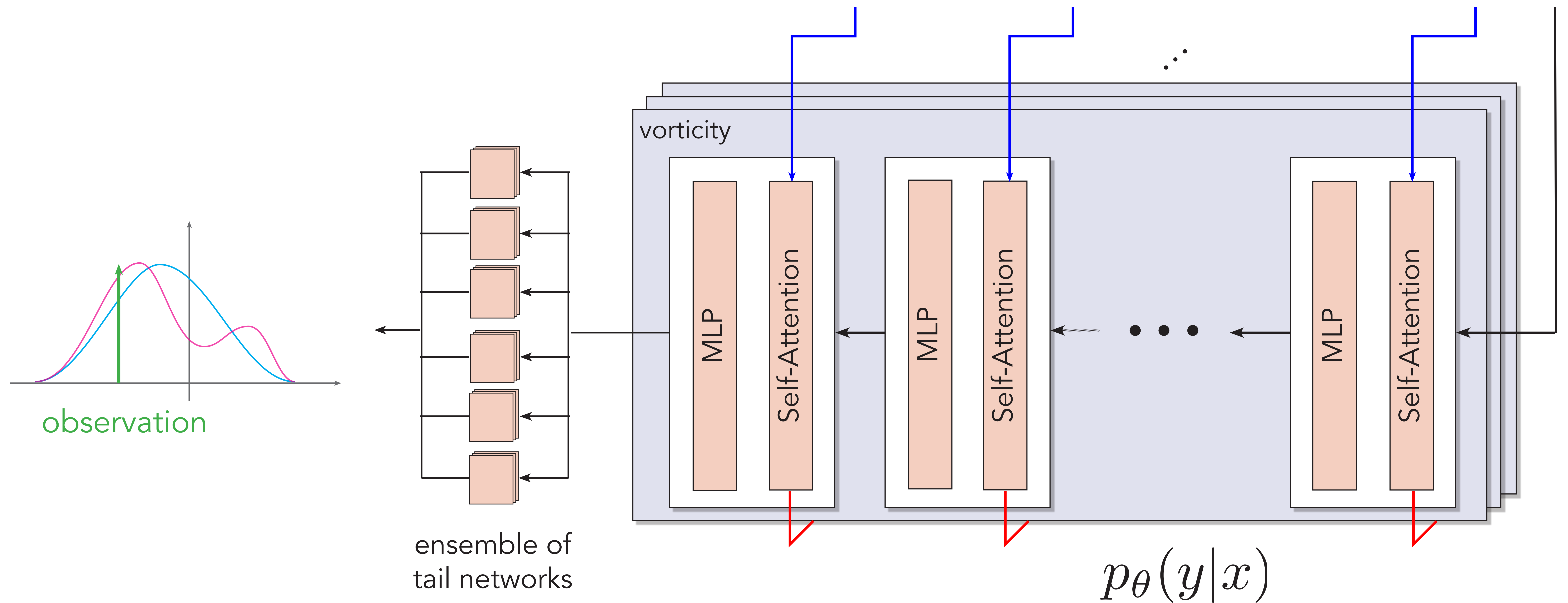
Statistical loss



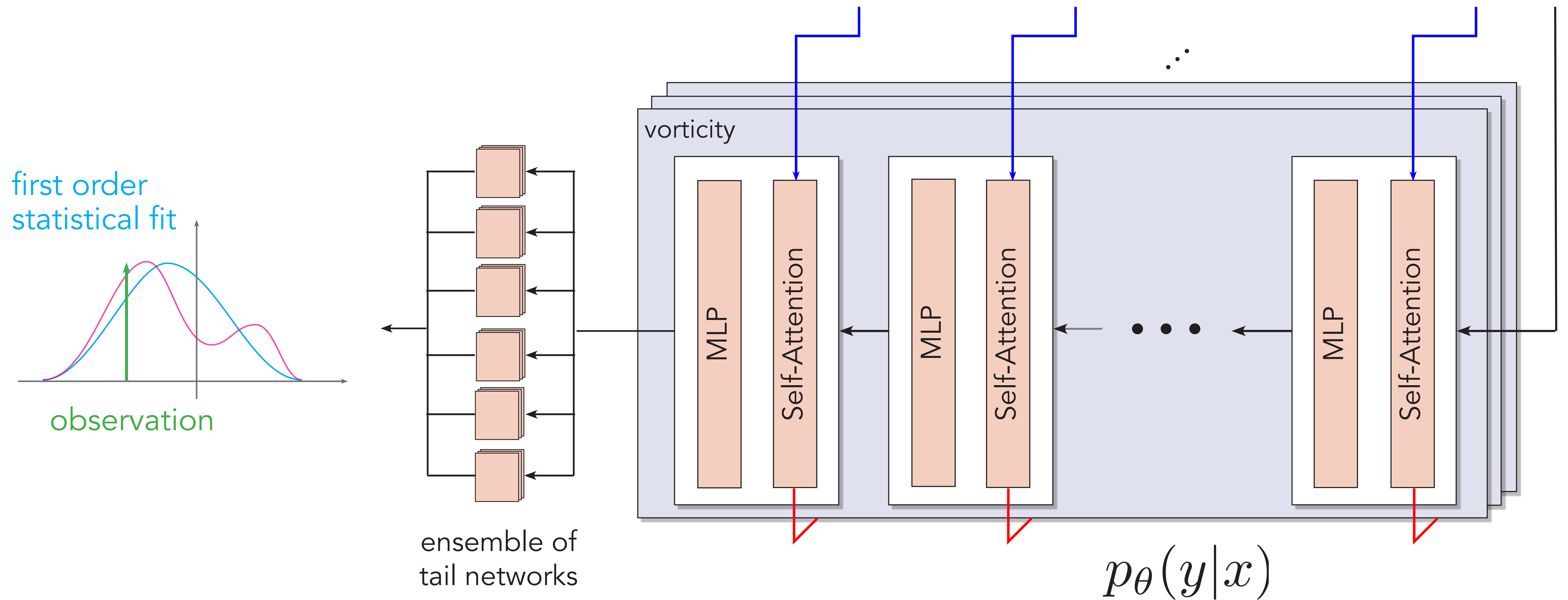
Statistical loss



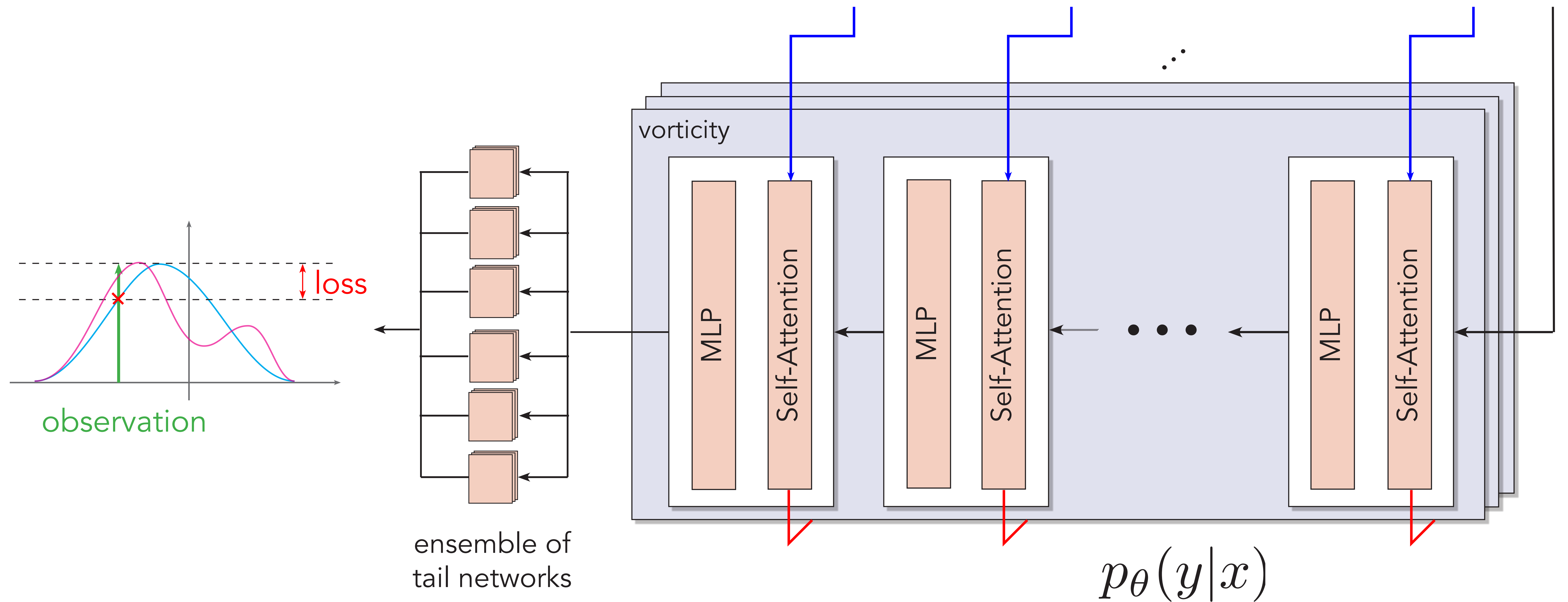
Statistical loss



Statistical loss

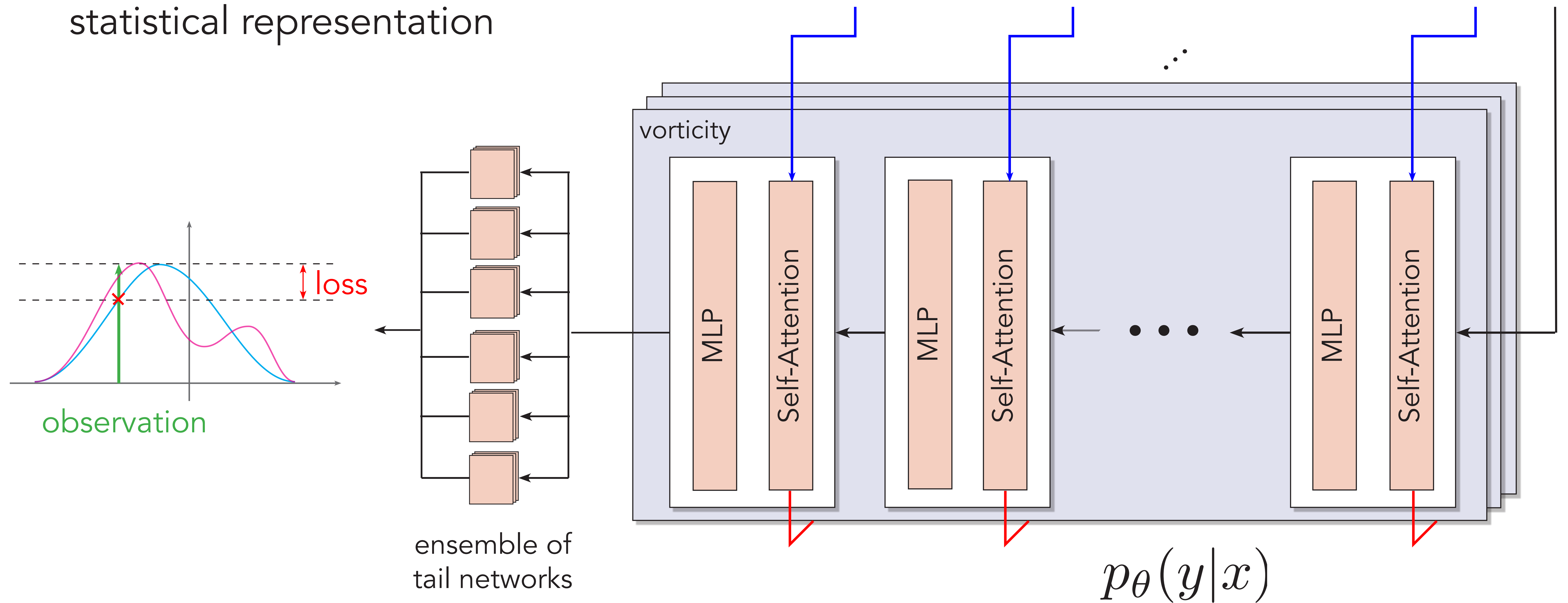


Statistical loss

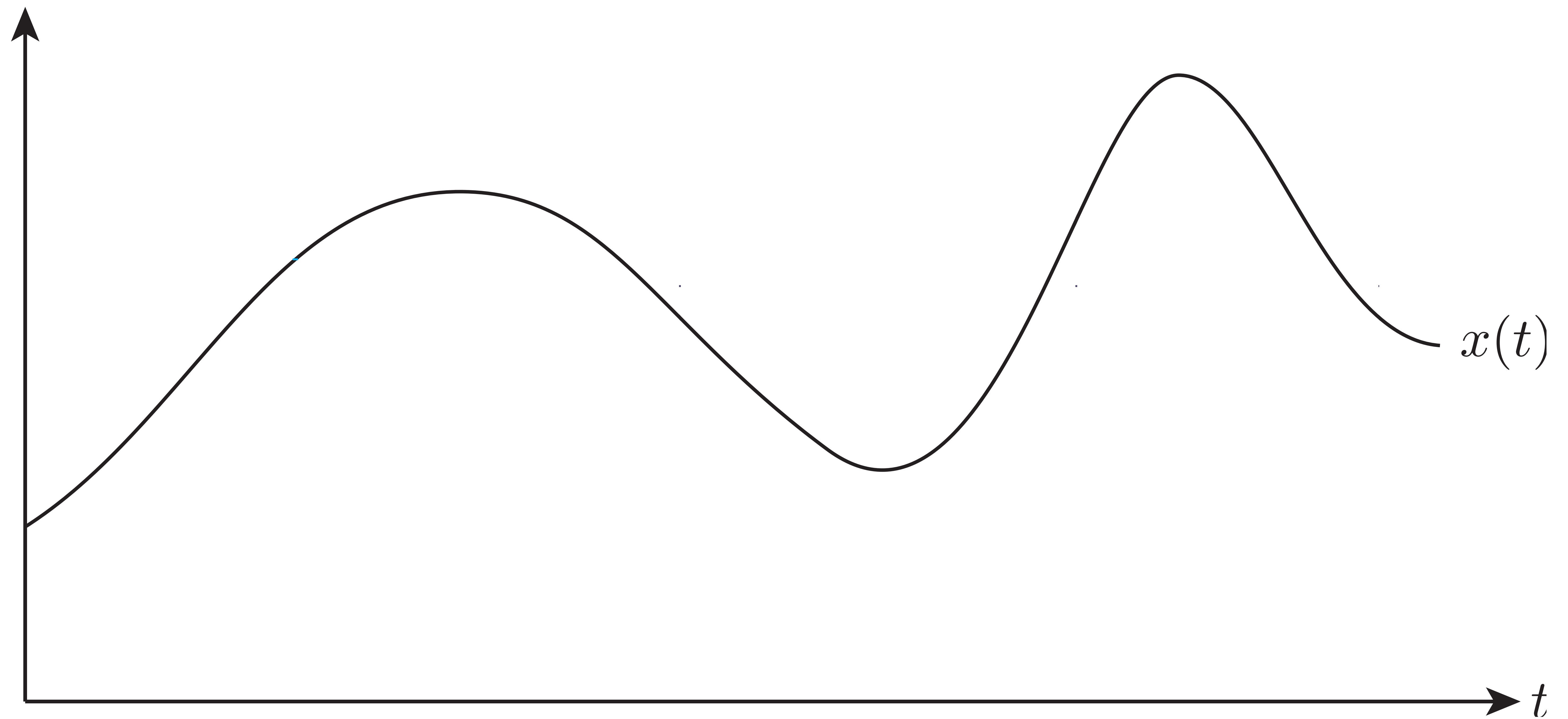


Statistical loss

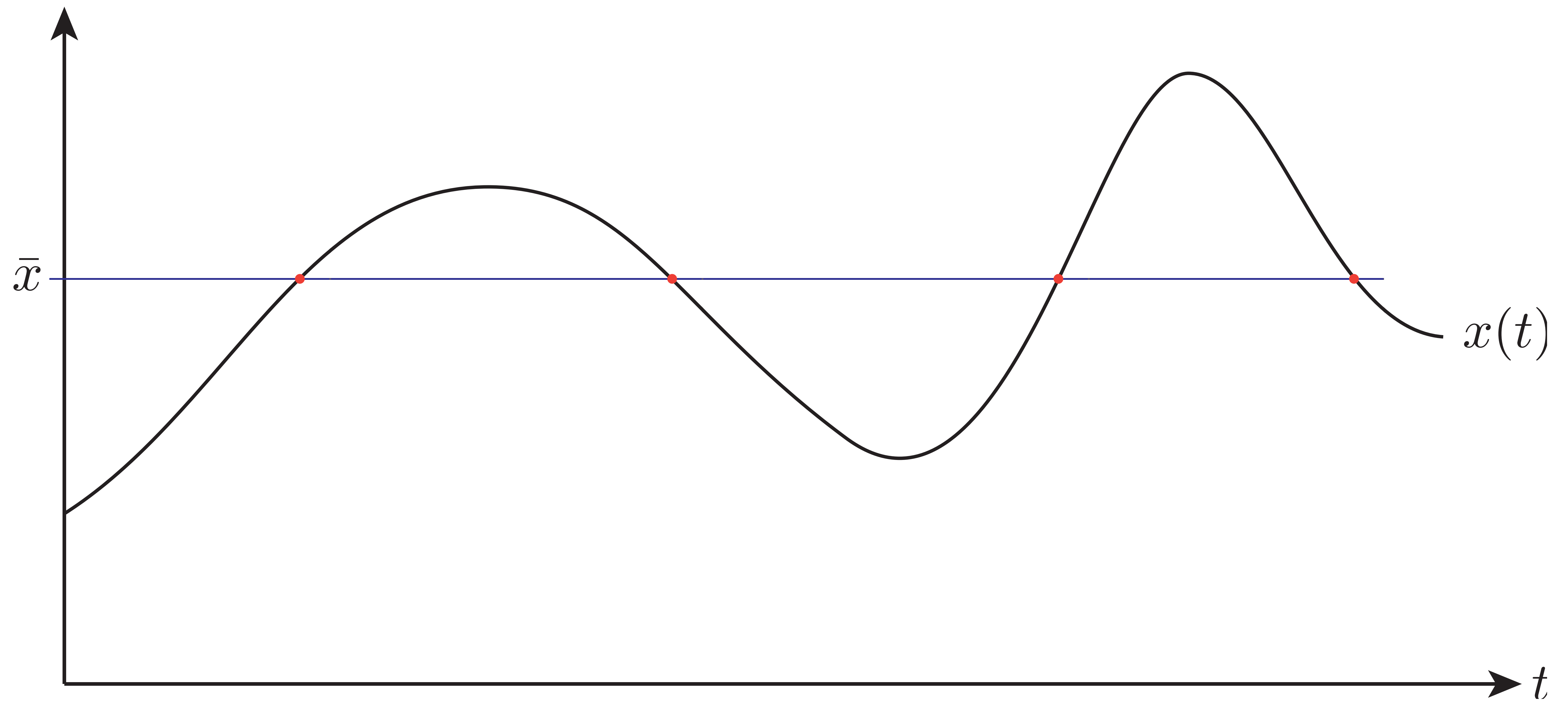
end-to-end training encourages statistical representation



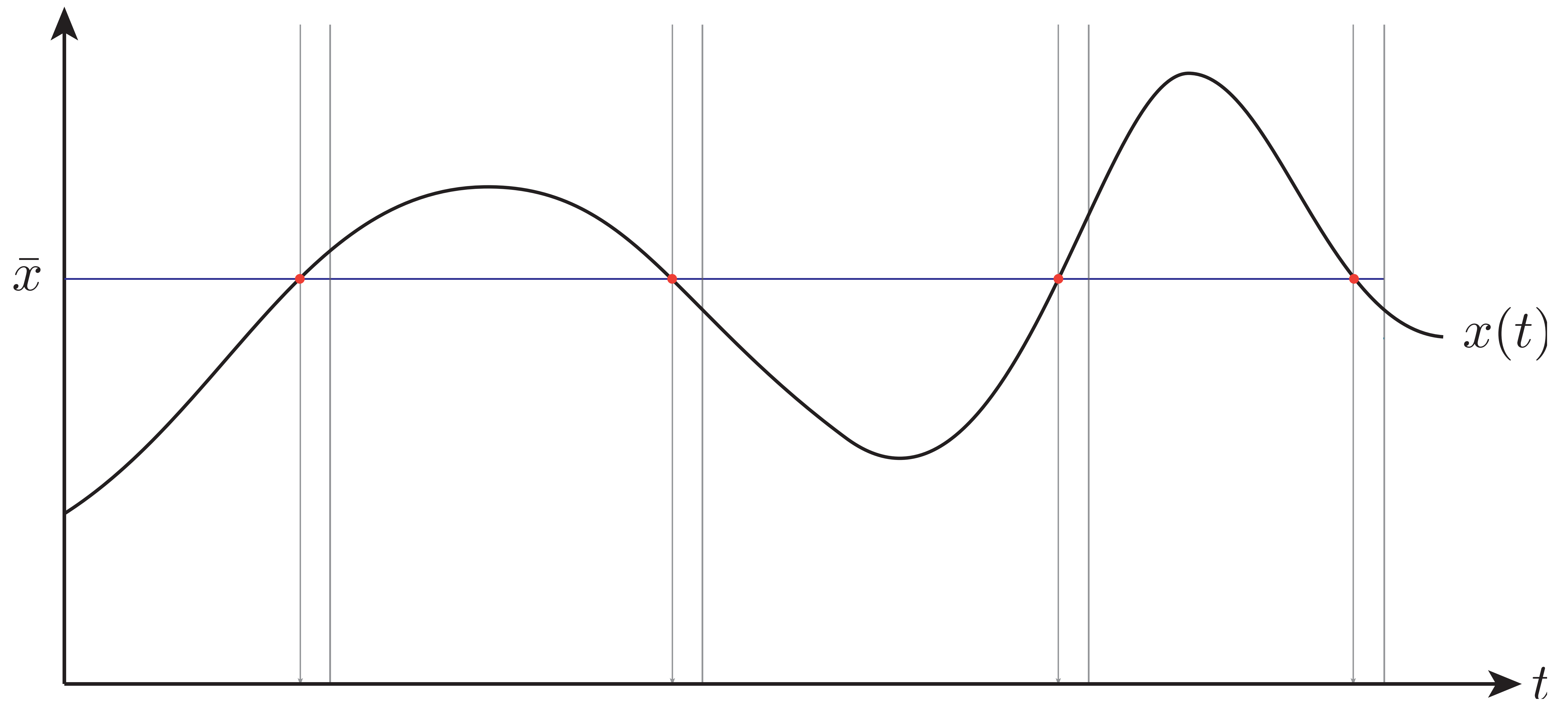
Statistical loss



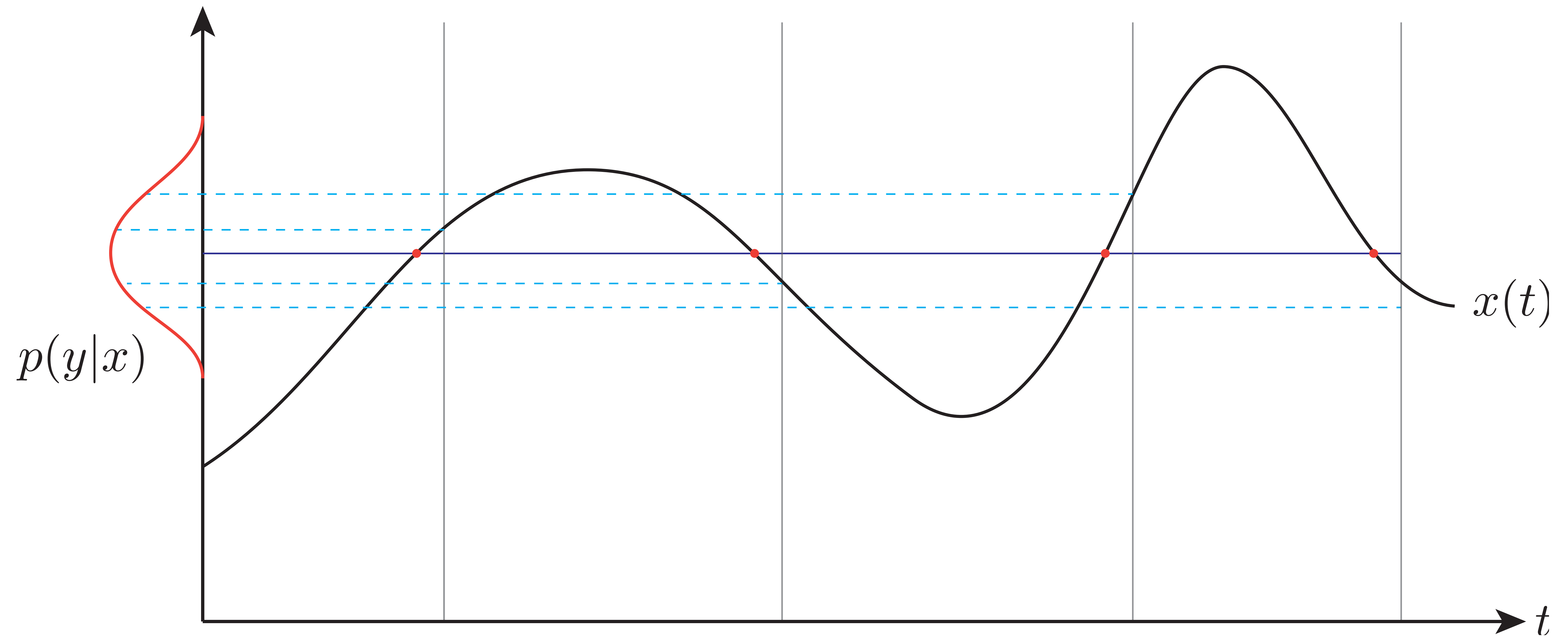
Statistical loss



Statistical loss

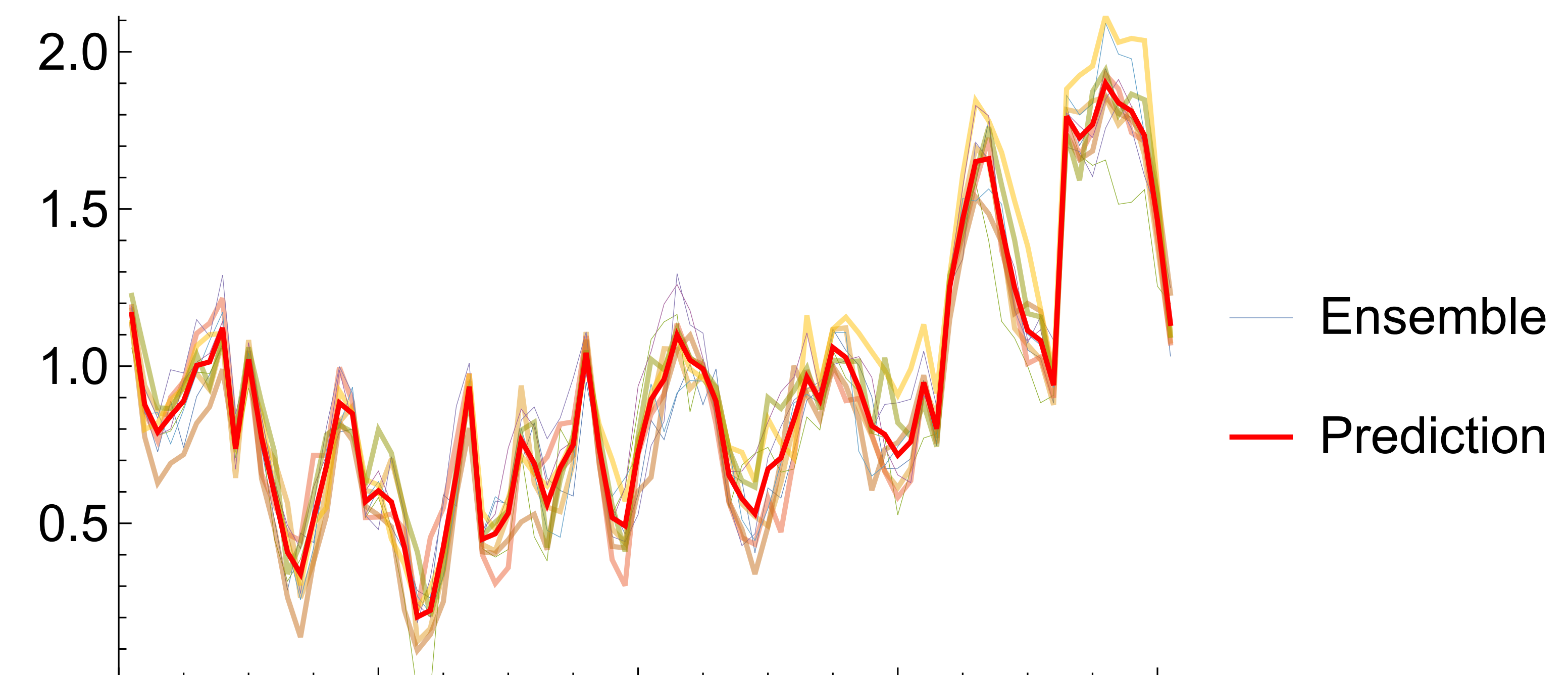
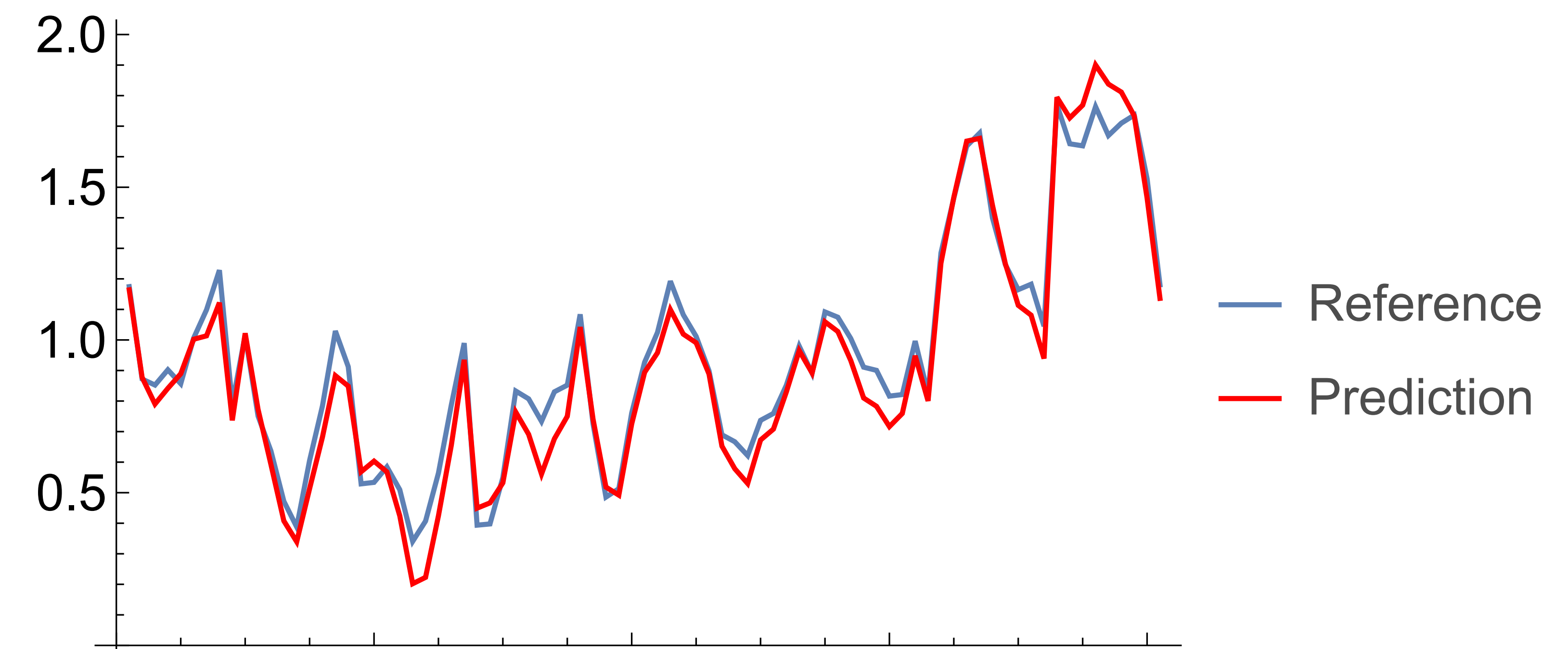
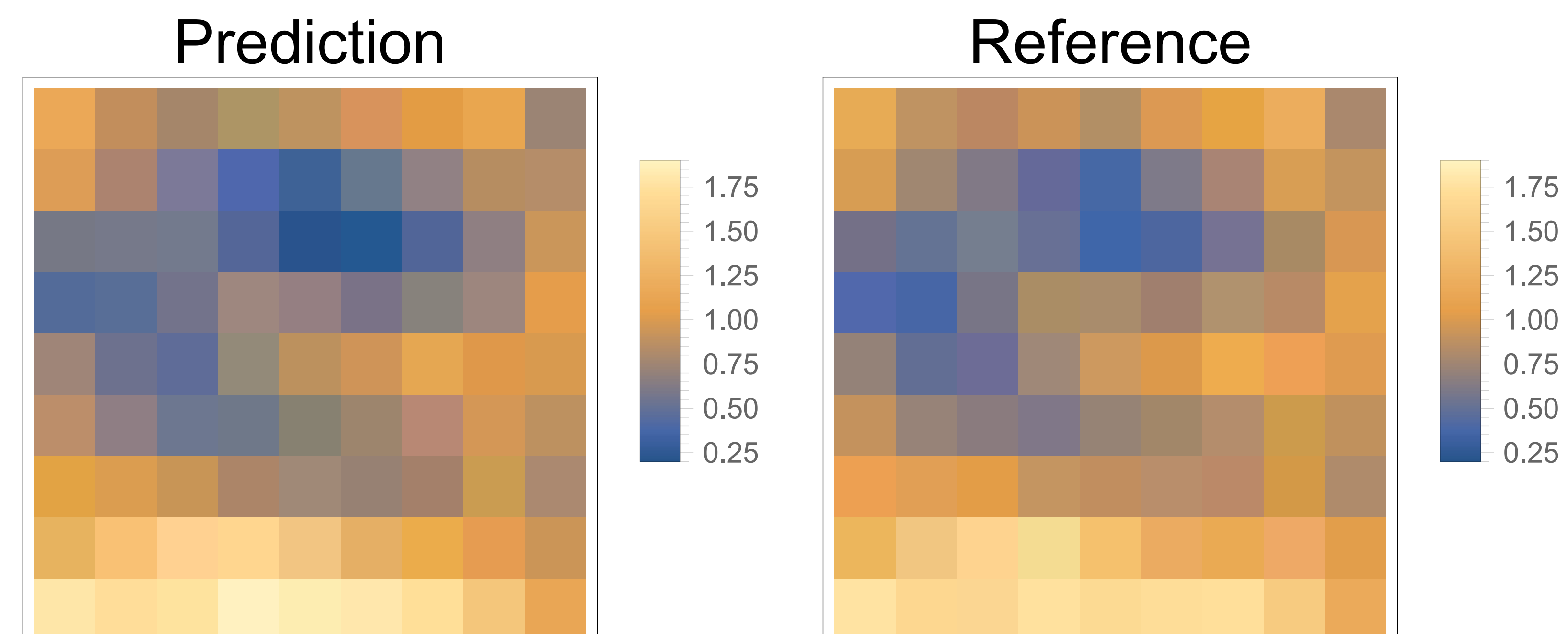


Statistical loss



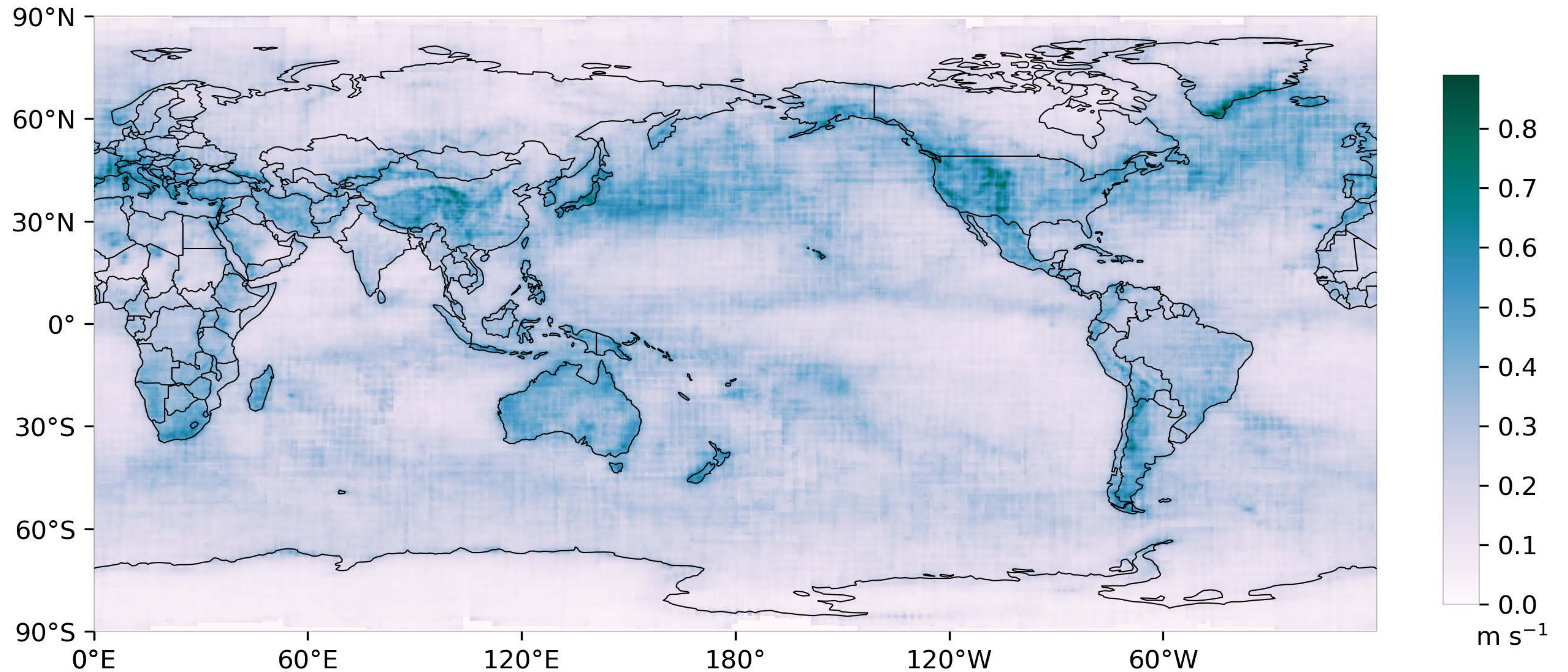
Statistical loss

- Predictions:

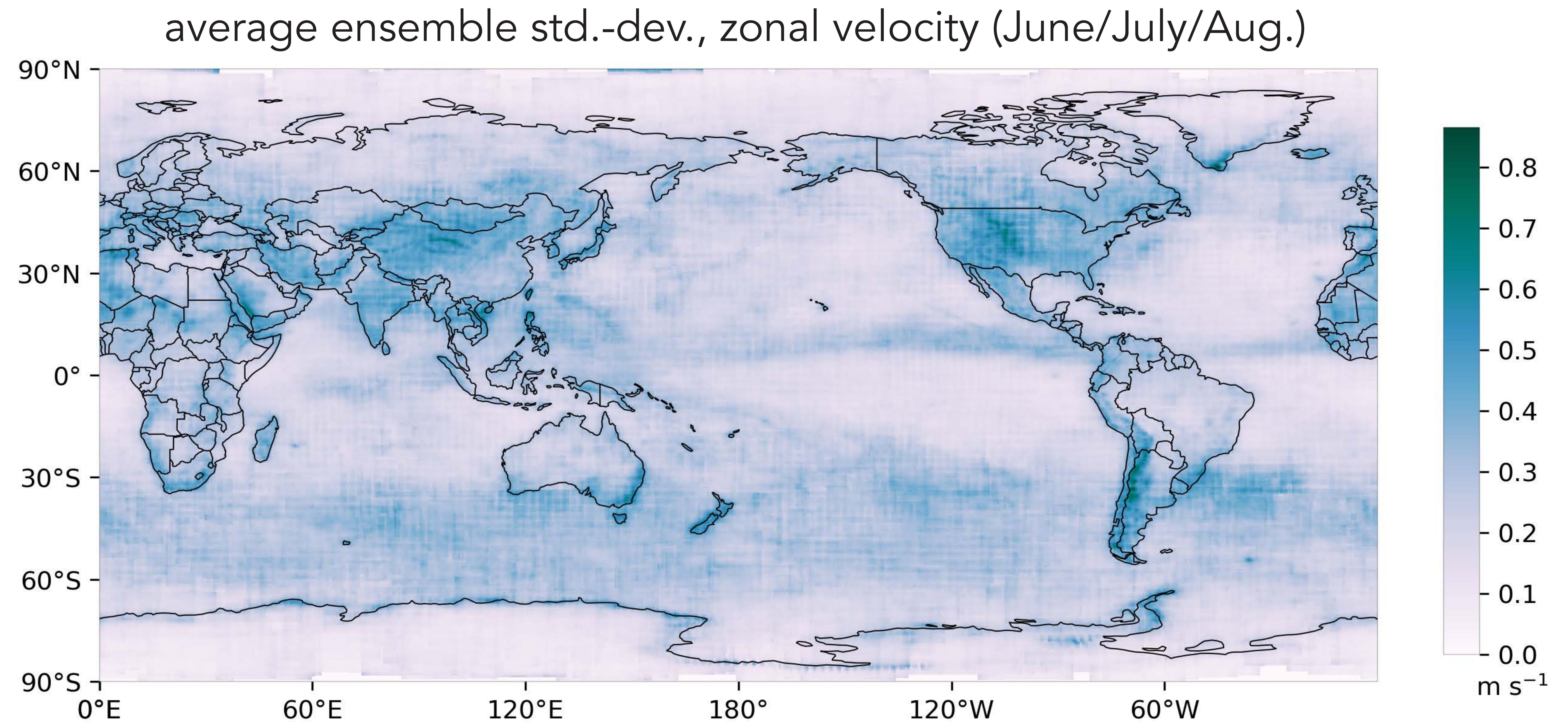


Statistical loss

average ensemble std.-dev., zonal velocity (Dec./Jan./Feb.)

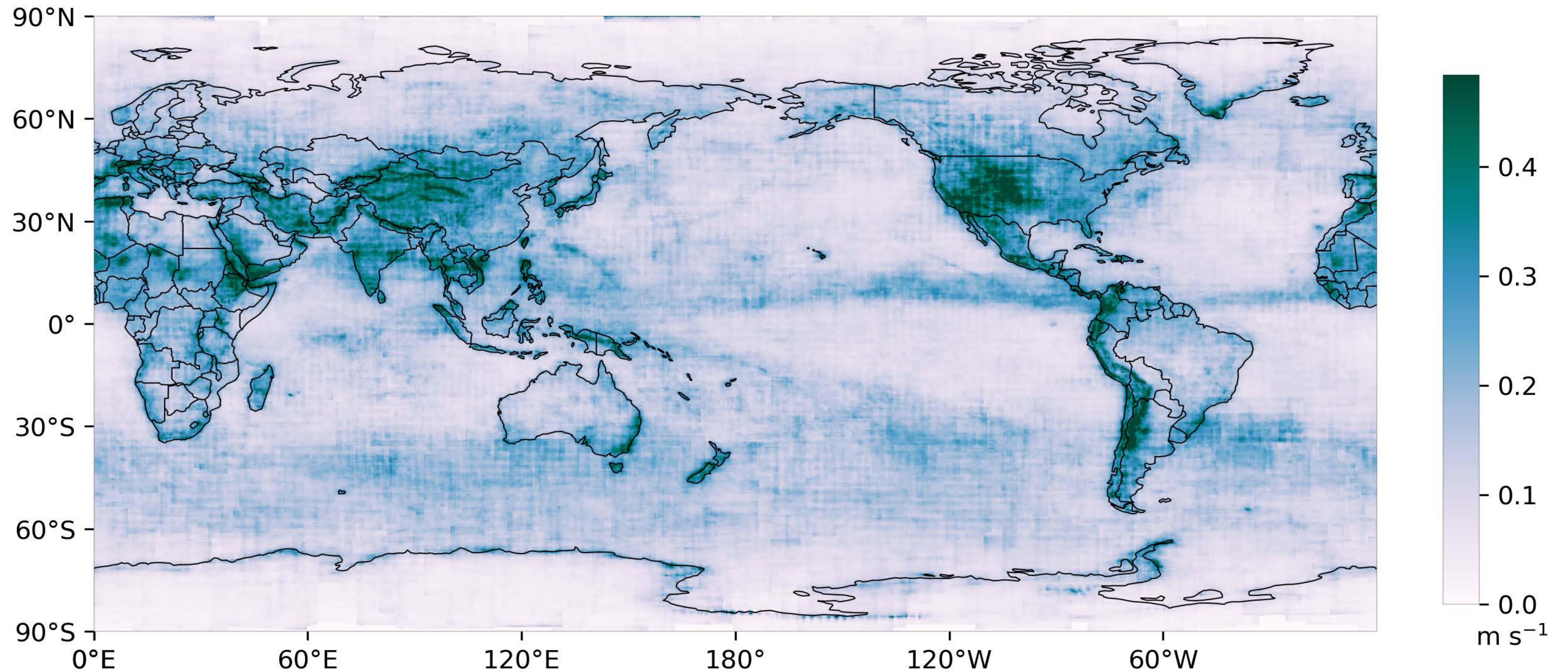


Statistical loss



Statistical loss

average error, zonal velocity (June/July/Aug.)



Intrinsic Capabilities

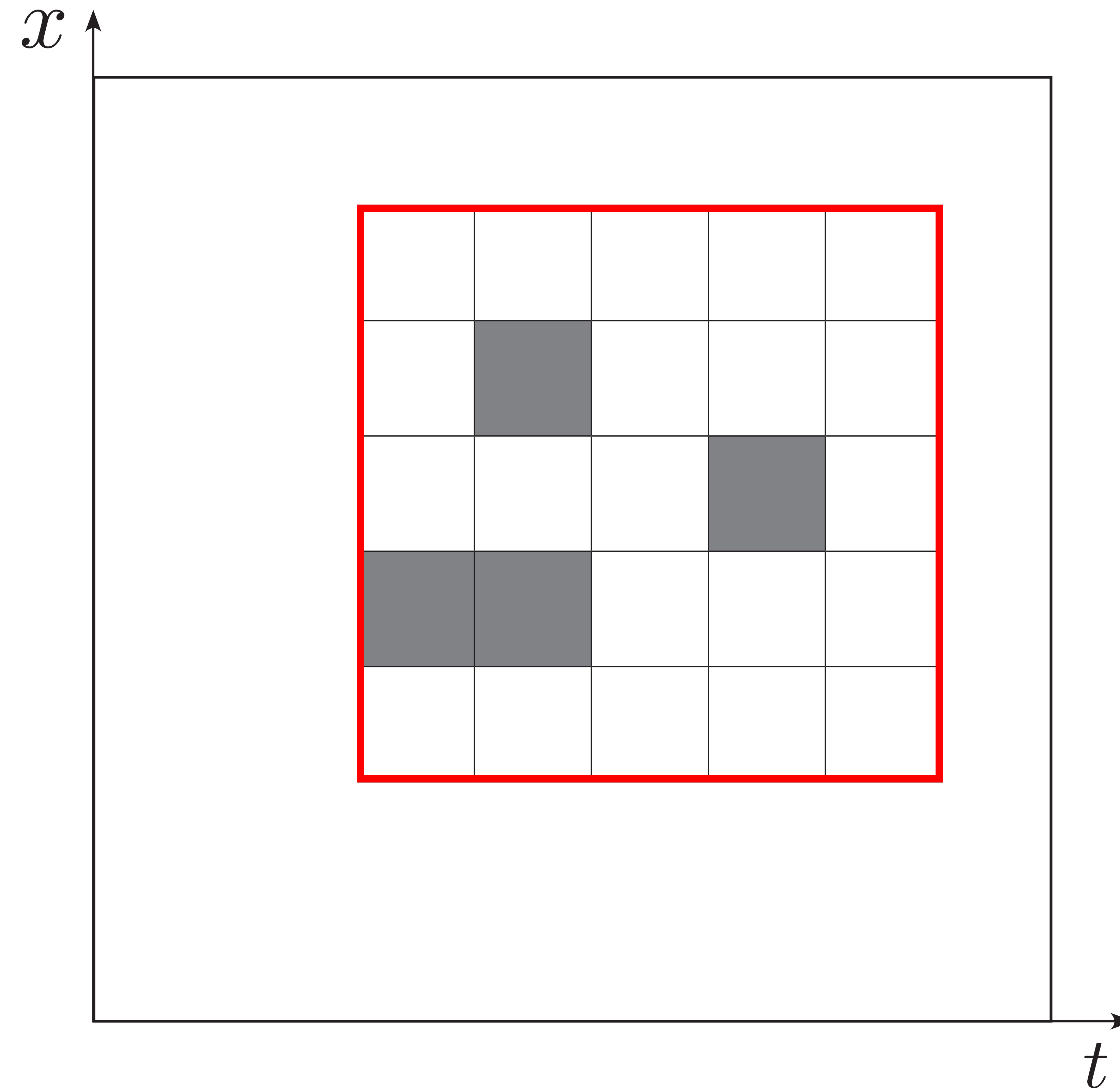
Zero-shot capabilities

- Numerical statistical atmospheric model:

$$p_{\theta}(y|x, \alpha)$$

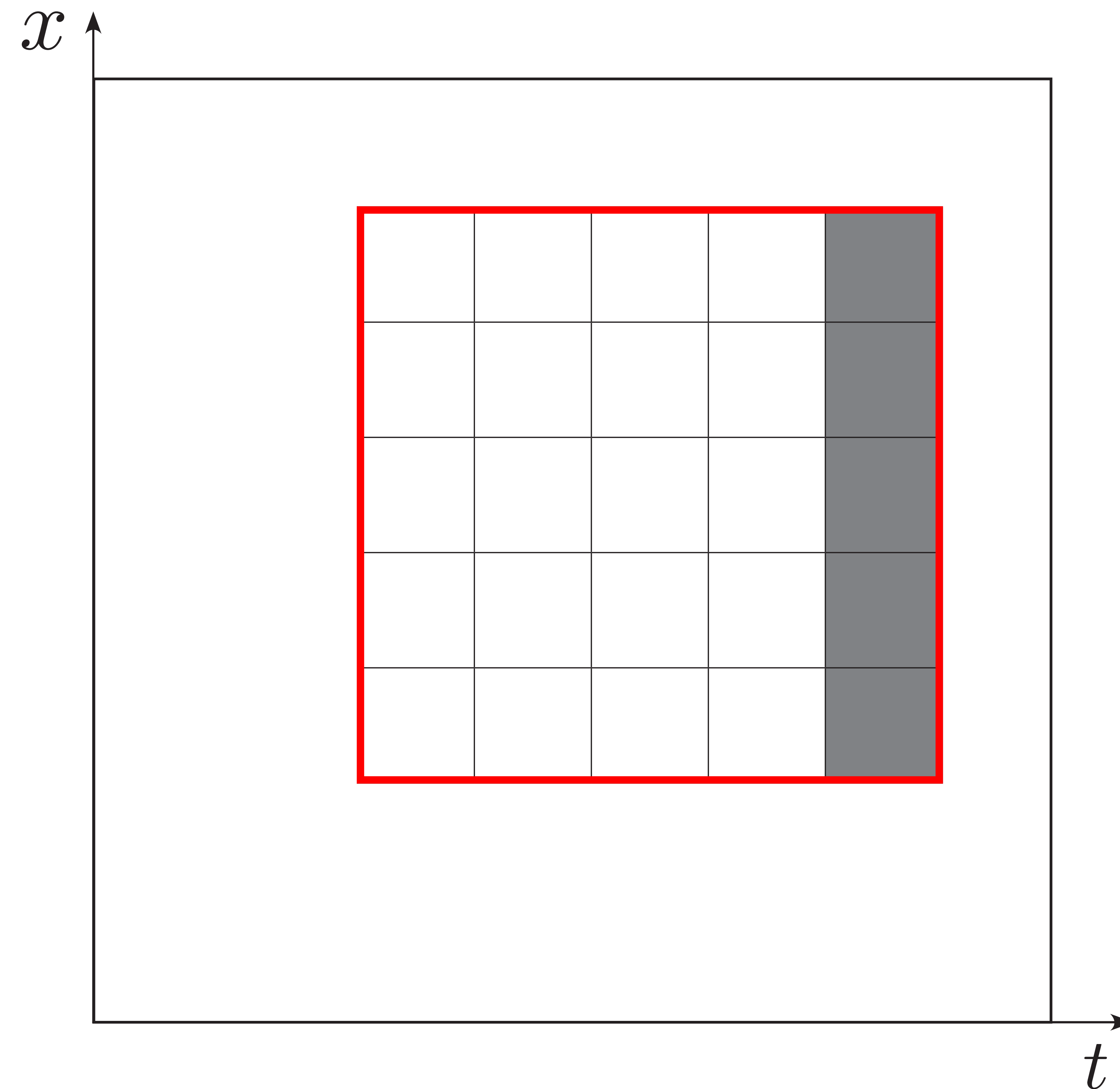
- Model directly includes important applications: forecasting, downscaling, temporal interpolation, ...

Zero-shot capabilities



Training task:
predict randomly
masked neighbor-
hoods in space-
time

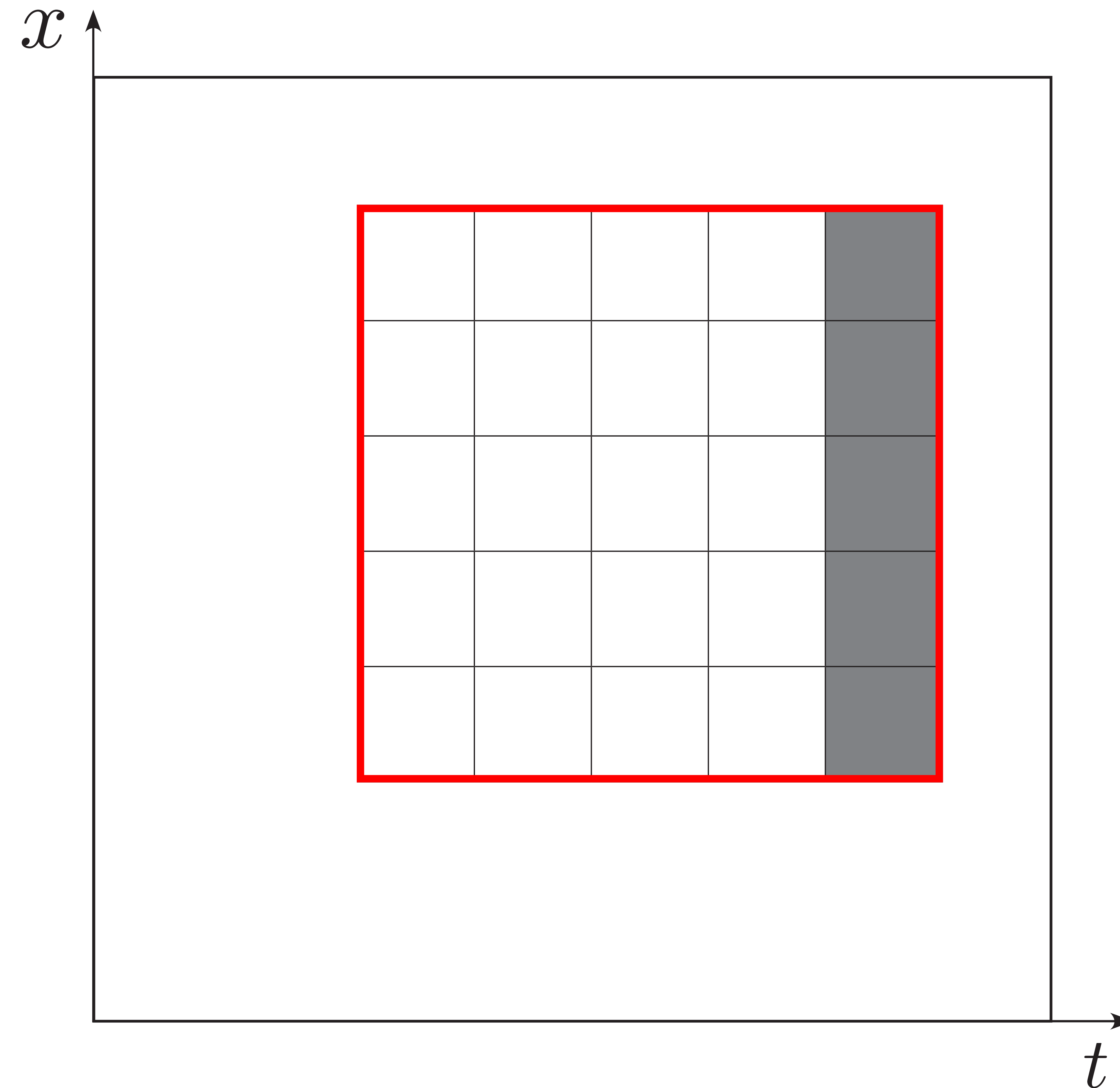
Zero-shot capabilities



Training task:
predict randomly
masked neighbor-
hoods in space-
time

Zero-shot capabilities

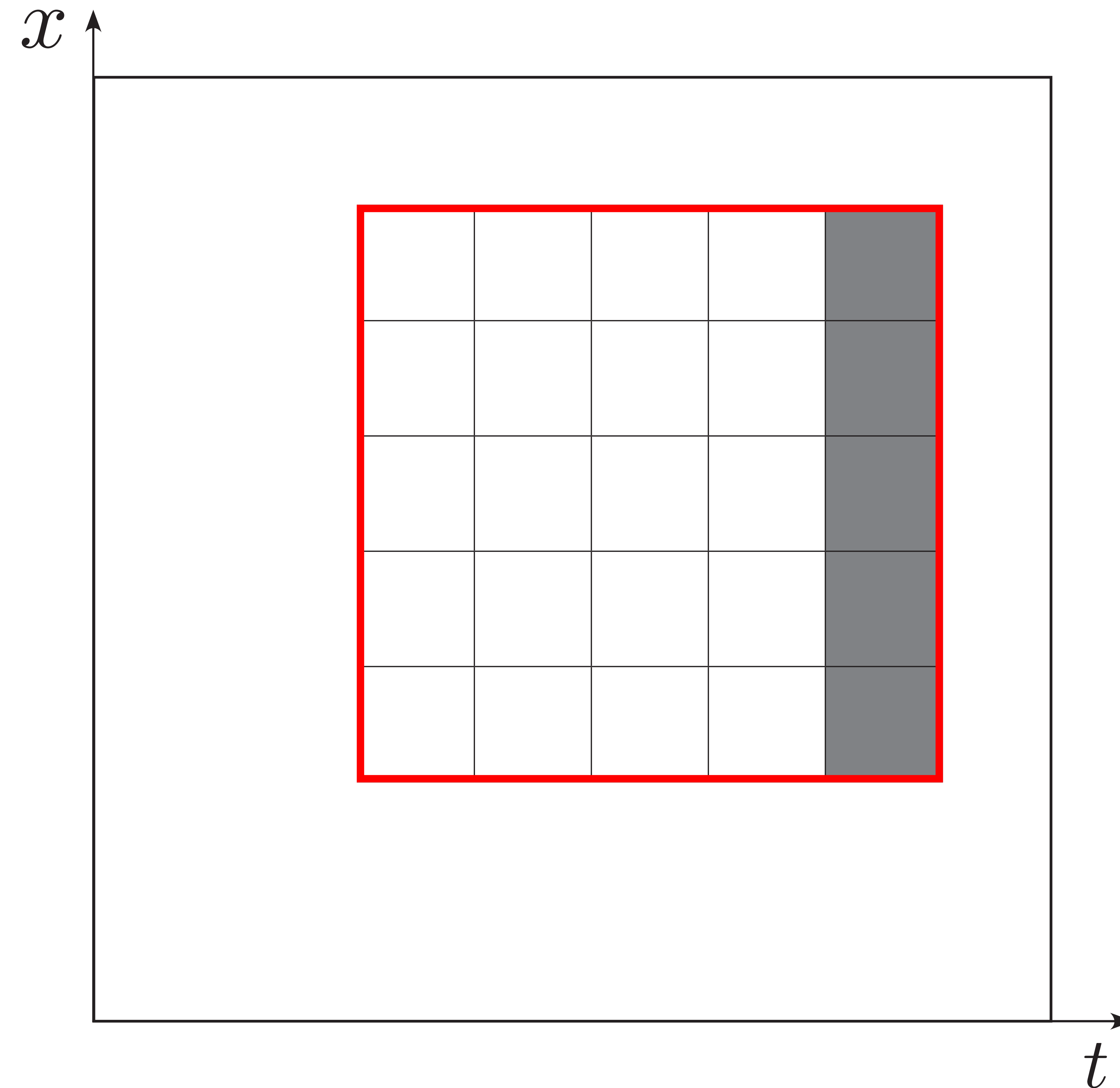
Forecasting



Training task:
predict randomly
masked neighbor-
hoods in space-
time

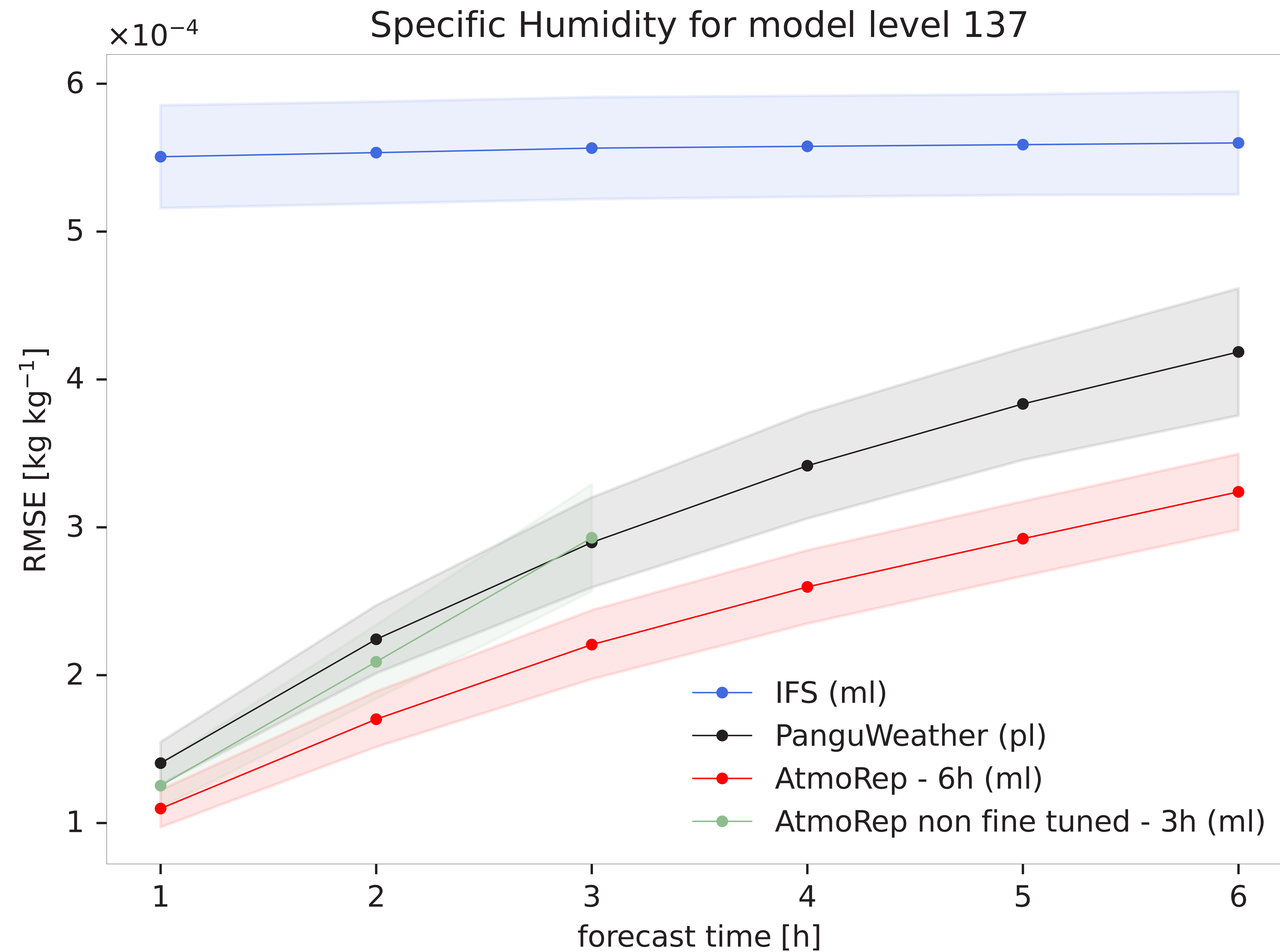
Zero-shot capabilities

Forecasting

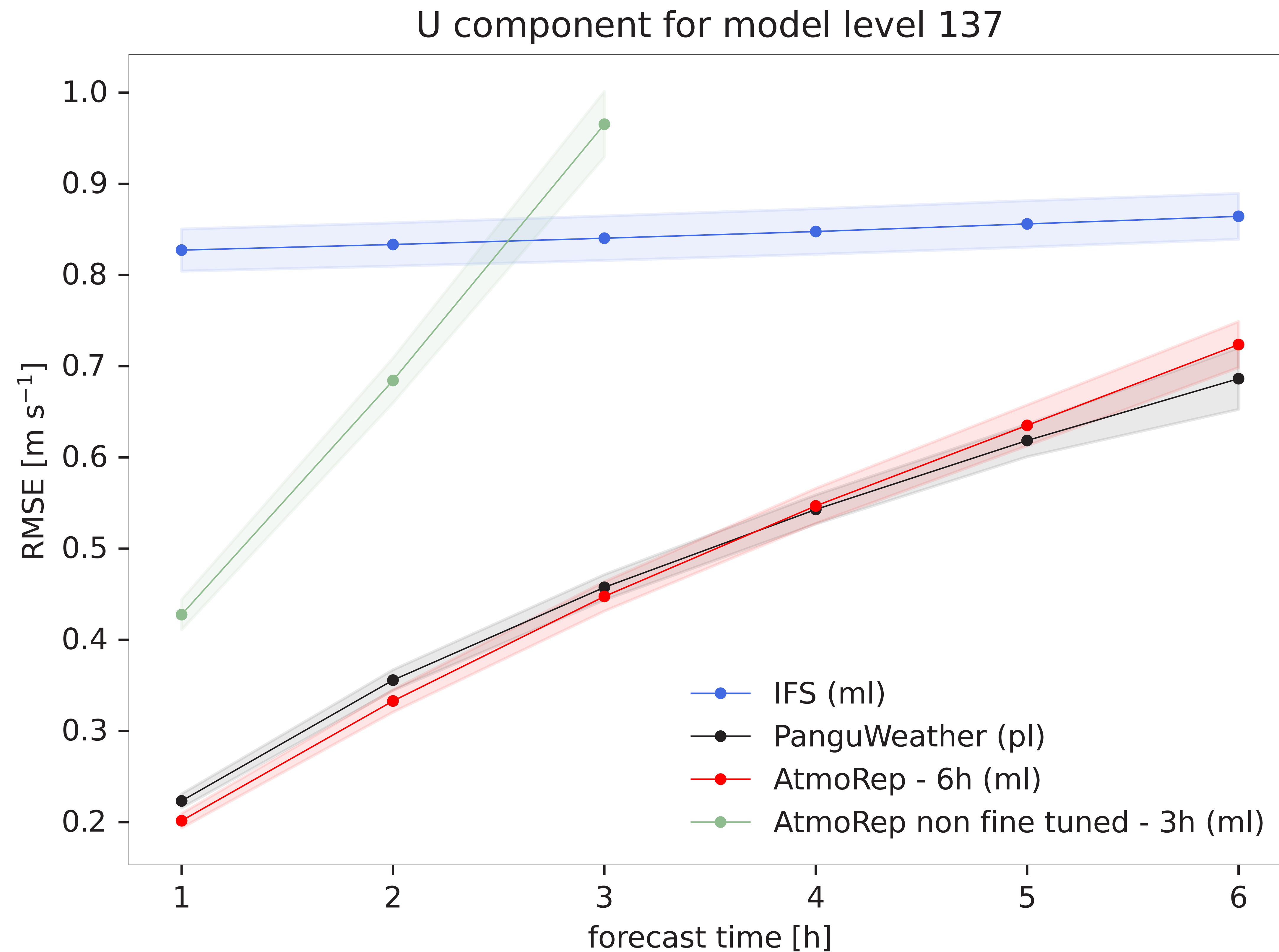


Training task:
predict randomly
masked neighbor-
hoods in space-
time

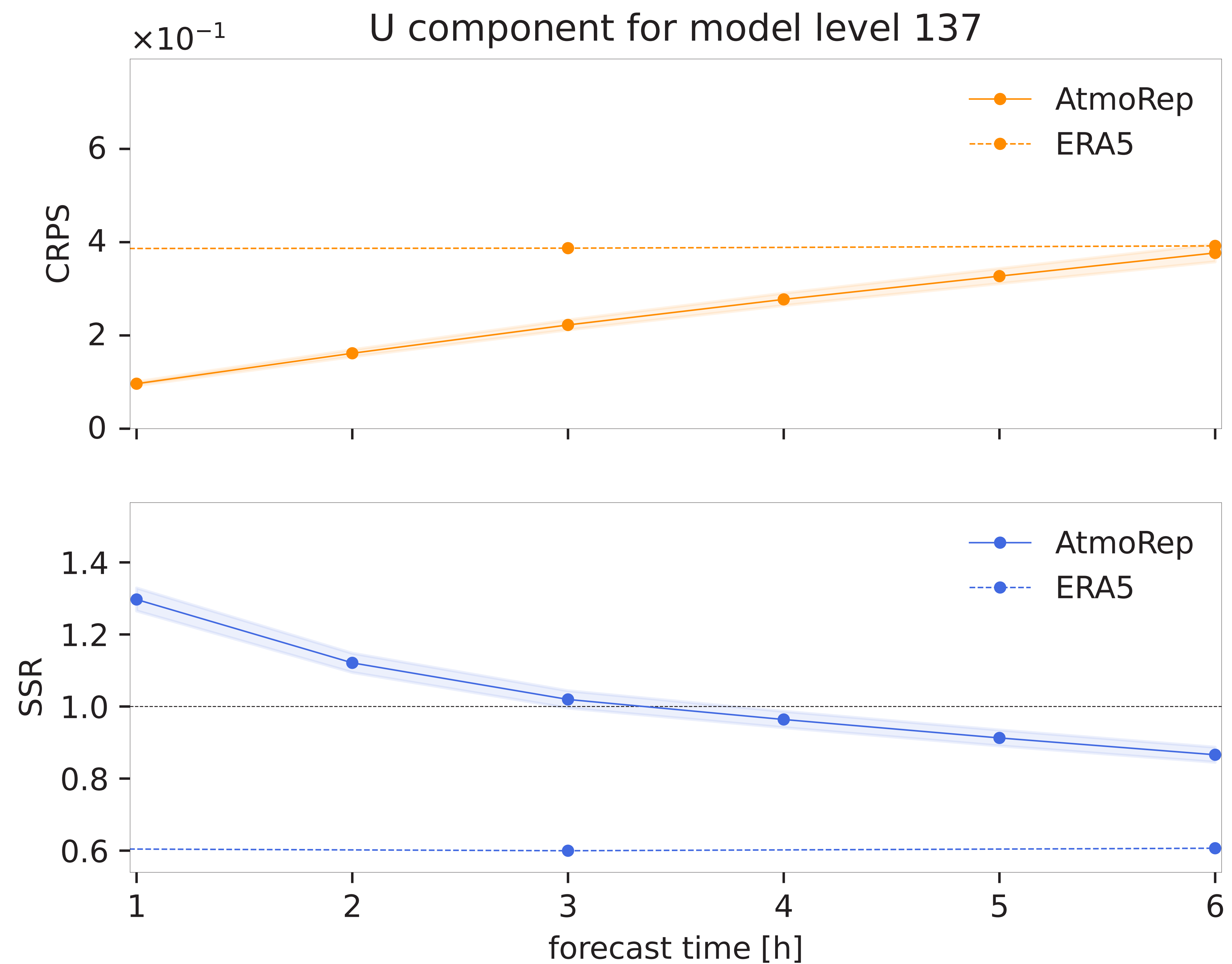
Zero-shot capabilities



Zero-shot capabilities

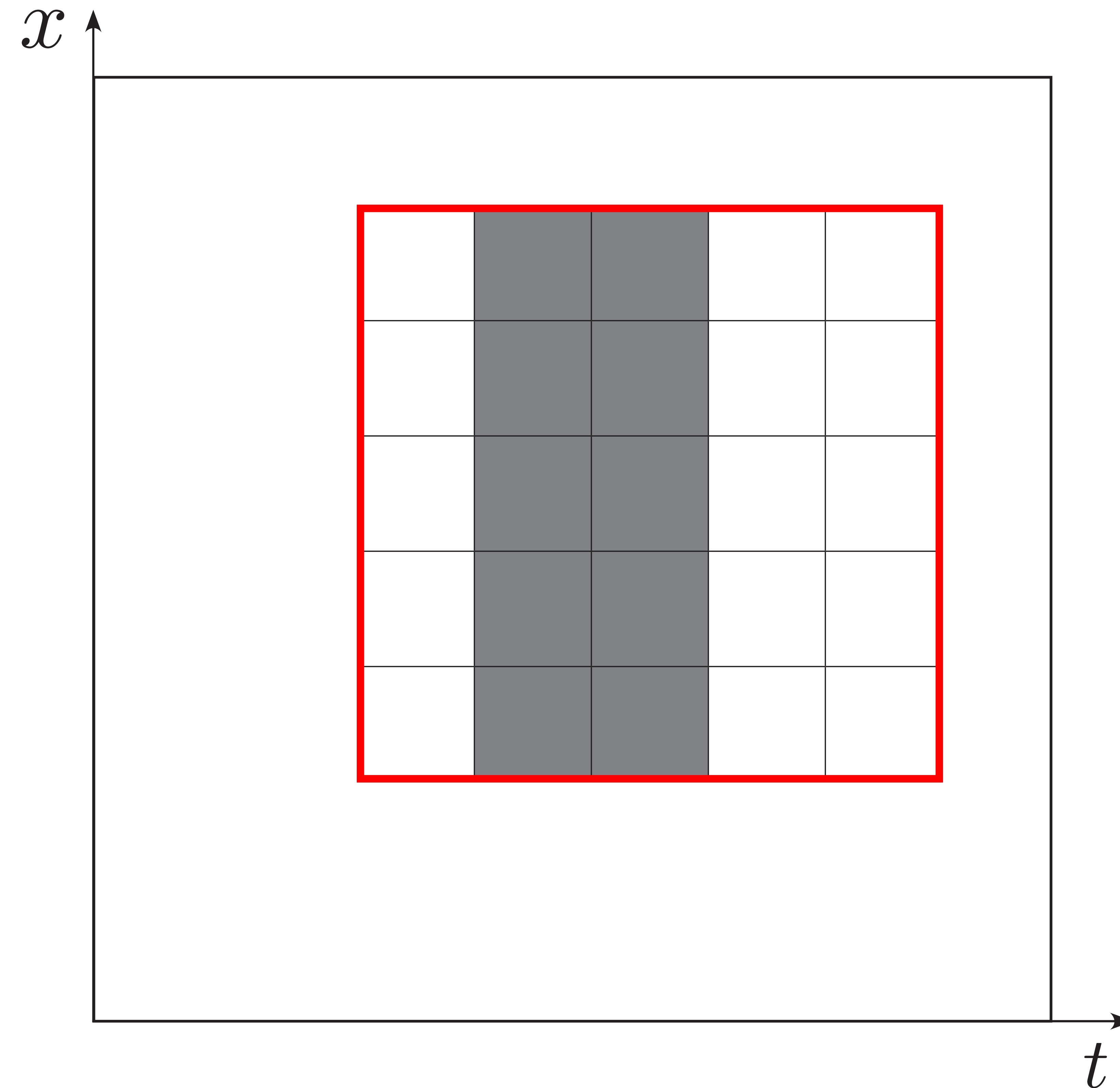


Zero-shot capabilities



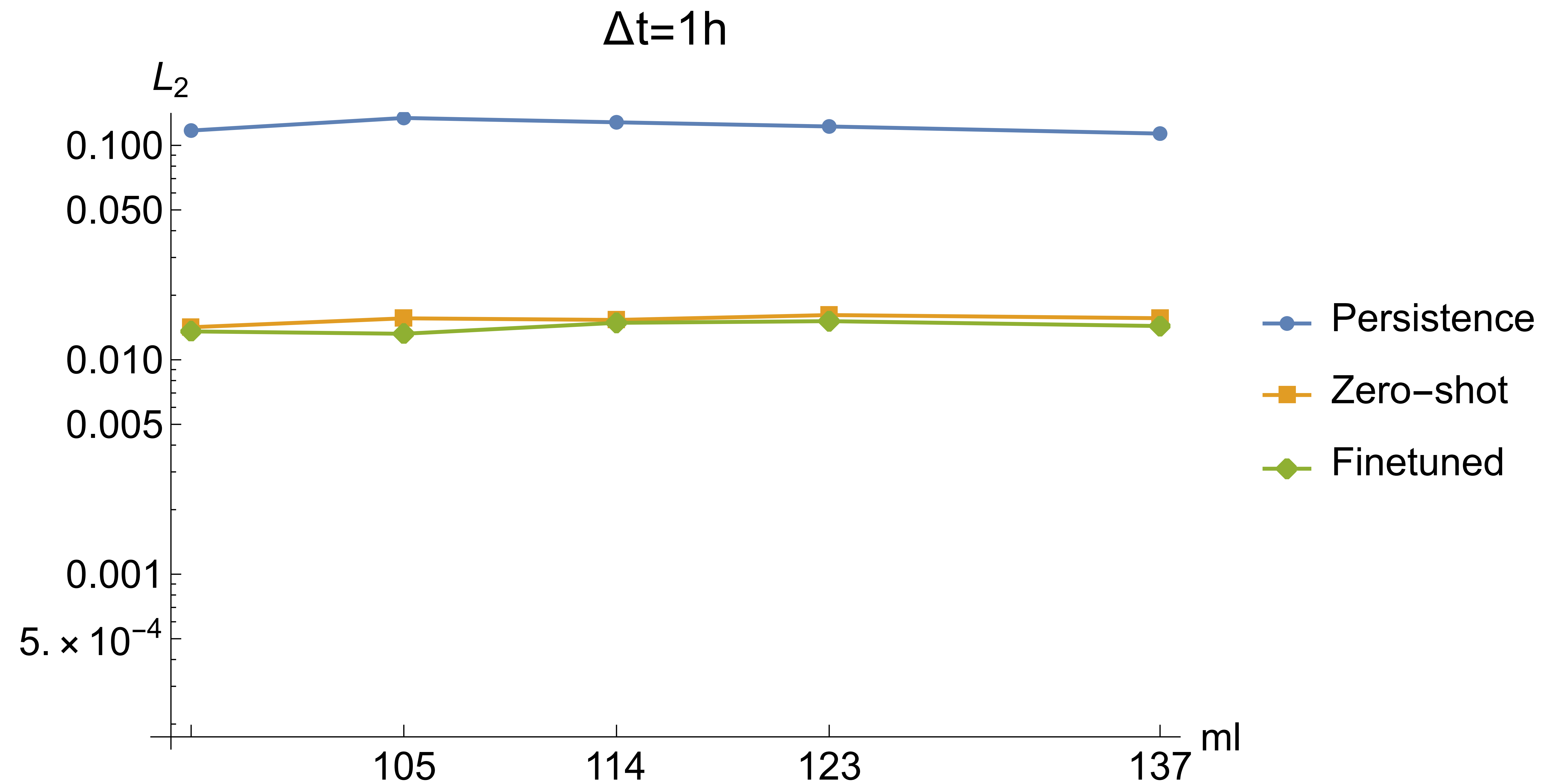
Zero-shot capabilities

Temporal
interpolation



Training task:
predict randomly
masked neighbor-
hoods in space-
time

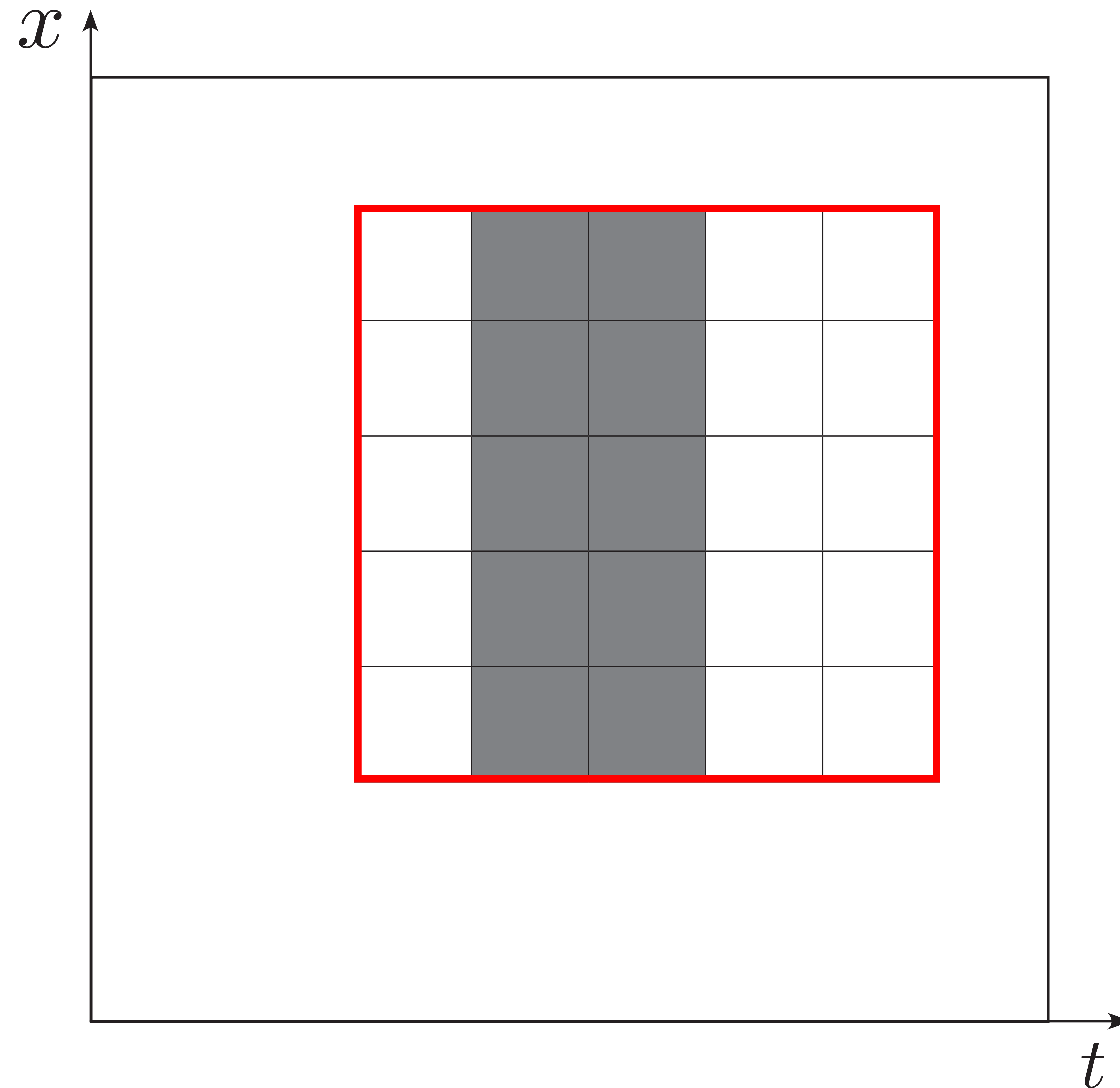
Zero-shot capabilities



Zero-shot capabilities

Also:

- spatial interpolation (missing data)
- downscaling
- ...



Training task:

predict randomly
masked neighbor-
hoods in space-
time

Model correction

- Numerical statistical atmospheric model:

$$p_{\theta}(y|x, \alpha)$$

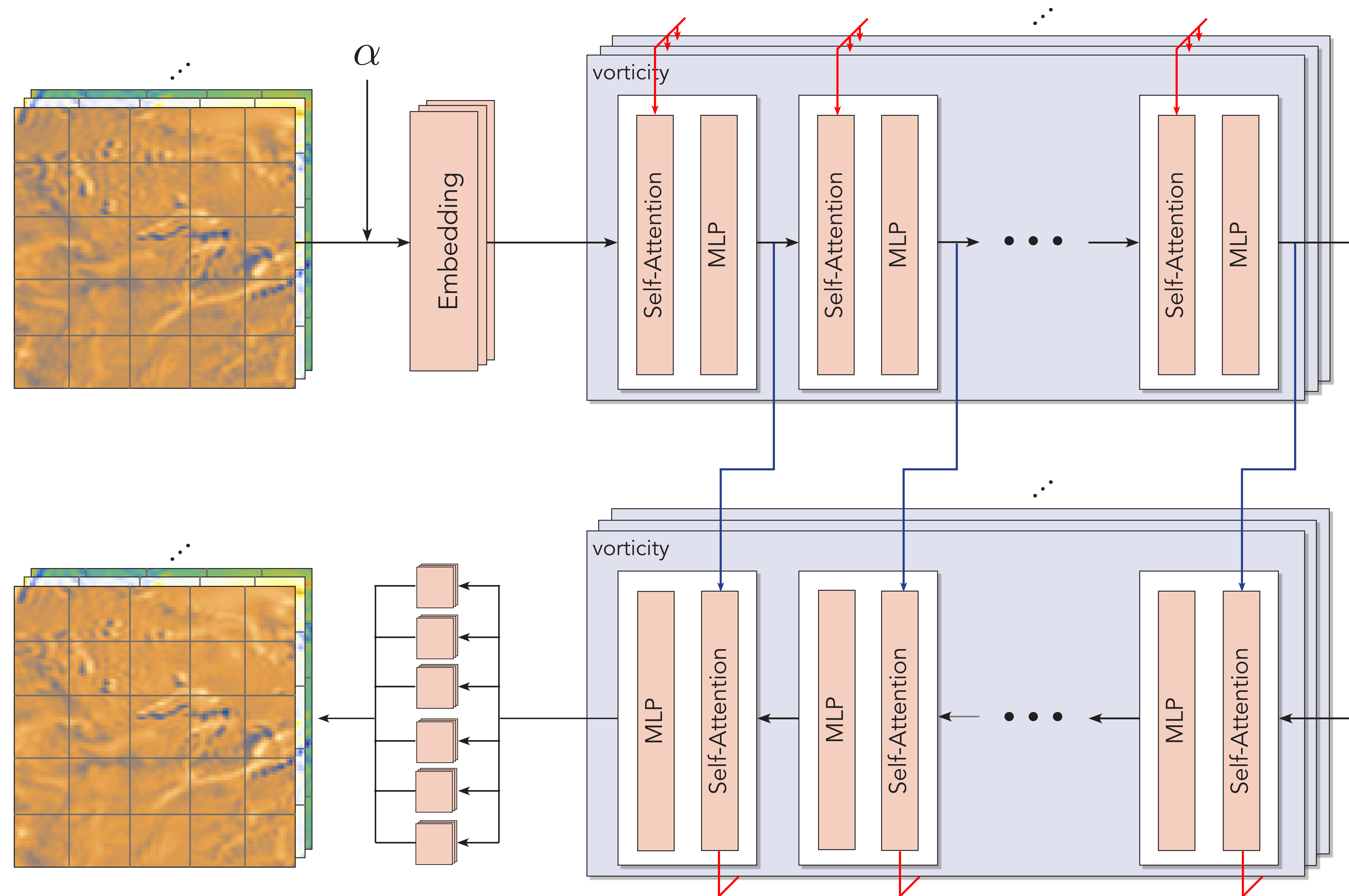
Model correction

- Numerical statistical atmospheric model:

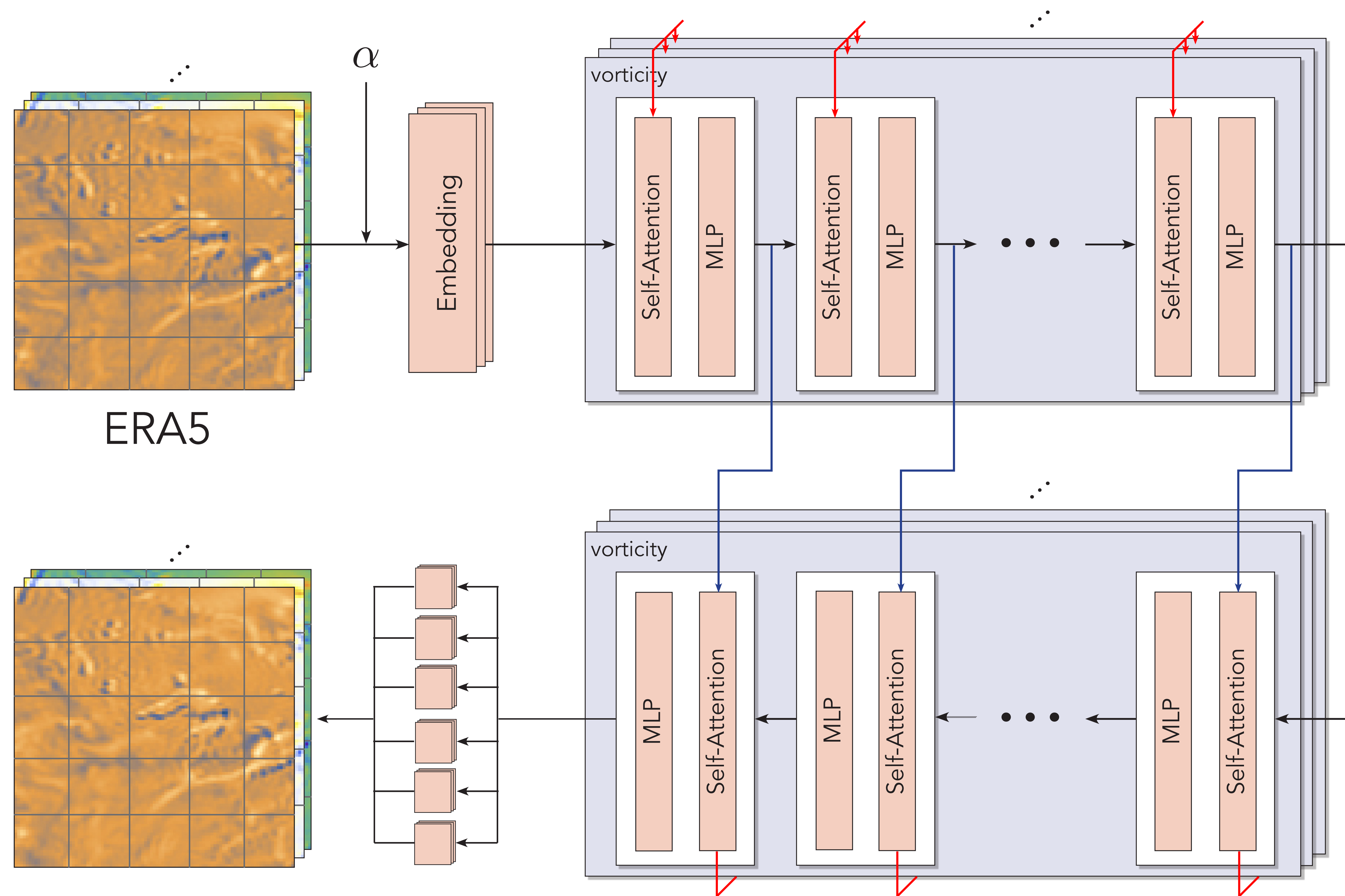
$$p_{\theta}(y|x, \alpha)$$

approximate initial state

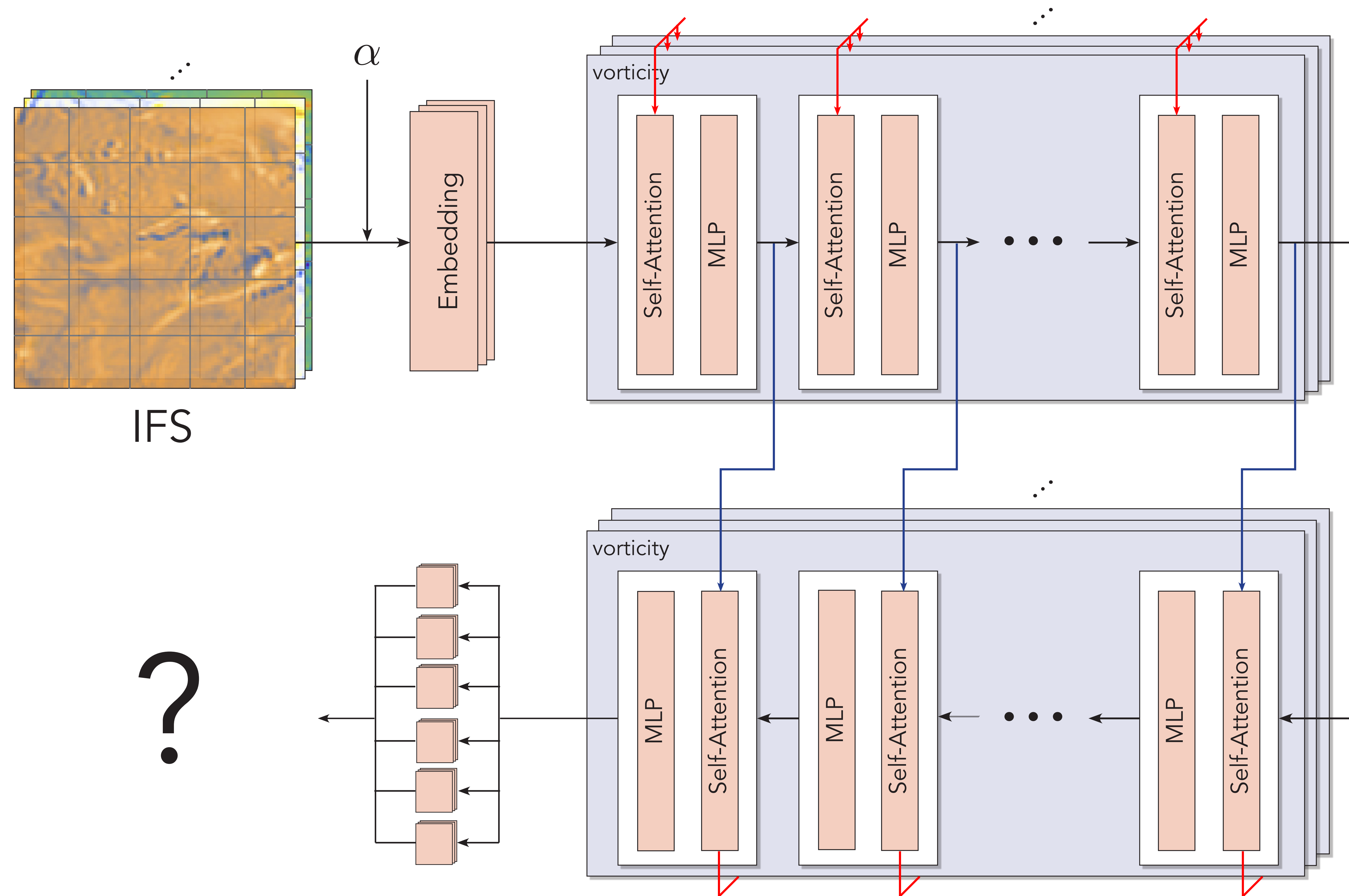
Model correction



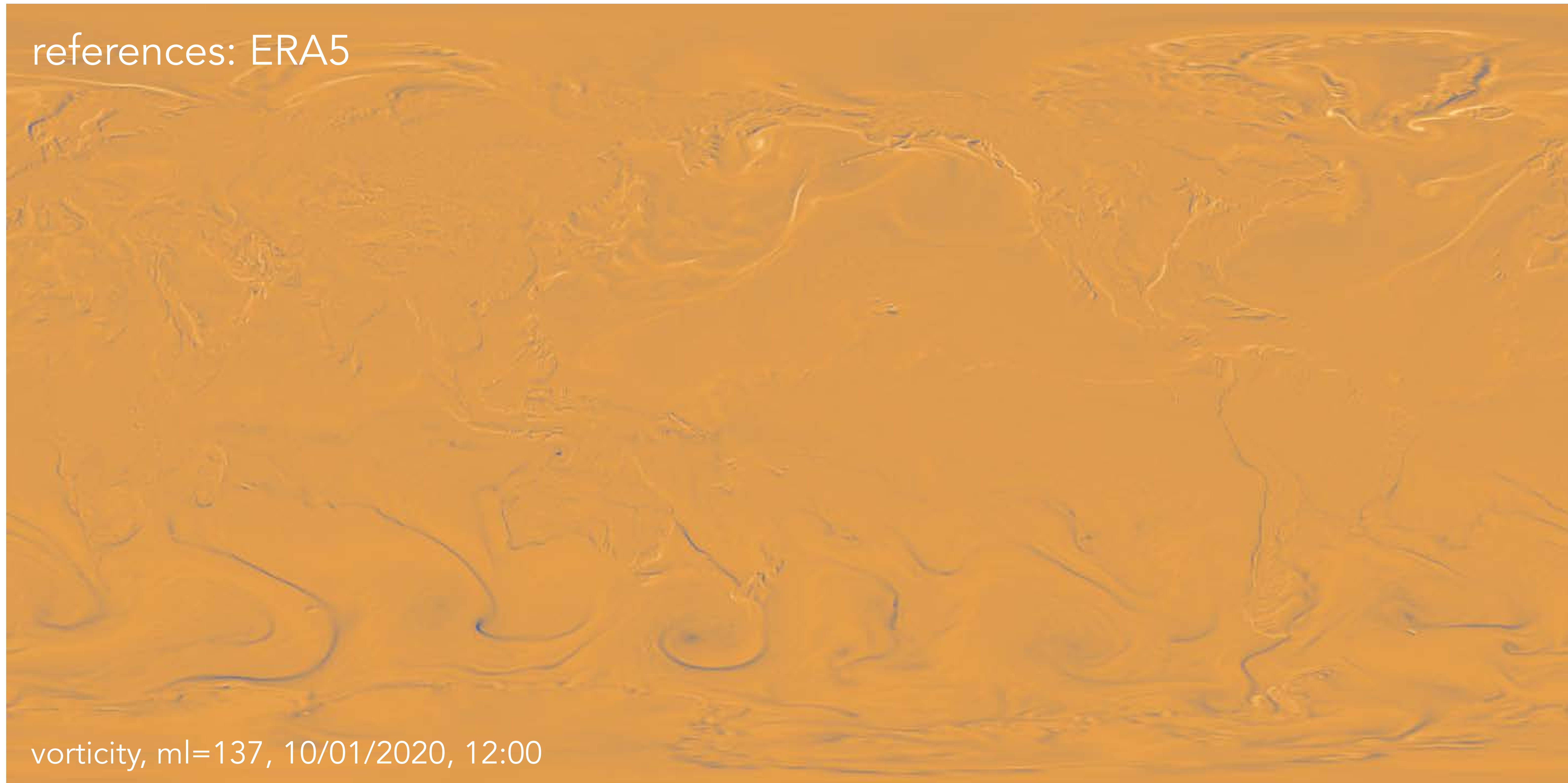
Model correction



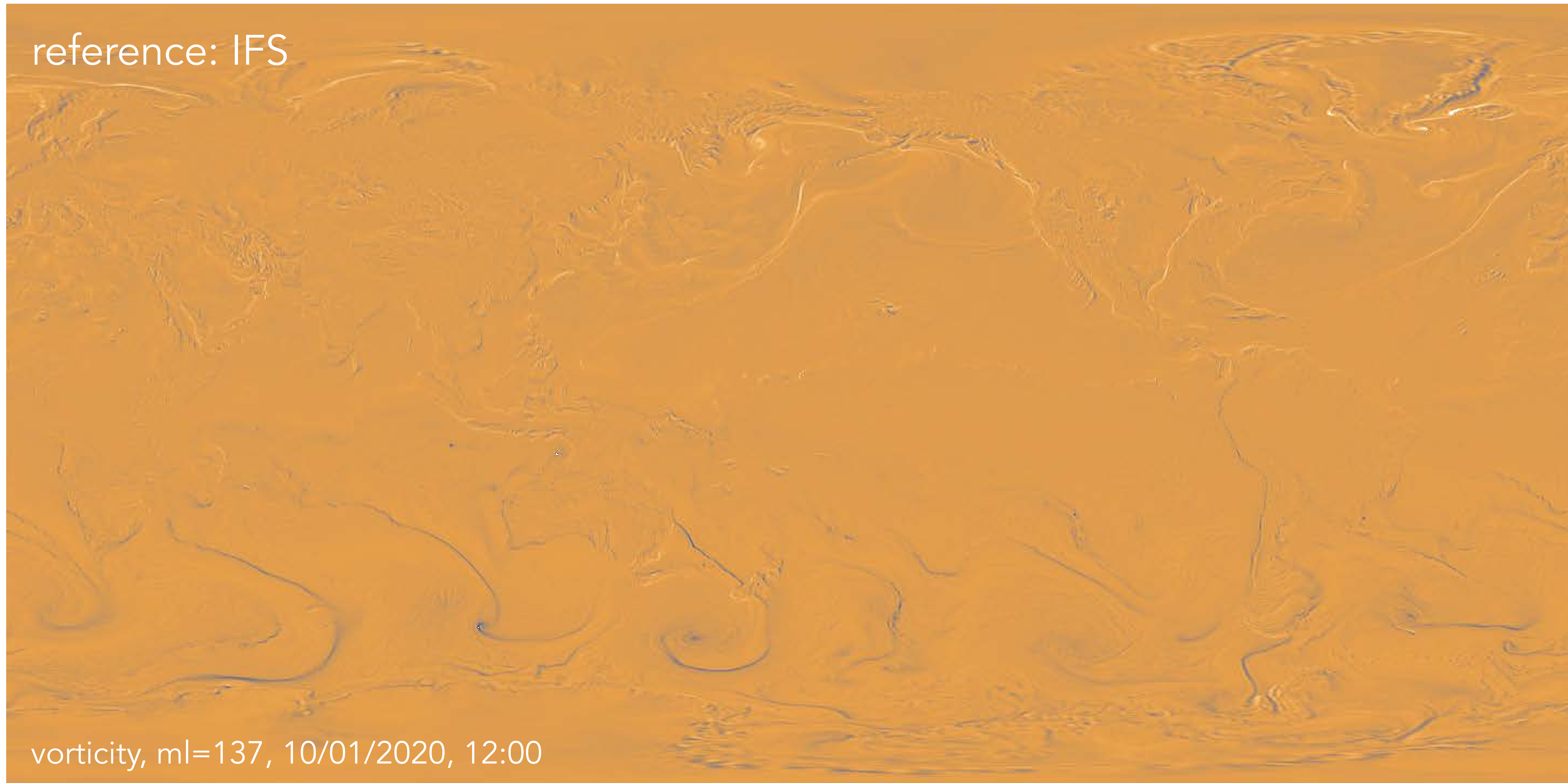
Model correction



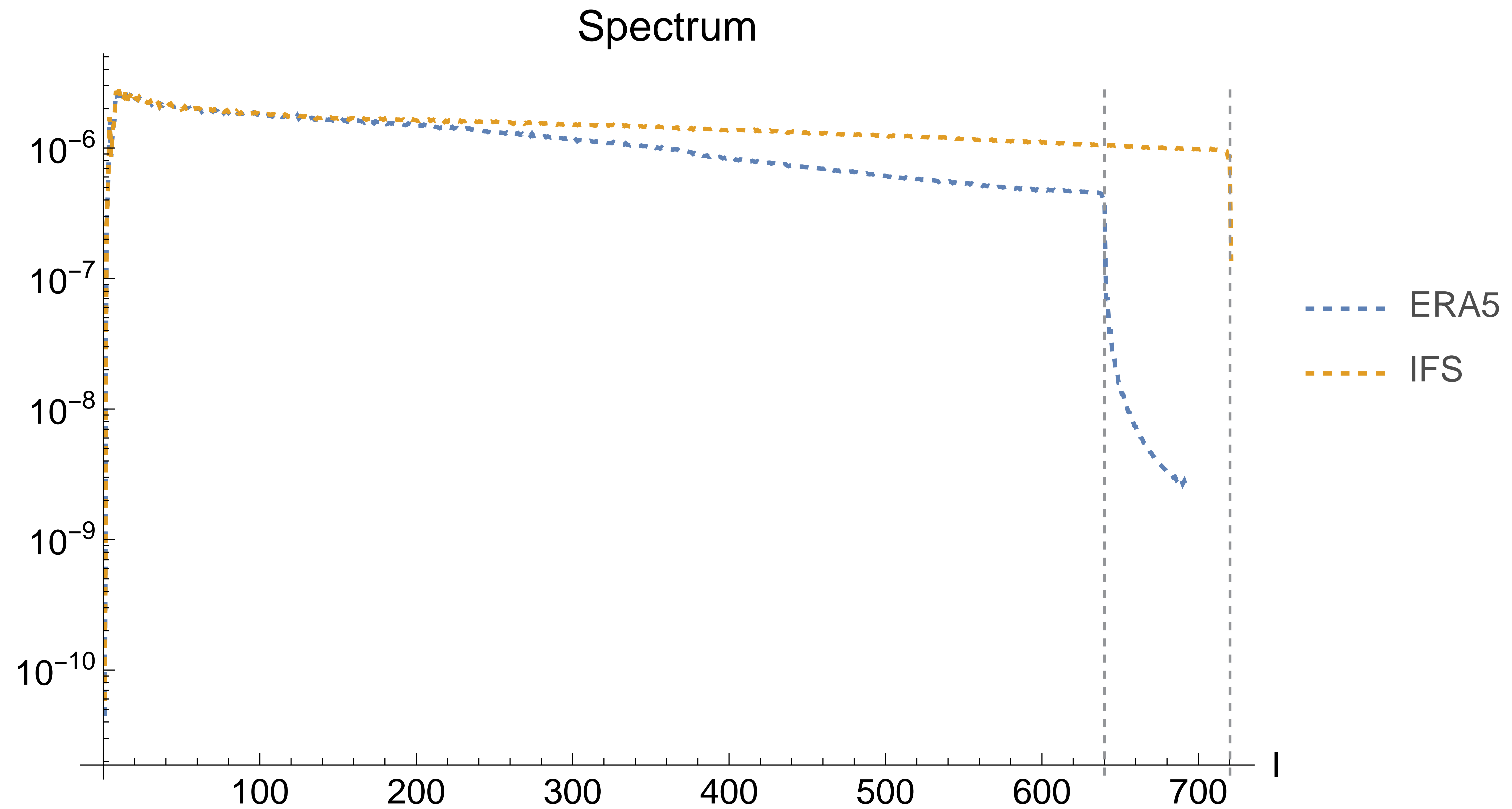
Model correction



Model correction



Model correction



Model correction

network input: ERA5, 1h prediction

vorticity, ml=137, 10/01/2020, 13:00

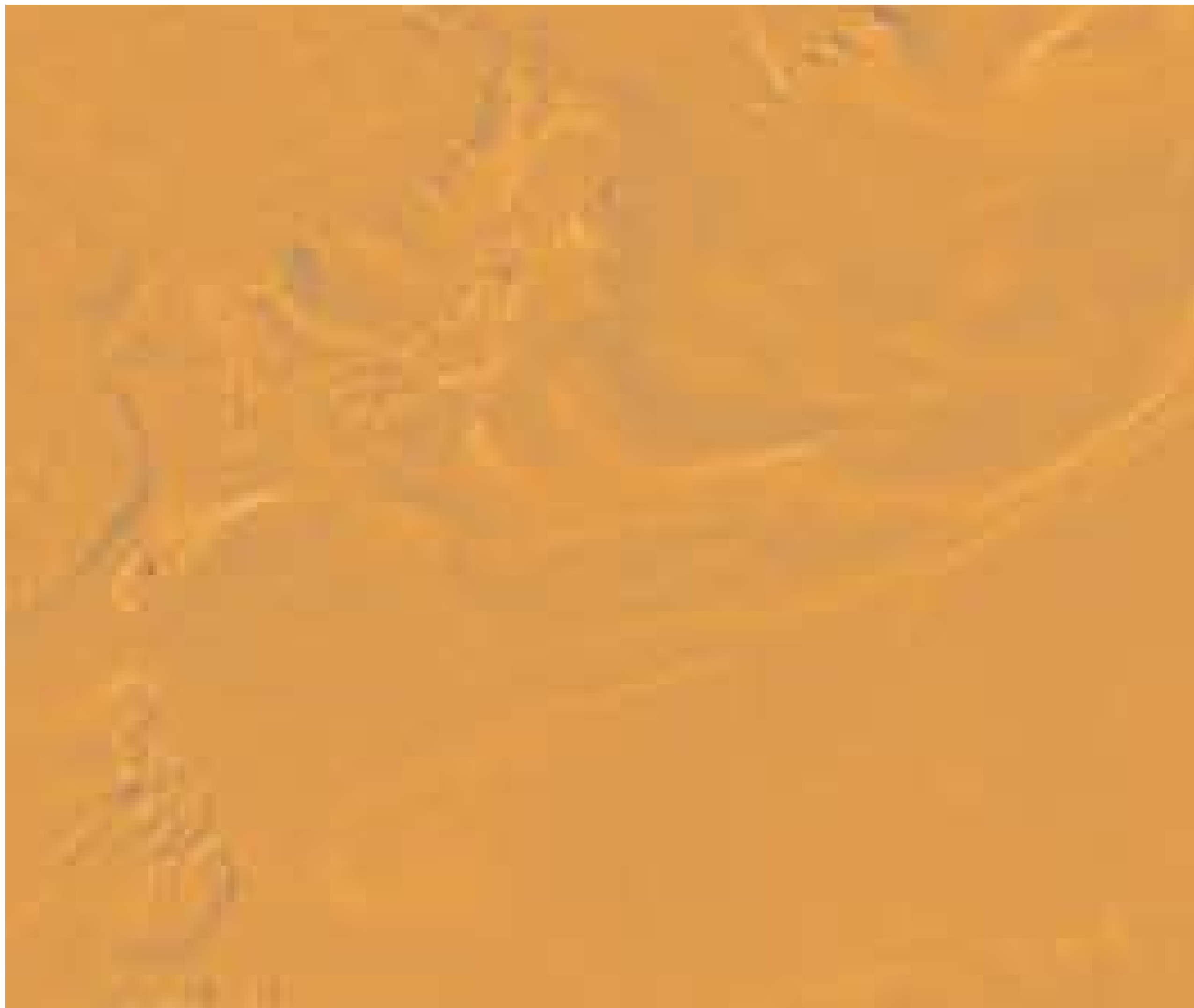
Model correction

network input: IFS, 1h prediction

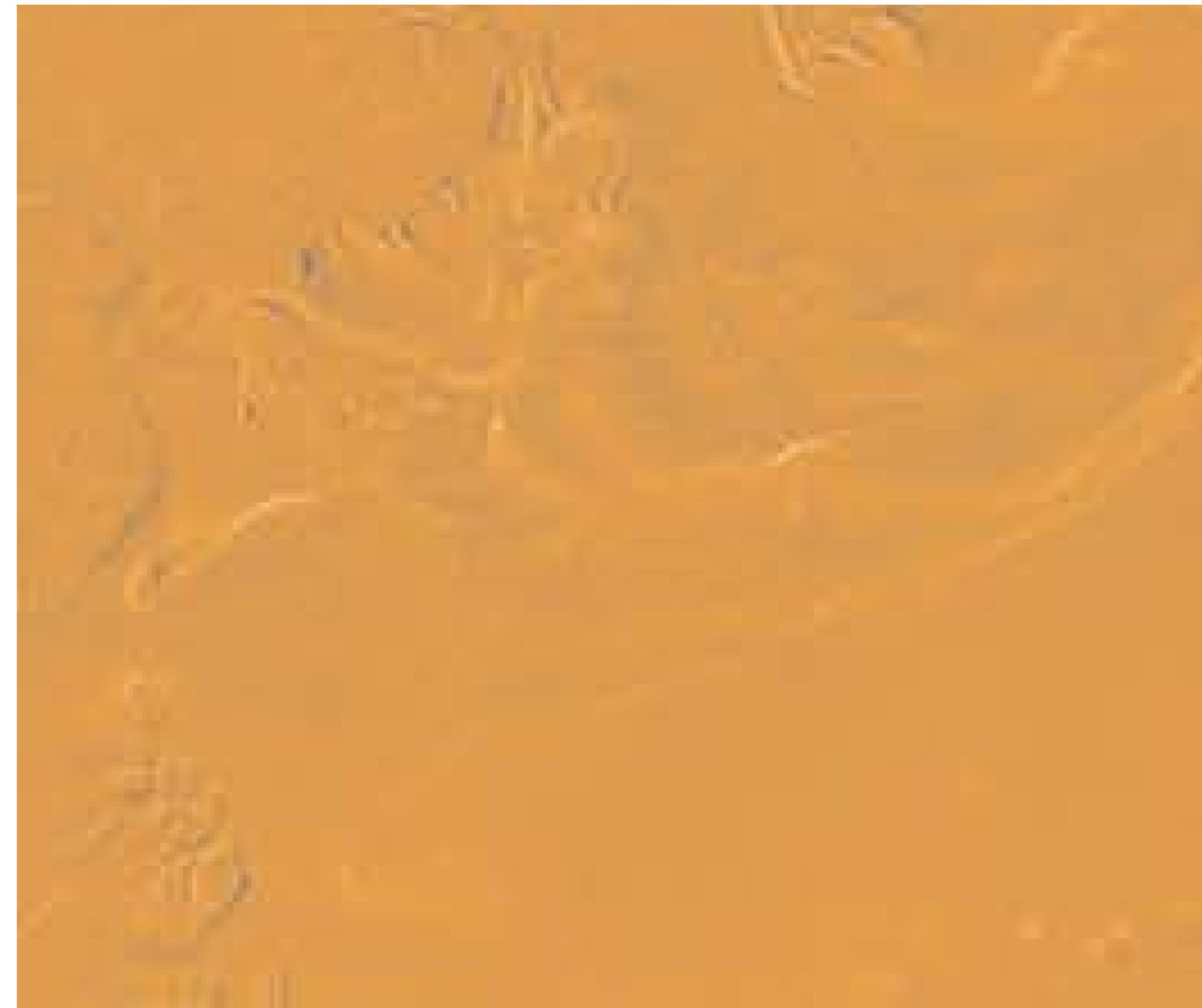
vorticity, ml=137, 10/01/2020, 13:00

Model correction

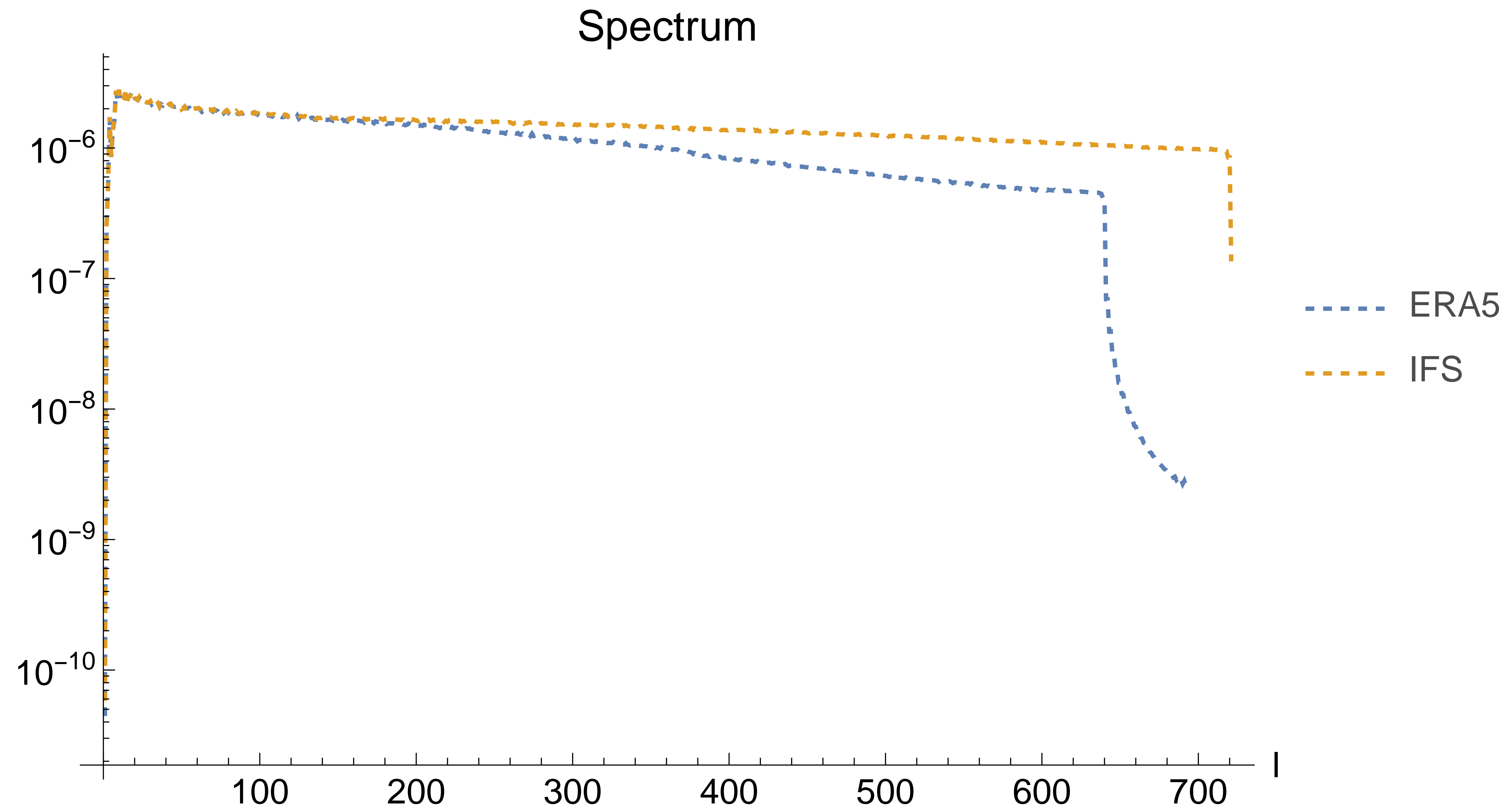
network input: ERA5



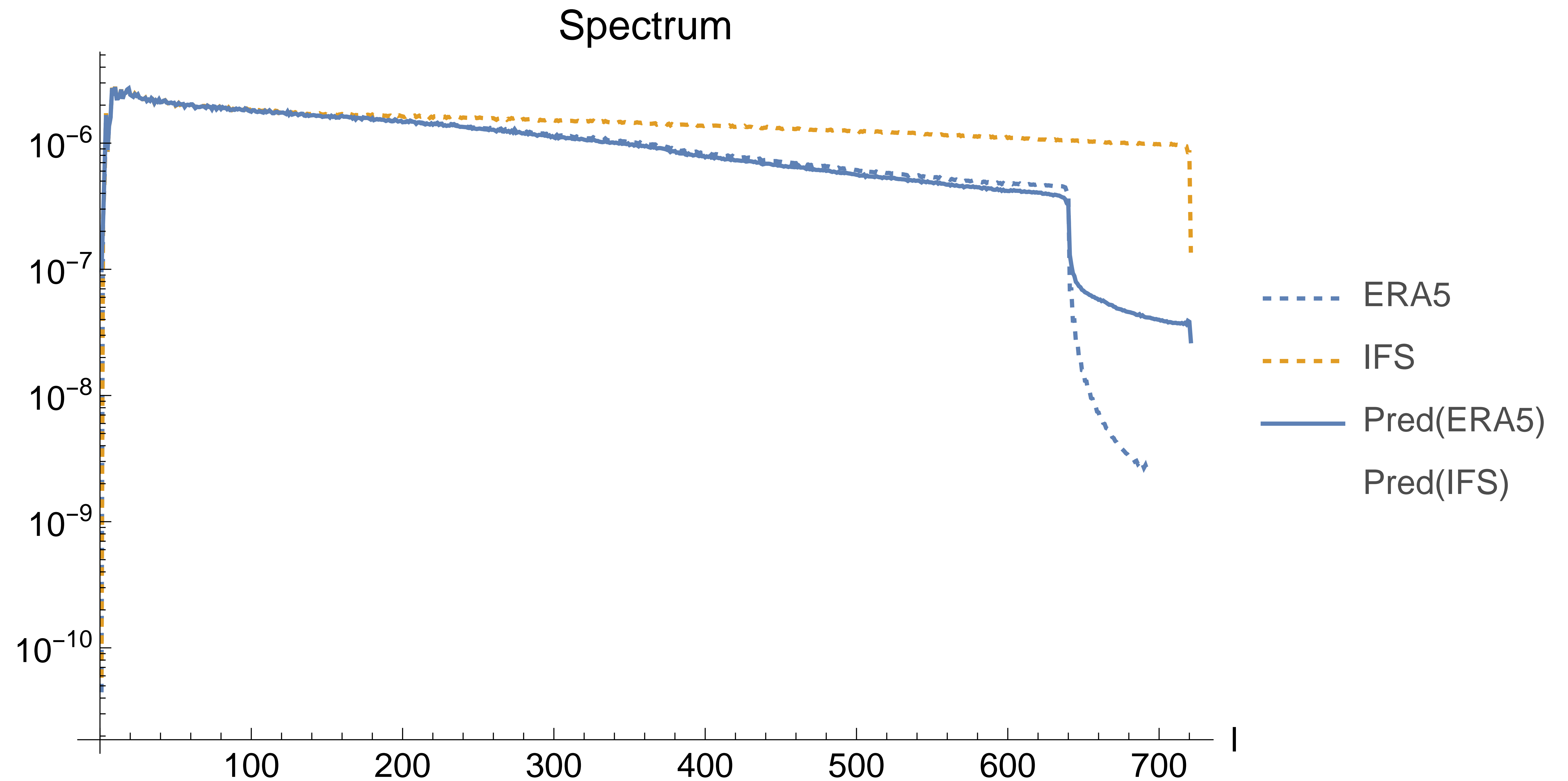
network input: IFS



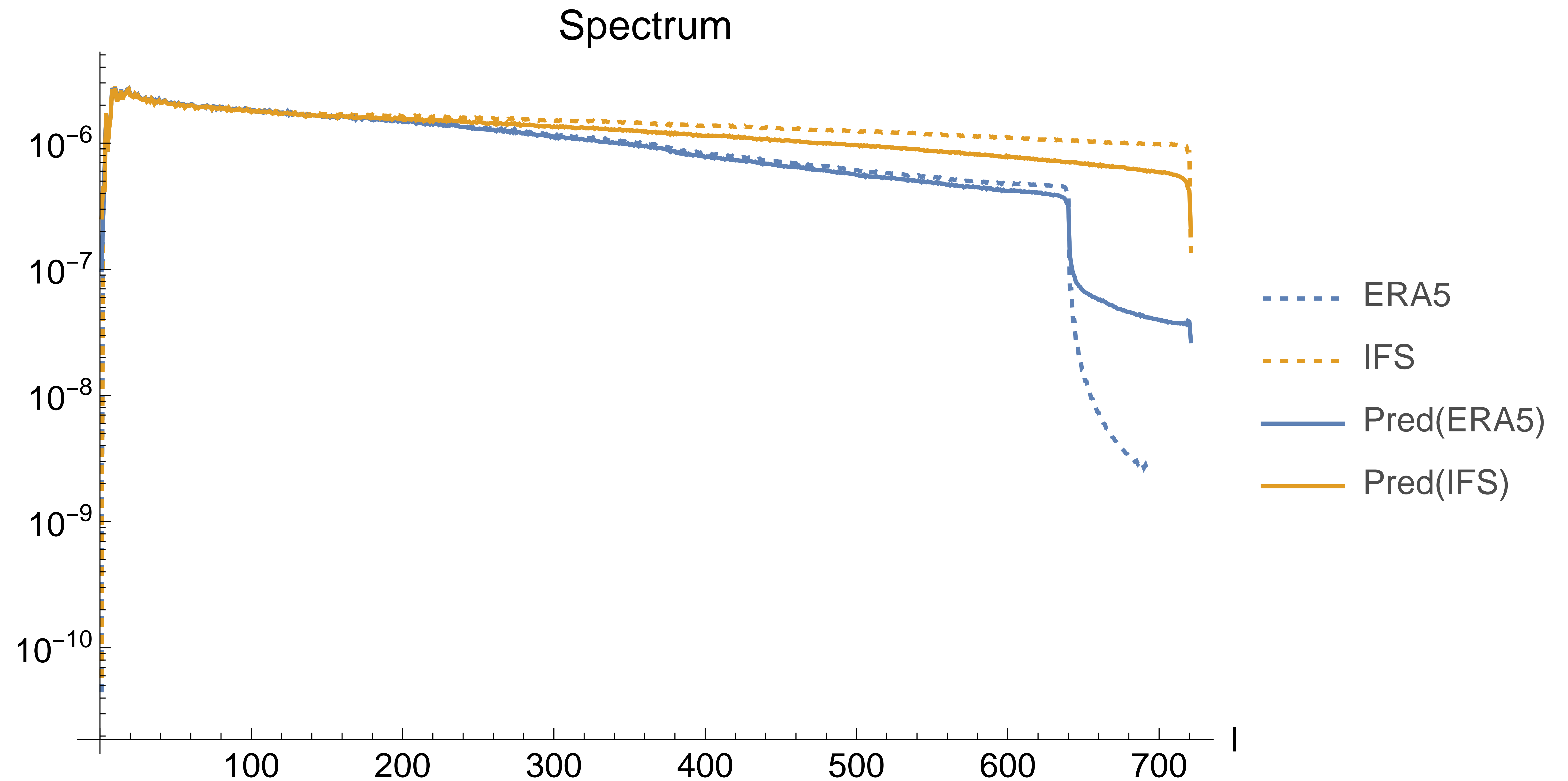
Model correction



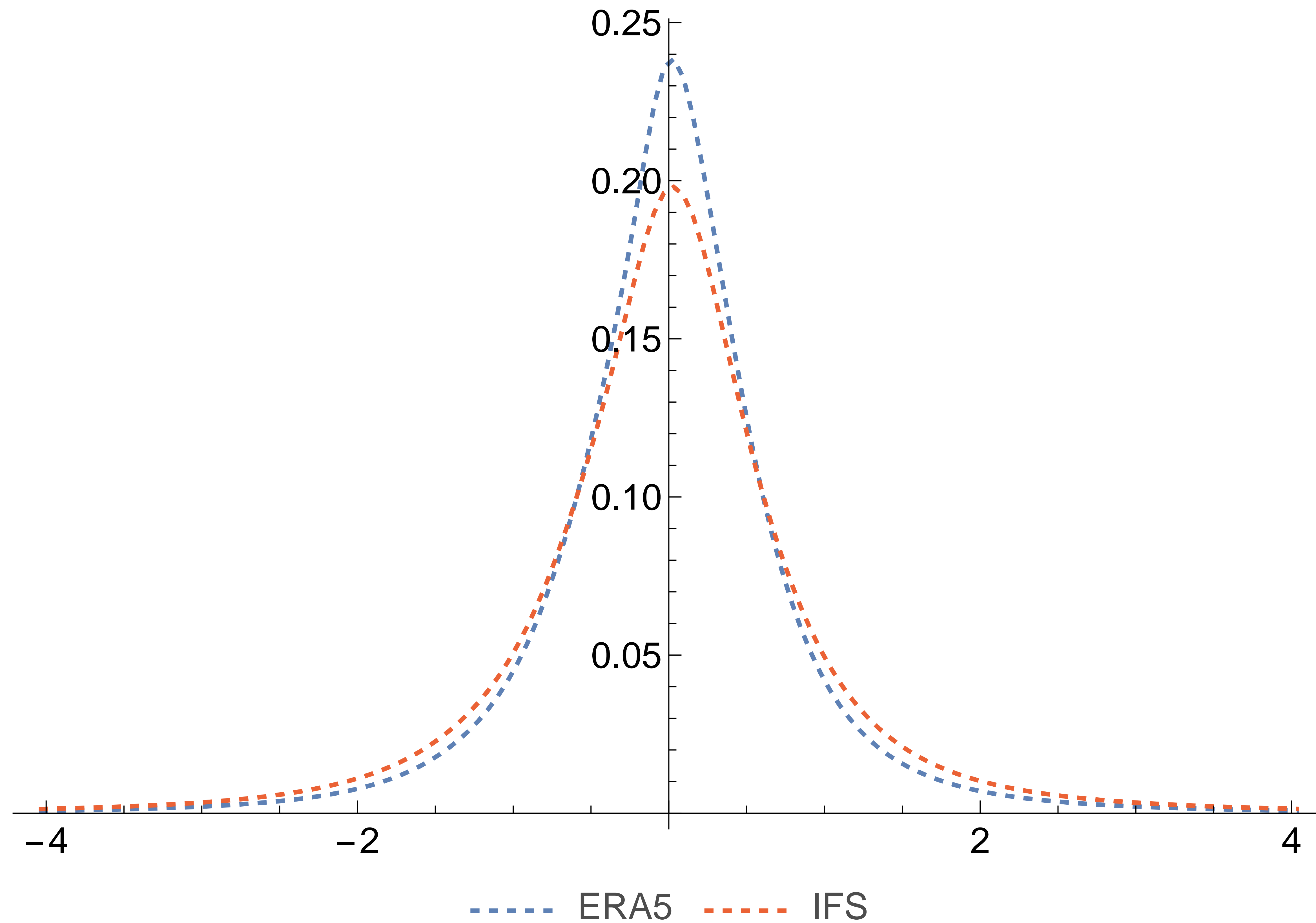
Model correction



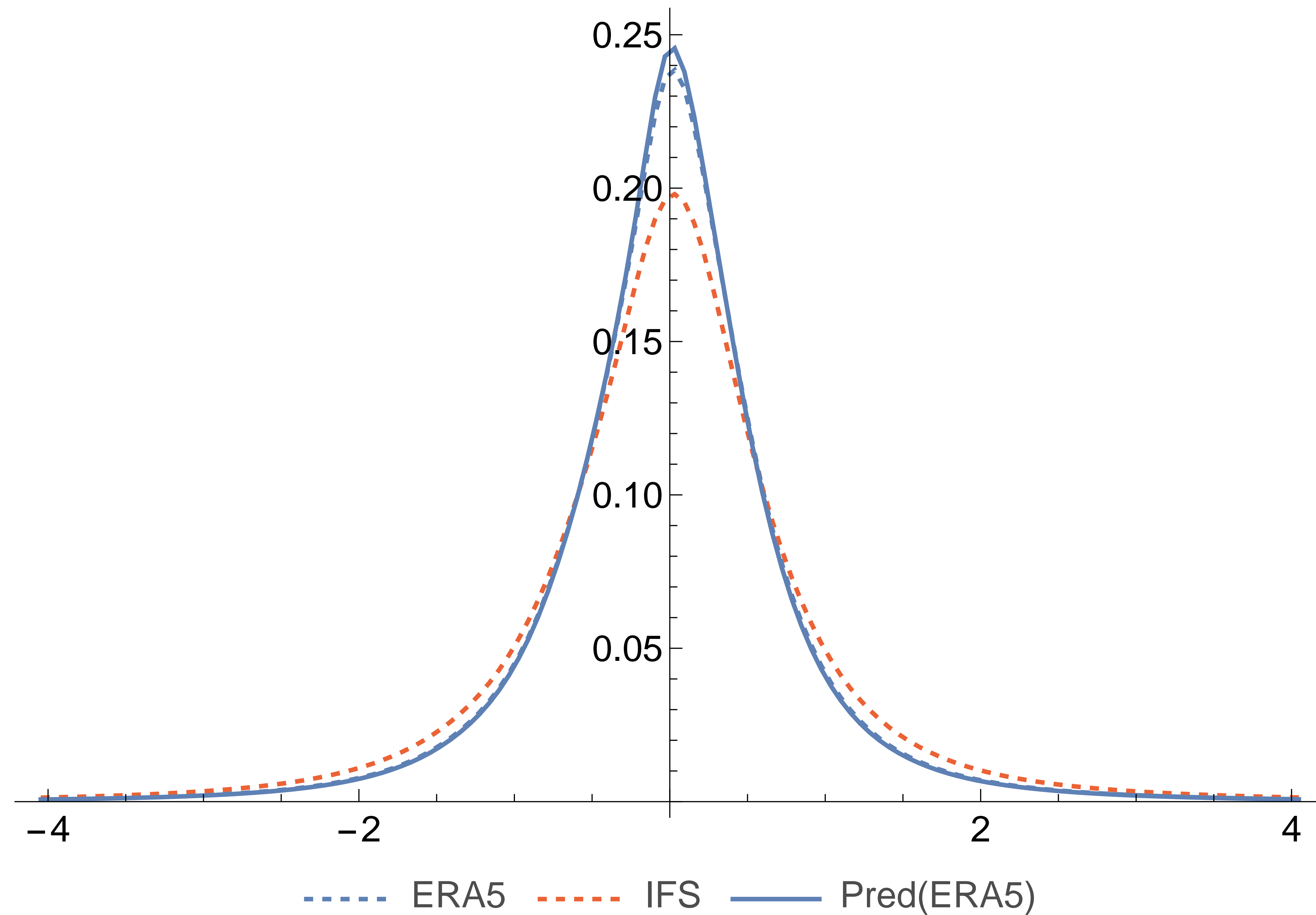
Model correction



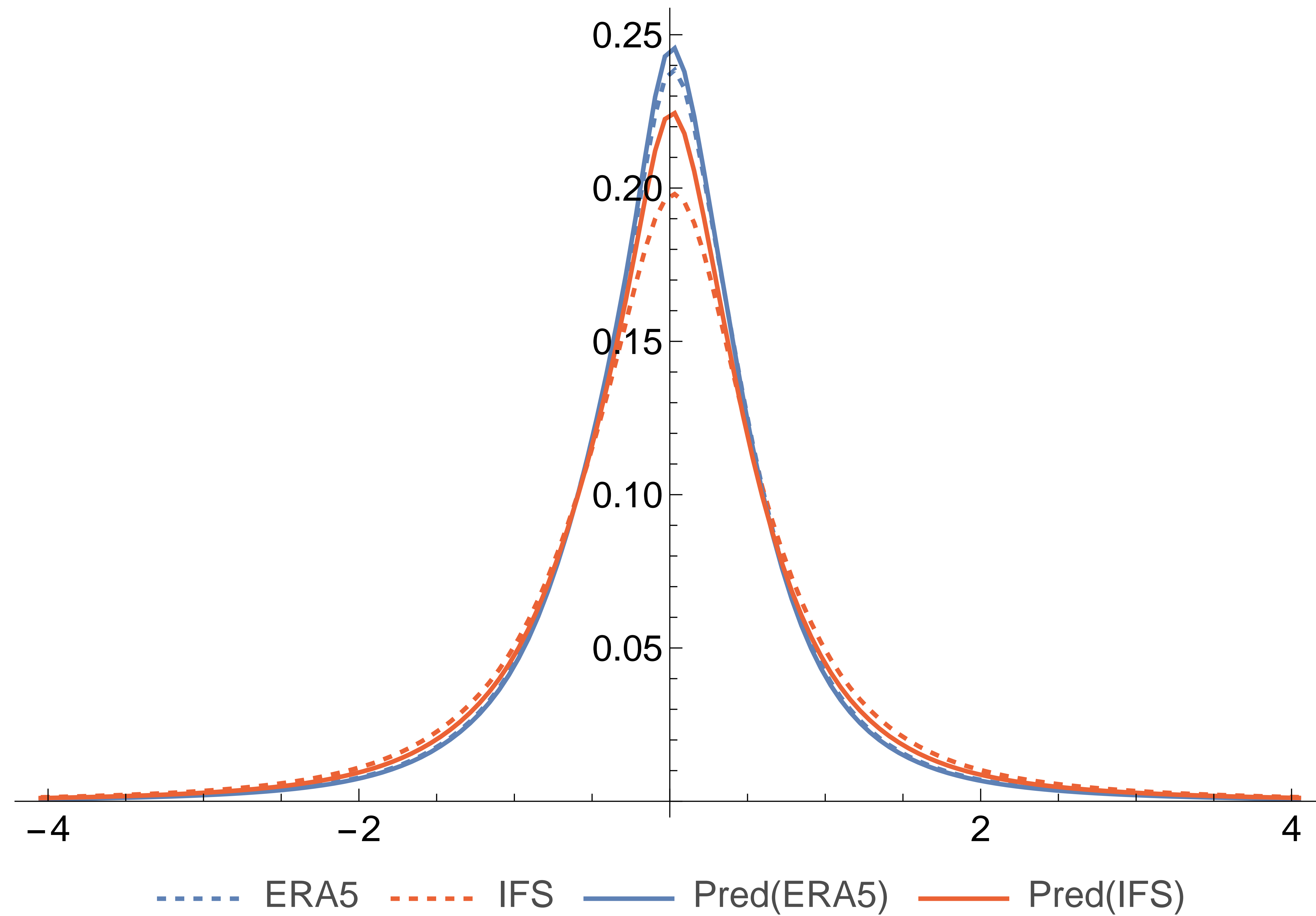
Model correction



Model correction



Model correction



Counterfactuals

- Numerical statistical atmospheric model:

$$p_{\theta}(y|x, \alpha)$$

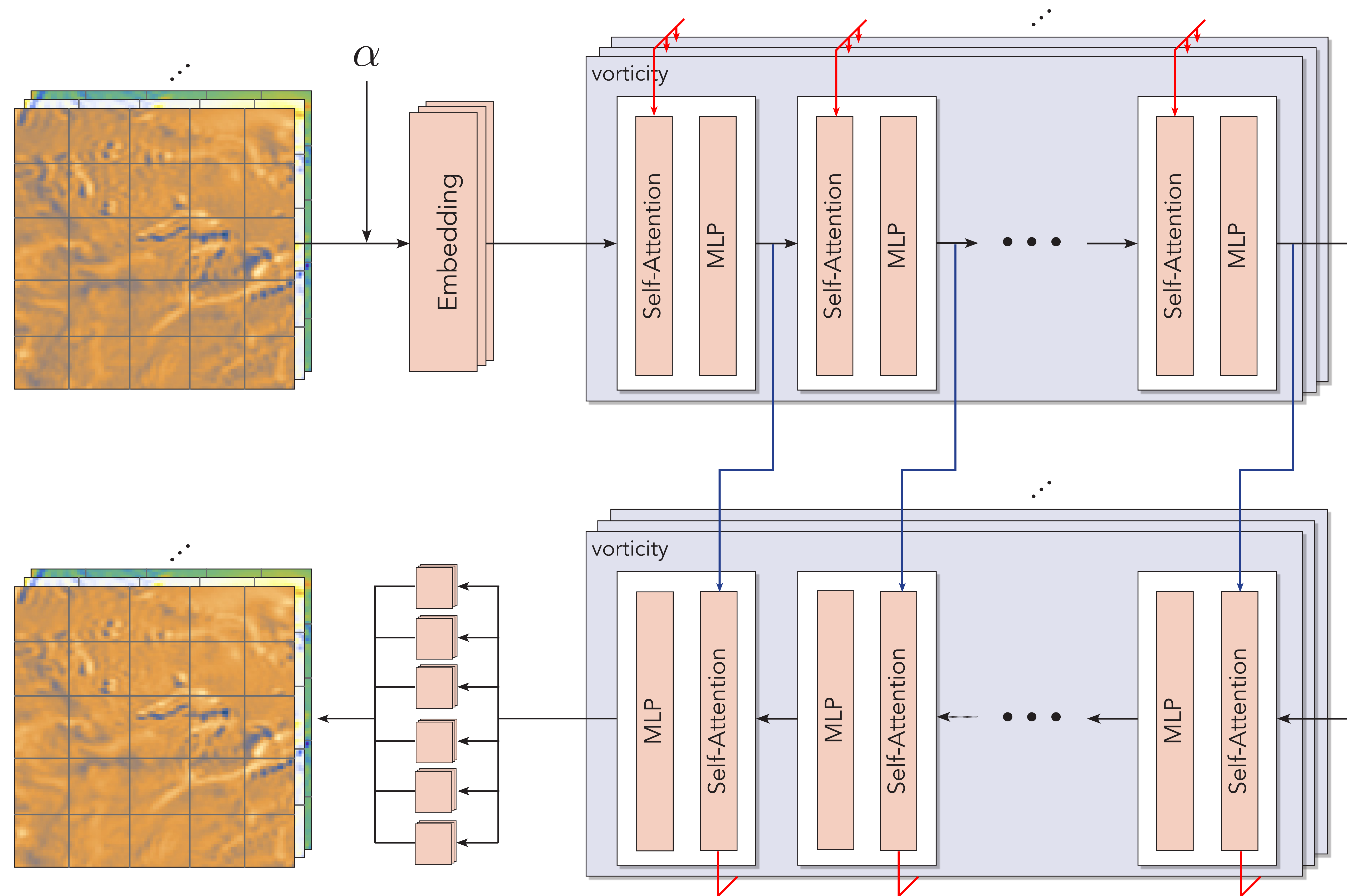
Counterfactuals

- Numerical statistical atmospheric model:

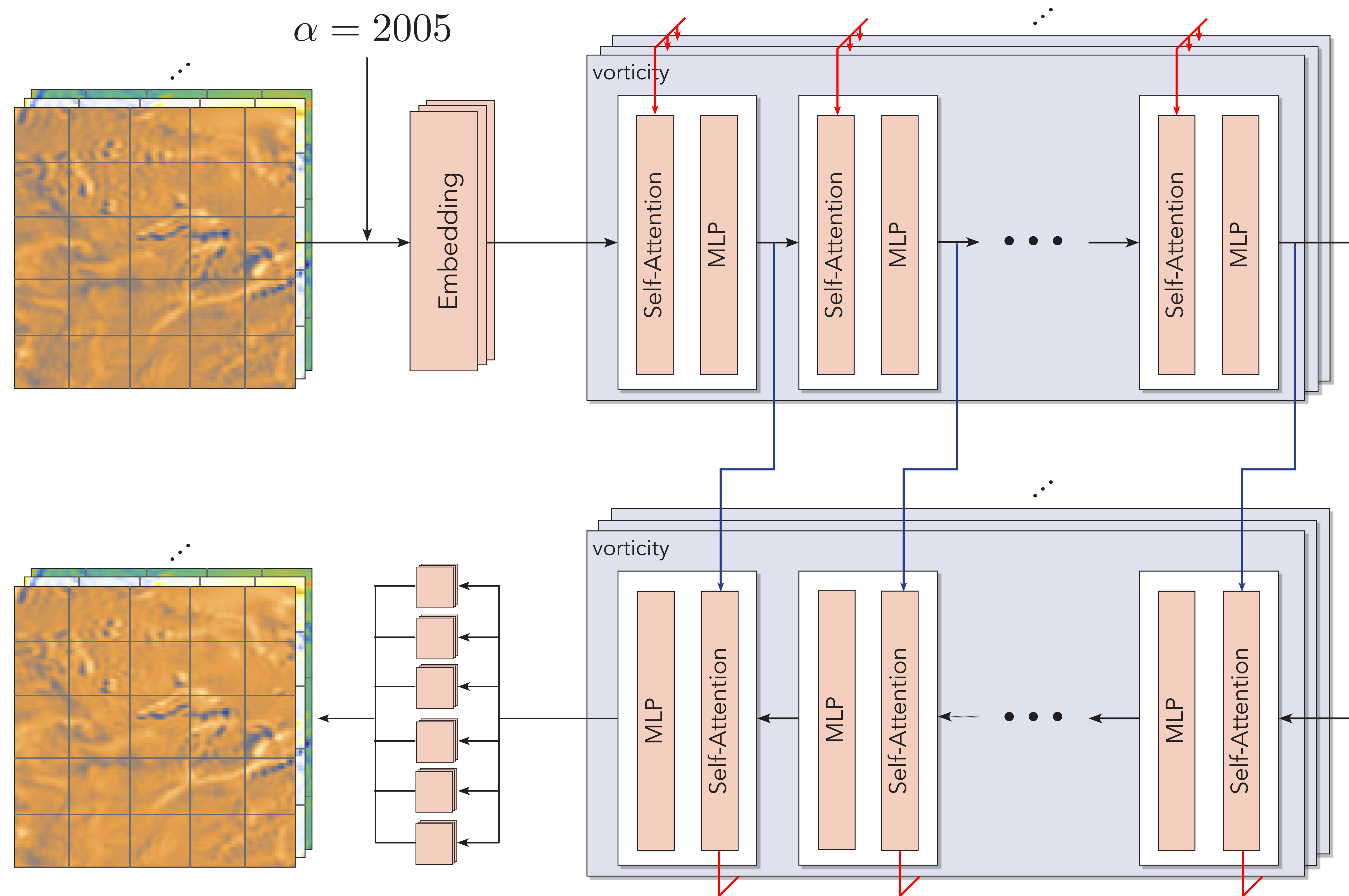
$$p_{\theta}(y|x, \alpha)$$

$$\alpha = (\text{year, day, hour, ml, } \theta, \phi, \text{res})$$

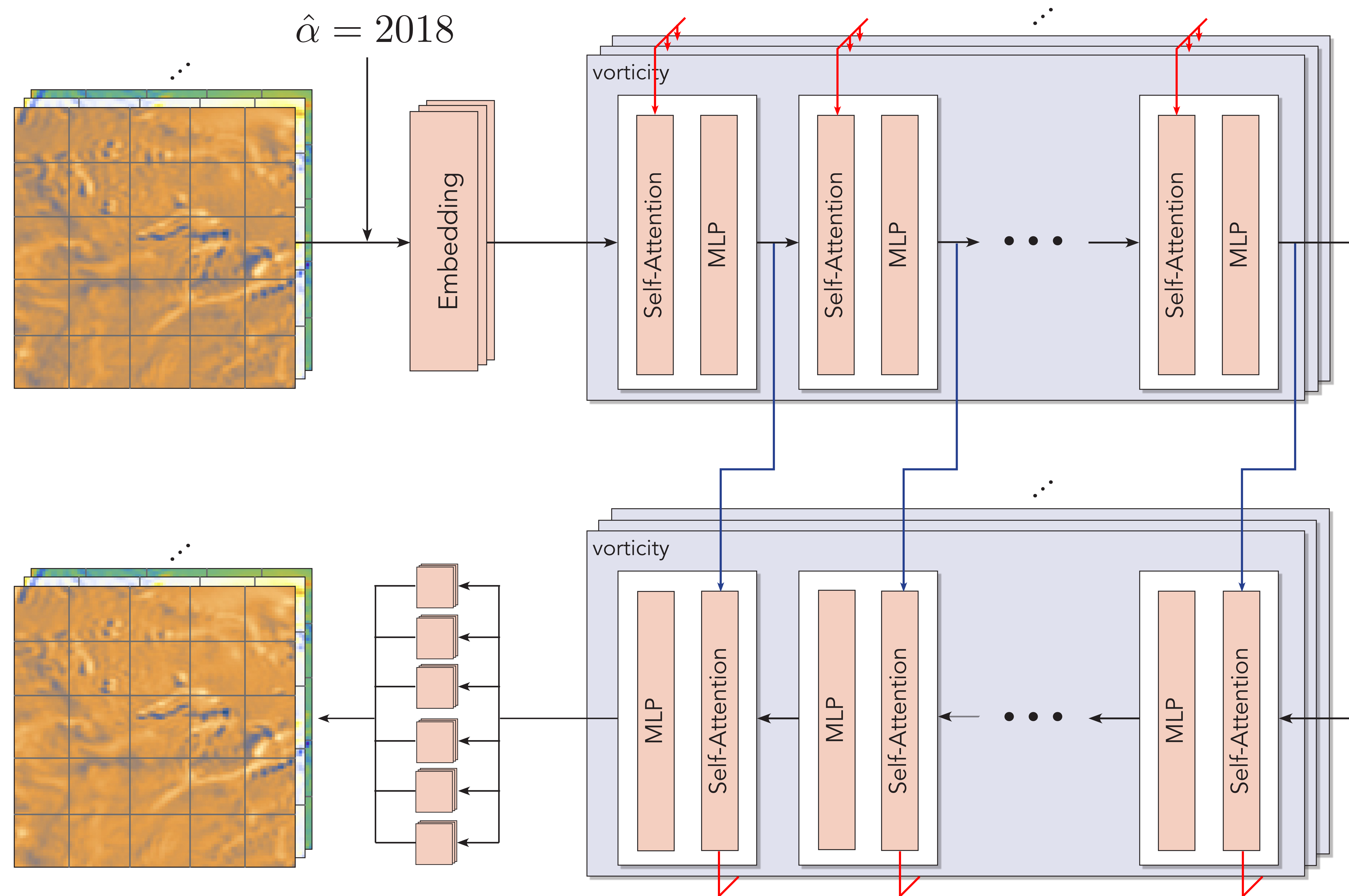
Counterfactuals



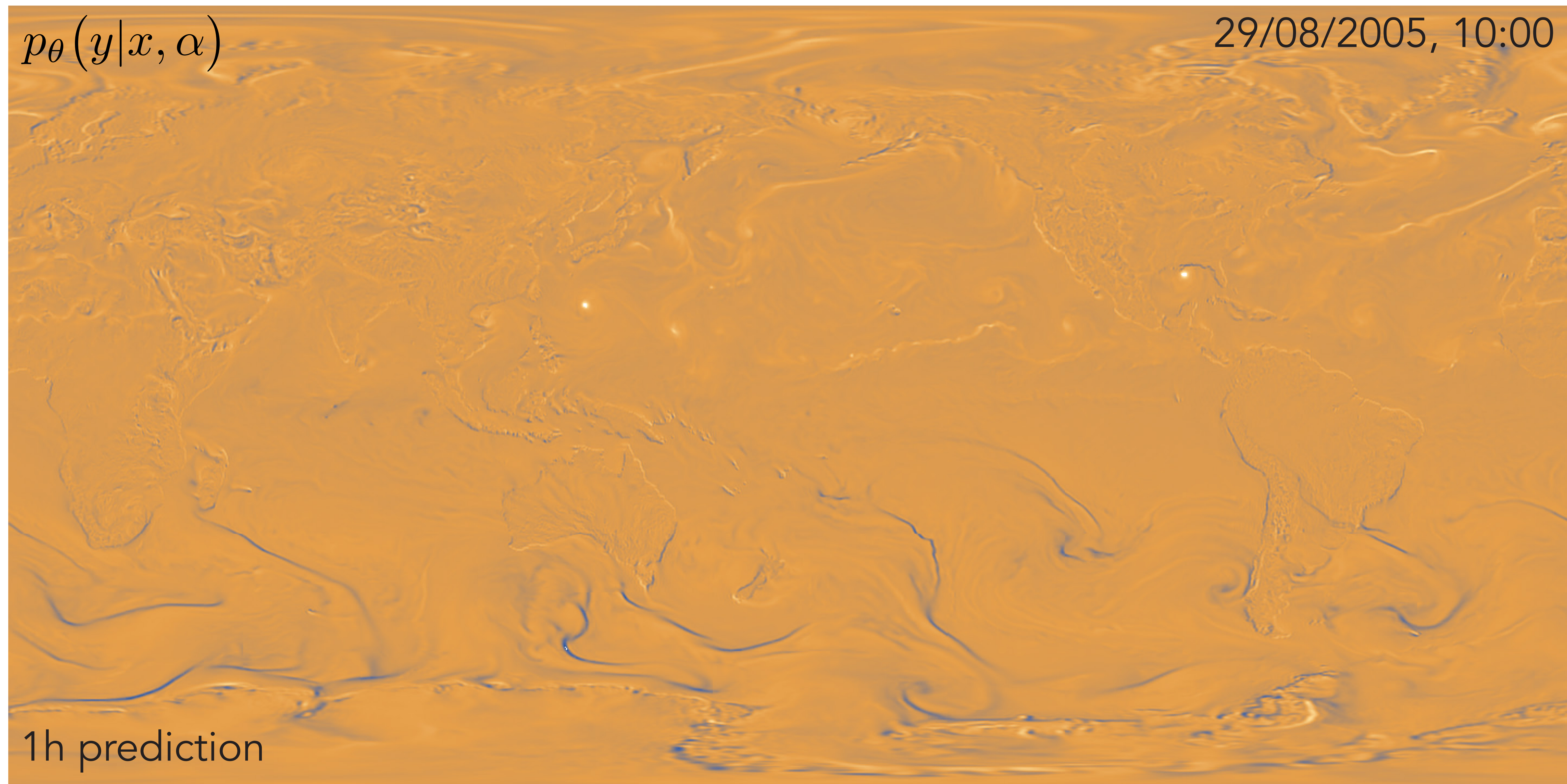
Counterfactuals



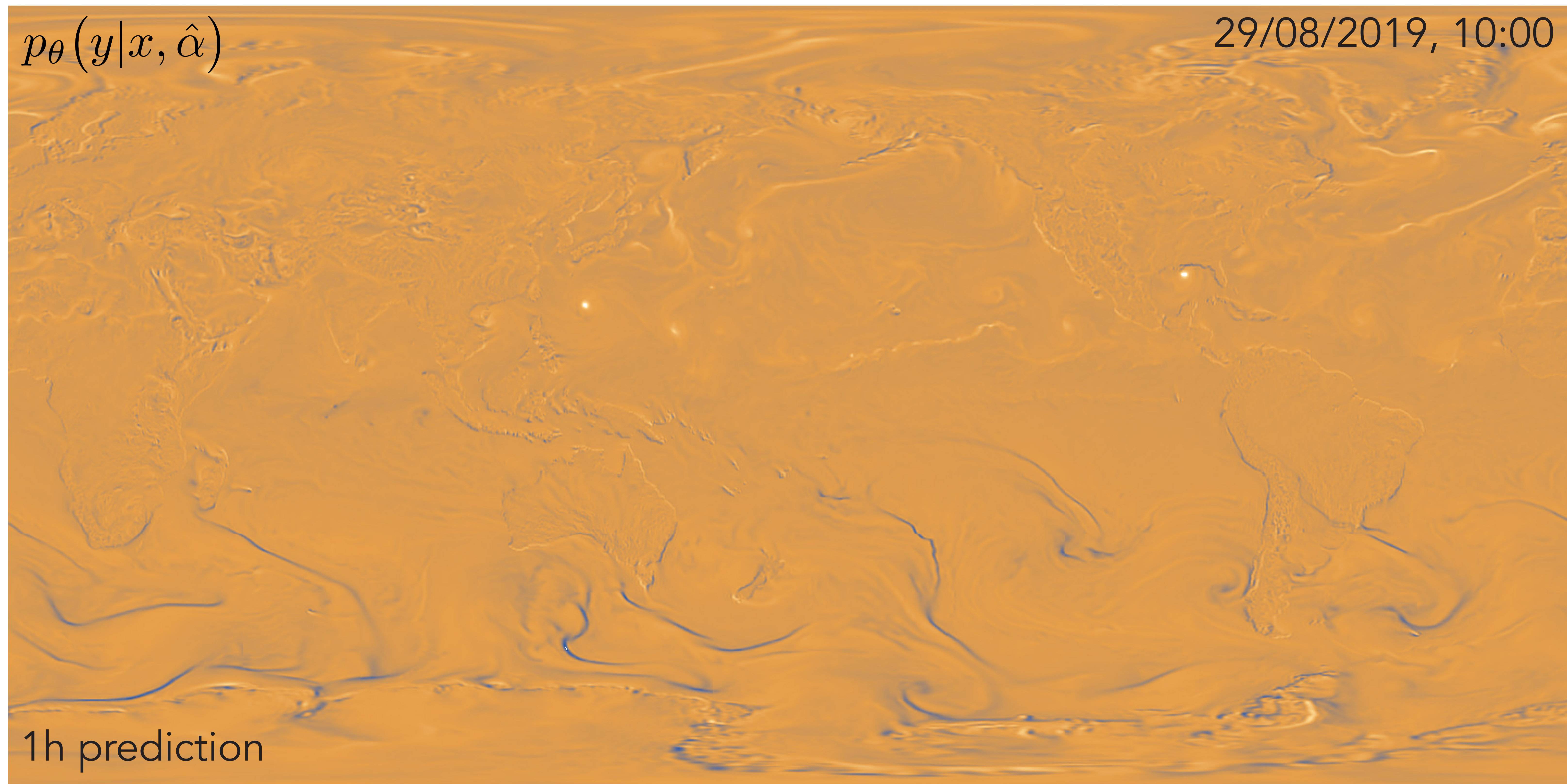
Counterfactuals



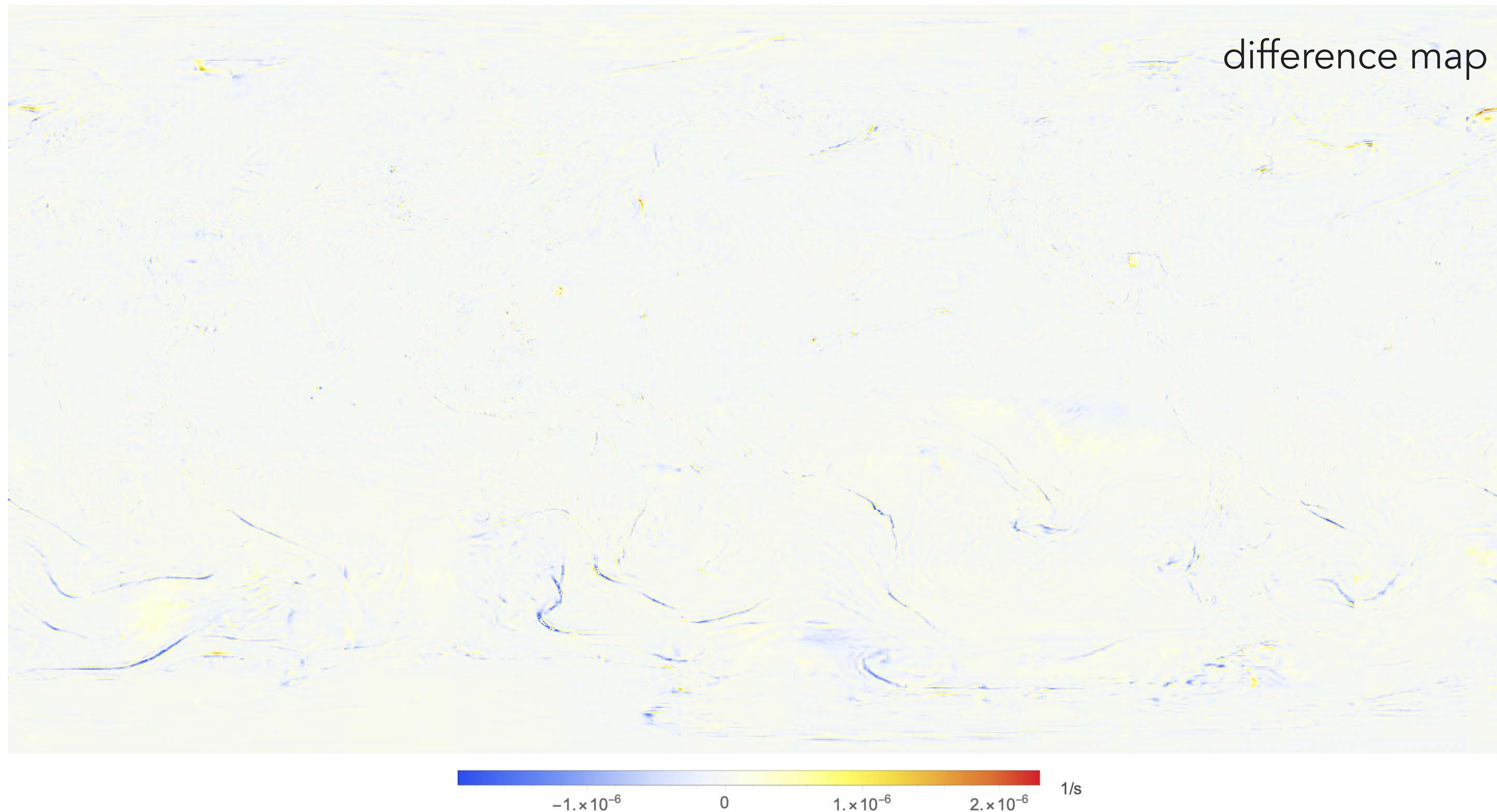
Counterfactuals



Counterfactuals



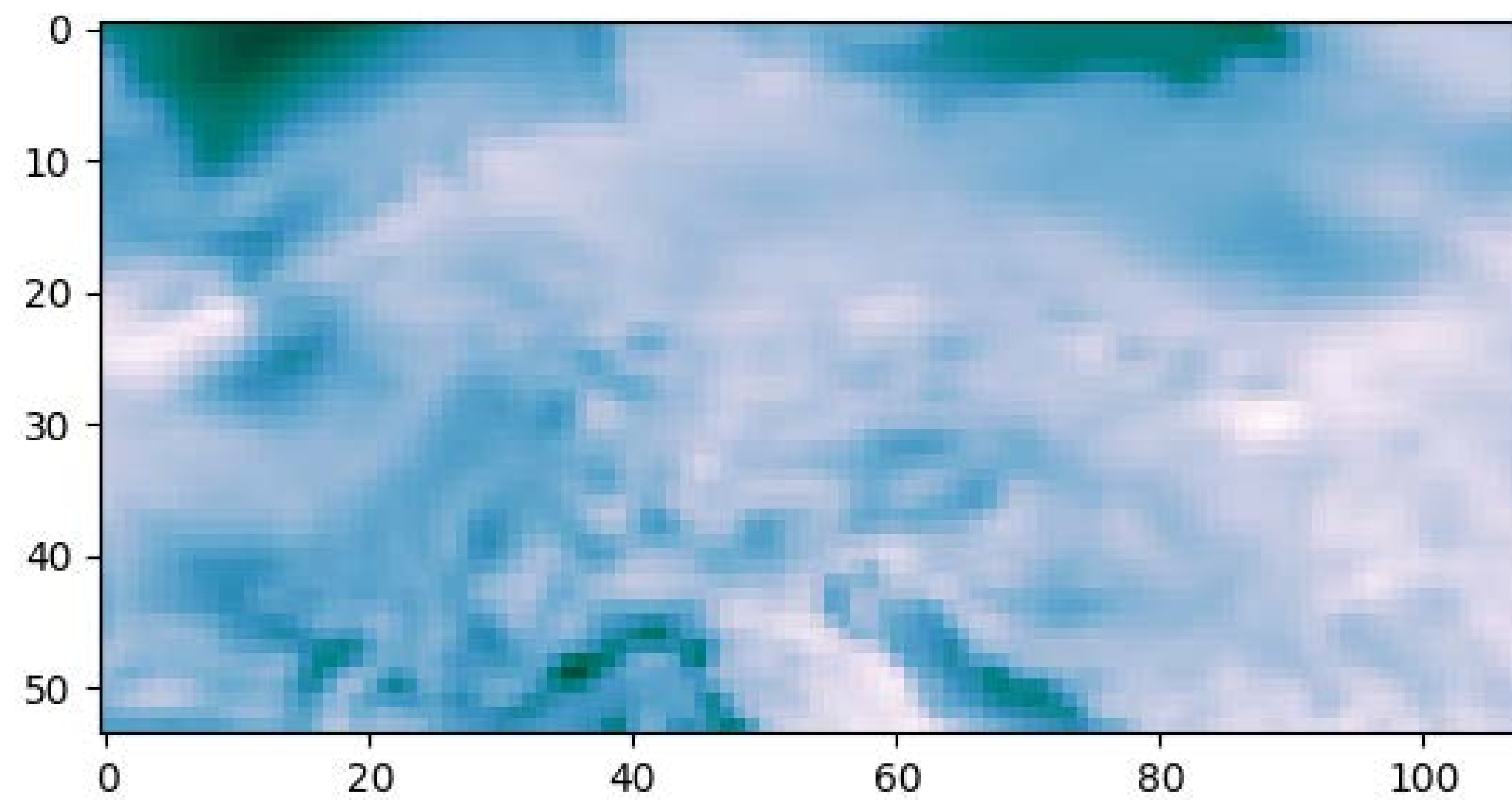
Counterfactuals



Downstream applications

Downscaling

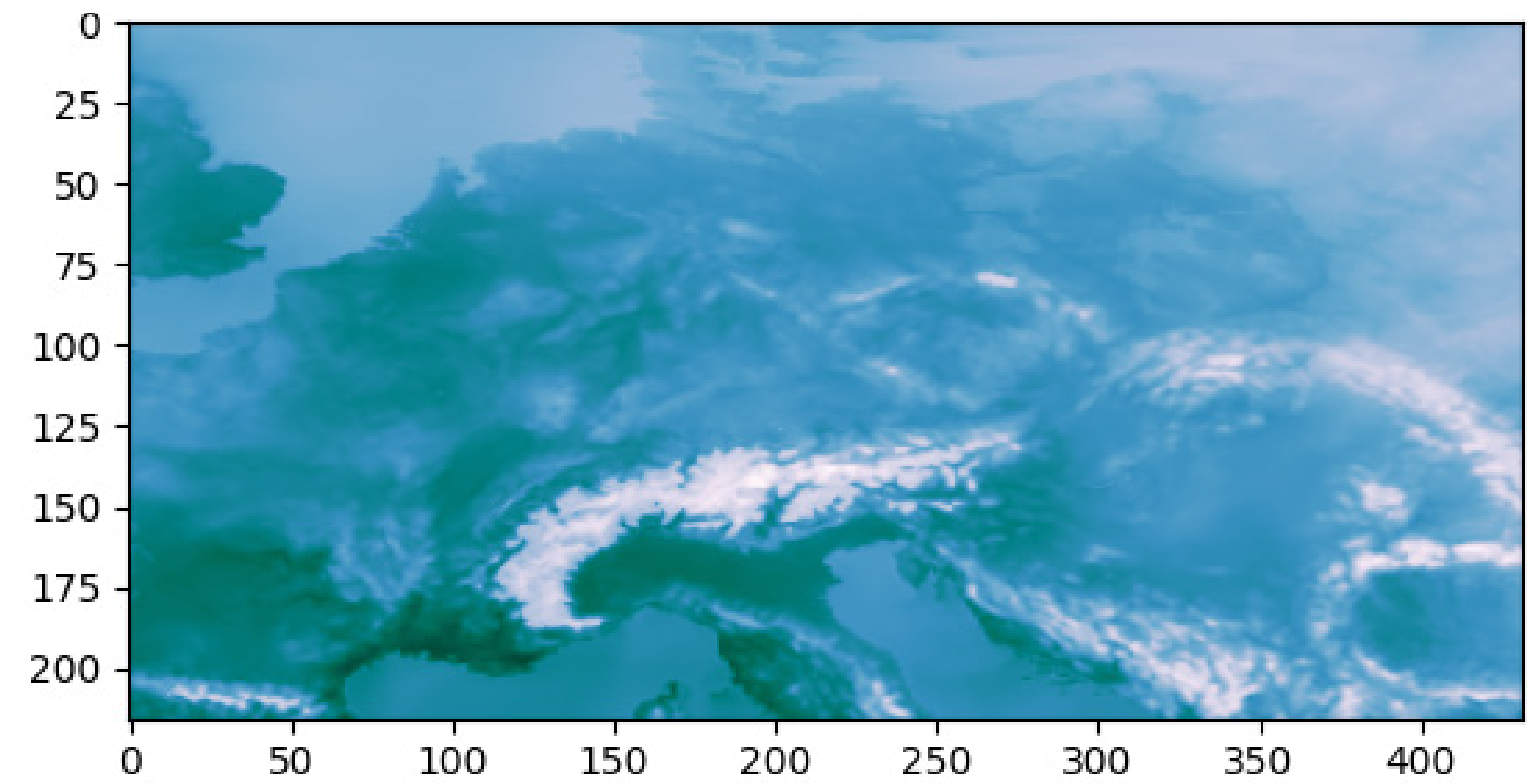
ERA5



temperature, ml=137

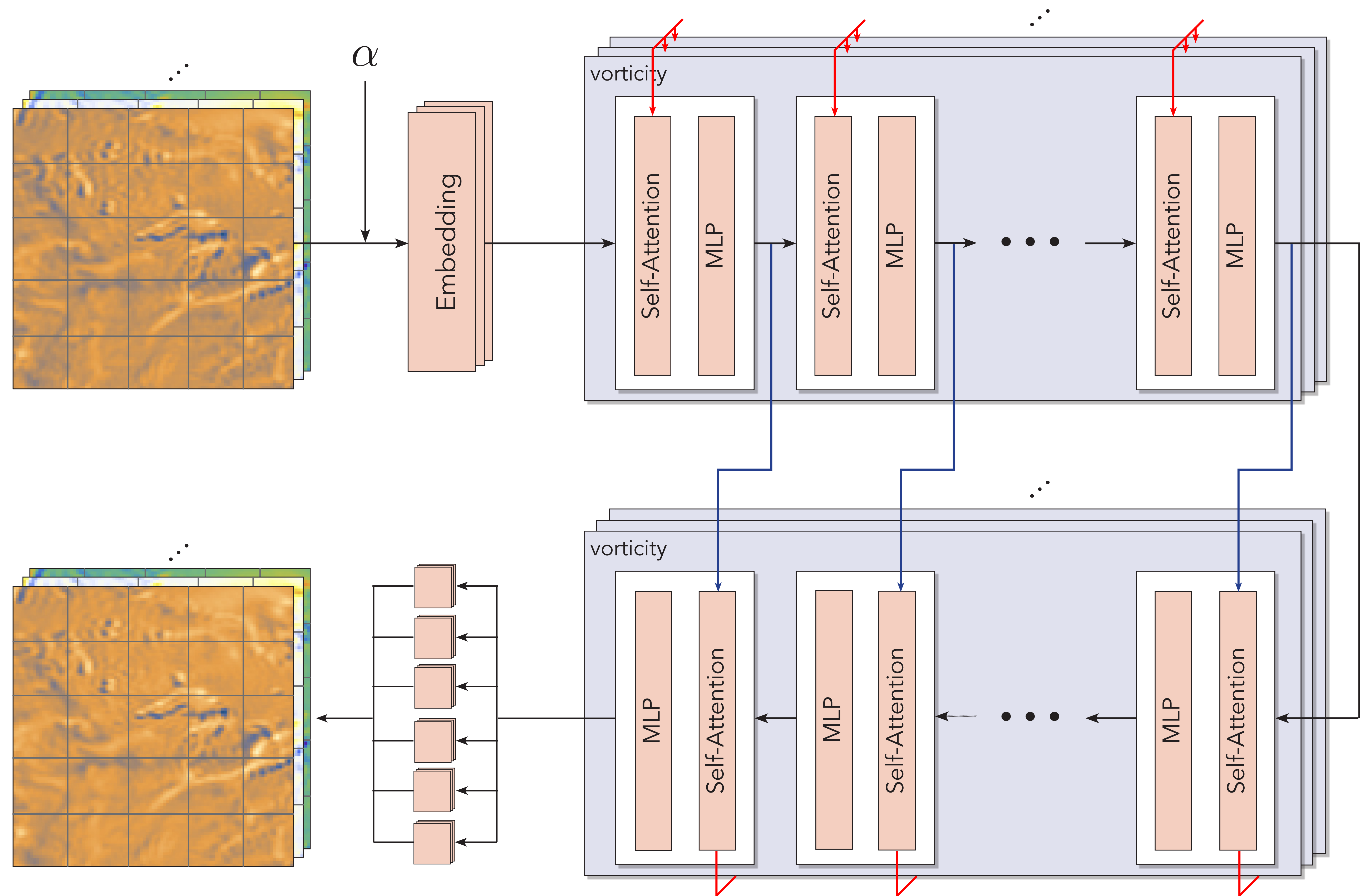


COSMO-REA6

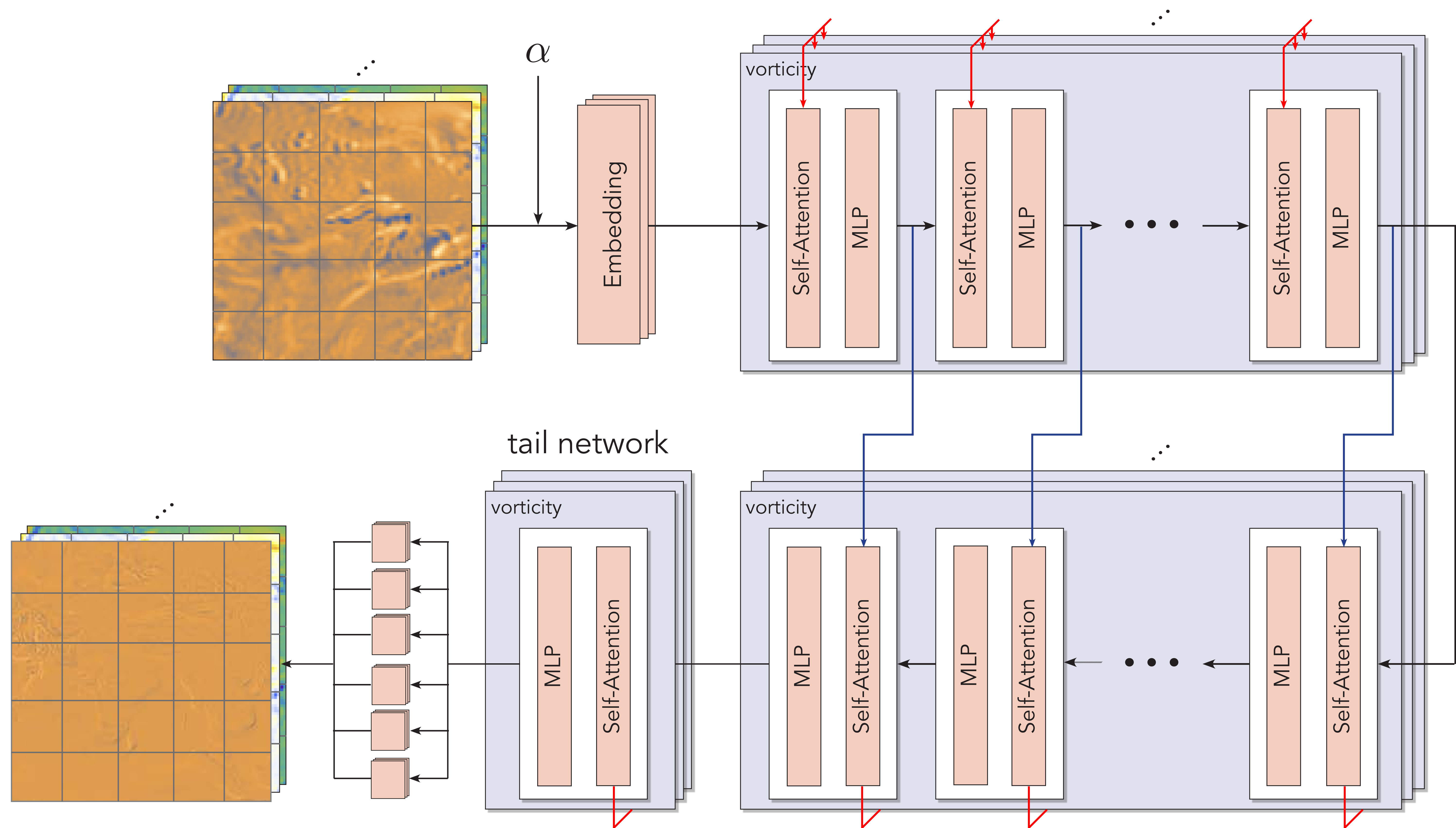


temperature, 2 m

Downscaling



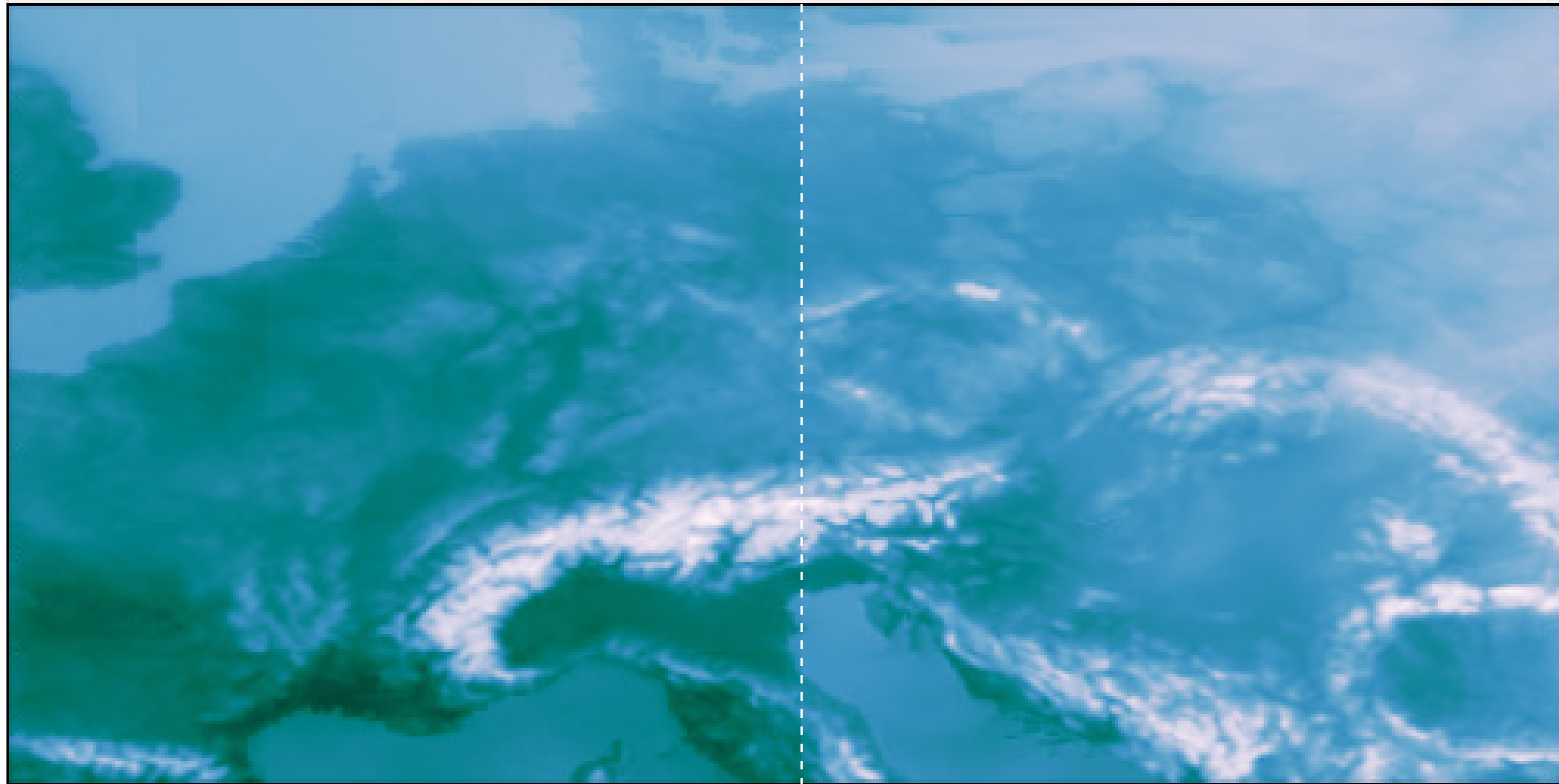
Downscaling



Downscaling

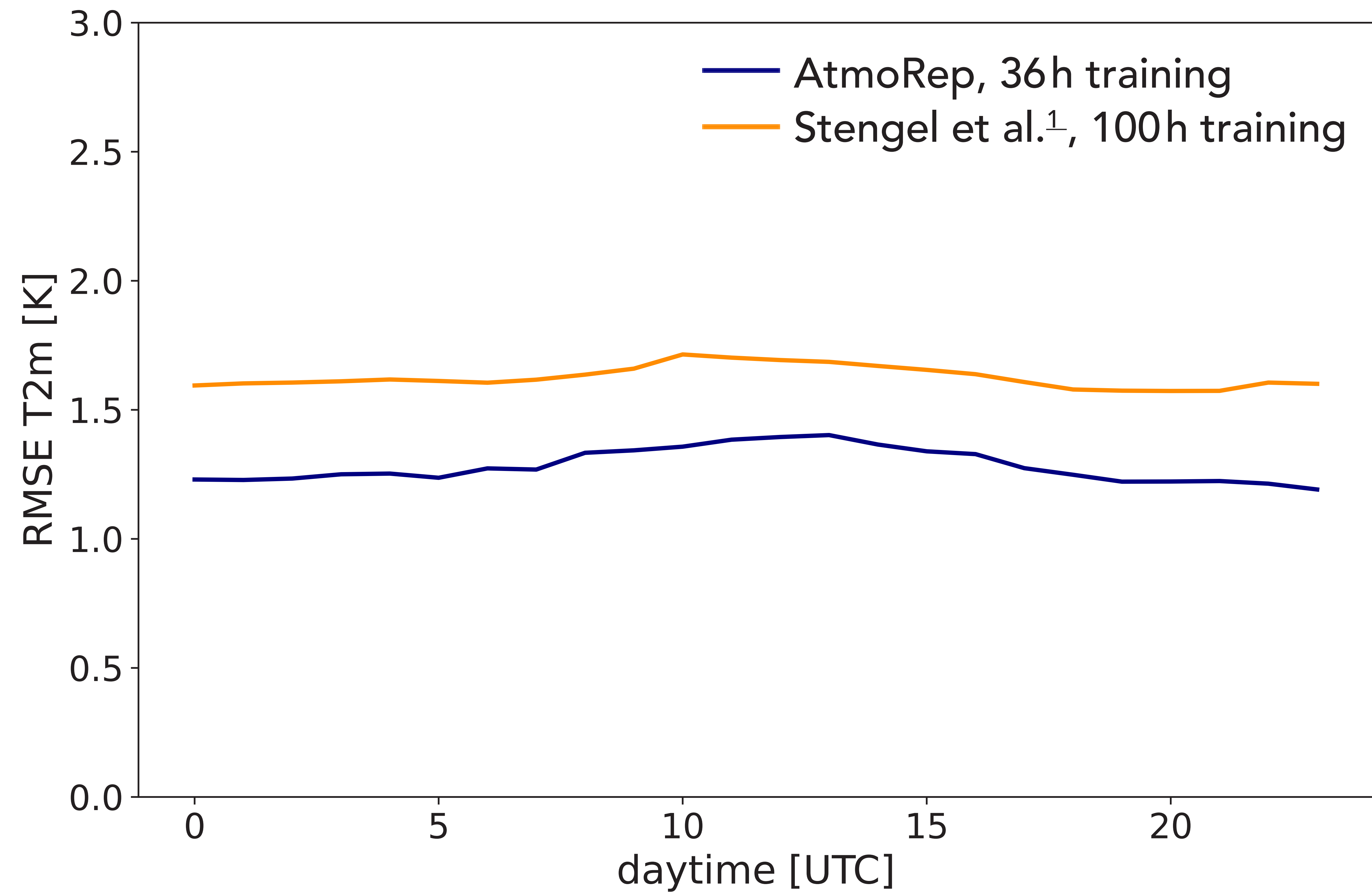
AtmoRep prediction

COSMO-REA6



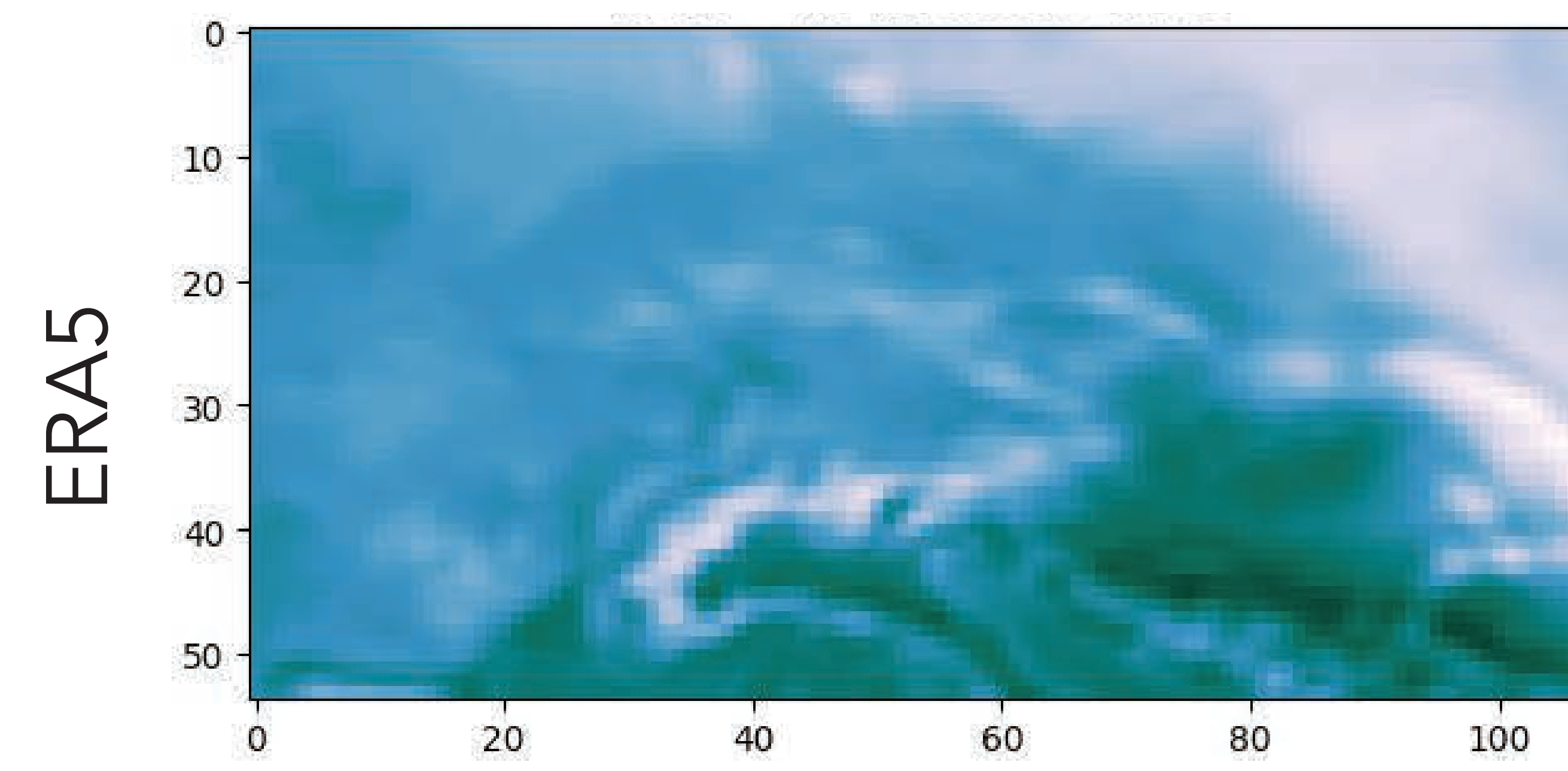
temperature, 2 m

Downscaling

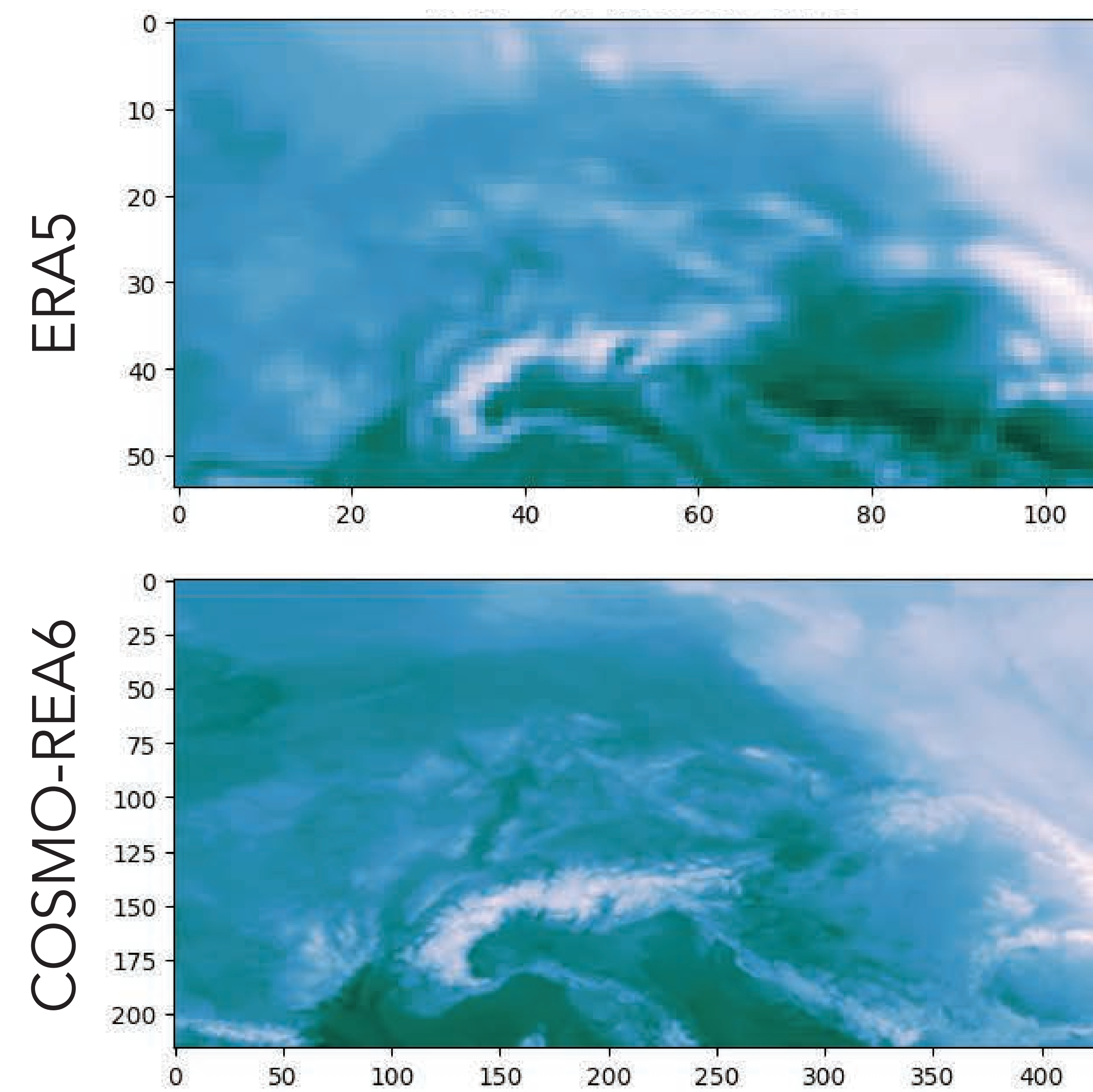


¹ K. Stengel, A. Glaws, D. Hettinger, and R. N. King. Adversarial super-resolution of climatological wind and solar data. Proceedings of the National Academy of Sciences, 117(29):16805–16815, 2020.

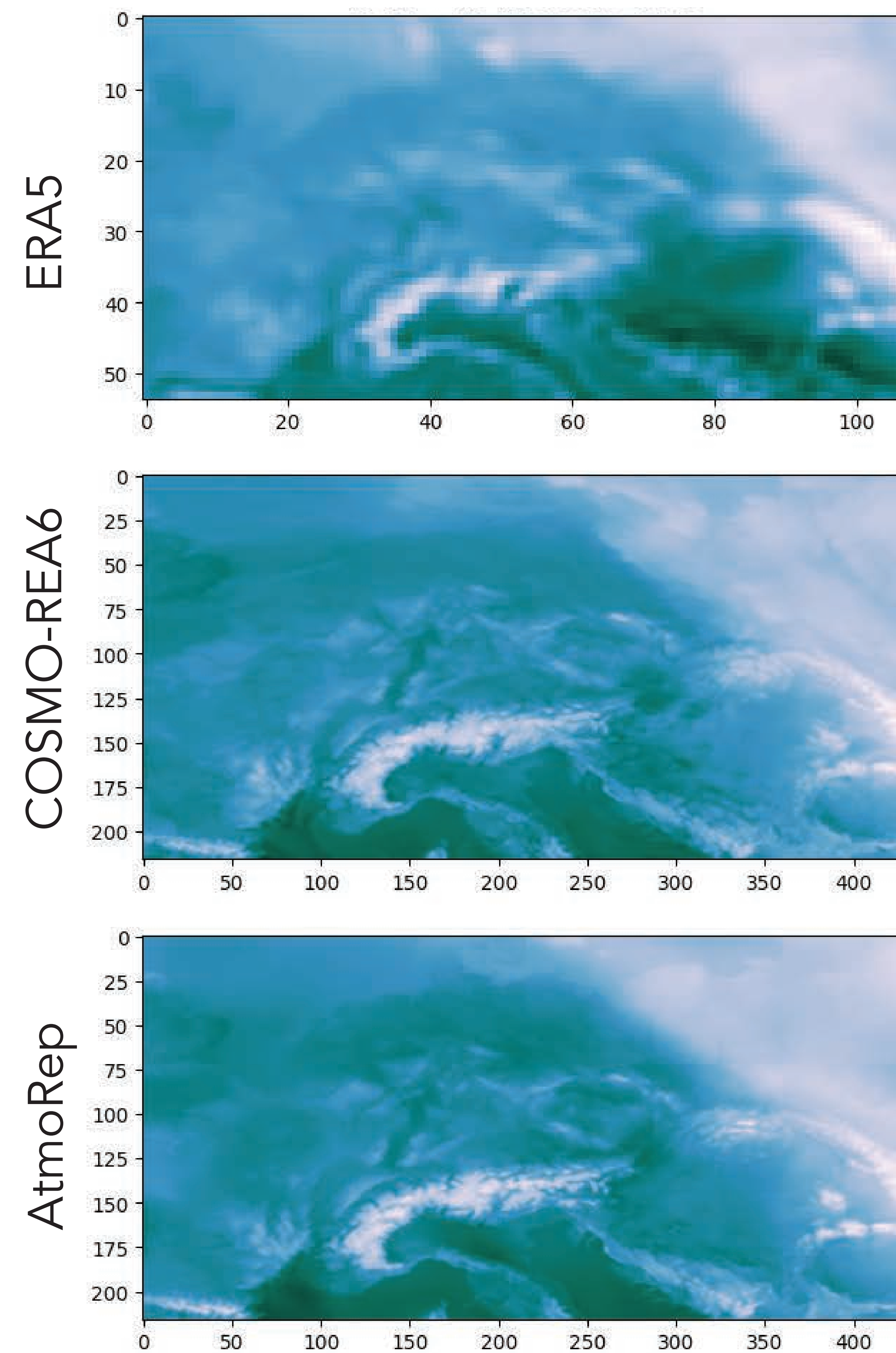
Downscaling



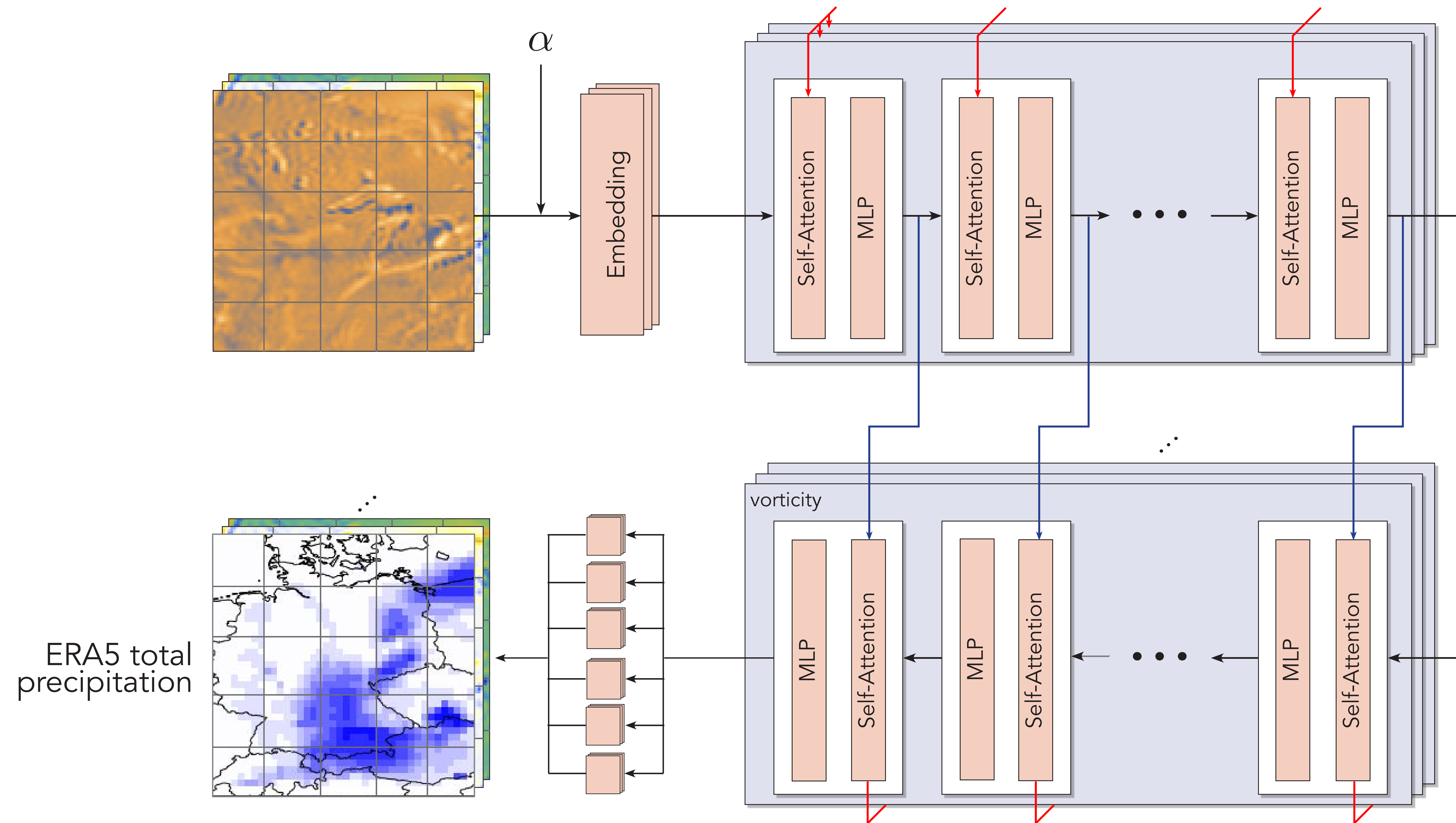
Downscaling



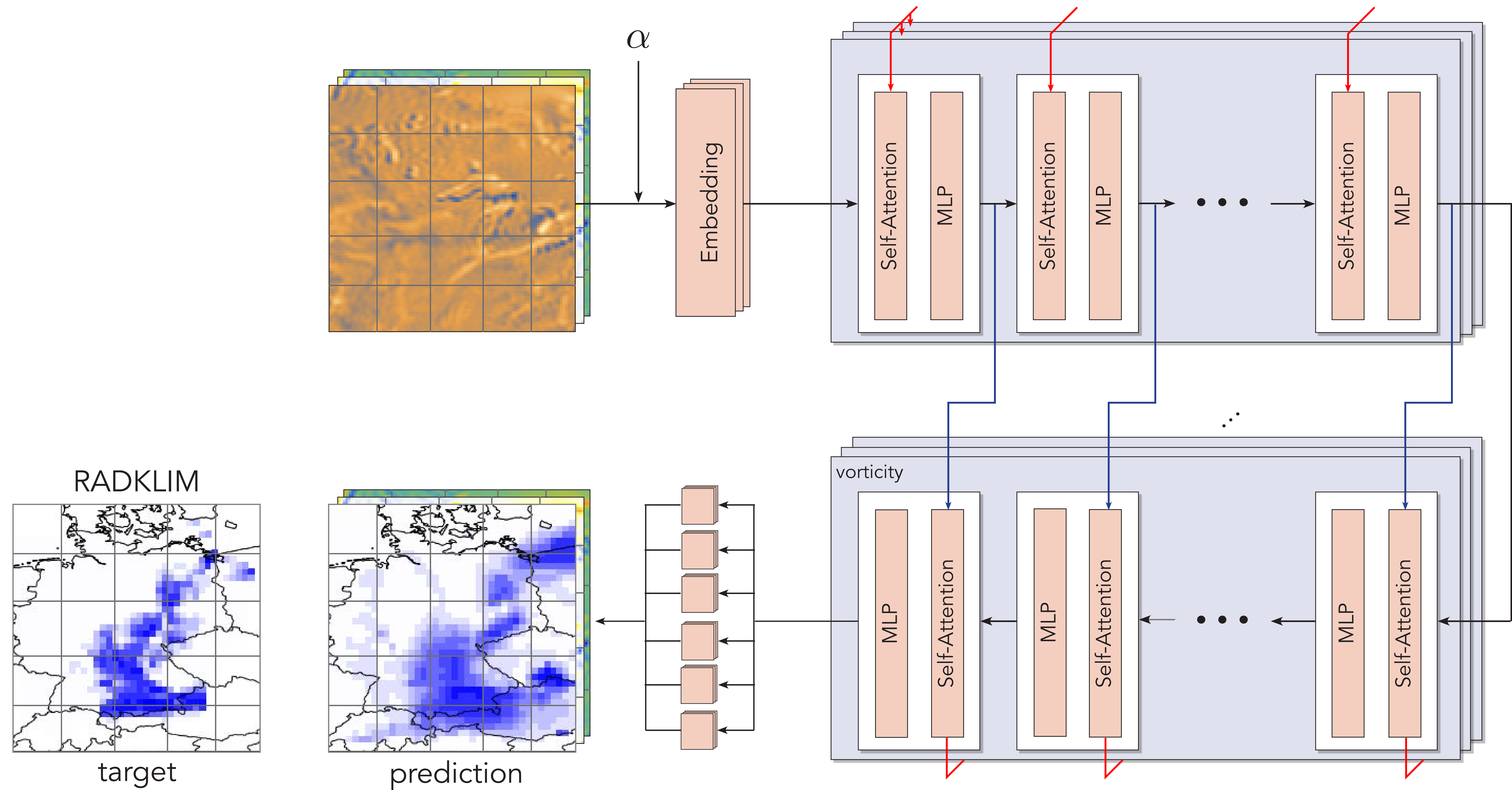
Downscaling



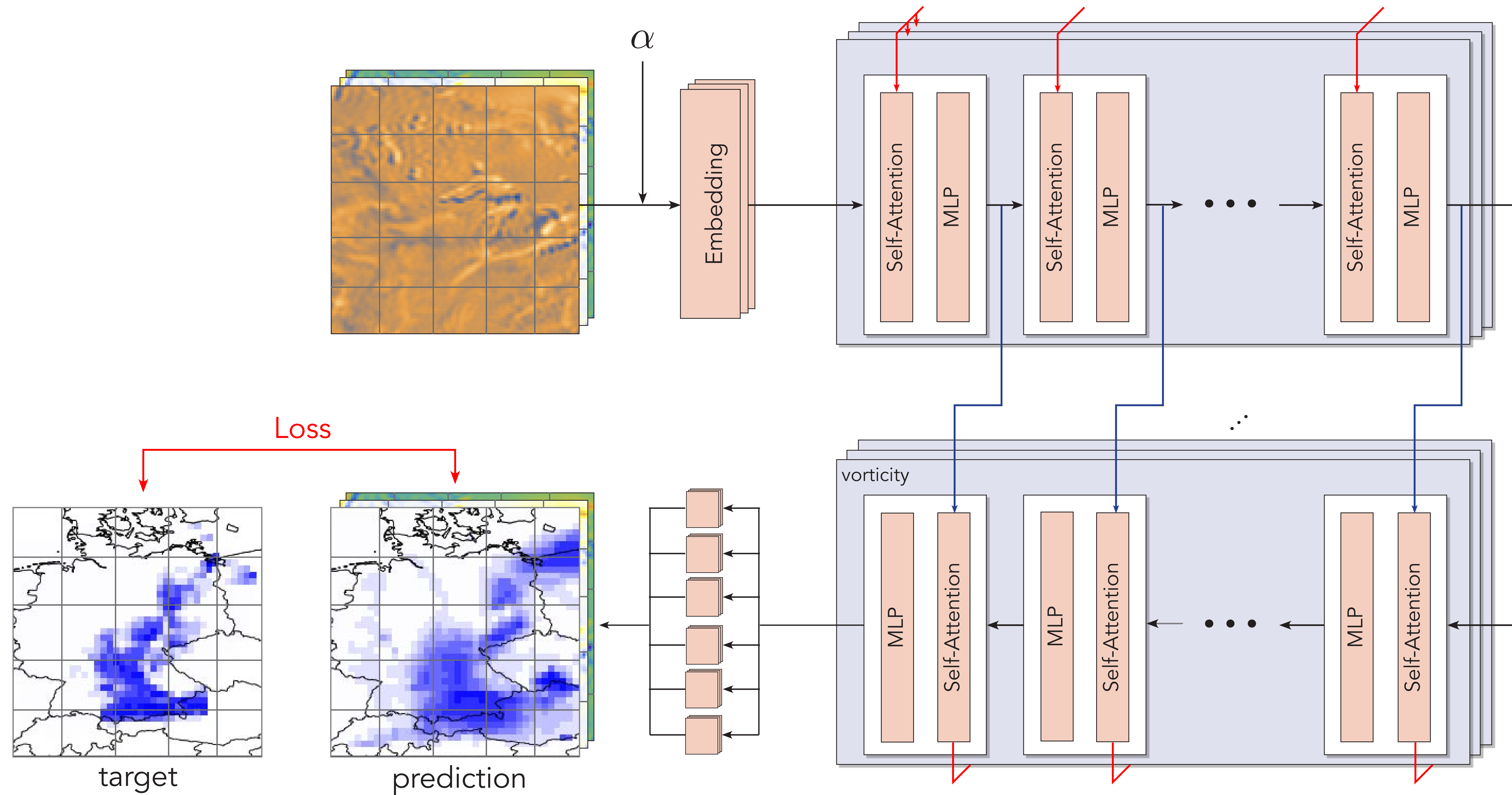
Bias correction



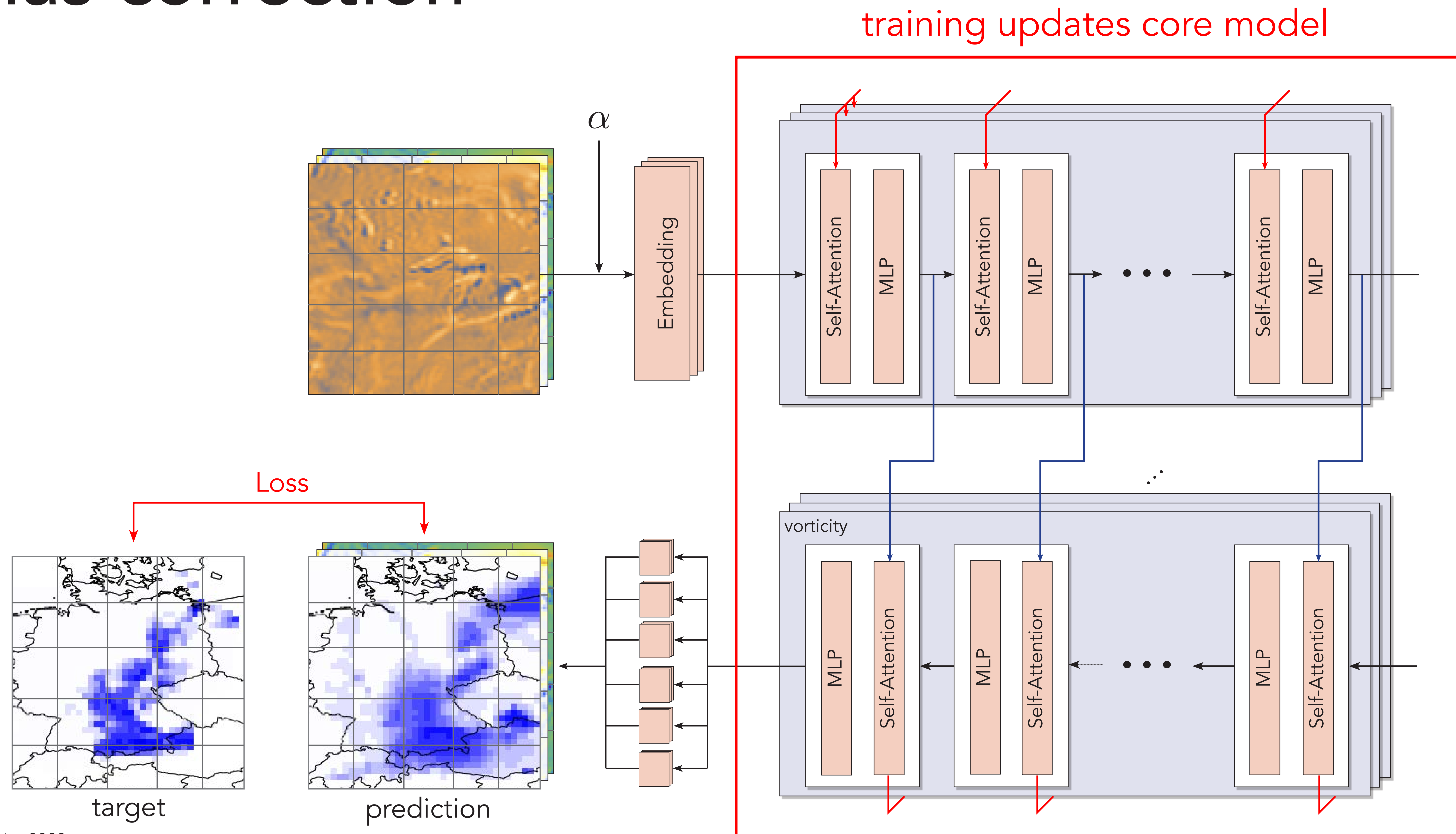
Bias correction



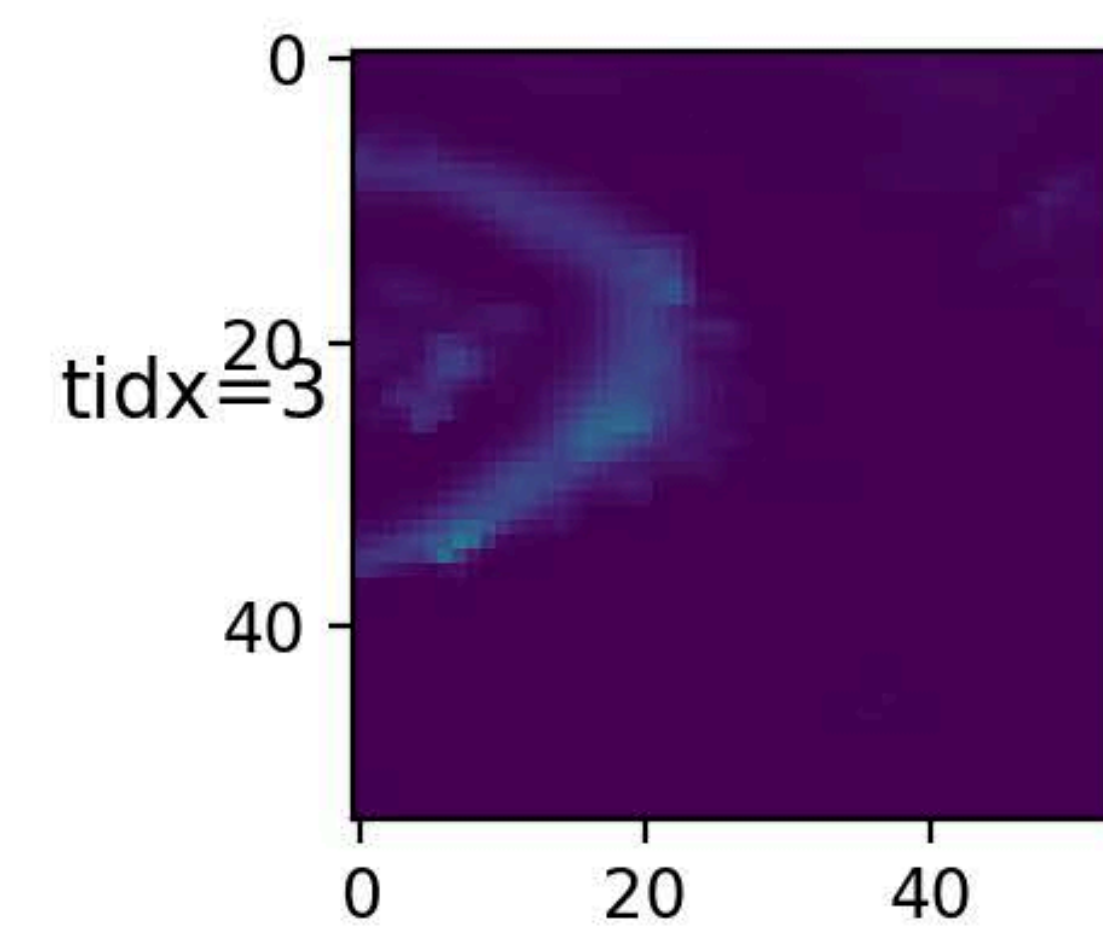
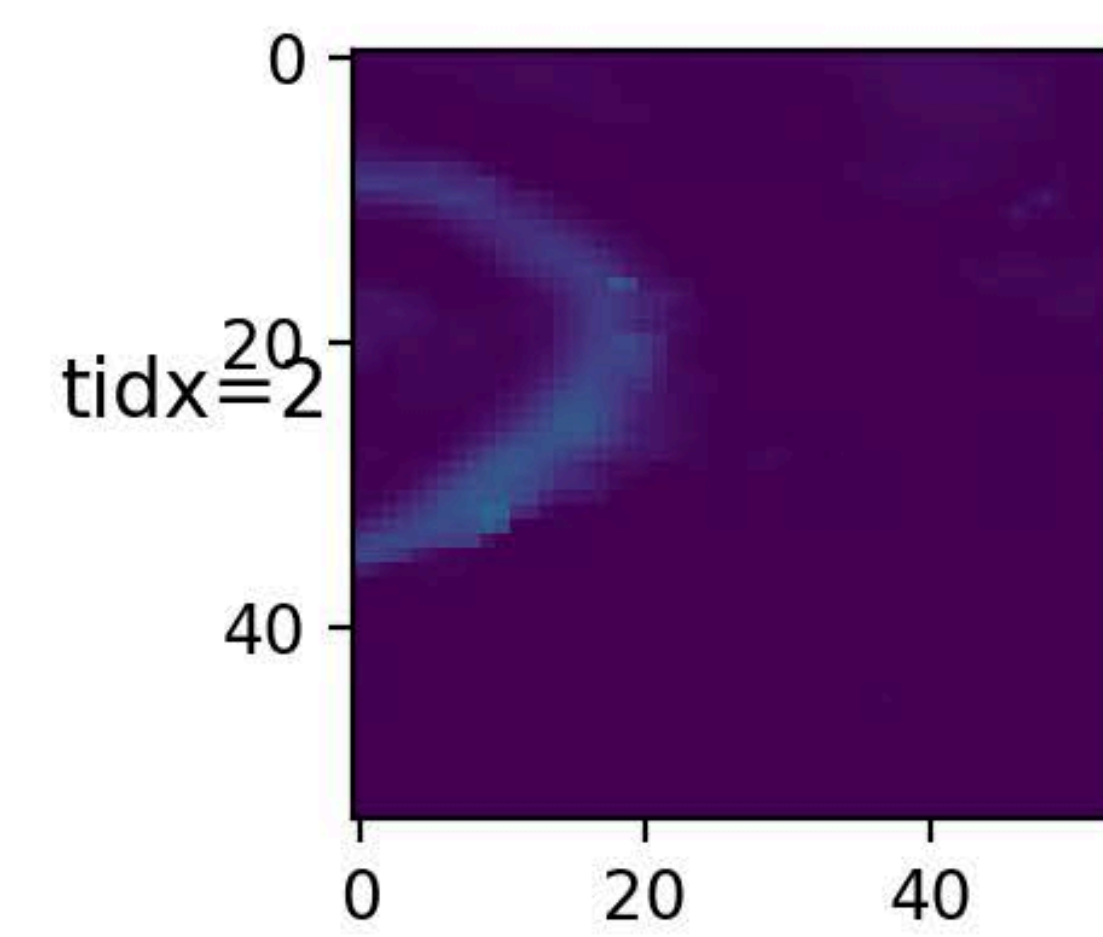
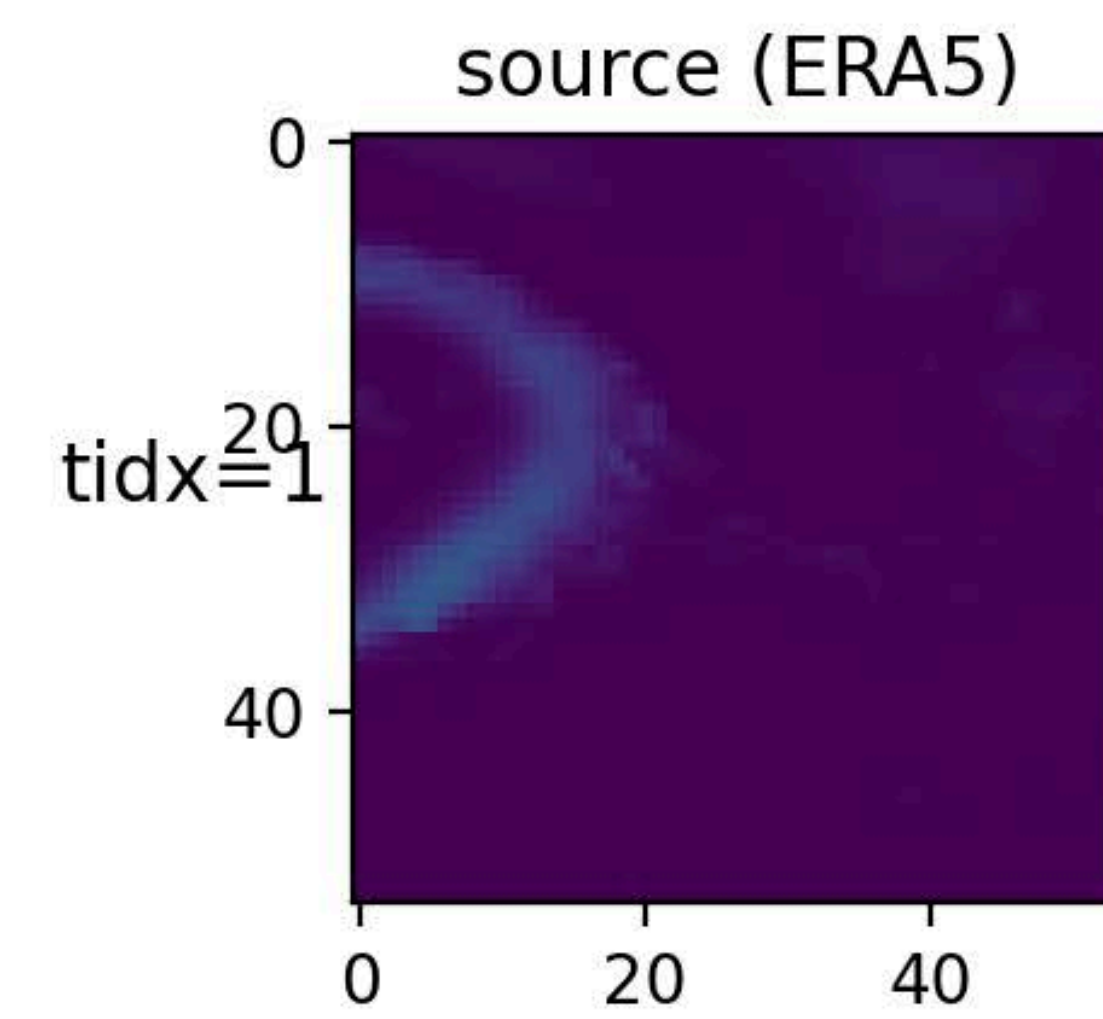
Bias correction



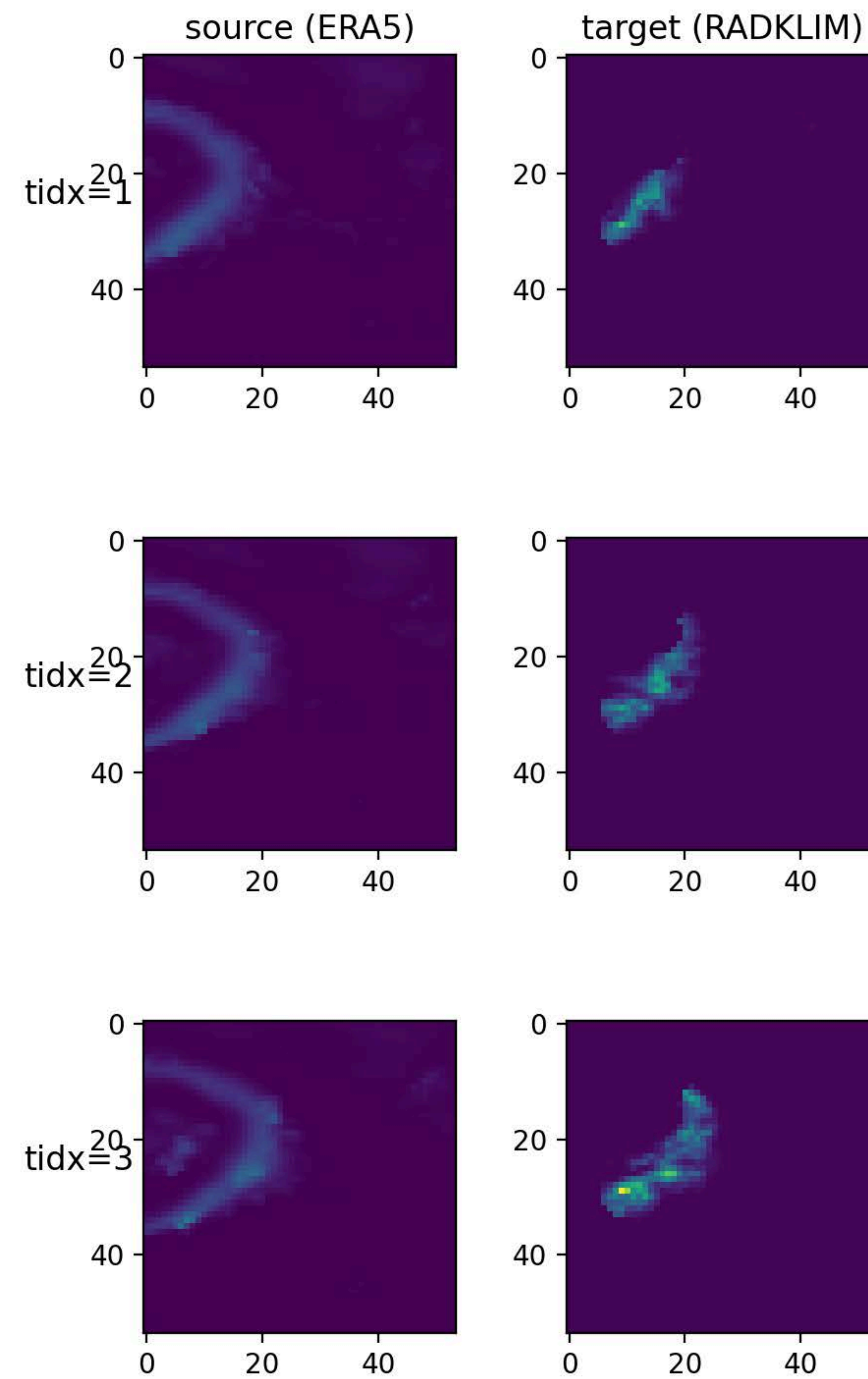
Bias correction



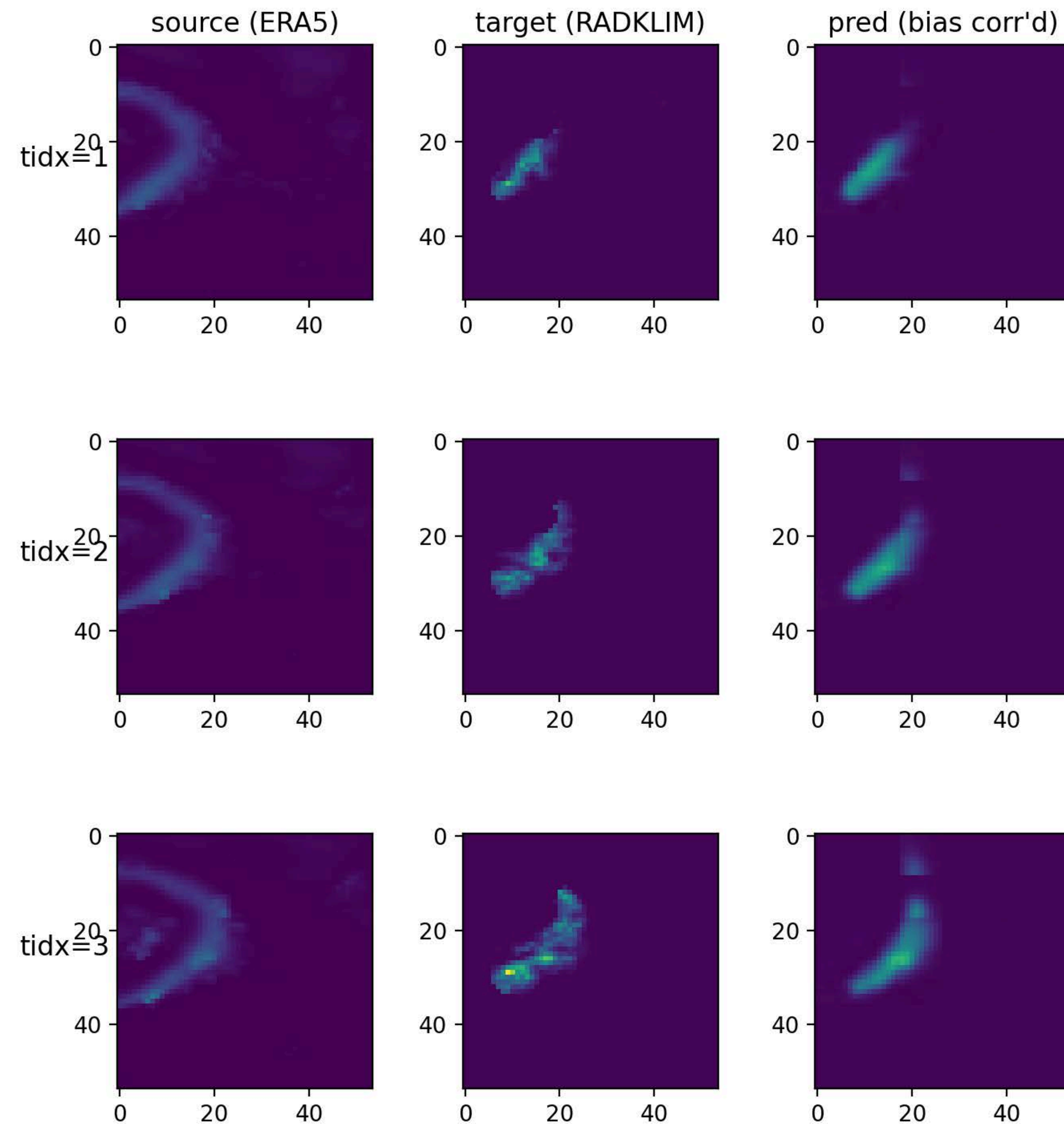
Bias correction



Bias correction



Bias correction



Outlook

Outlook

- Train and integrate many different data sources
 - › CERRA, NexRAD, ...
 - › IFS, ICON, ...
 - › Microwave sounders, radiosondes, SYNOP, ...

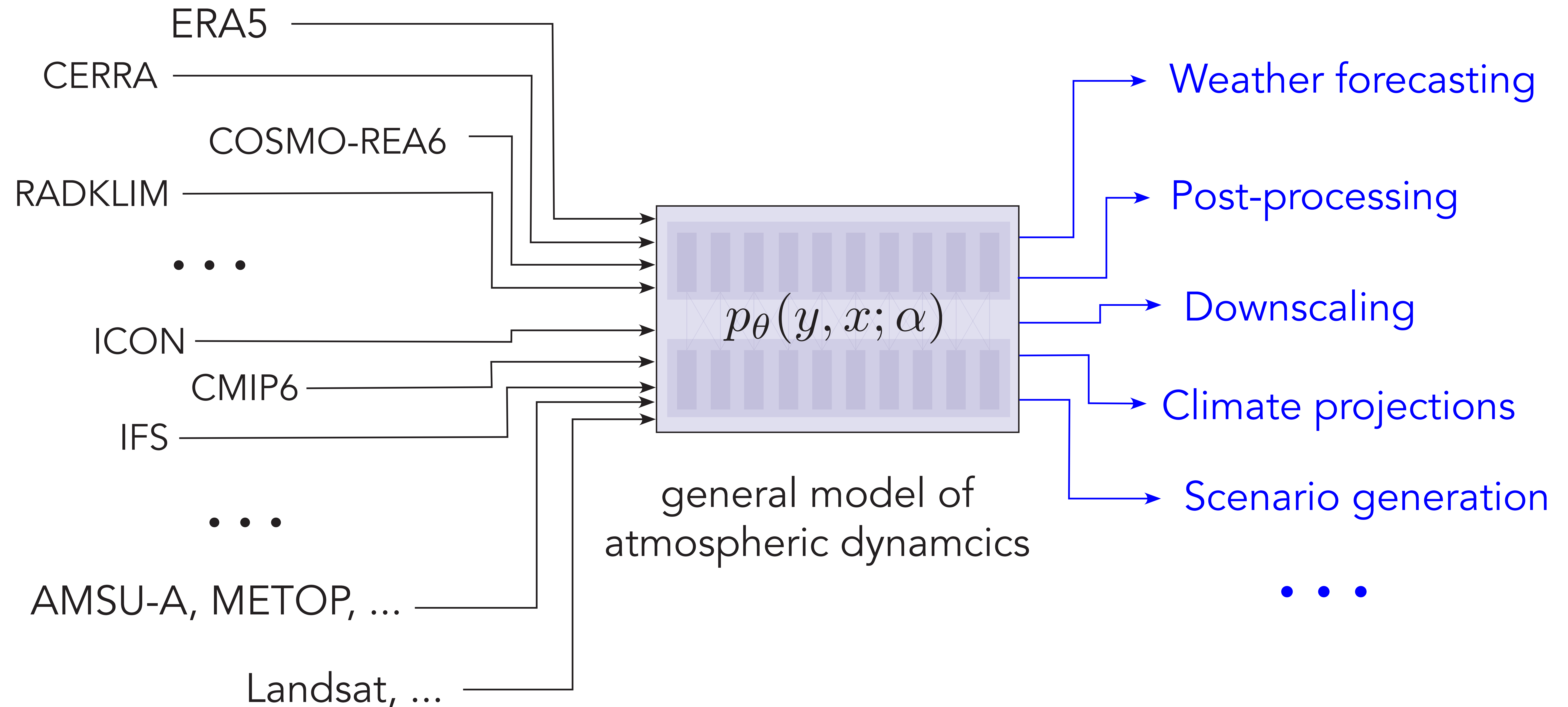
Outlook

- Train and integrate many different data sources
 - › CERRA, NexRAD, ...
 - › IFS, ICON, ...
 - › Microwave sounders, radiosondes, SYNOP, ...
- Consider more applications
 - › Medium-range weather forecasting

Outlook

- Train and integrate many different data sources
 - › CERRA, NexRAD, ...
 - › IFS, ICON, ...
 - › Microwave sounders, radiosondes, SYNOP, ...
- Consider more applications
 - › Medium-range weather forecasting
- Can one continuously update a model?

Outlook



Summary

AtmoRep



<https://arxiv.org/abs/2308.13280>

- Numerical statistical atmospheric model
 - › Complementary to classical GCMs and ESMs
 - › Represented by very large neural network
 - › Very long training leads to continuous improvement

AtmoRep

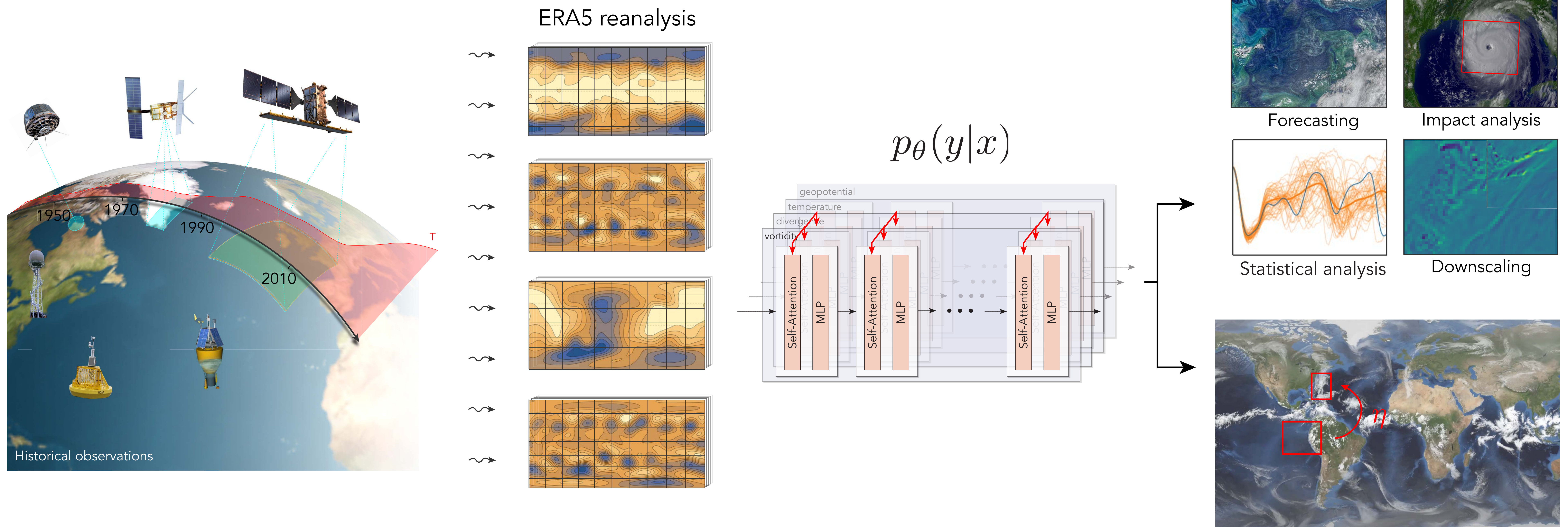


<https://arxiv.org/abs/2308.13280>

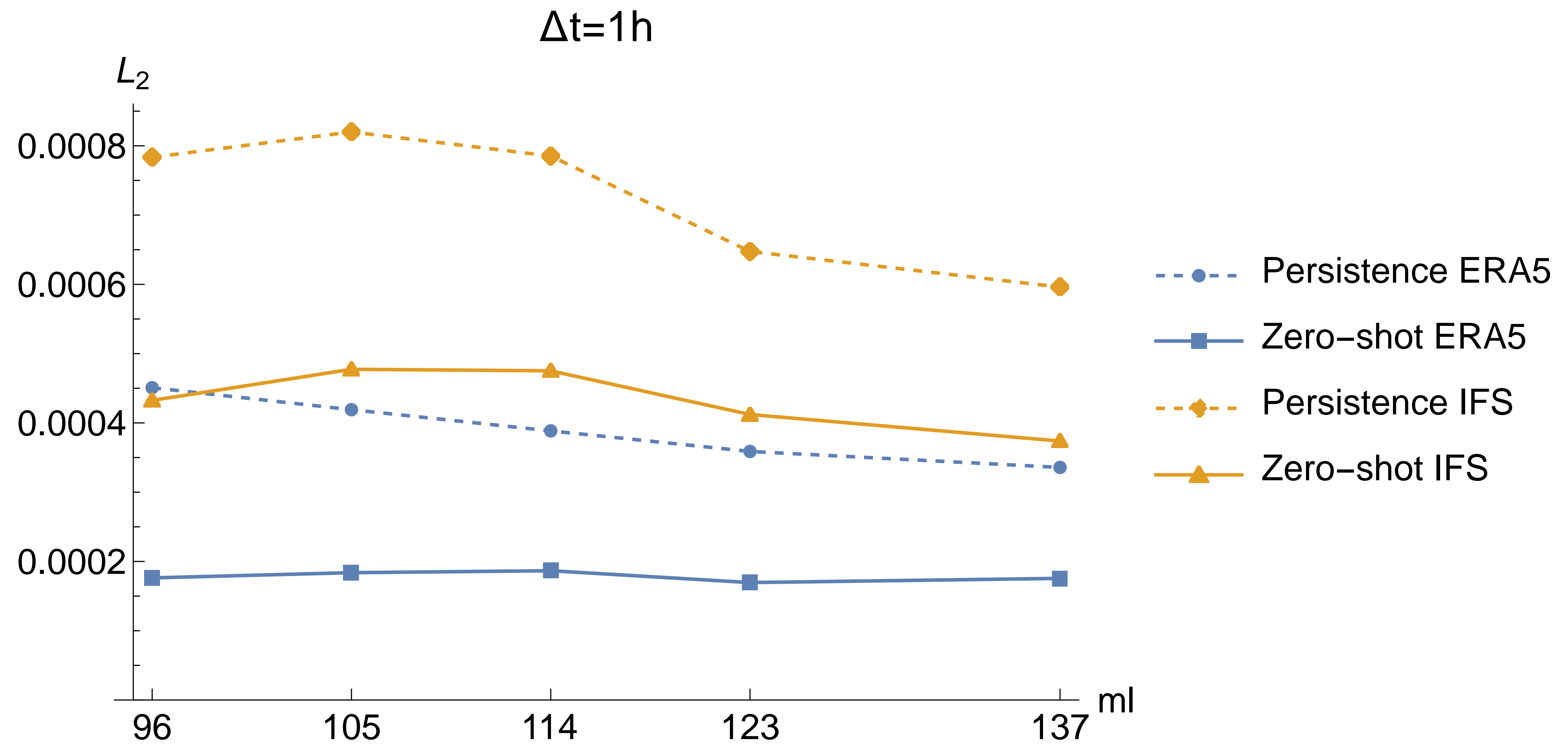
- Numerical statistical atmospheric model
 - › Complementary to classical GCMs and ESMs
 - › Represented by very large neural network
 - › Very long training leads to continuous improvement
- BERT-type training leads to versatile intrinsic capabilities
 - › Forecasting, temporal interpolation, model correction, ...
 - › Straight forward extension to various applications

AtmoRep

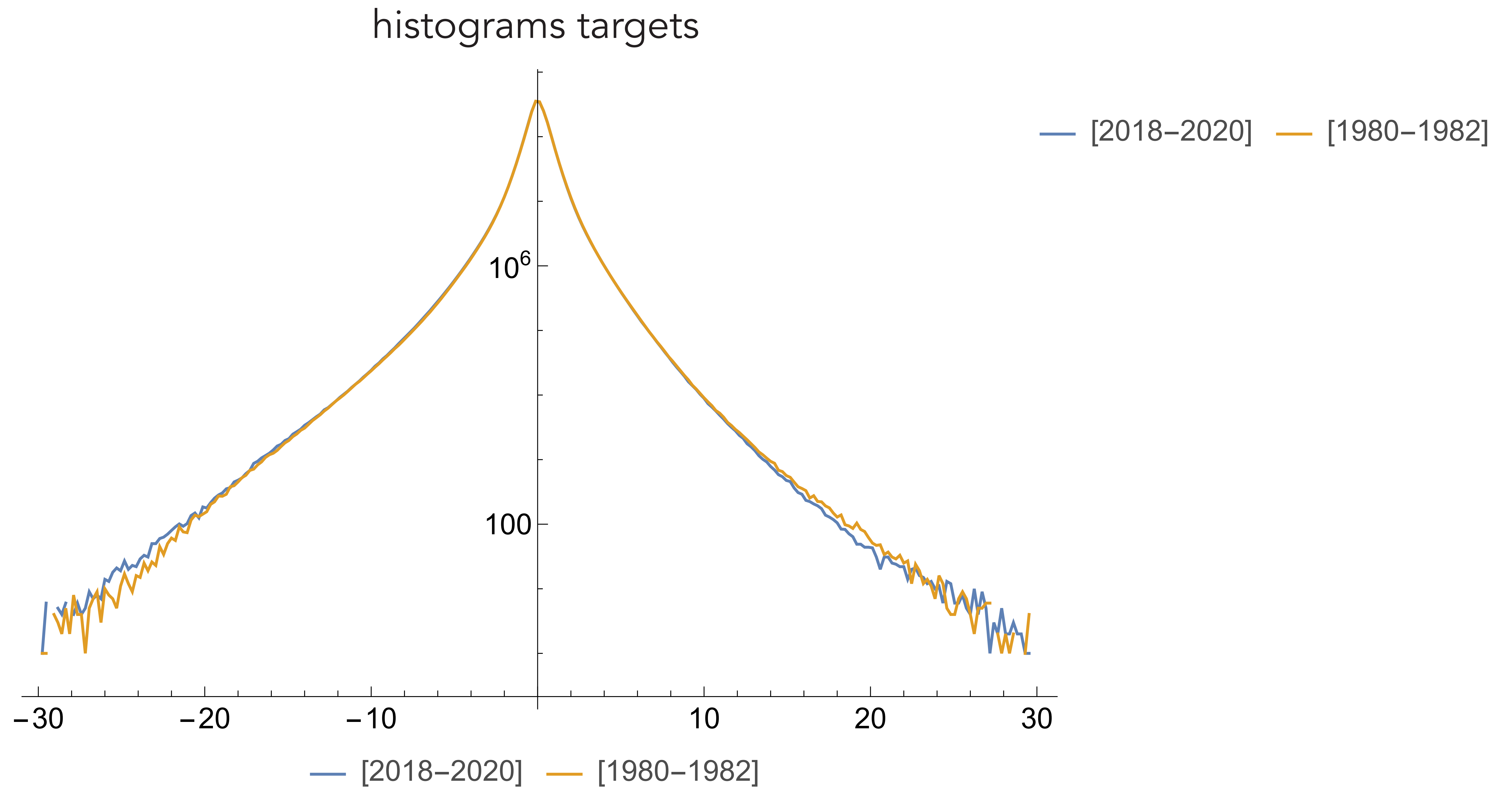
<https://arxiv.org/abs/2308.13280>



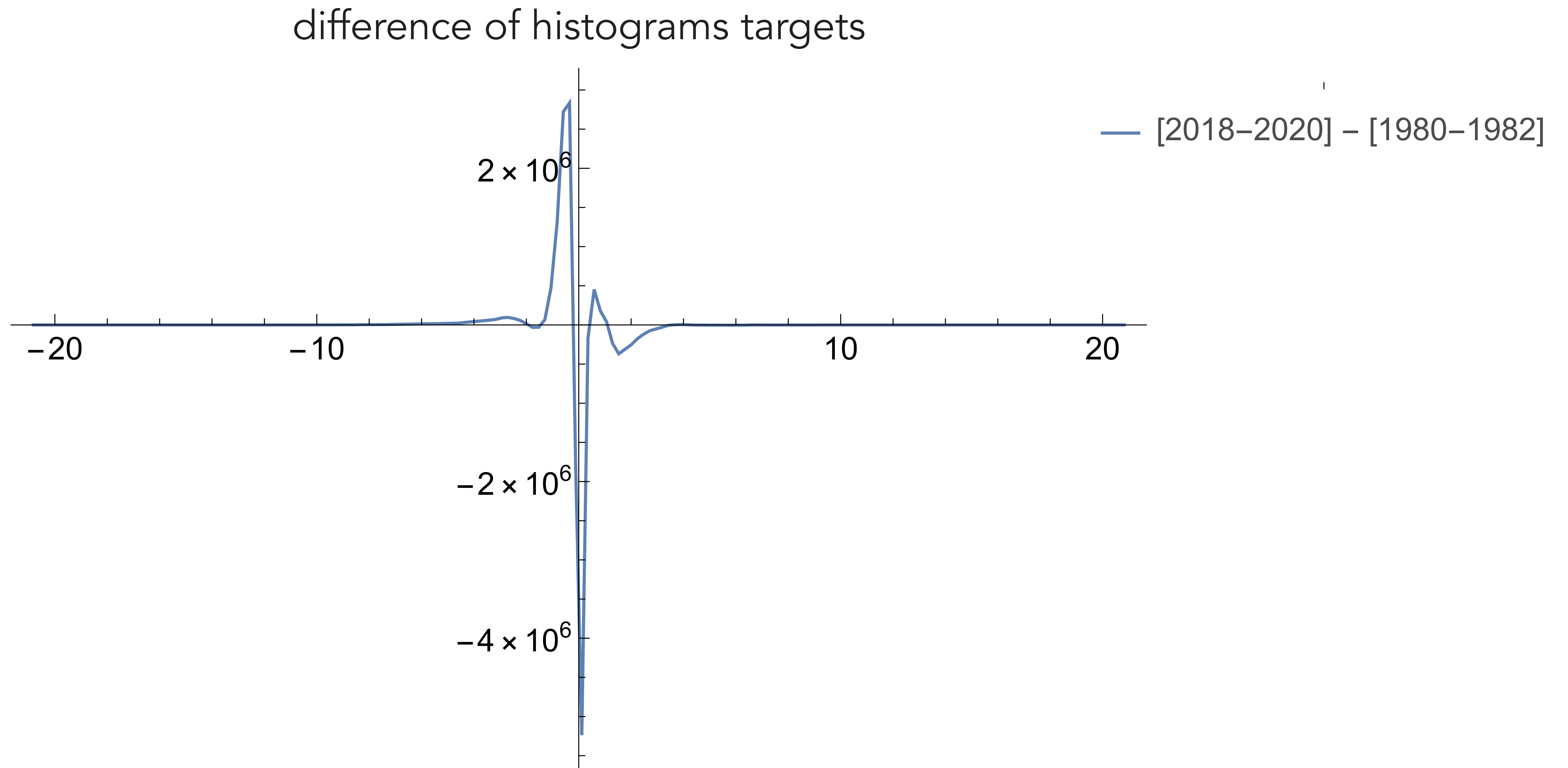
Model correction



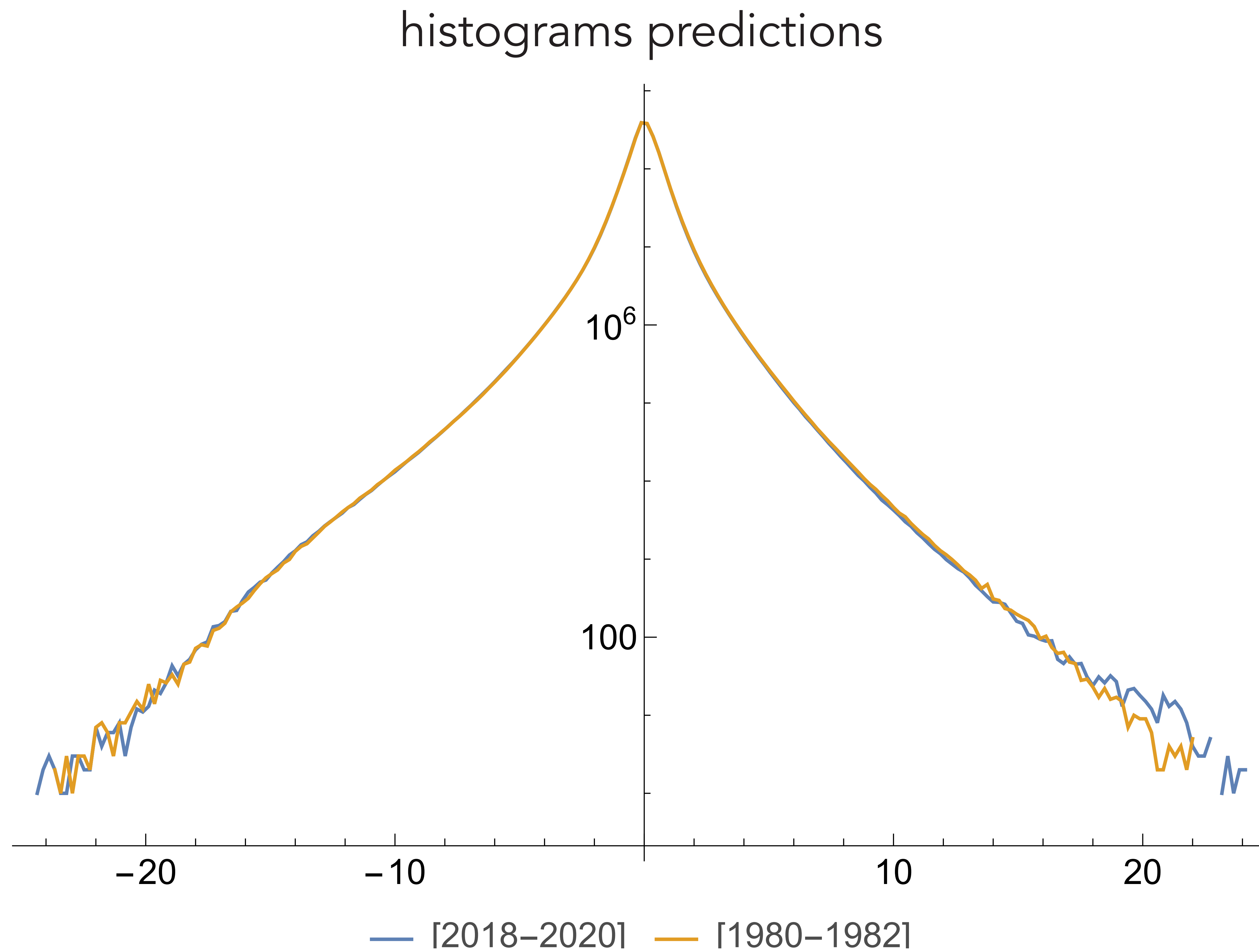
Counterfactuals



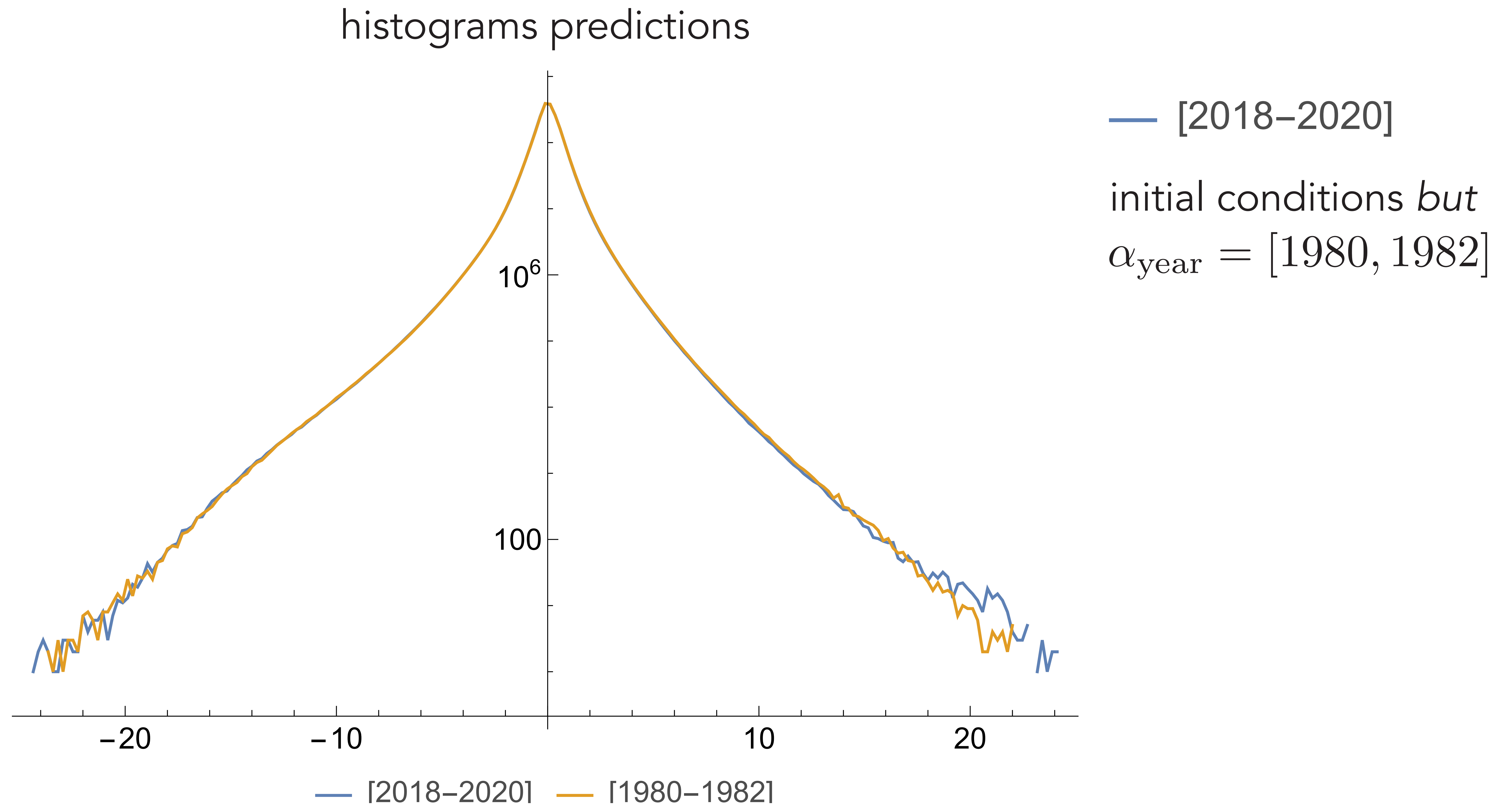
Counterfactuals



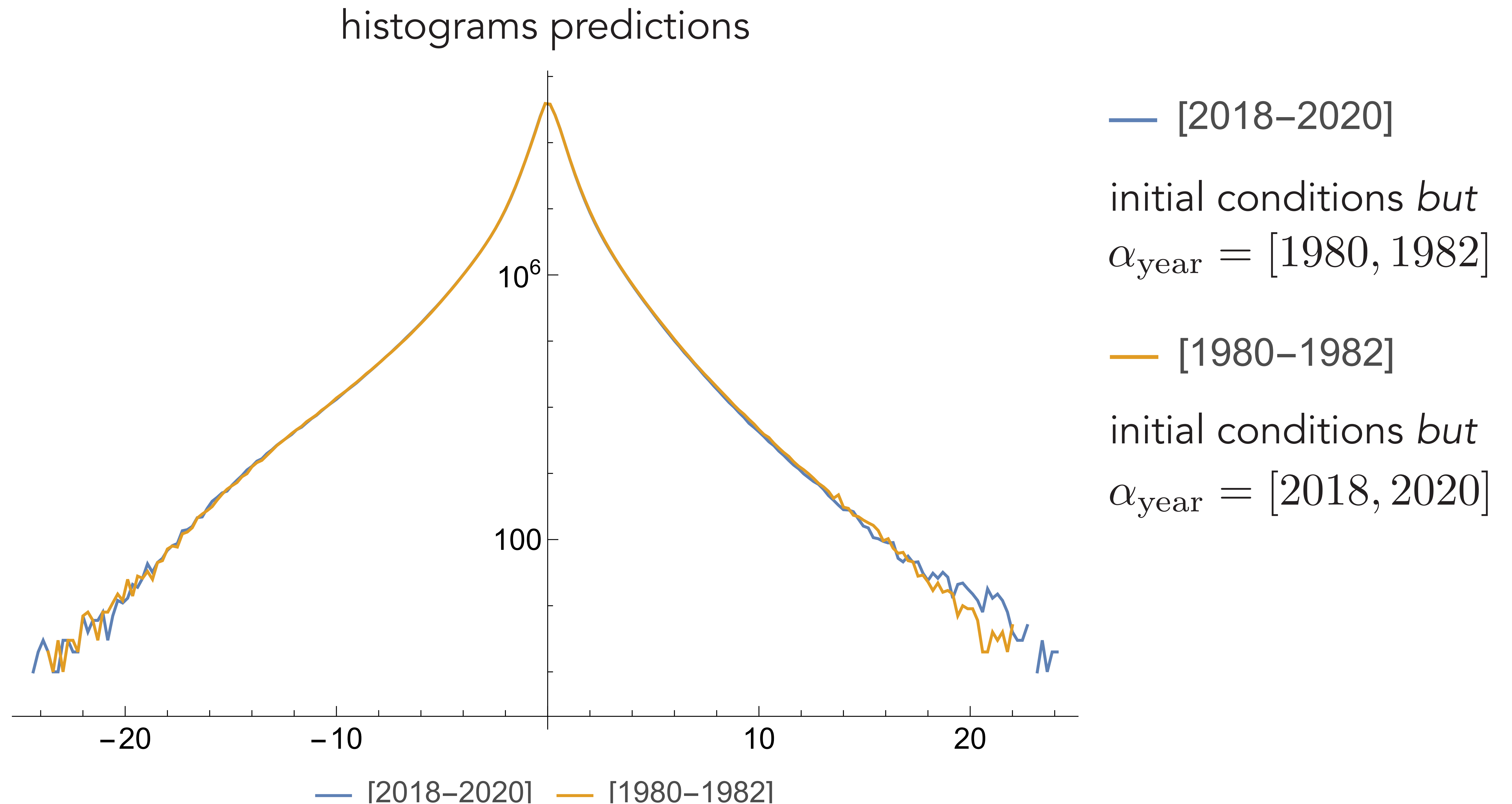
Counterfactuals



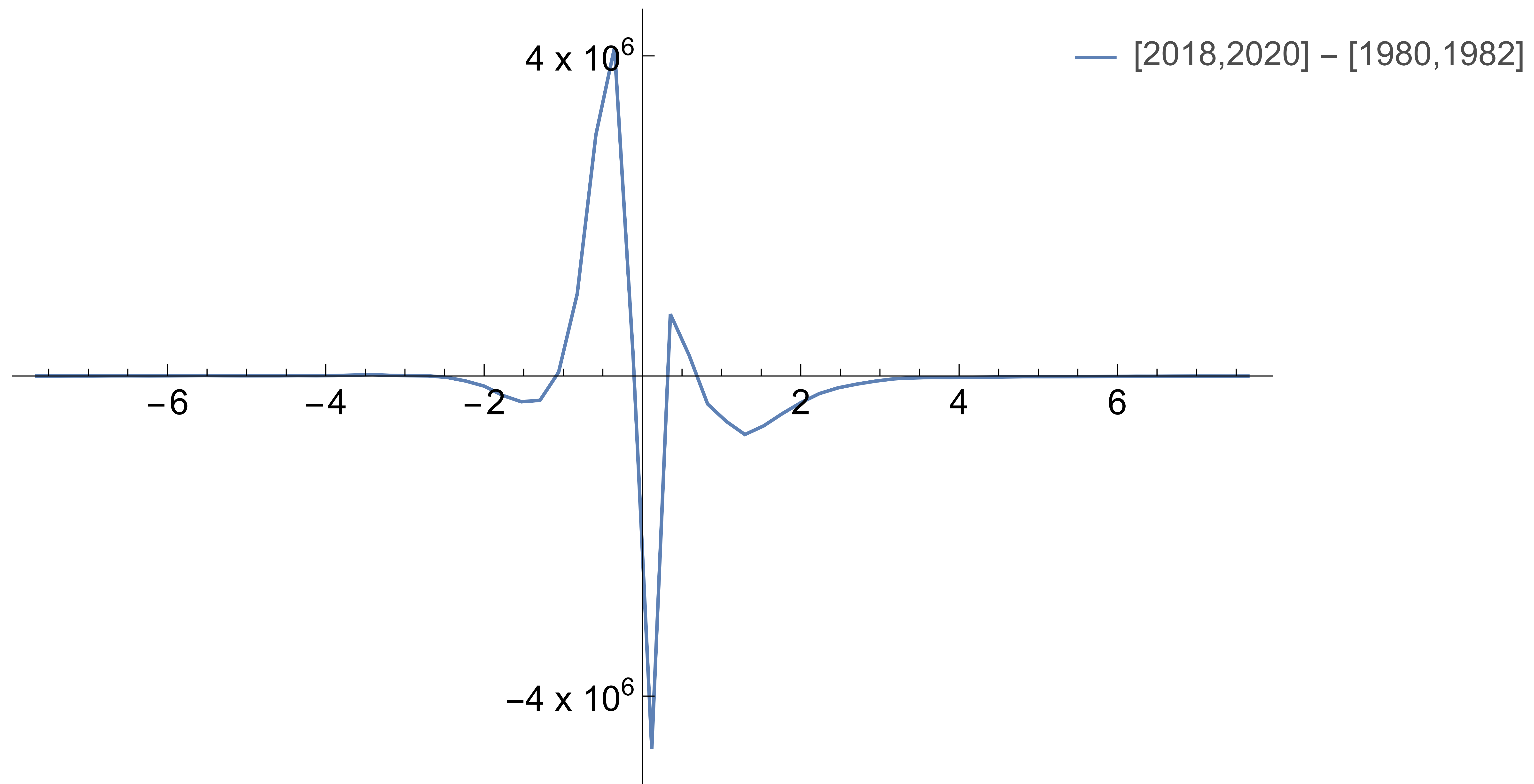
Counterfactuals



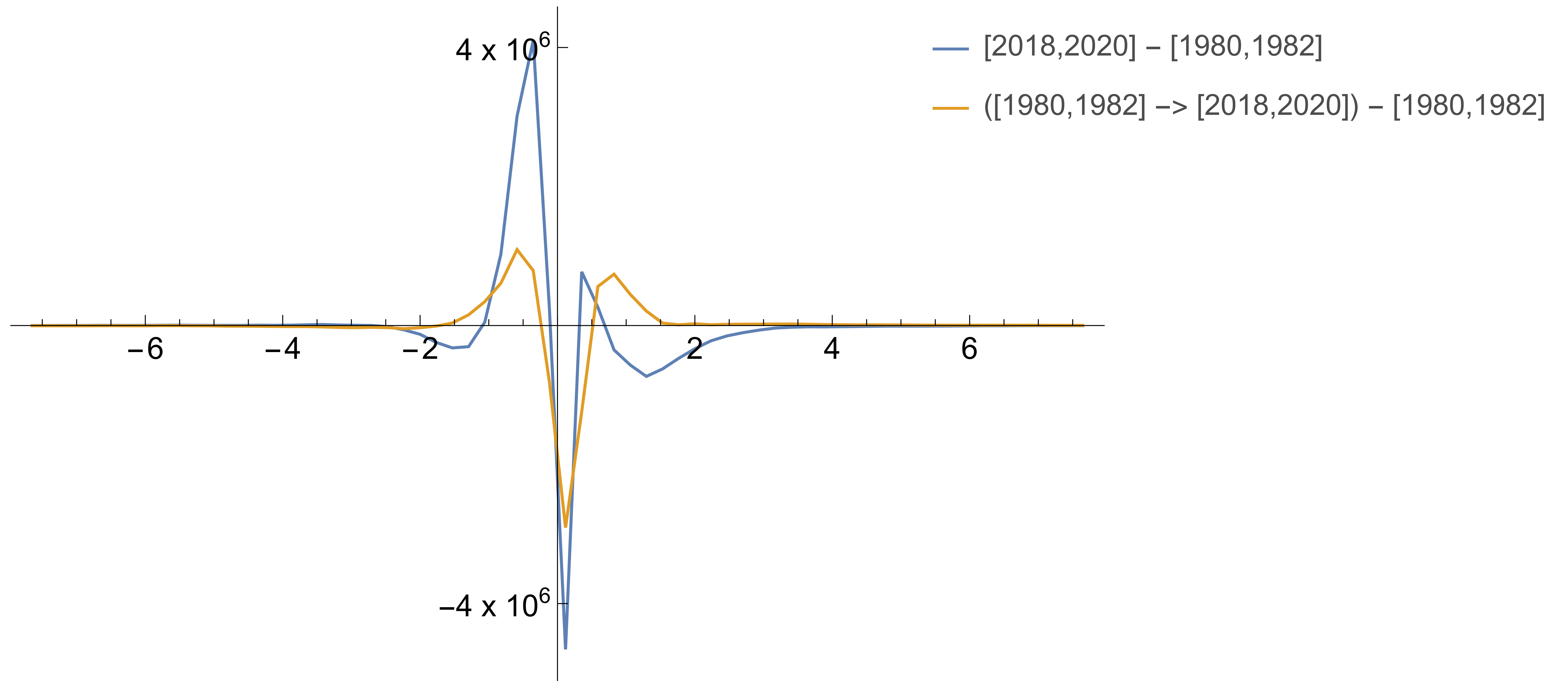
Counterfactuals



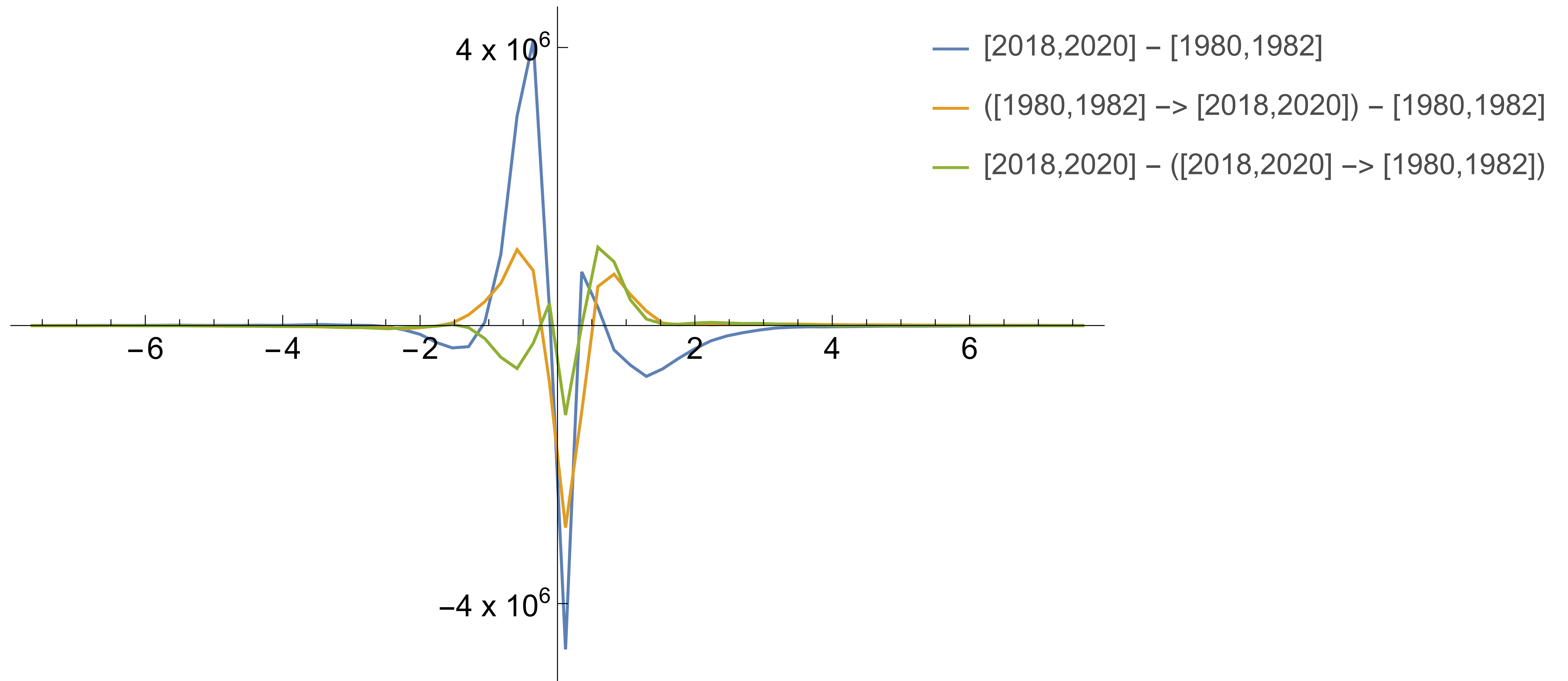
Counterfactuals



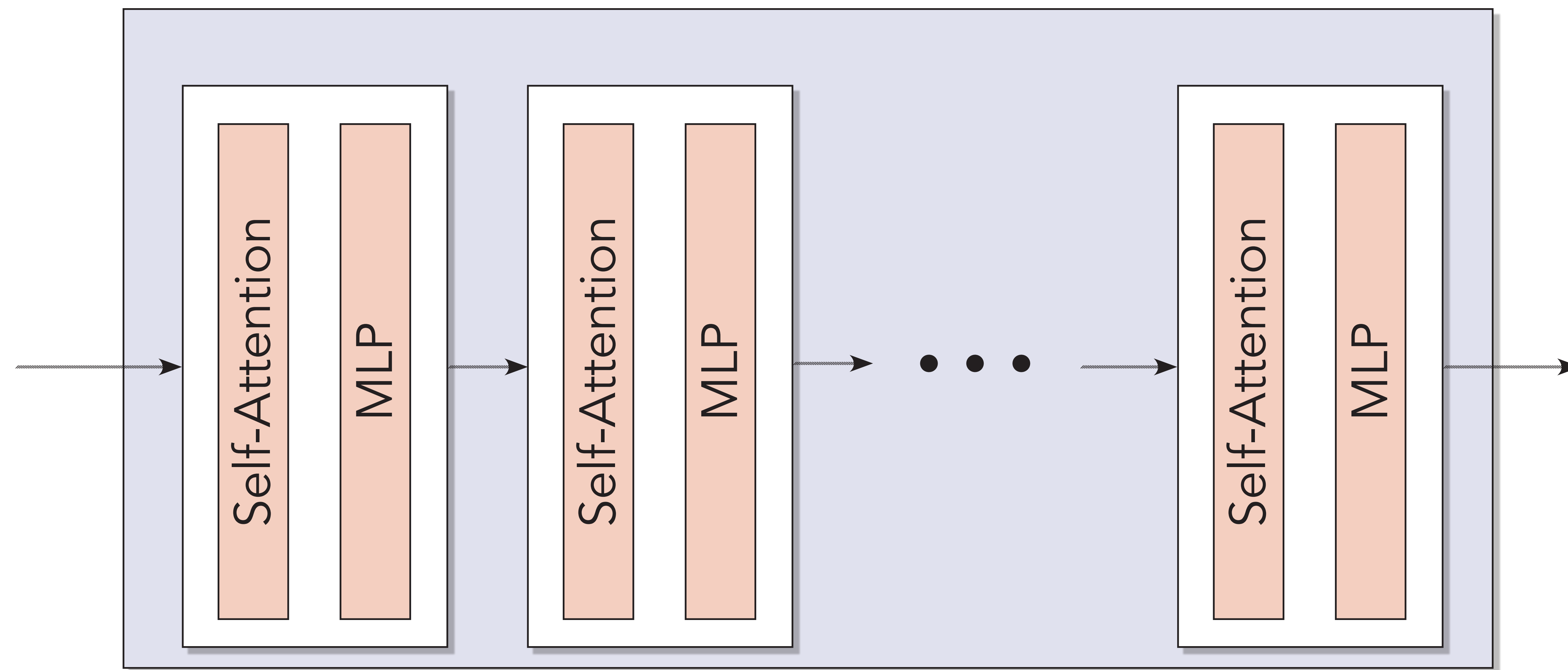
Counterfactuals



Counterfactuals



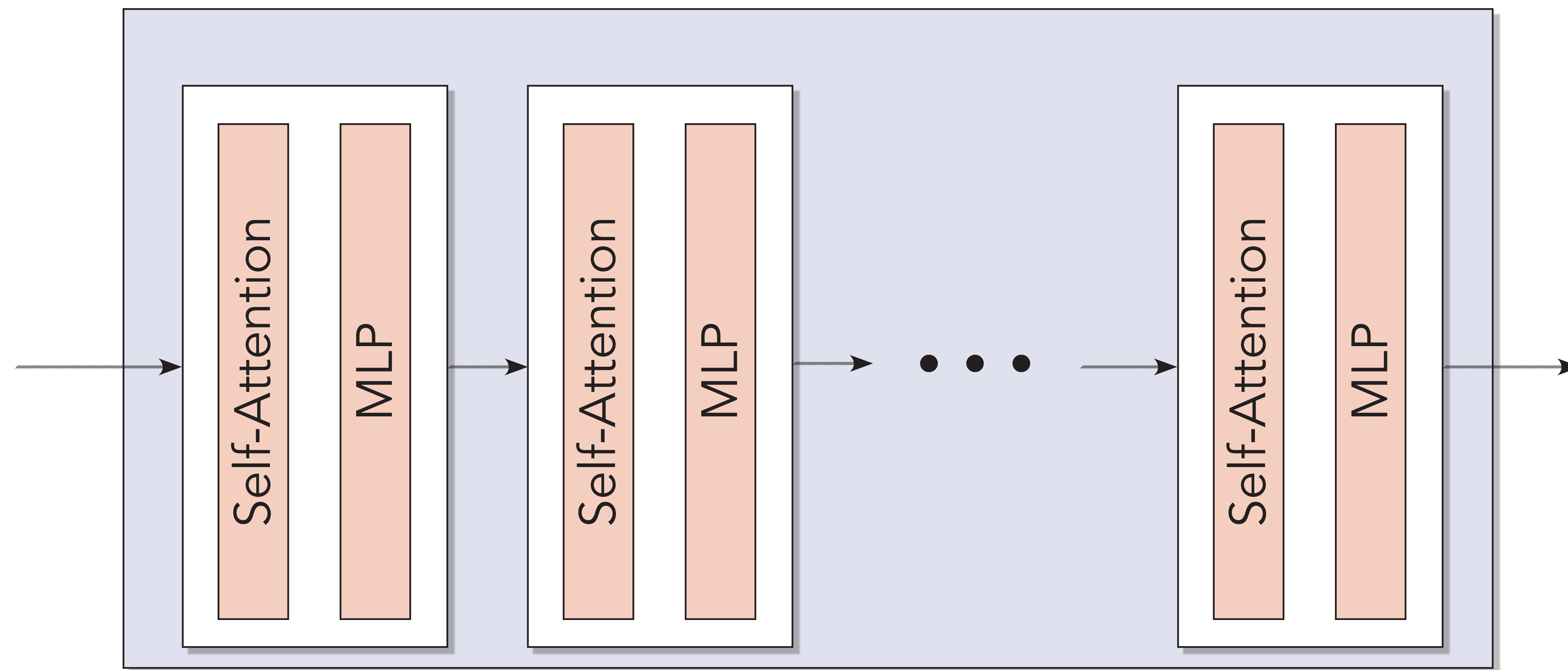
Multiformer



Multiformer

Self
attention

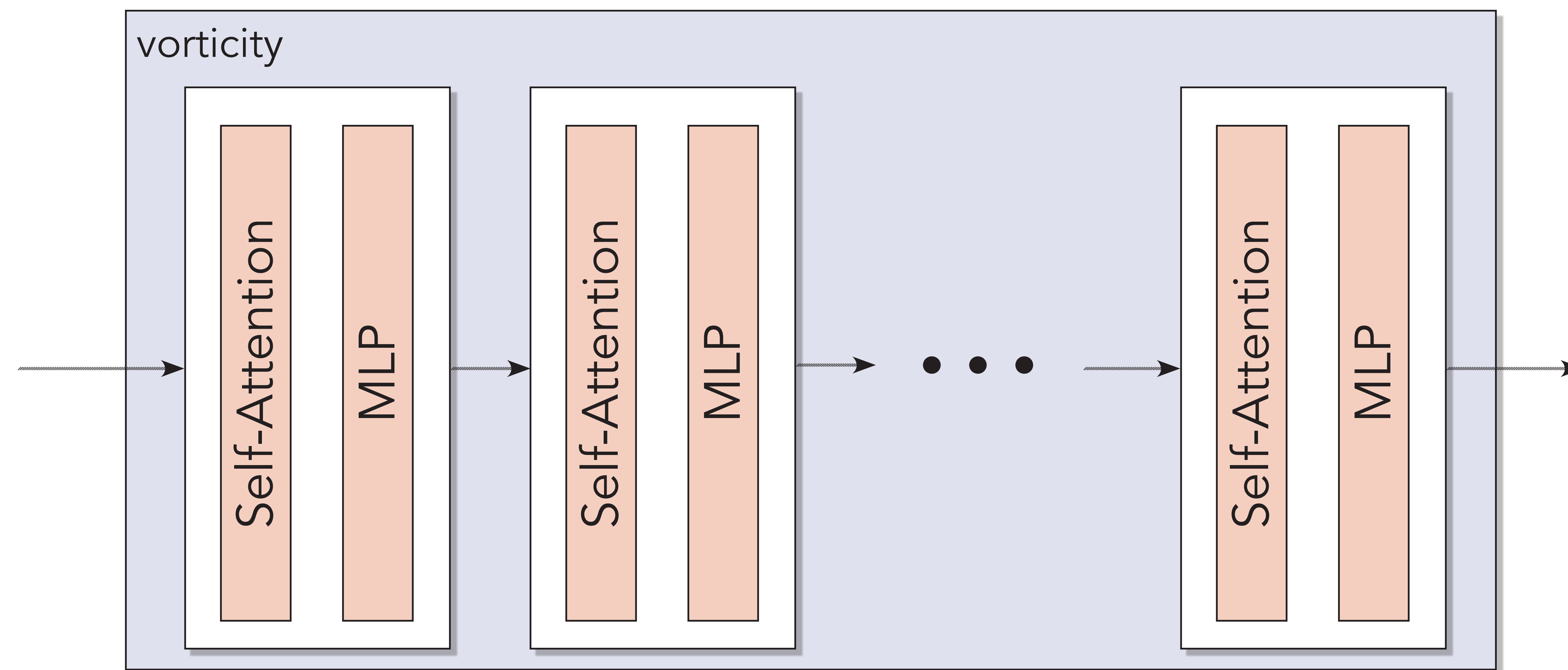
$$\sigma(Q K^T) V$$



Multiformer

Self
attention

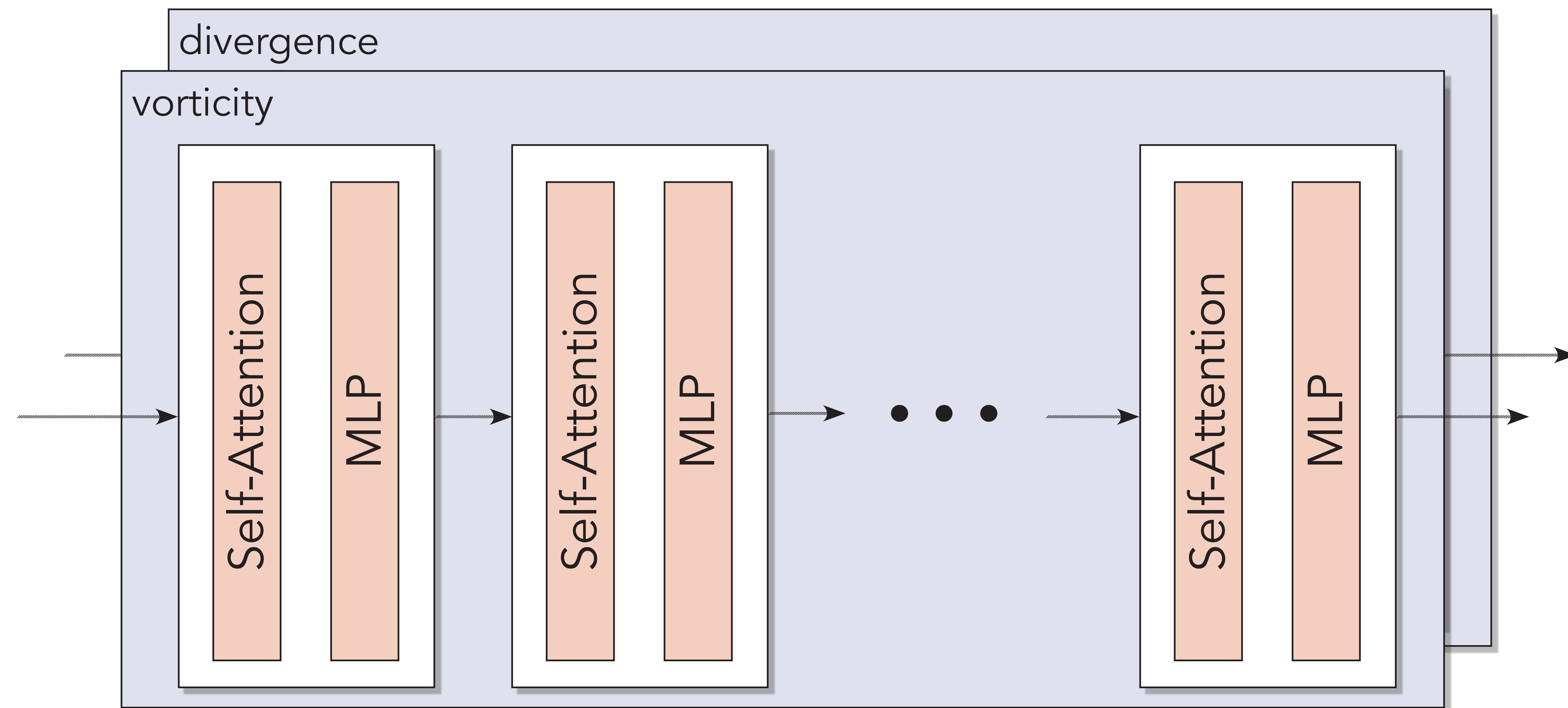
$$\sigma(Q K^T) V$$



Multiformer

Self
attention

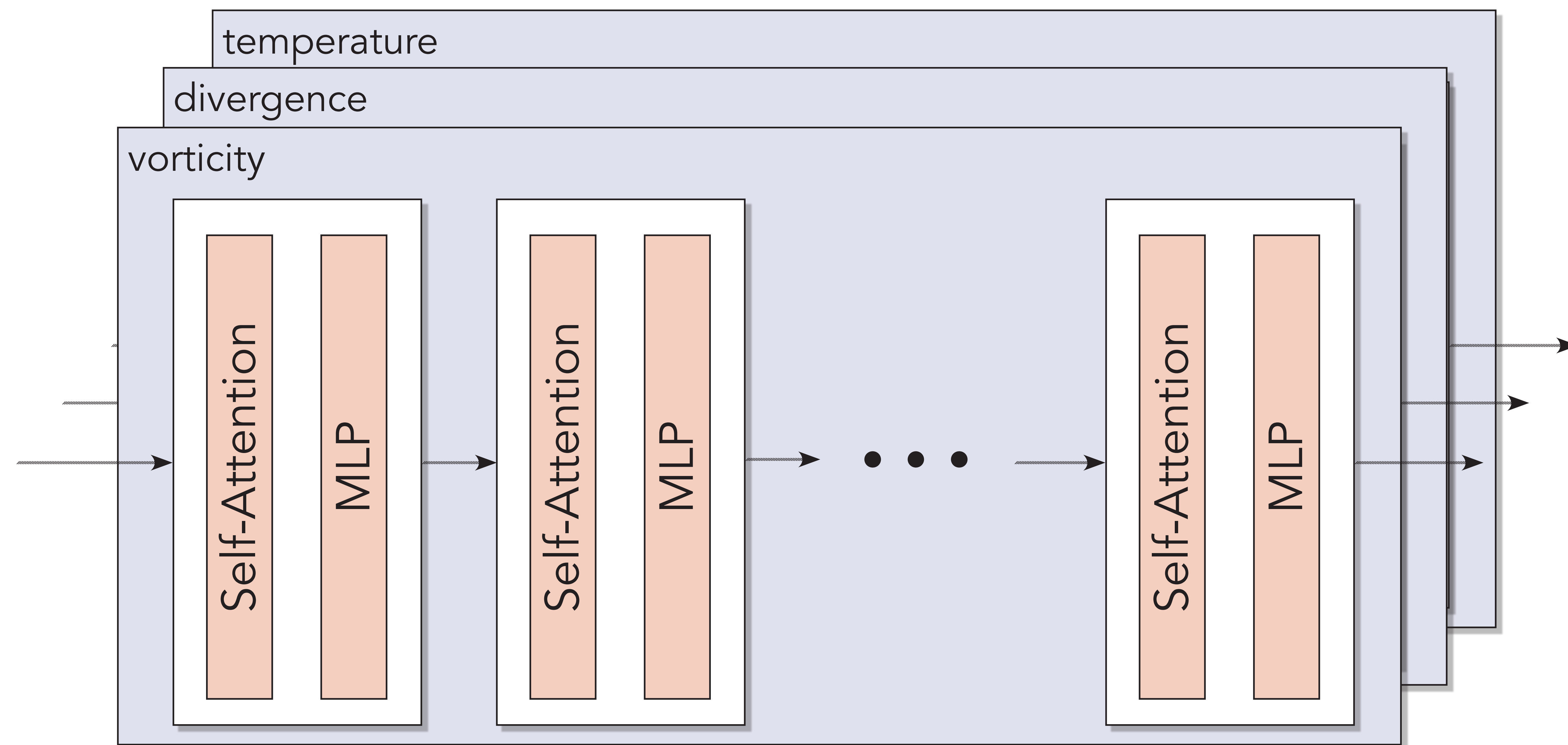
$$\sigma(Q K^T) V$$



Multiformer

Self
attention

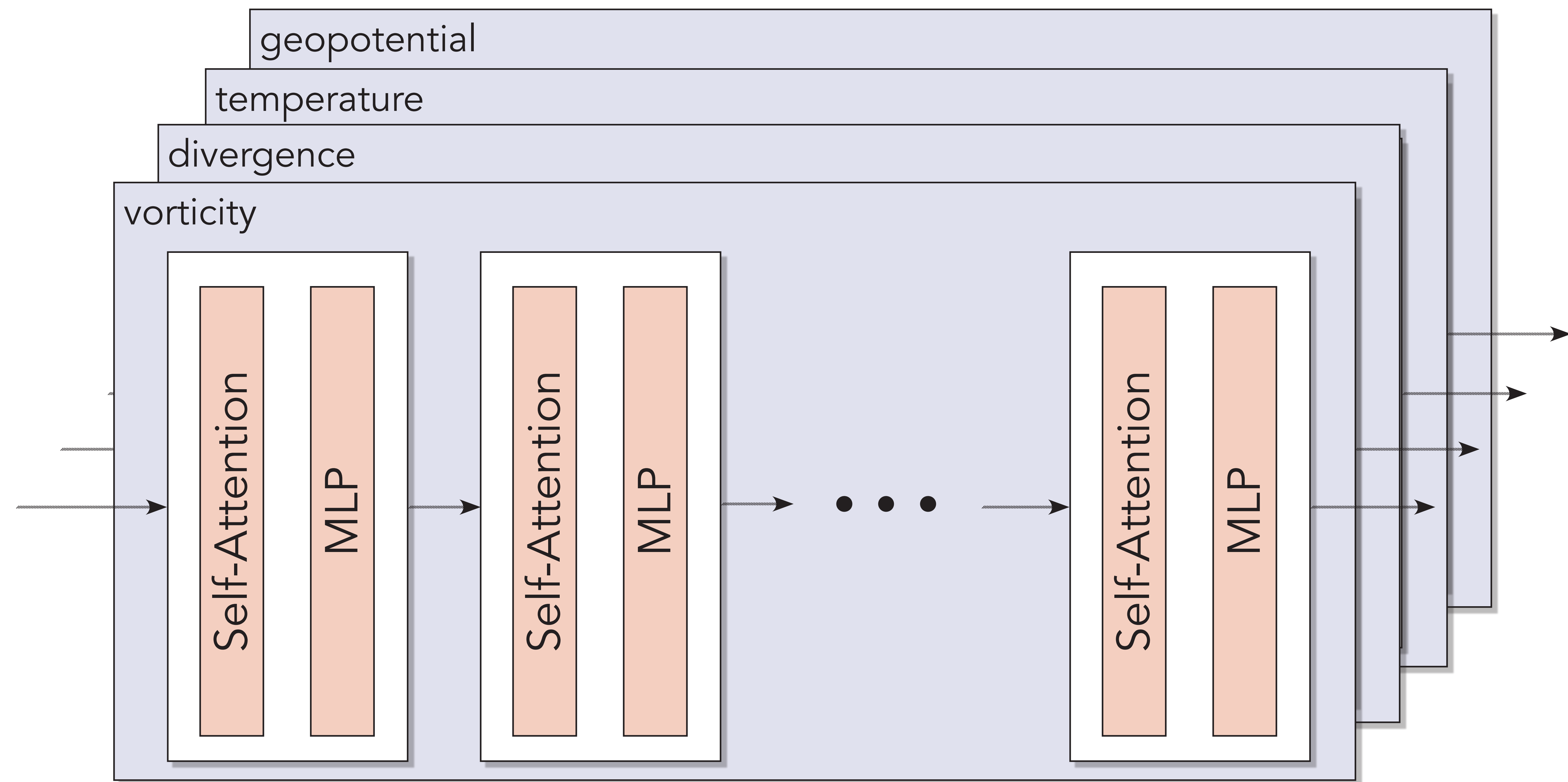
$$\sigma(Q K^T) V$$



Multiformer

Self
attention

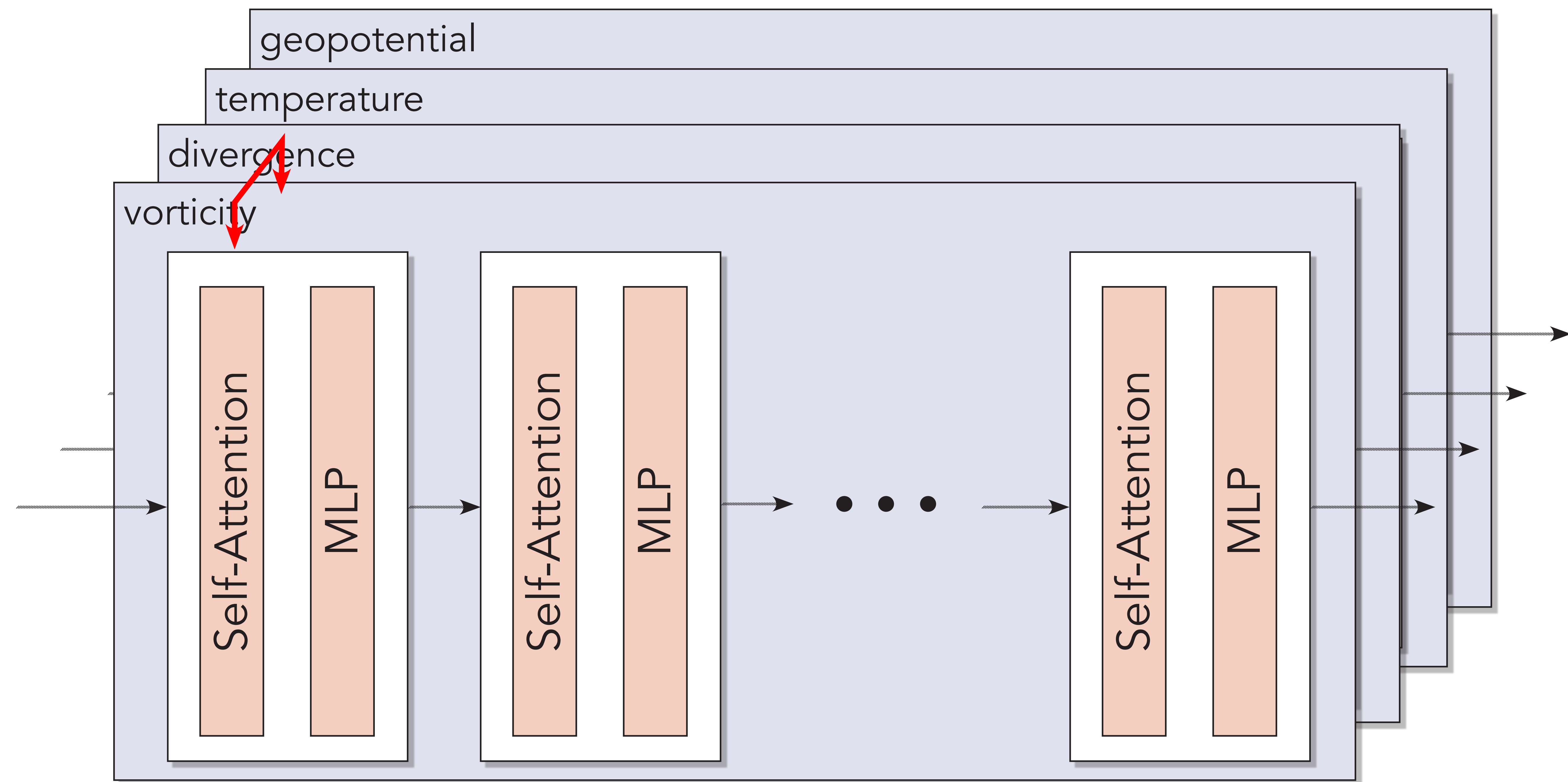
$$\sigma(Q K^T) V$$



Multiformer

Self
attention

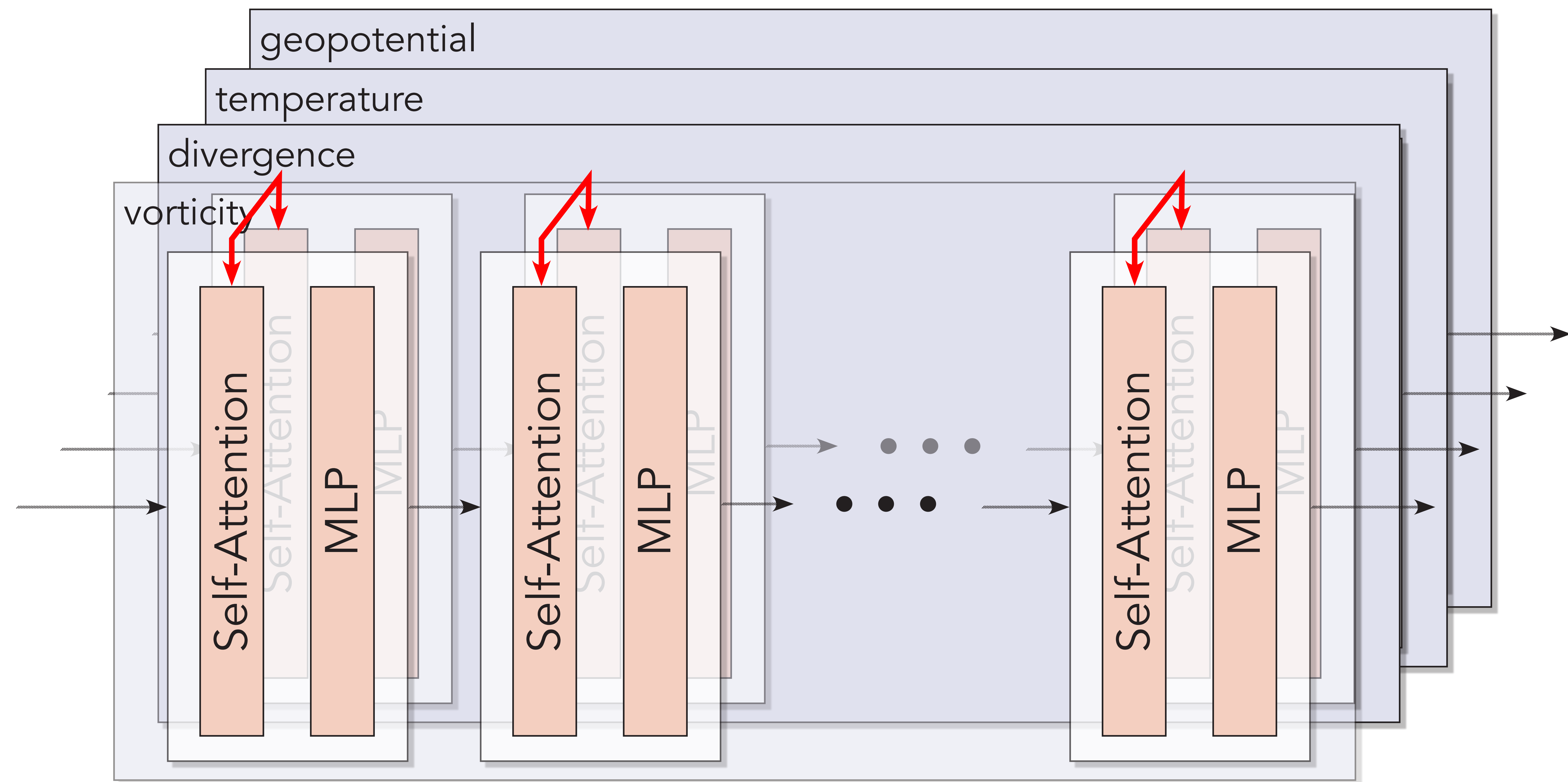
$$\sigma(Q K^T) V$$



Multiformer

Self
attention

$$\sigma(Q K^T) V$$



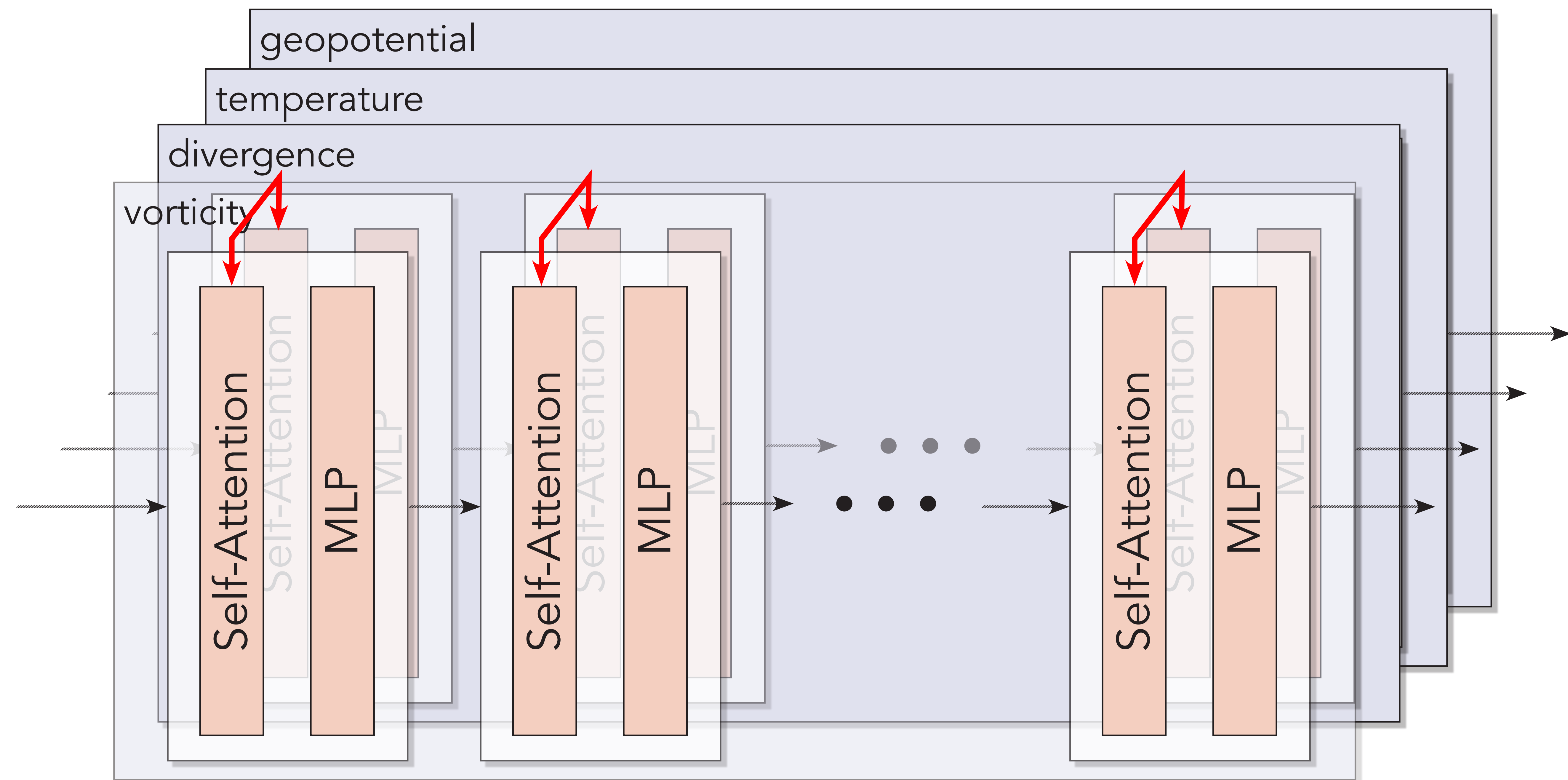
Multiformer

Self
attention

$$\sigma(Q K^T) V$$

Cross
attention

$$\sigma(Q_{\zeta} K_{\mu}^T) V_{\mu}$$



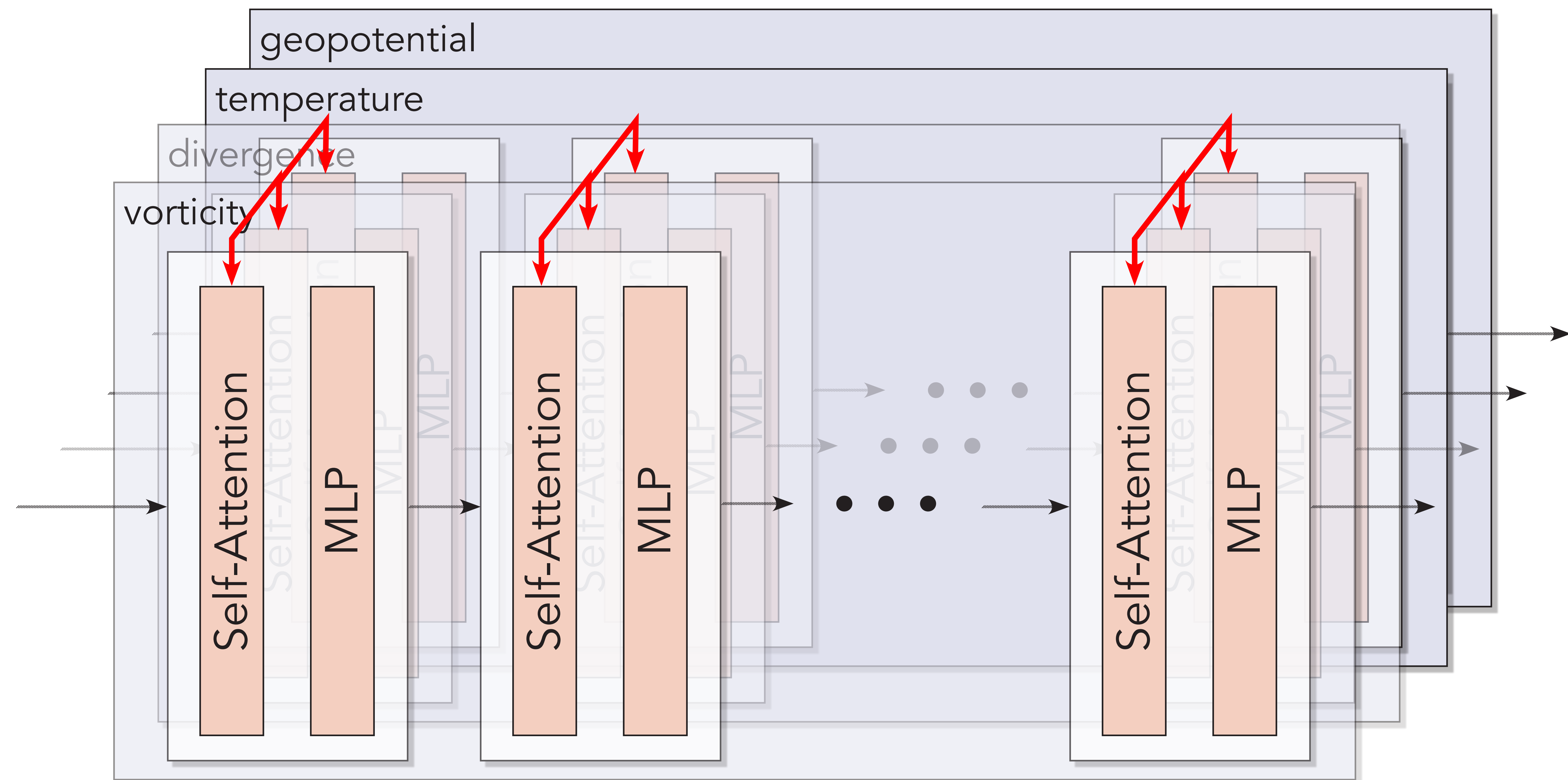
Multiformer

Self
attention

$$\sigma(Q K^T) V$$

Cross
attention

$$\sigma(Q_{\zeta} K_{\mu}^T) V_{\mu}$$



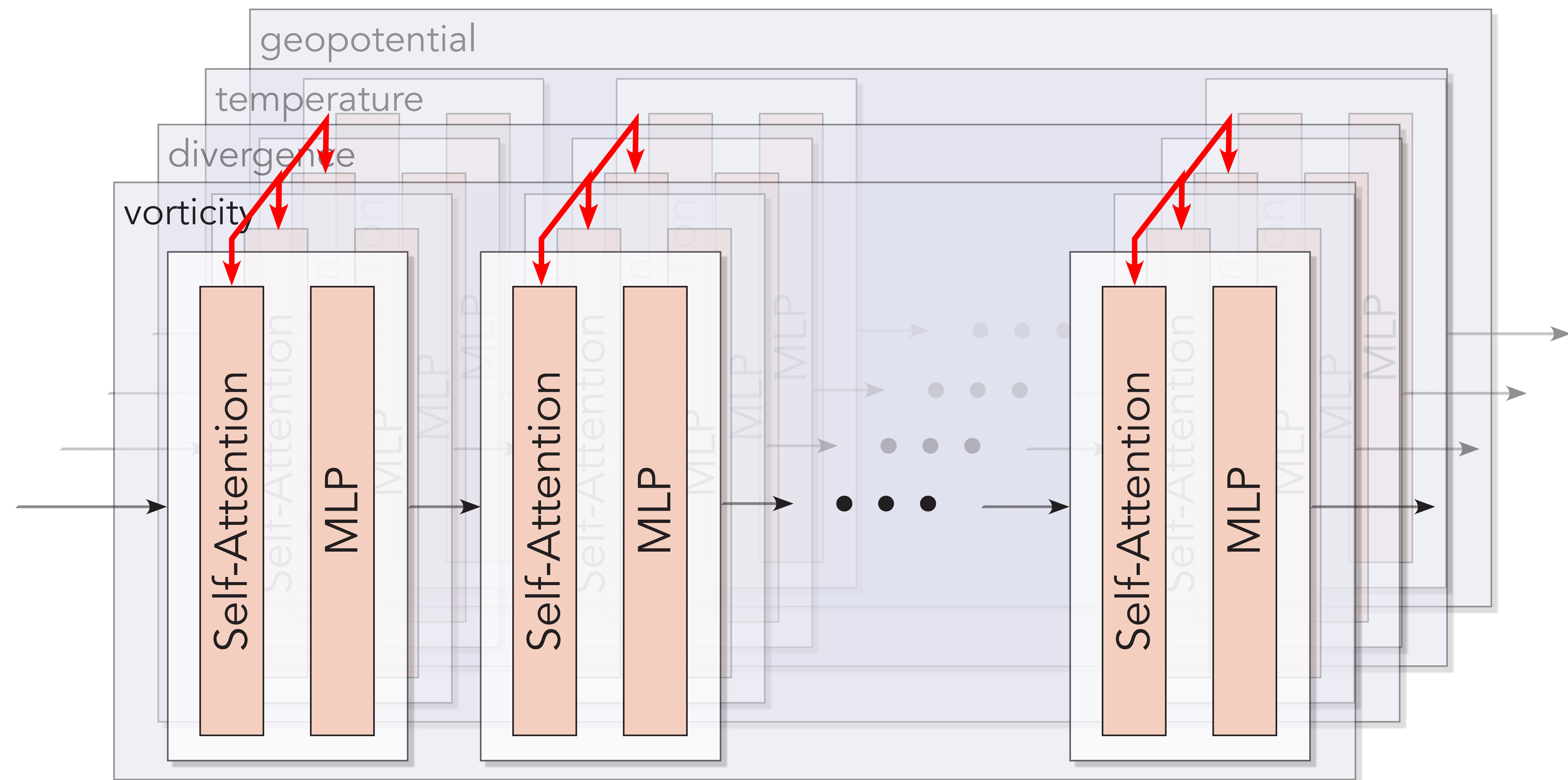
Multiformer

Self
attention

$$\sigma(Q K^T) V$$

Cross
attention

$$\sigma(Q_{\zeta} K_{\mu}^T) V_{\mu}$$



Multiformer

- Plug-and-play of fields
 - › Fields can be added/removed with limited (or no) computational effort

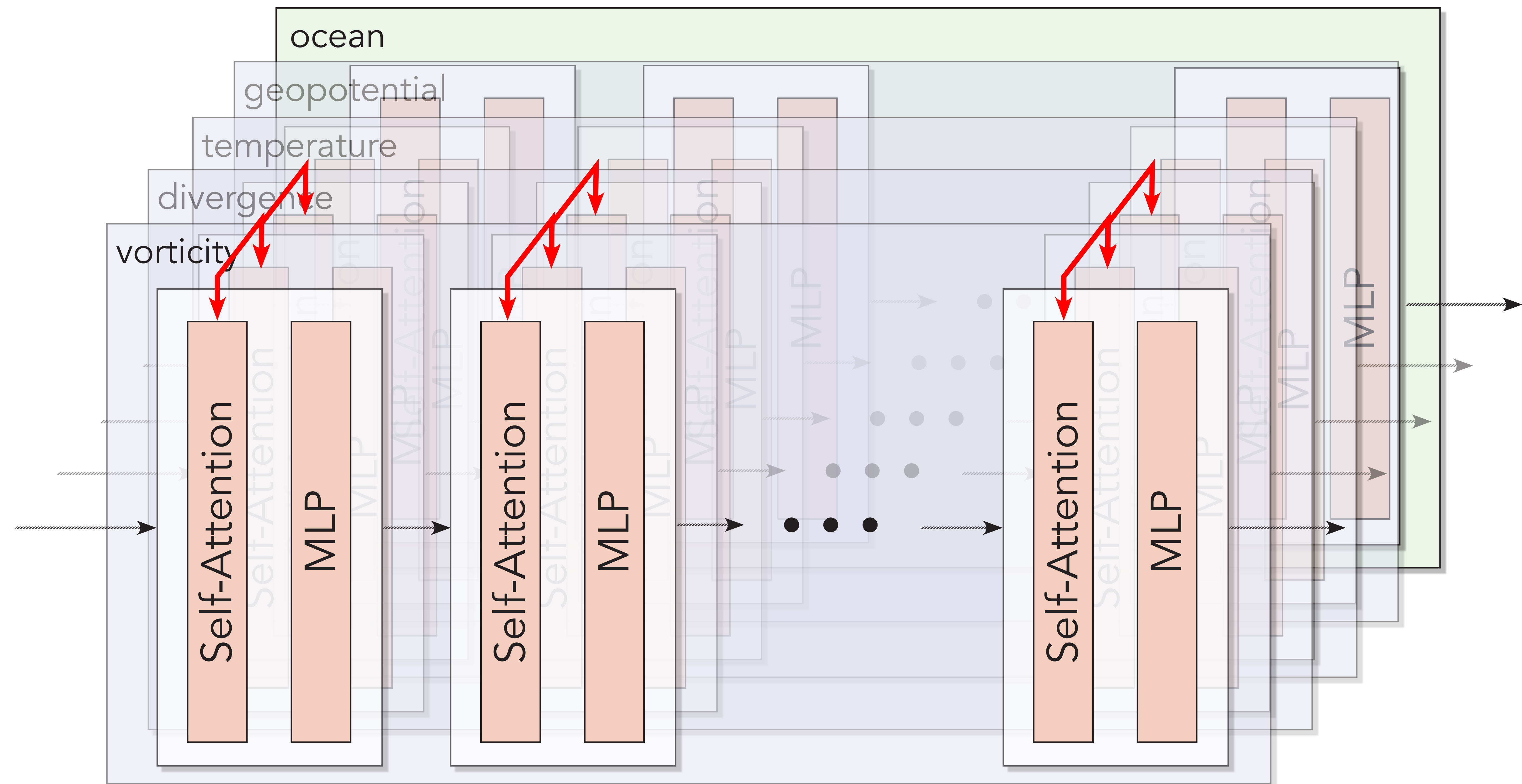
Multiformer

Self
attention

$$\sigma(Q K^T) V$$

Cross
attention

$$\sigma(Q_{\zeta} K_{\mu}^T) V_{\mu}$$



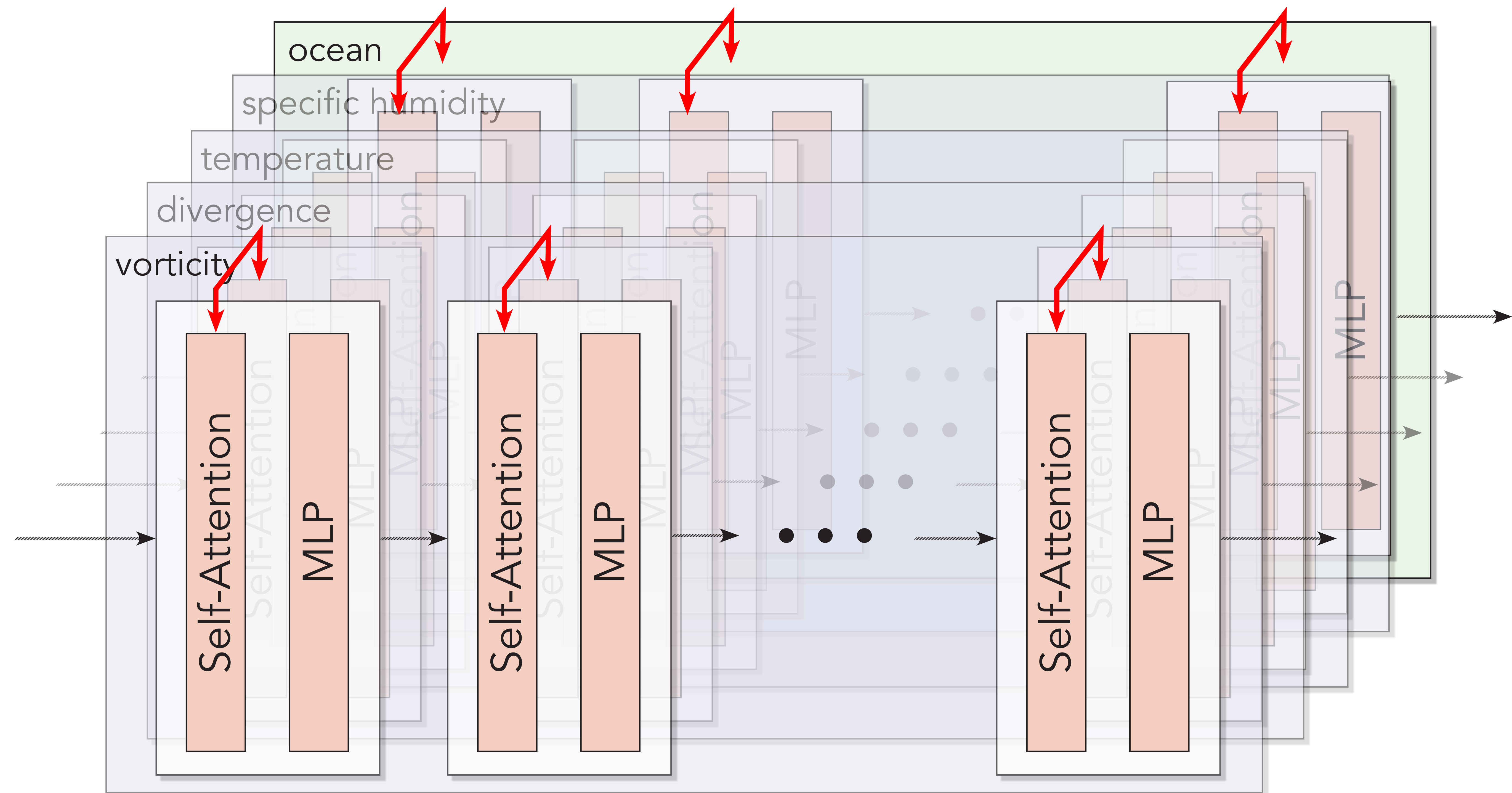
Multiformer

Self
attention

$$\sigma(Q K^T) V$$

Cross
attention

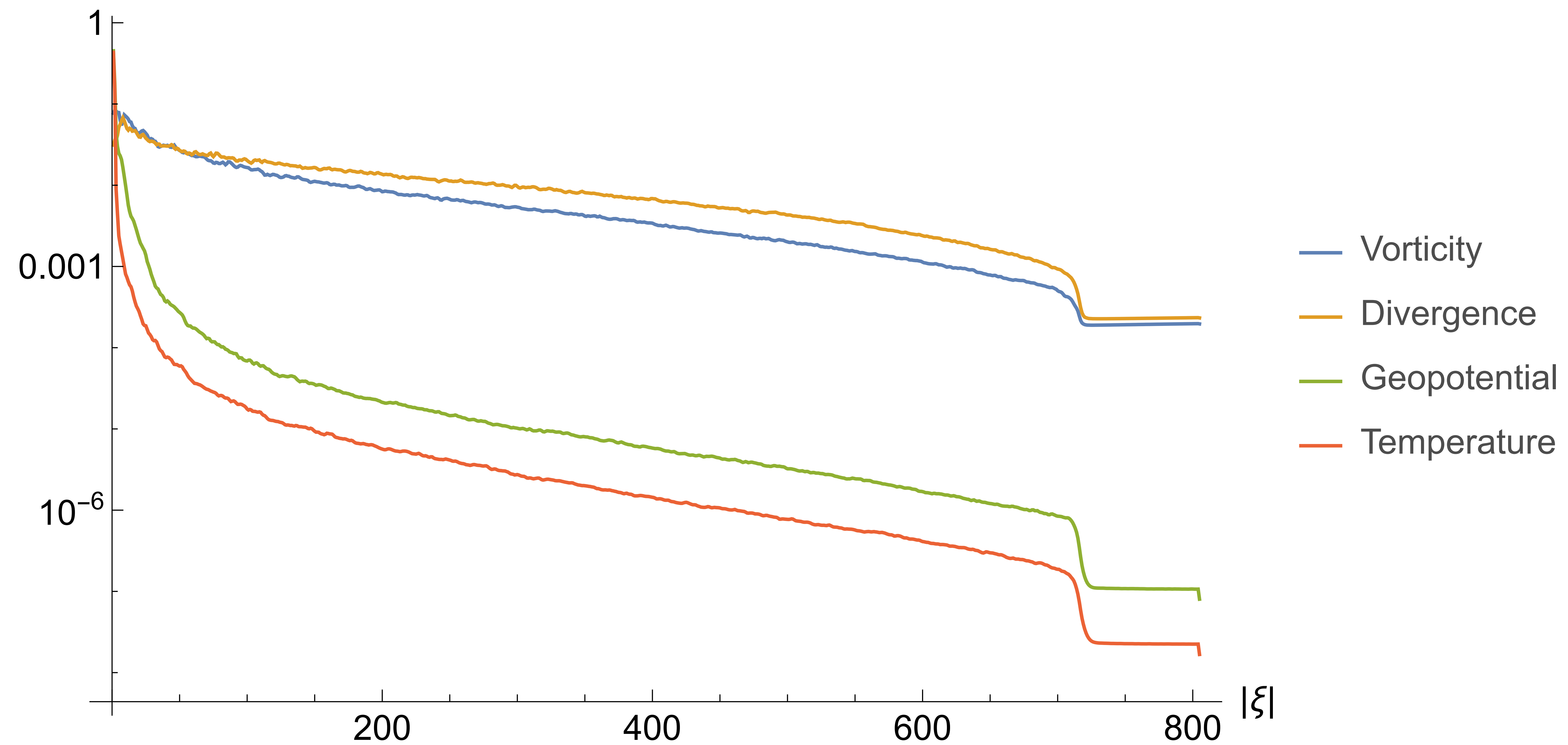
$$\sigma(Q_{\zeta} K_{\mu}^T) V_{\mu}$$



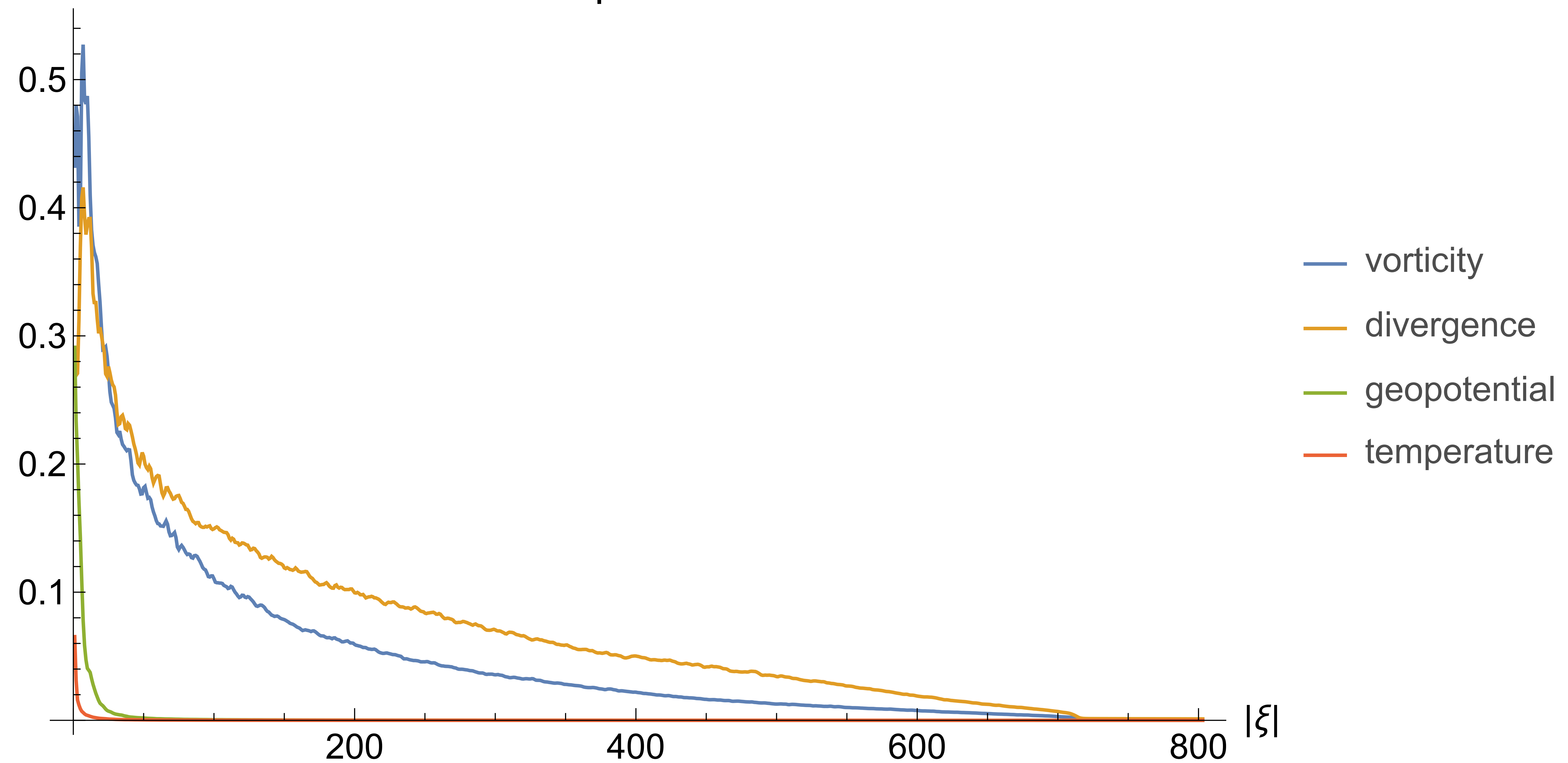
Multiformer

- Plug-and-play of fields
 - › Fields can be added/removed with very limited computational effort
- Cross-attention allows for explicit introspection of interaction between fields
- Different physical fields with different properties have separate latent spaces (and transformations for these)

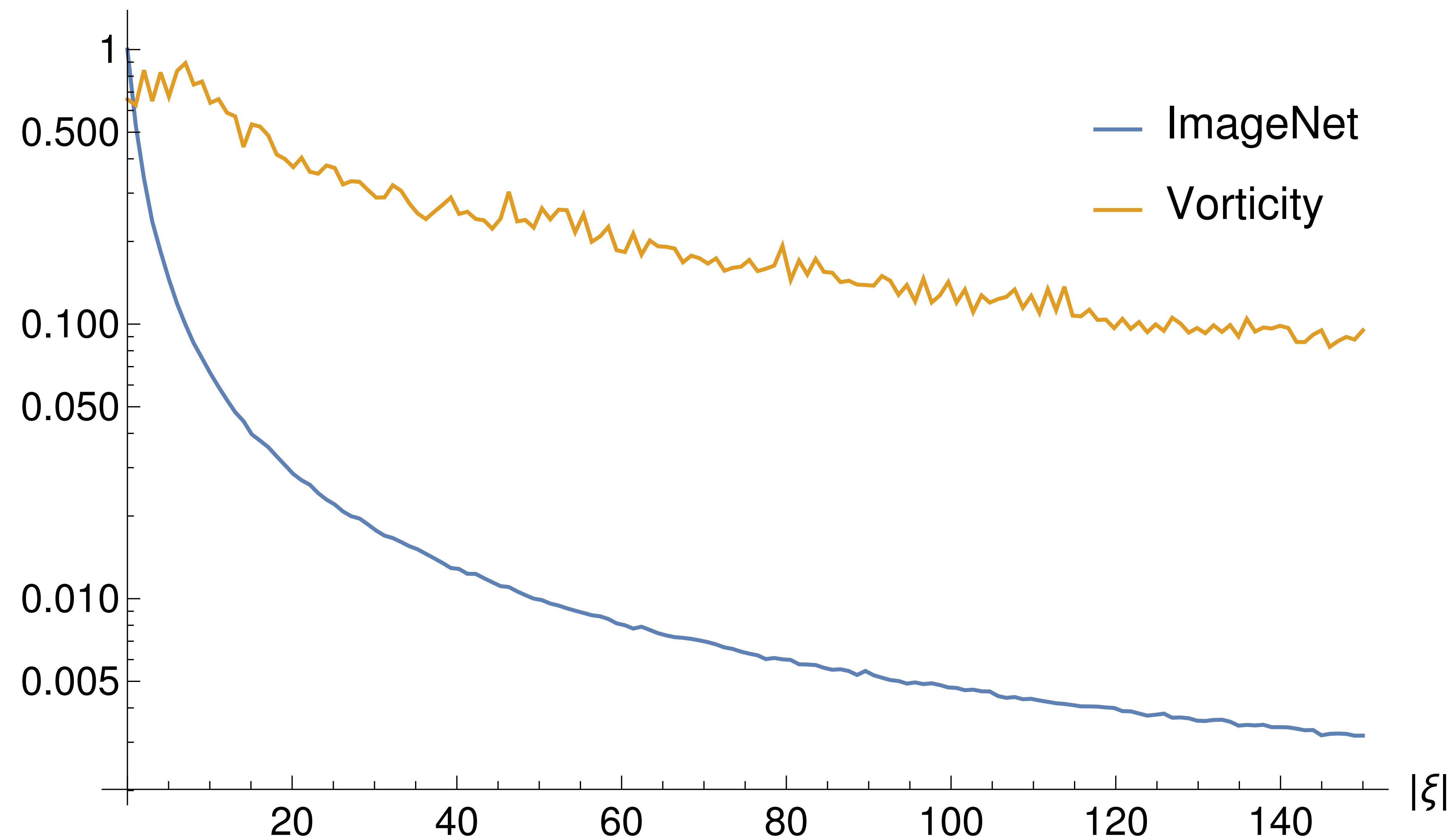
AtmoRep data



AtmoRep data

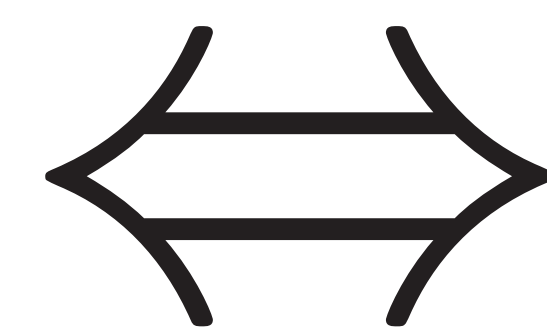


ERA5 versus ImageNet

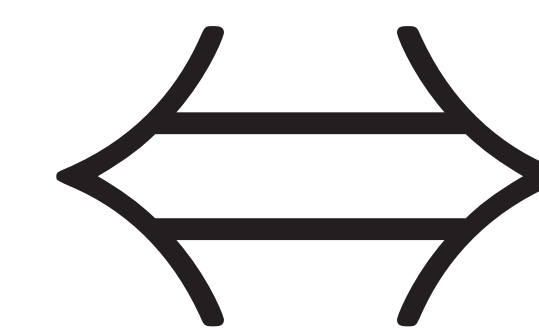


ERA5 versus ImageNet

stream function
velocity potential

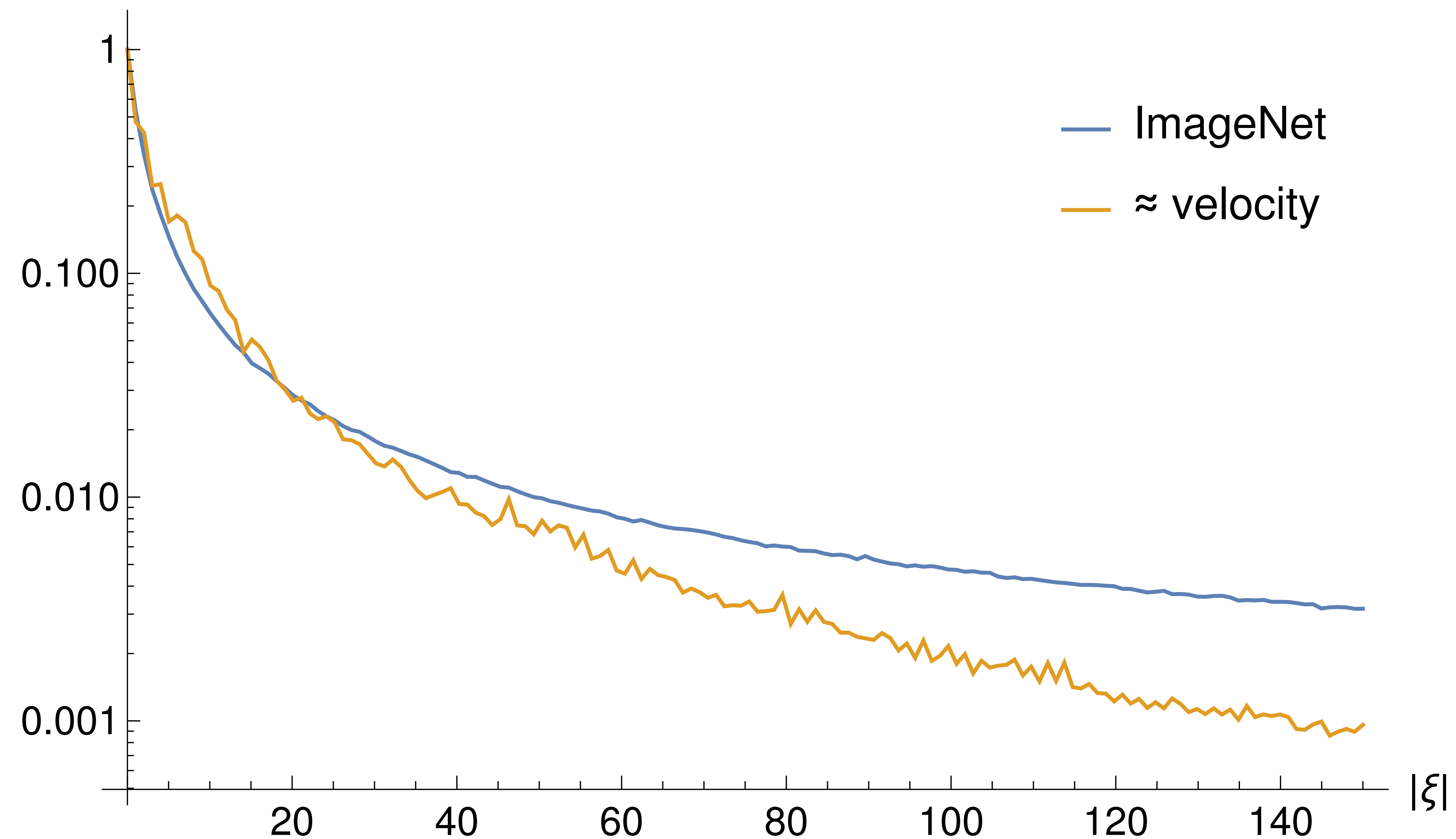


velocity
vector field

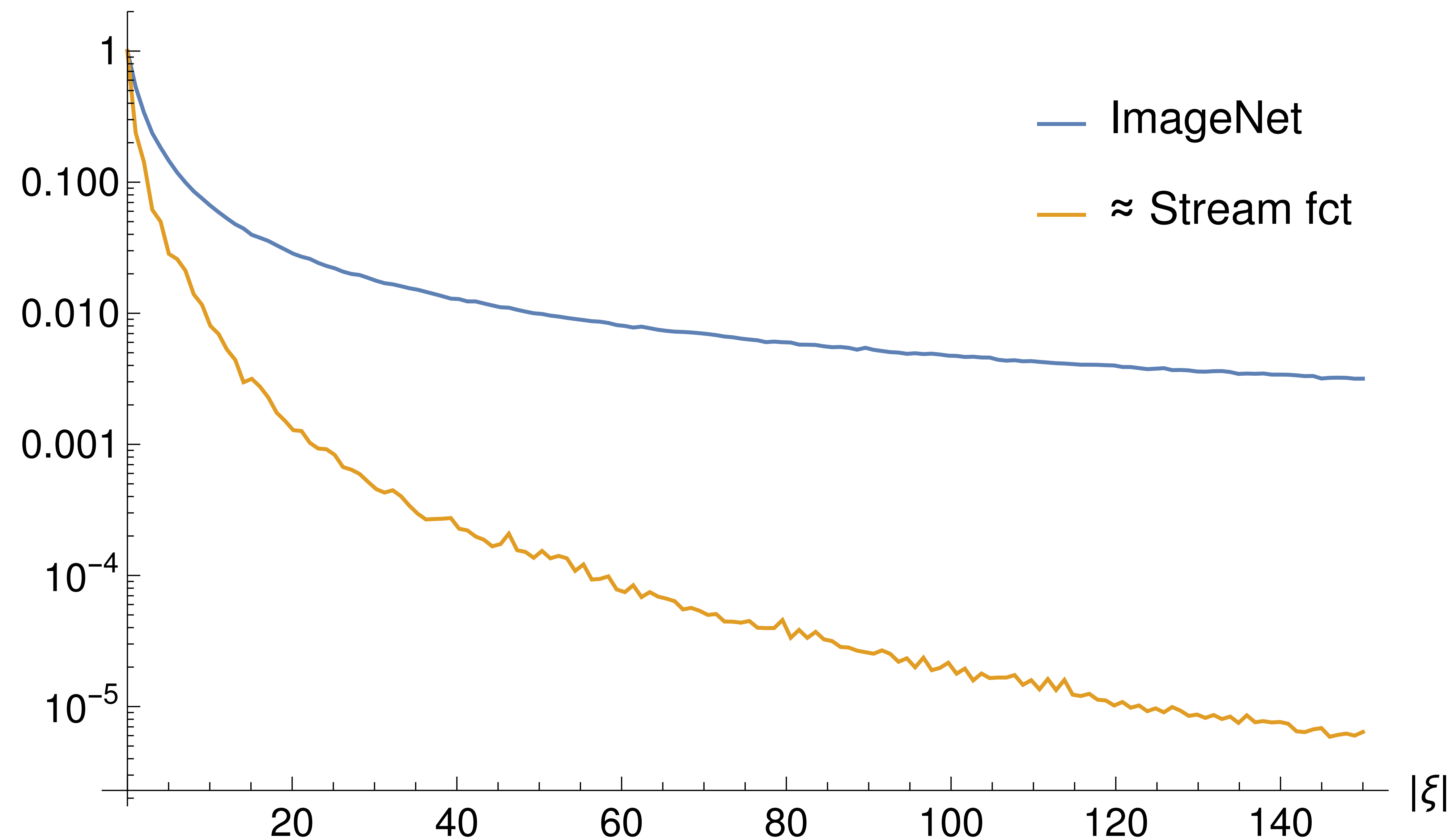


vorticity
divergence

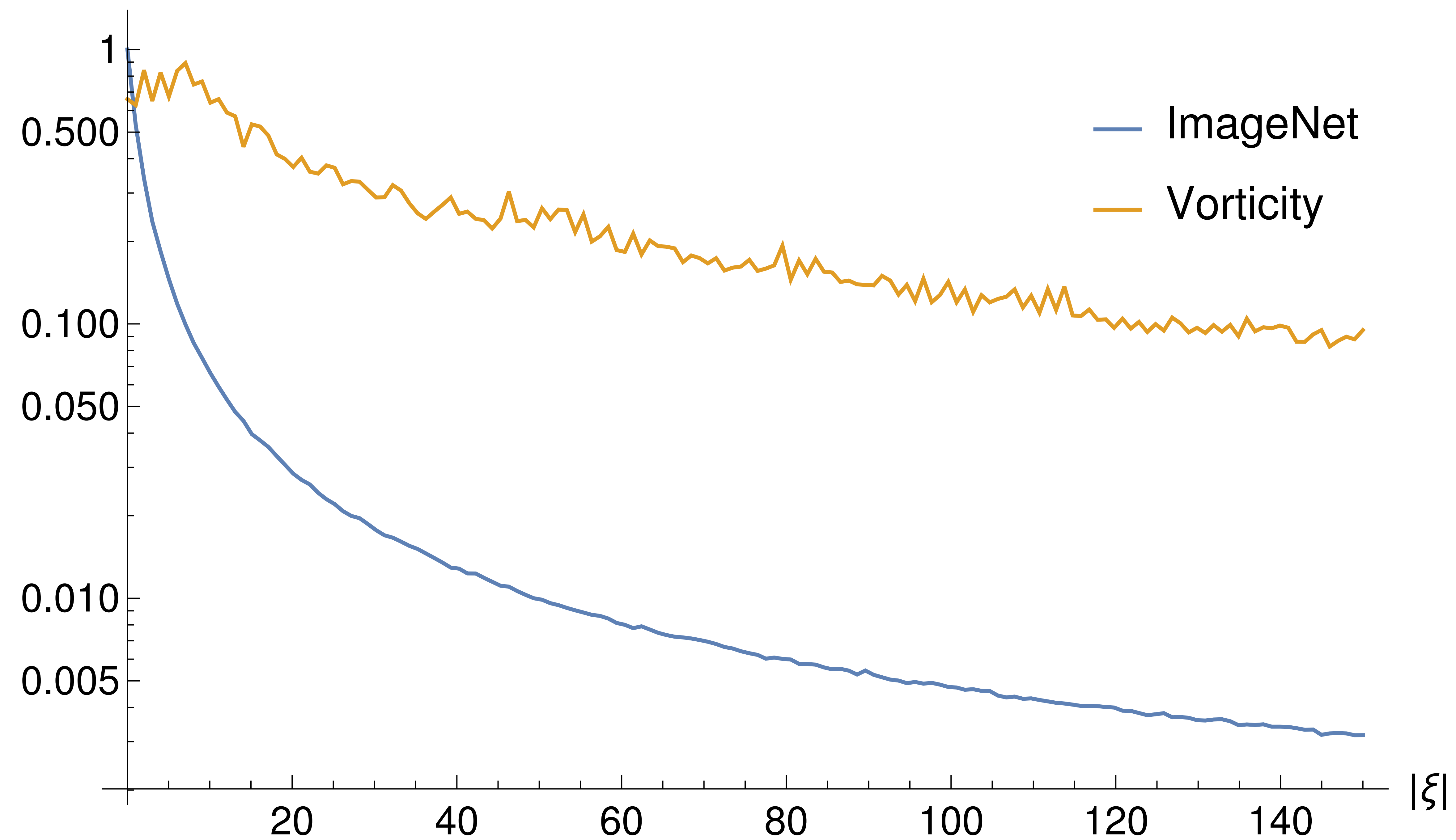
ERA5 versus ImageNet



ERA5 versus ImageNet



AtmoRep data



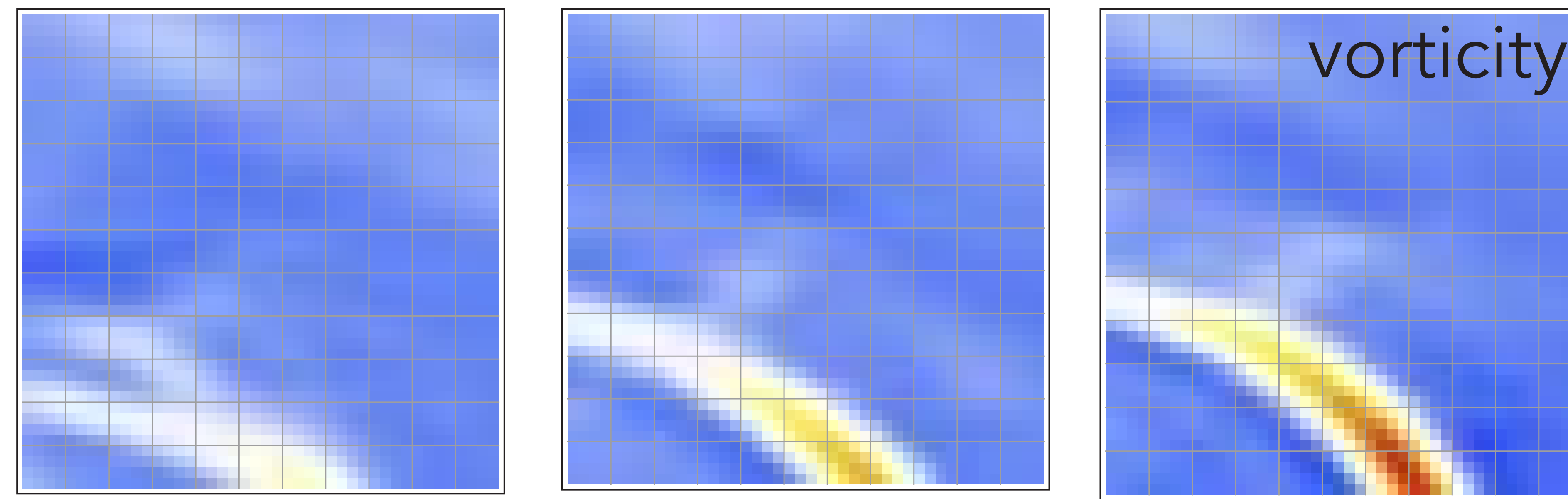
Statistical loss

- Attention maps:

attention maps



vorticity



$t-2$

$t-1$

t

Statistical loss

- Statistical loss:

$$\mathcal{L}_{\text{stats}} = \left| 1 - \int_{\mathbb{R}} \delta_y(x) G_{\tilde{\mu}, \tilde{\sigma}}(x) \, dx \right|^2$$

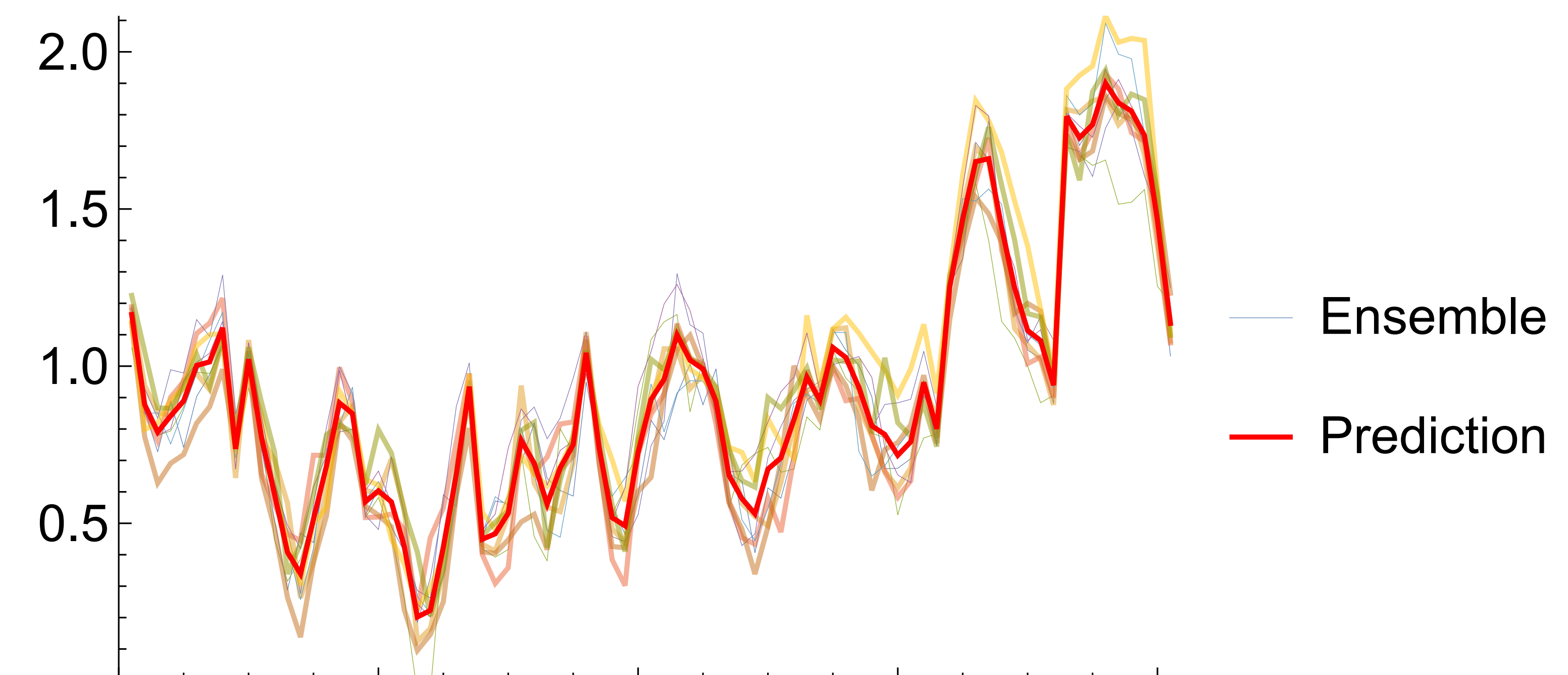
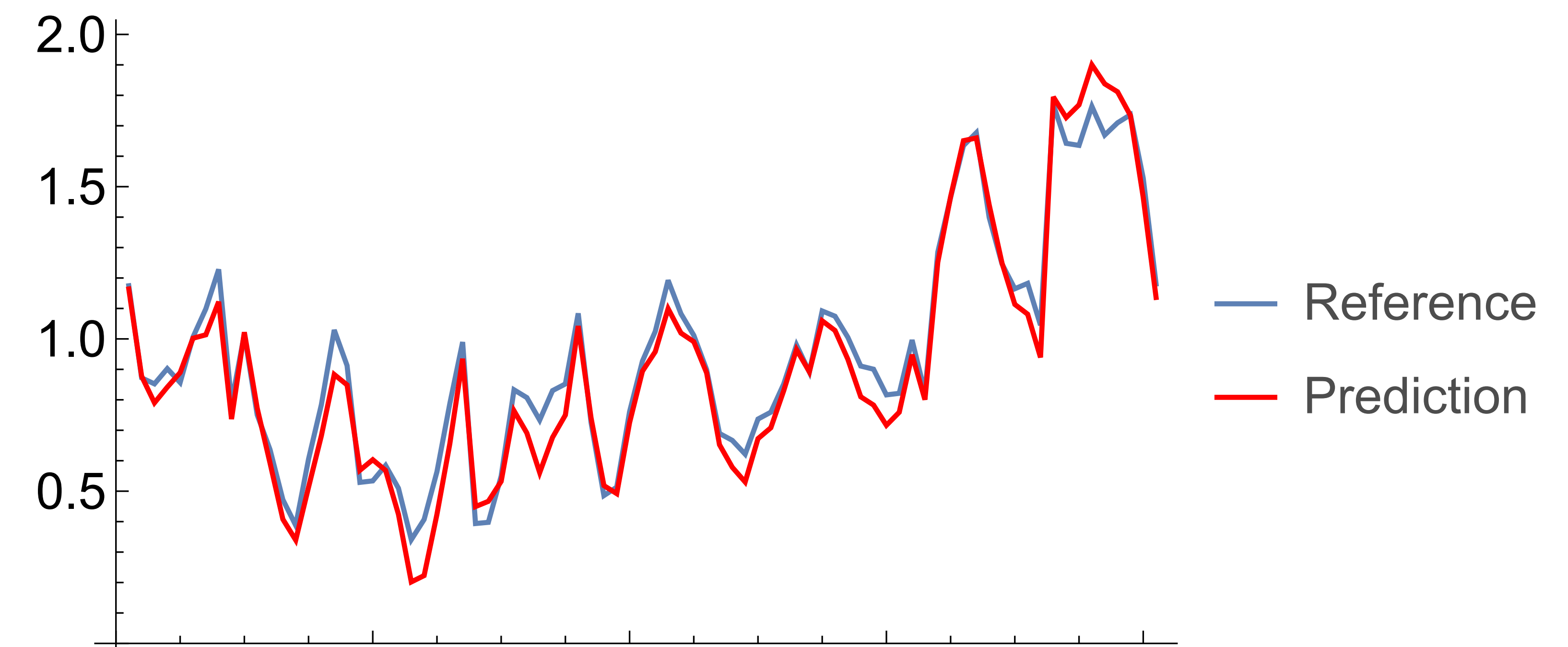
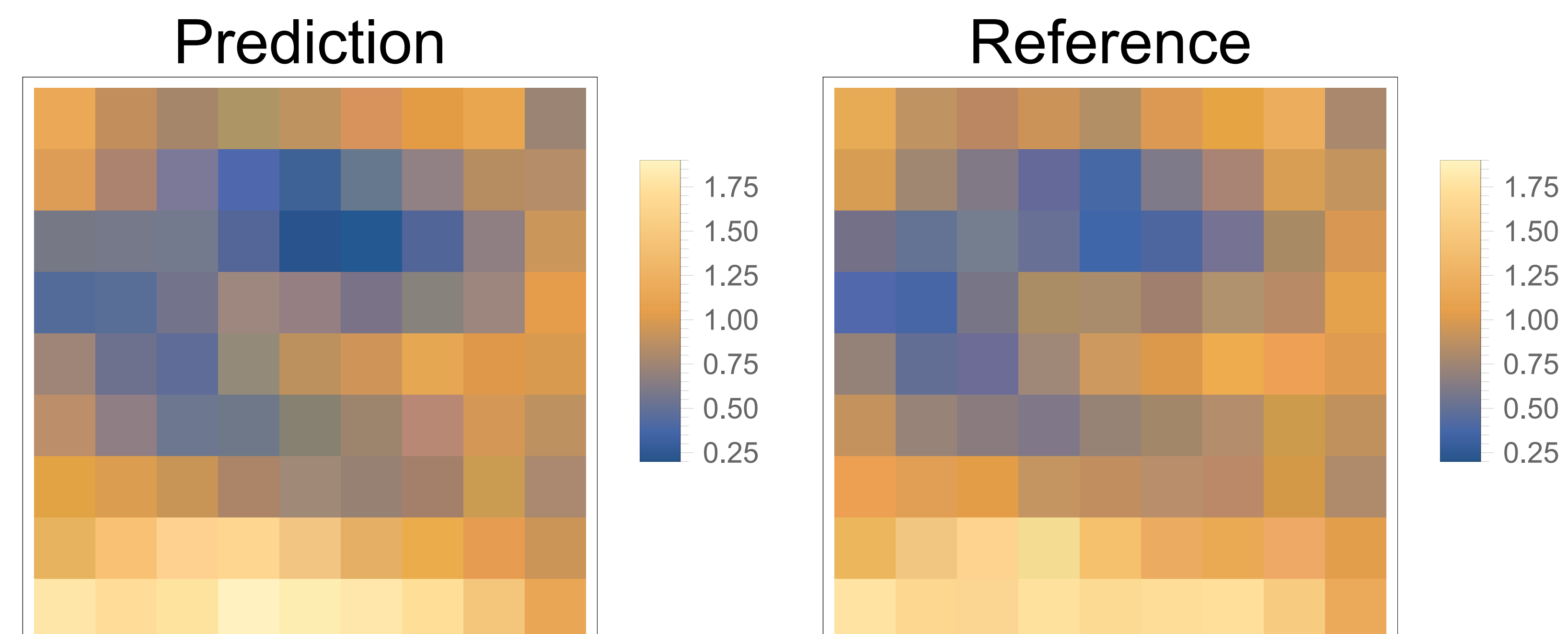
- CRPS:¹

$$\mathcal{L}_{\text{CRPS}} = \int_{\mathbb{R}} \left| H_y(x) \operatorname{erf}_{\tilde{\mu}, \tilde{\sigma}}(x) \right|^2 \, dx$$

¹ S. Rasp and S. Lerch. Neural networks for postprocessing ensemble weather forecasts. Monthly Weather Review, 146(11):3885 – 3900, 2018.

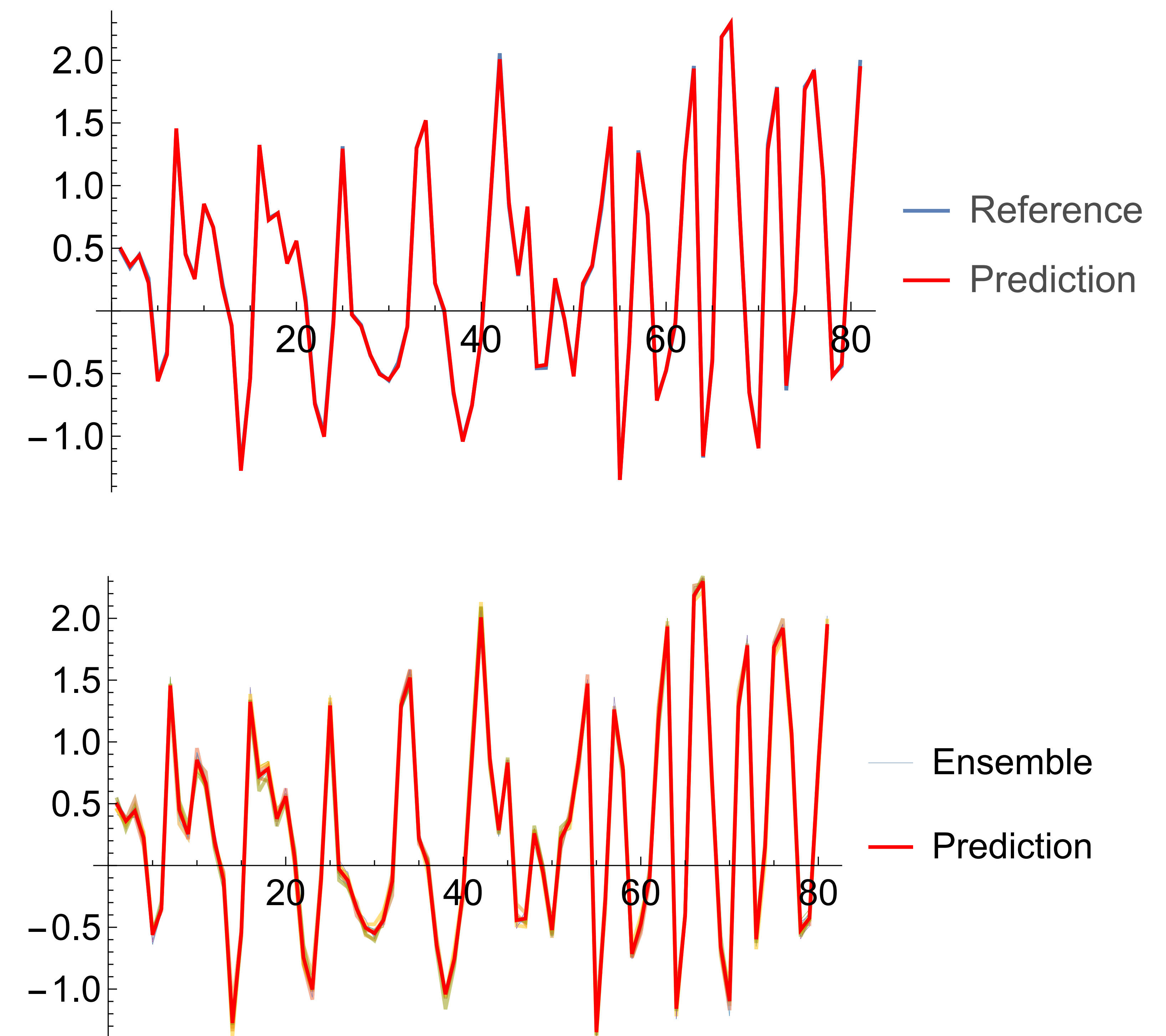
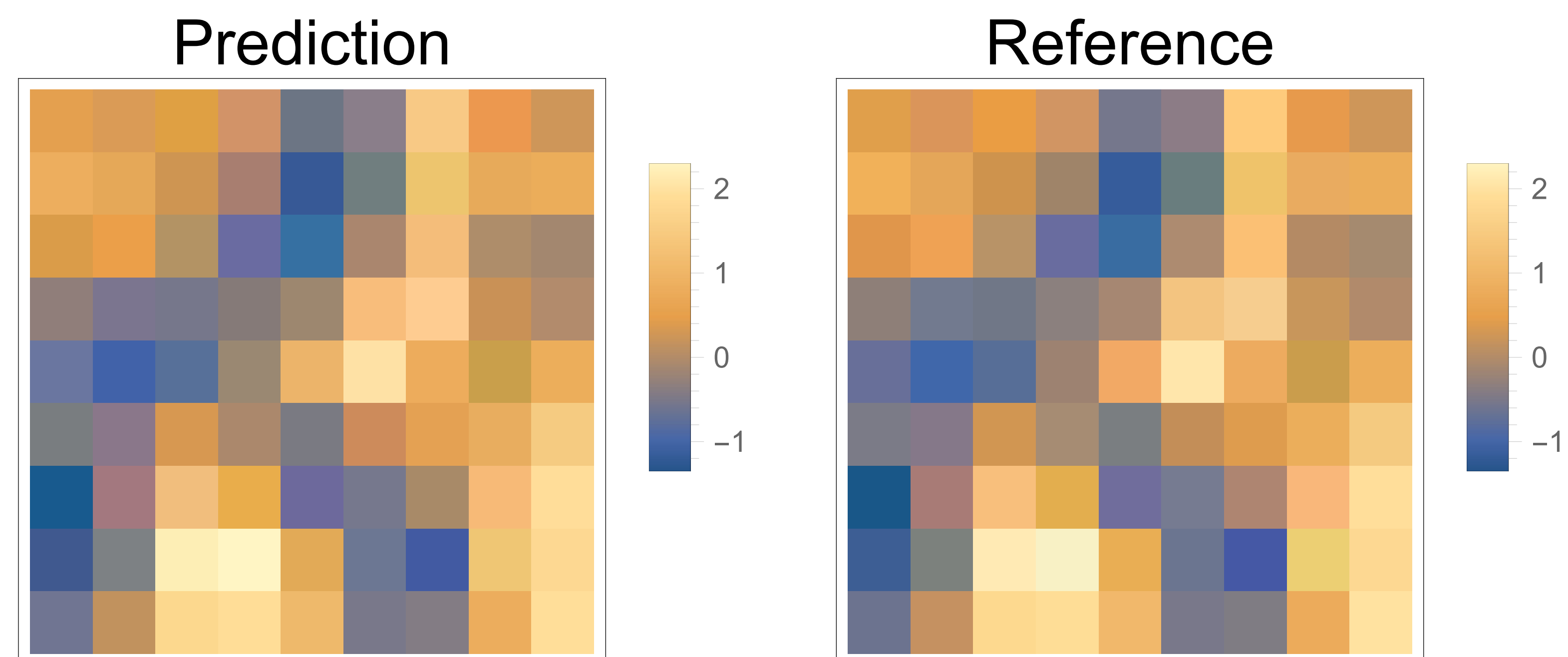
Statistical loss

- Predictions:



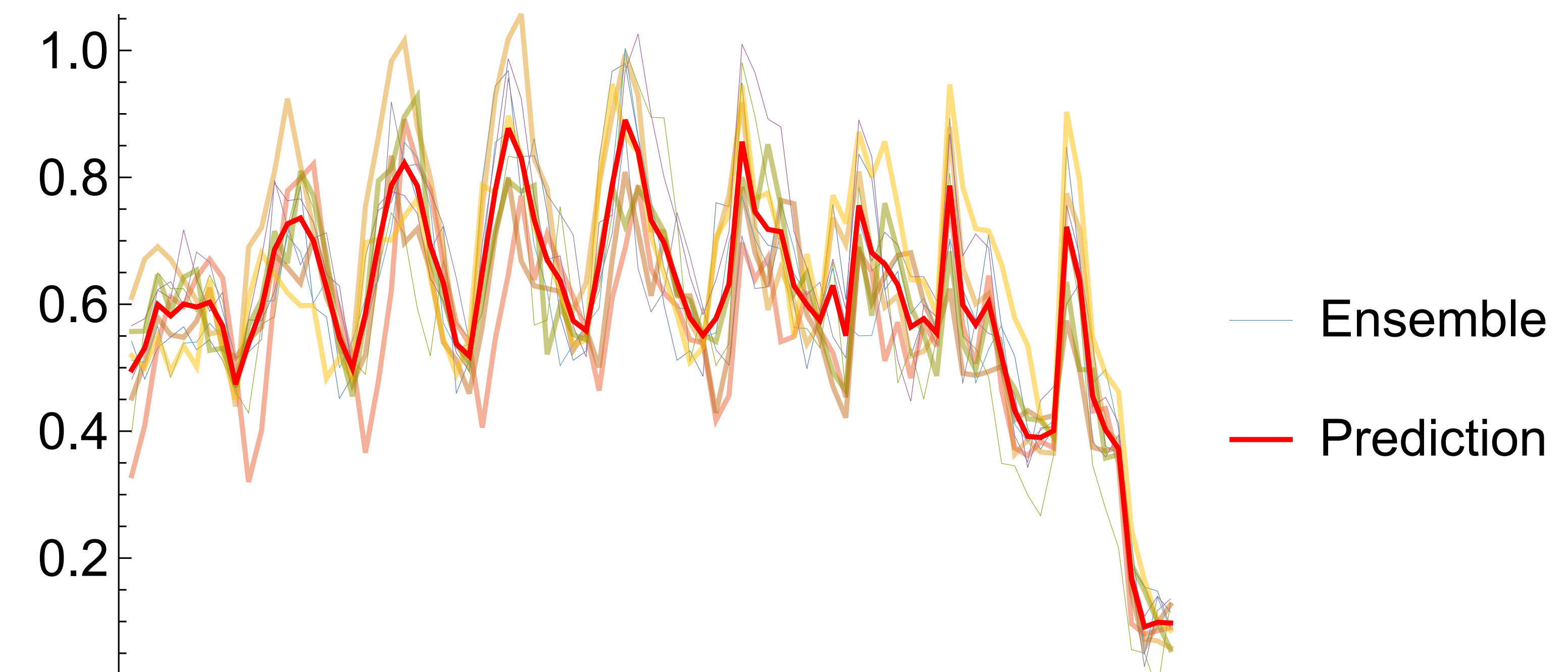
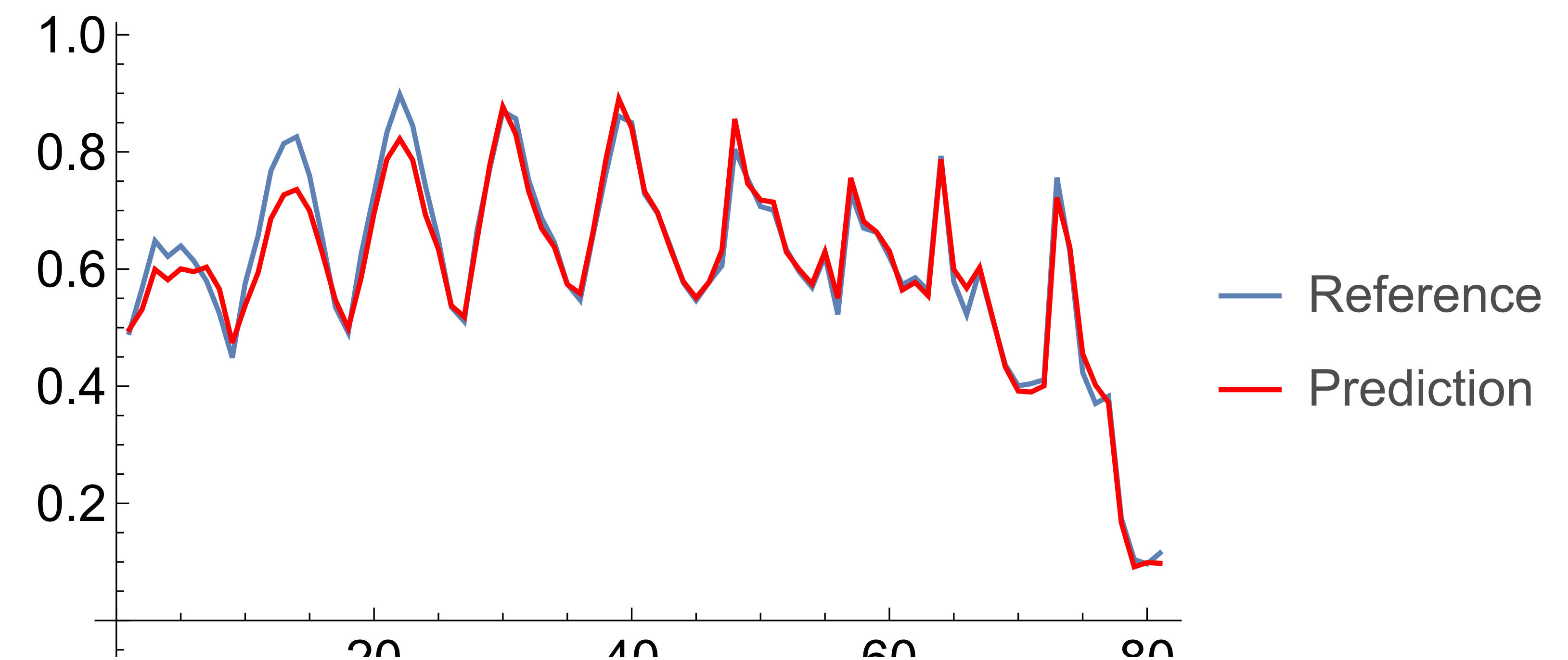
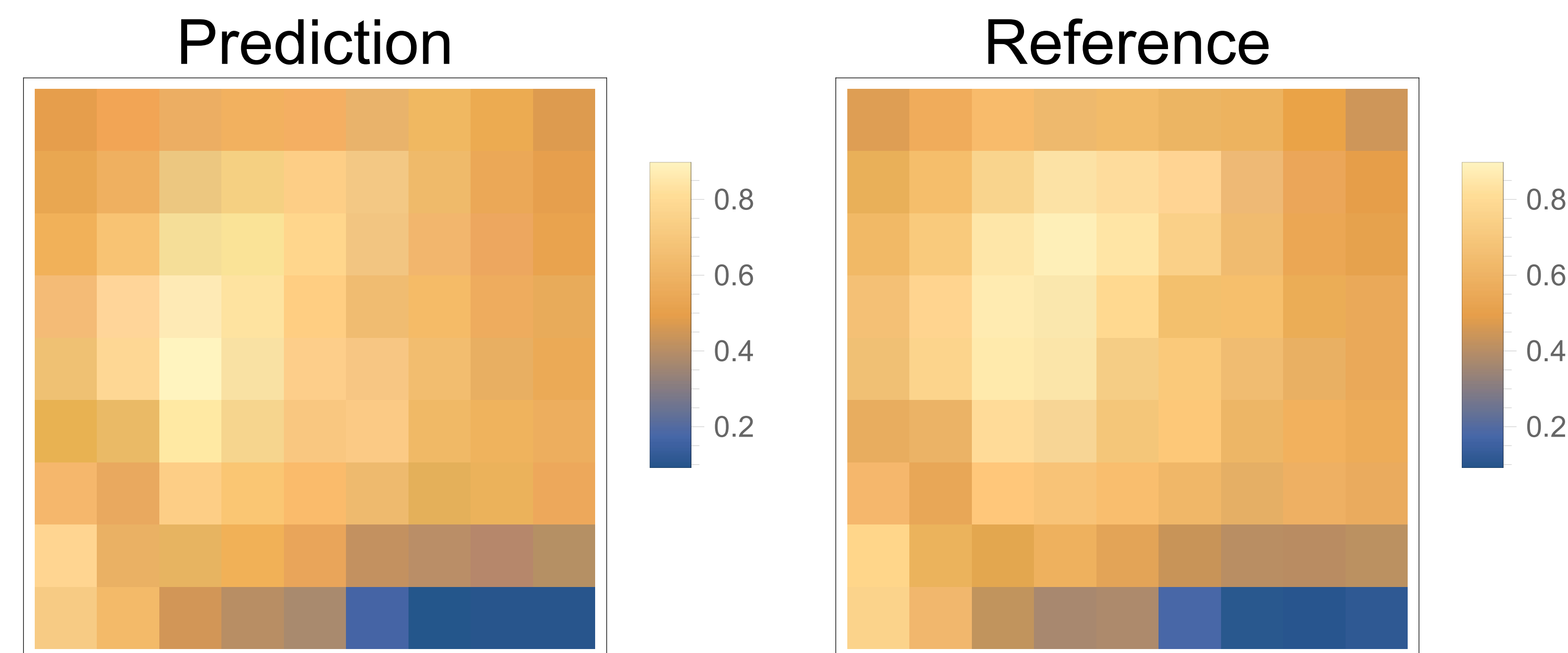
Statistical loss

- Predictions:

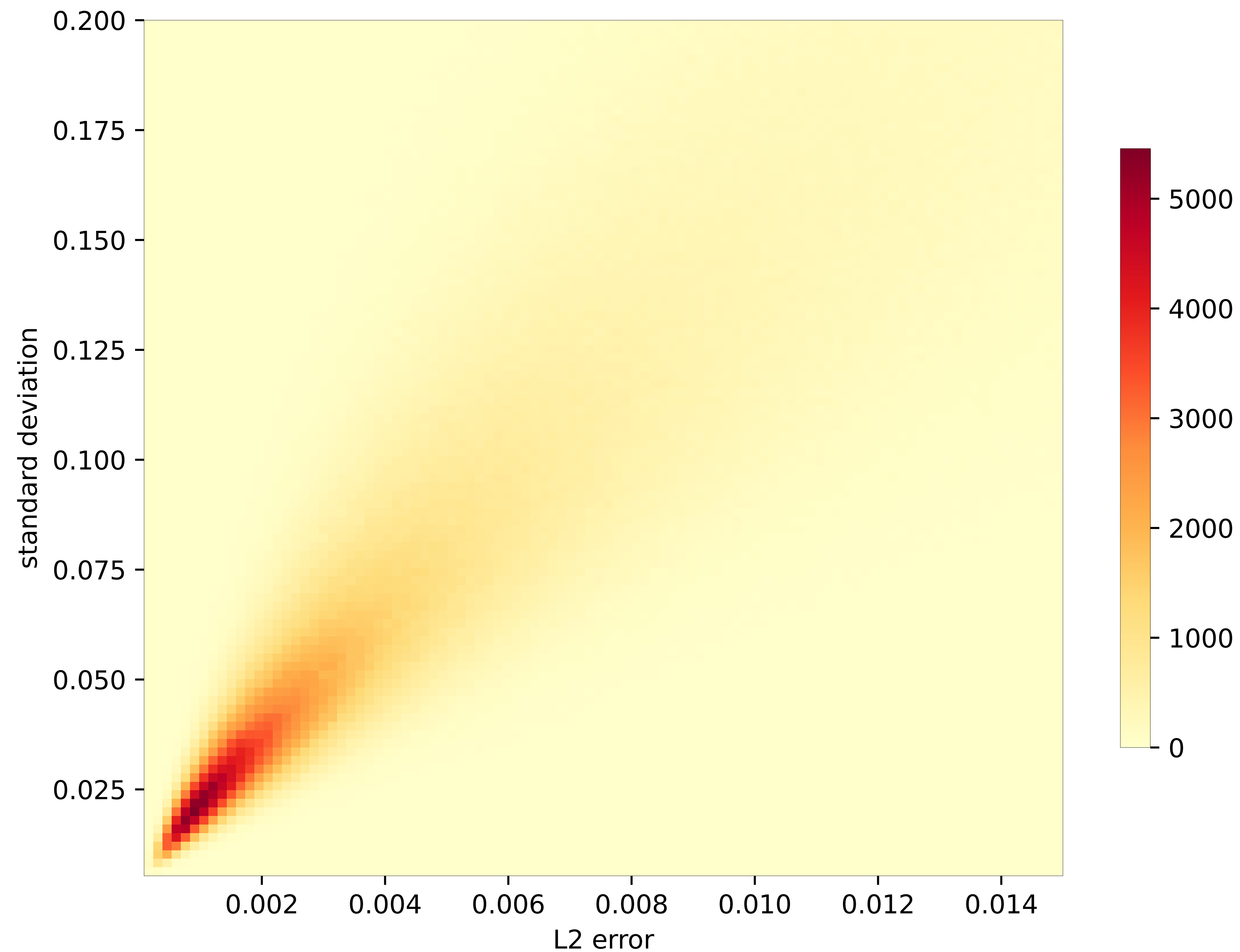


Statistical loss

- Predictions:



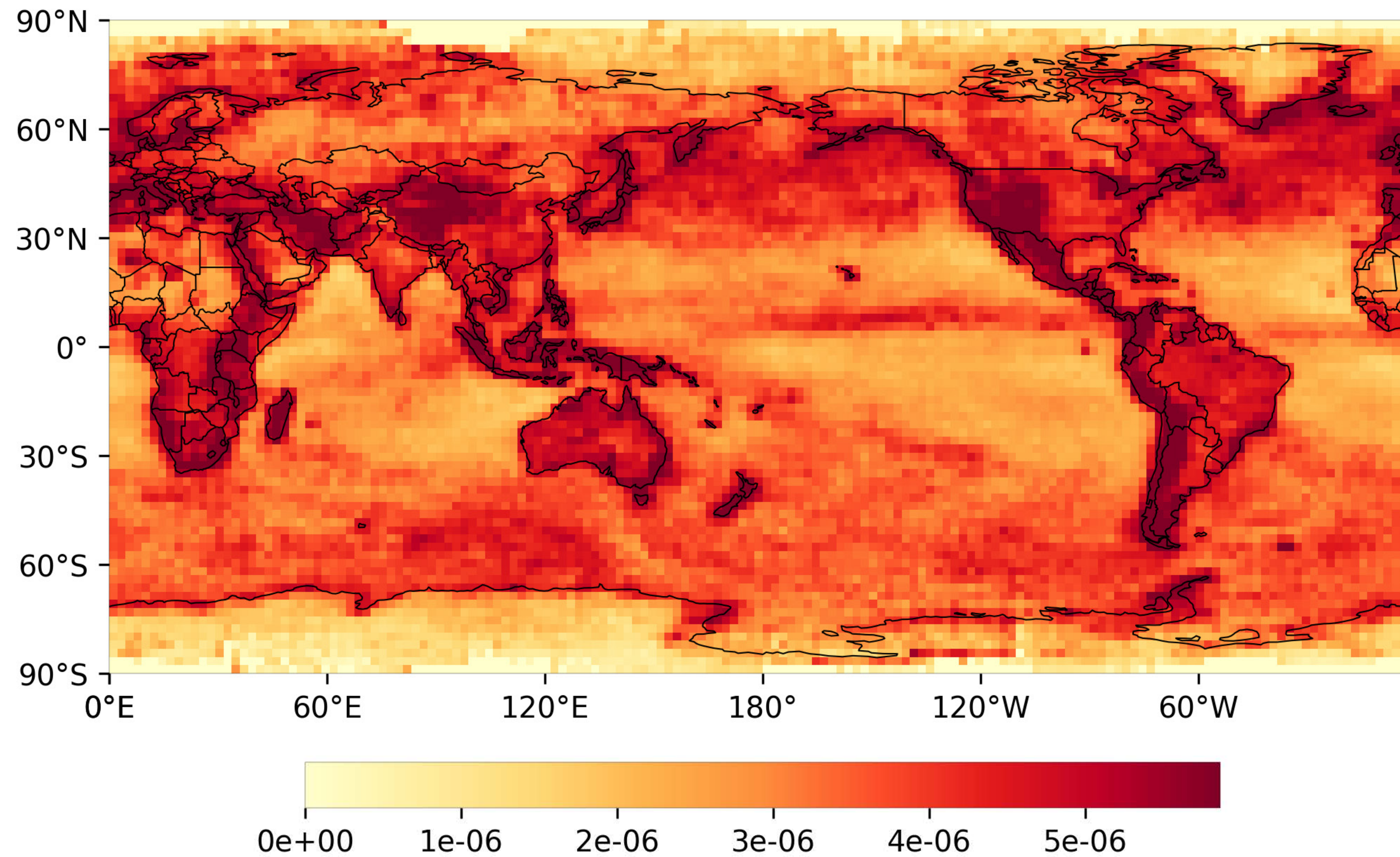
Statistical loss



2D Histogram
of L_2 error vs.
std. dev.
(temperature)

Statistical loss

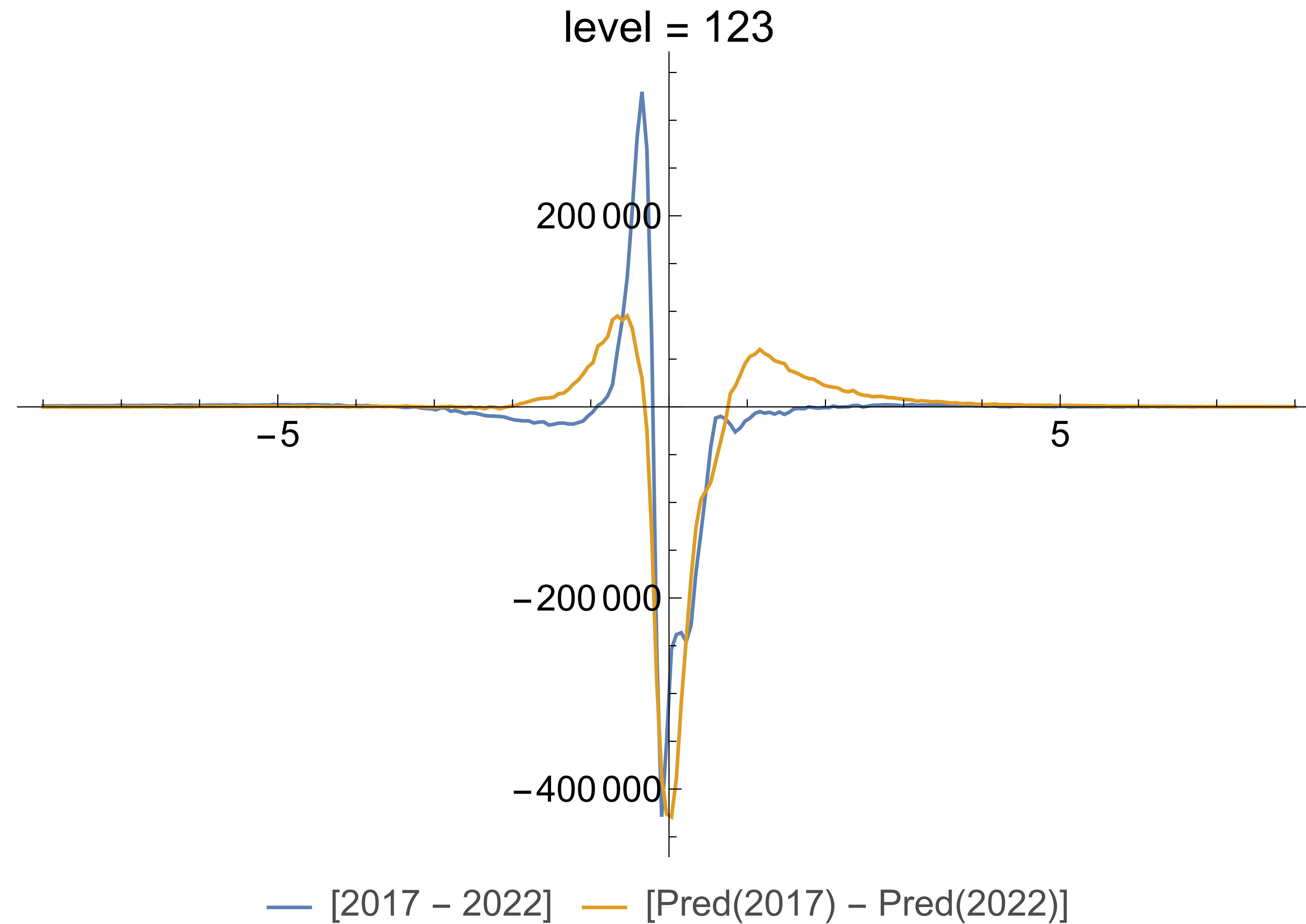
standard deviation (vorticity)



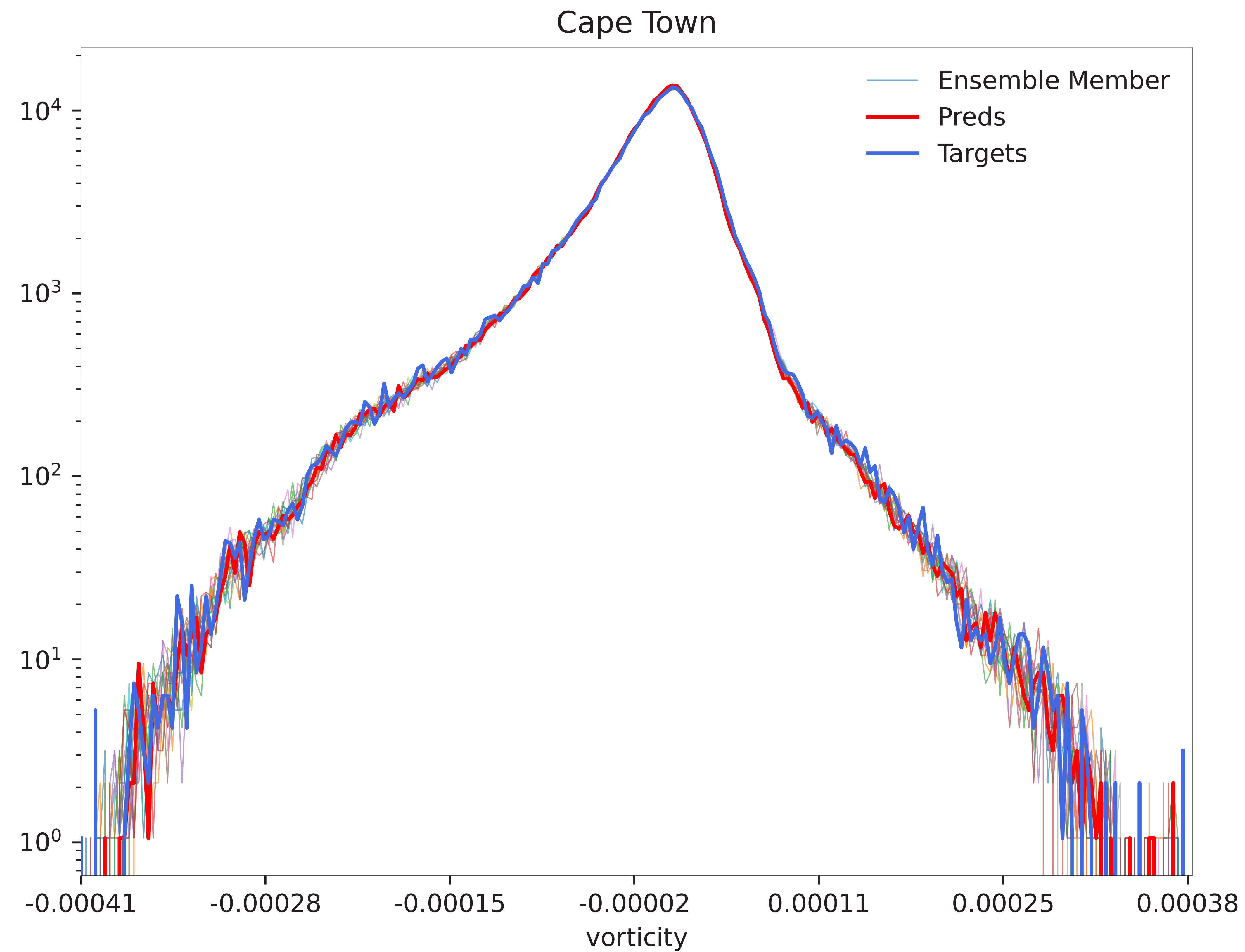
Some observations on training

- No overfitting
 - › No difference between train and test set even with few months and limited spatial domain
- Different fields behave rather differently
 - › Masking ratios have to be chosen differently
 - › Statistical loss performs differently
- MSE loss accross fields not comparable and misleading

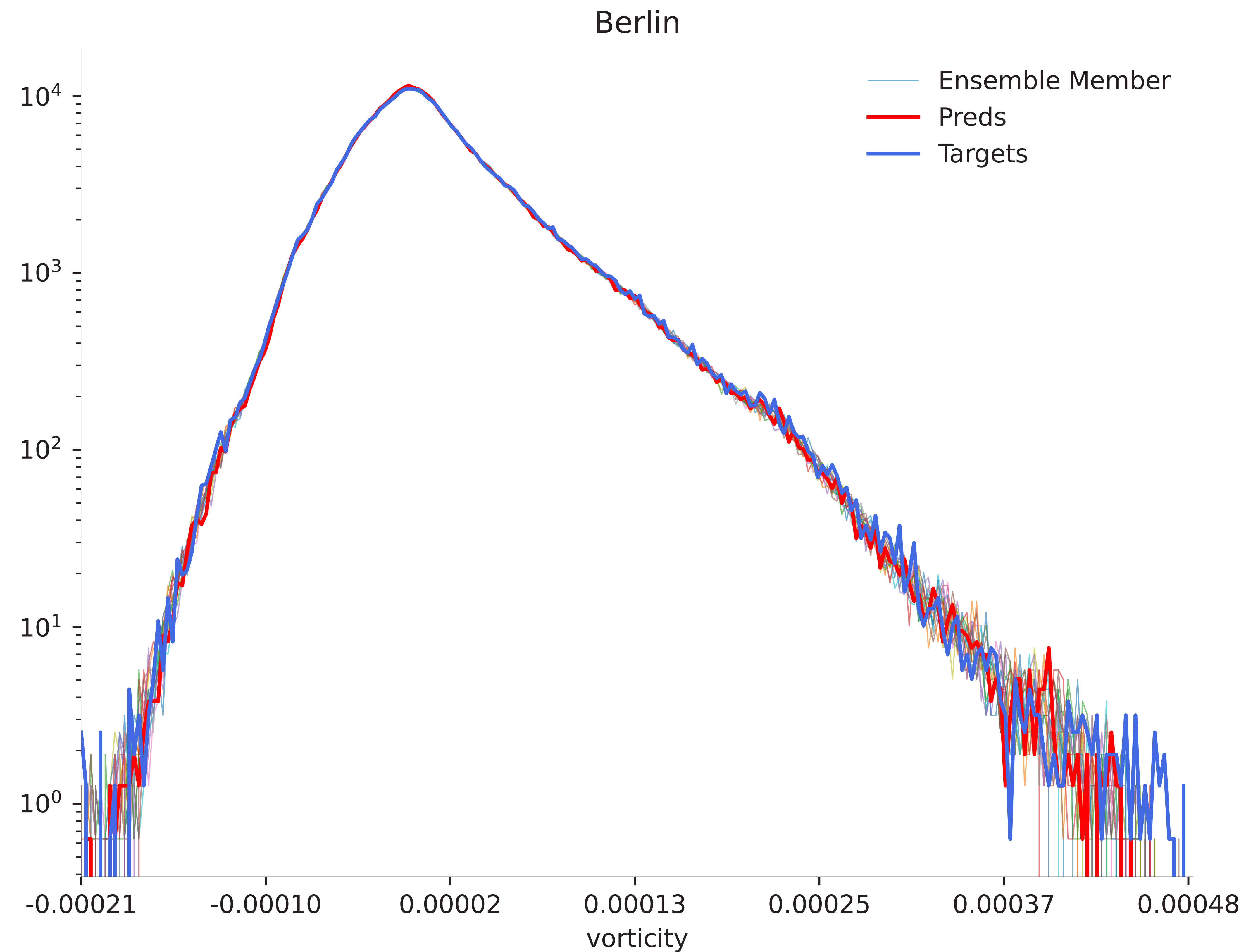
Counterfactuals: extrapolation



Pre-training results



Pre-training results



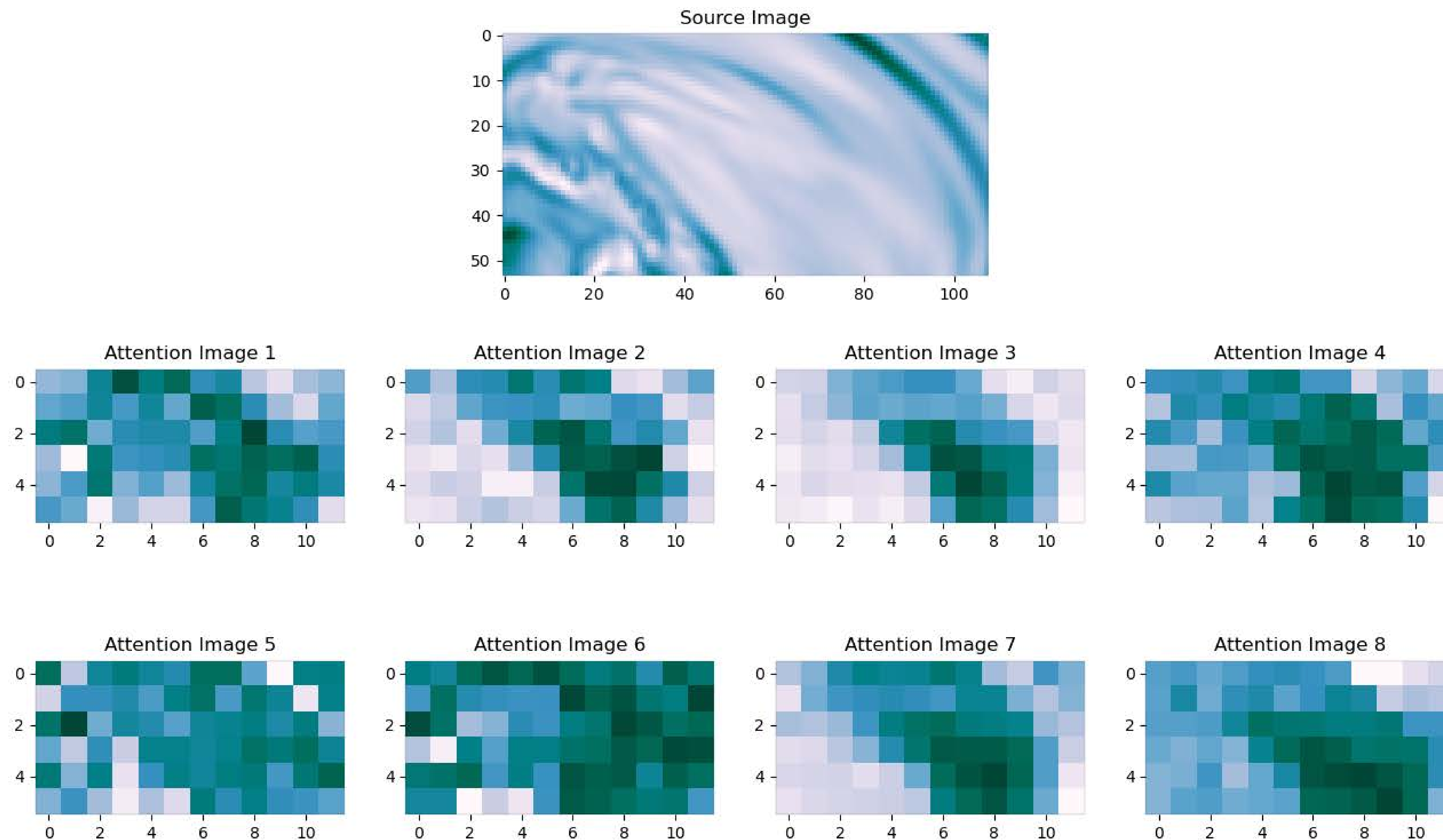
Training on observations

- Neural networks are models that work well on heterogeneous and noisy data
- Fine-tune/bias correct a pre-trained model with observations (instead of training from scratch)

Can one continuously update a model with observations?

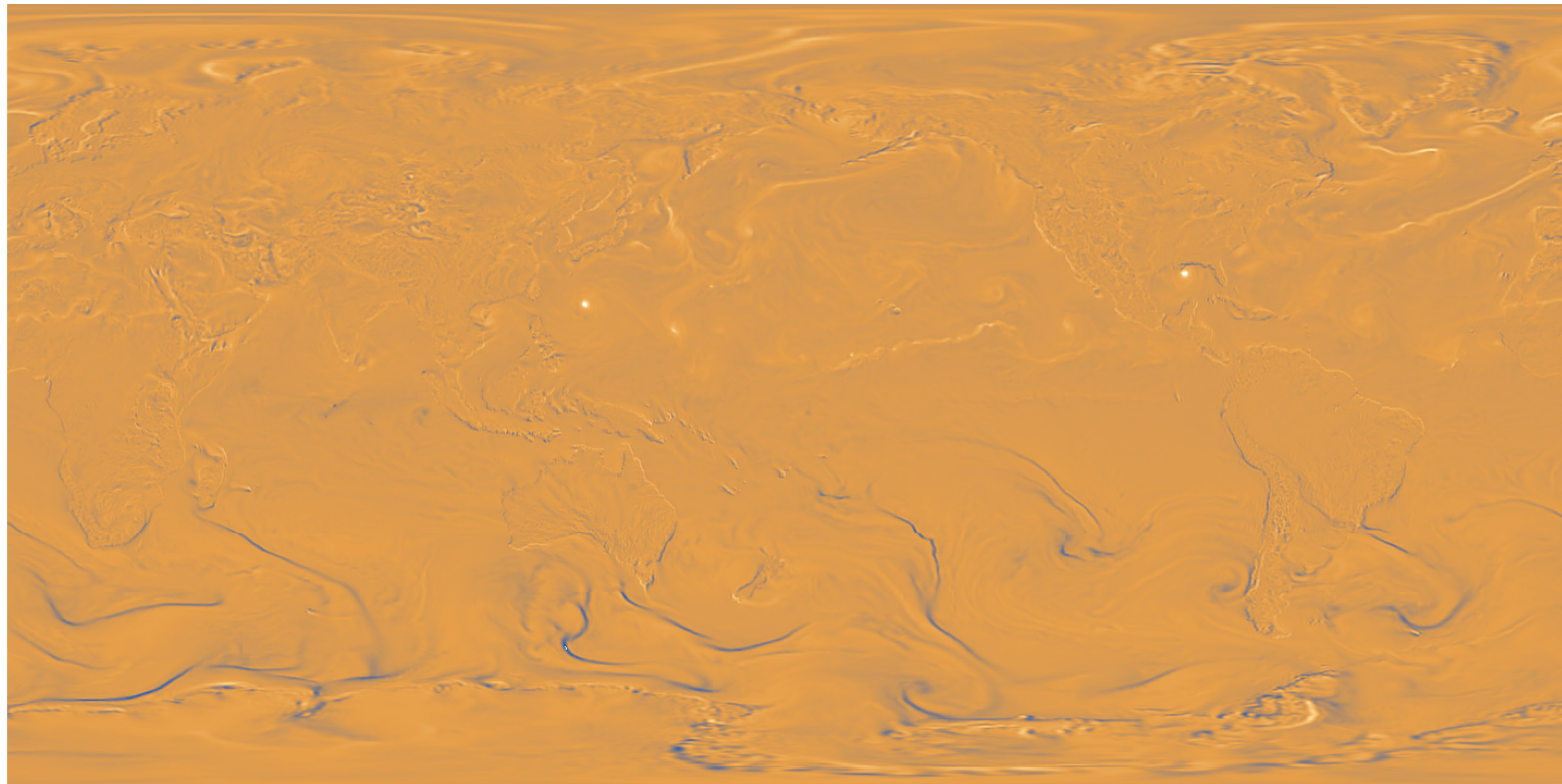
How to handle and propagate uncertainties?

Physics of trained network



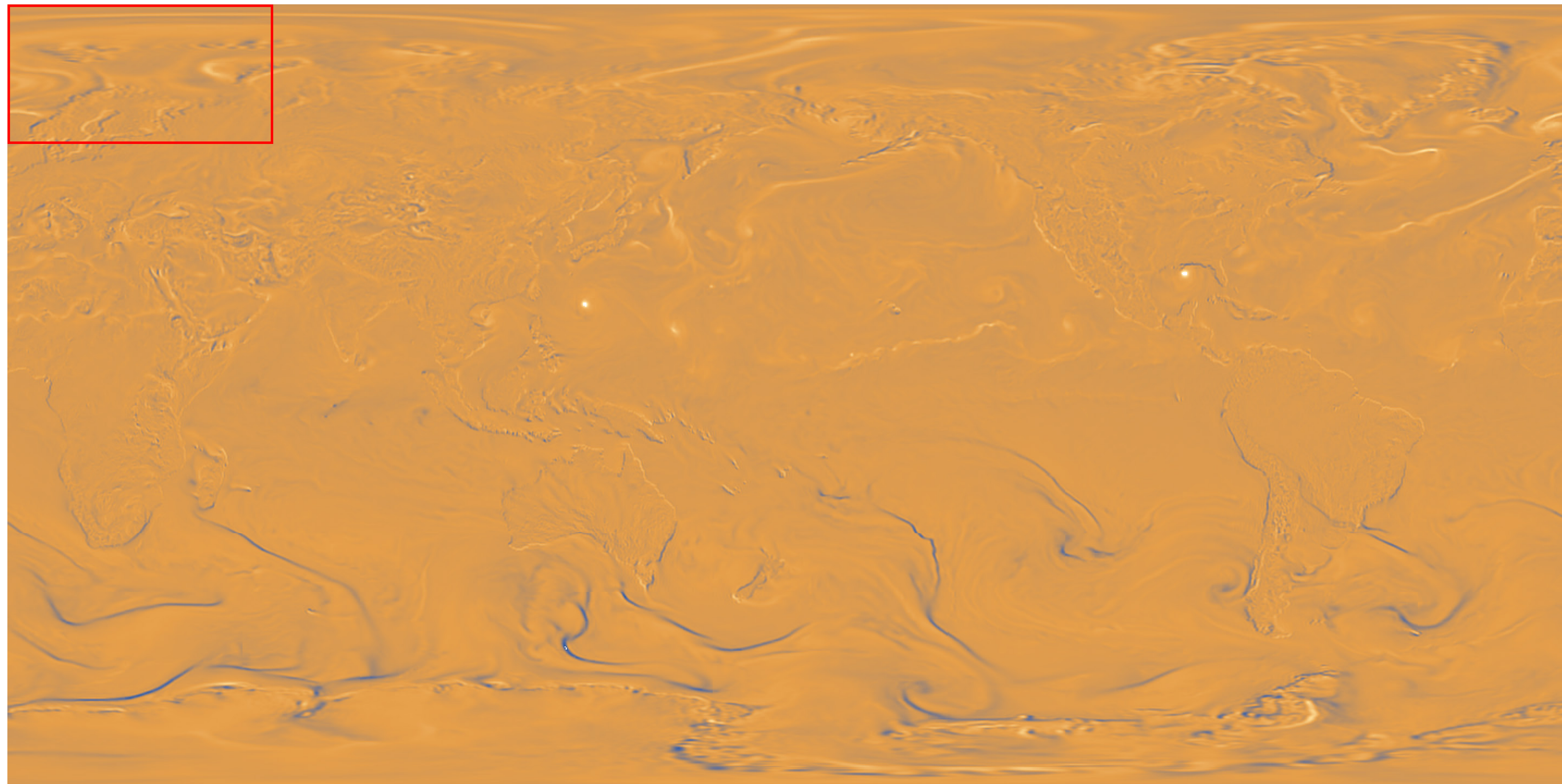
Medium range forecasting

- How to do global forecasts with a local model?



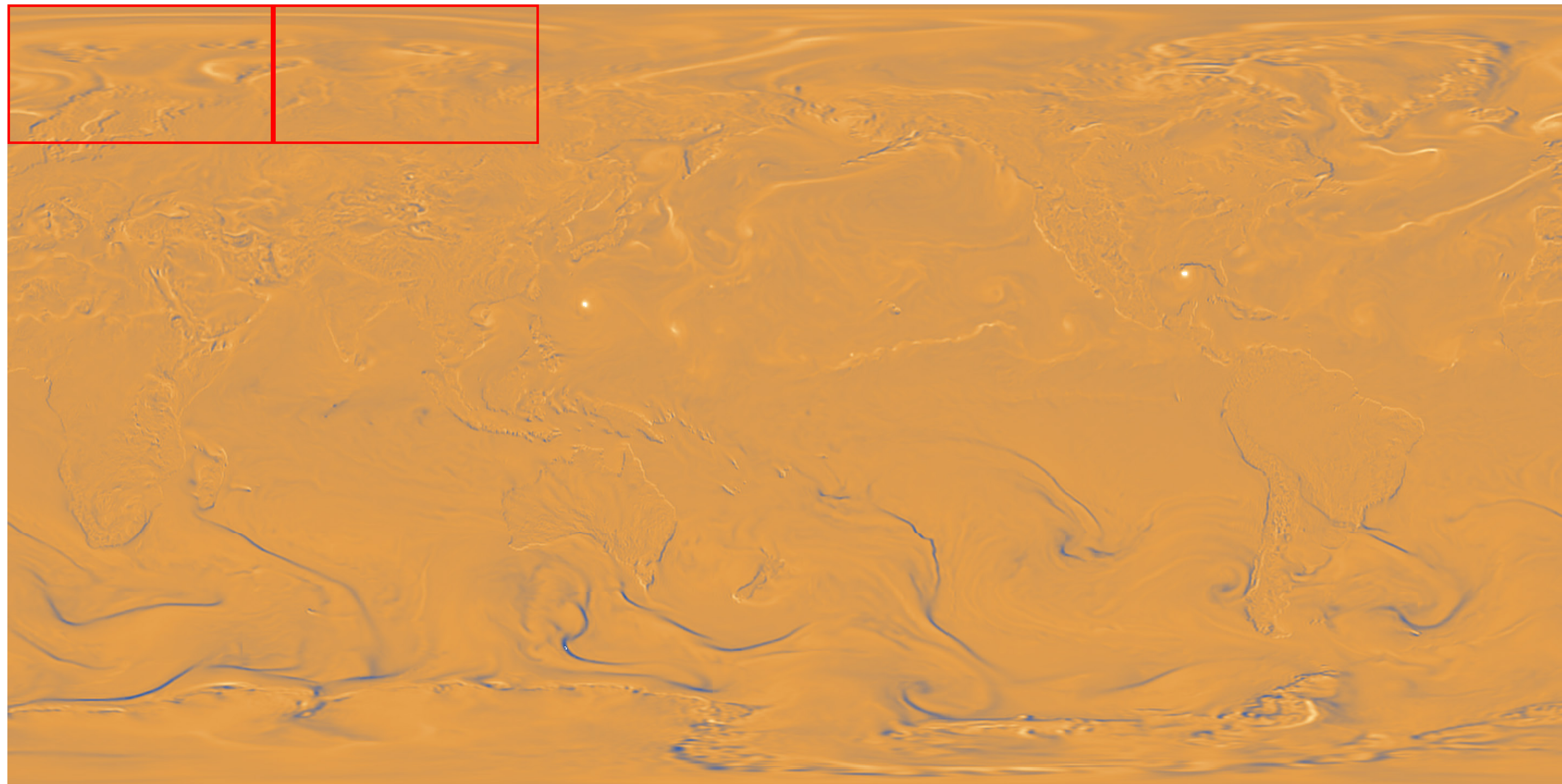
Medium range forecasting

- How to do global forecasts with a local model?



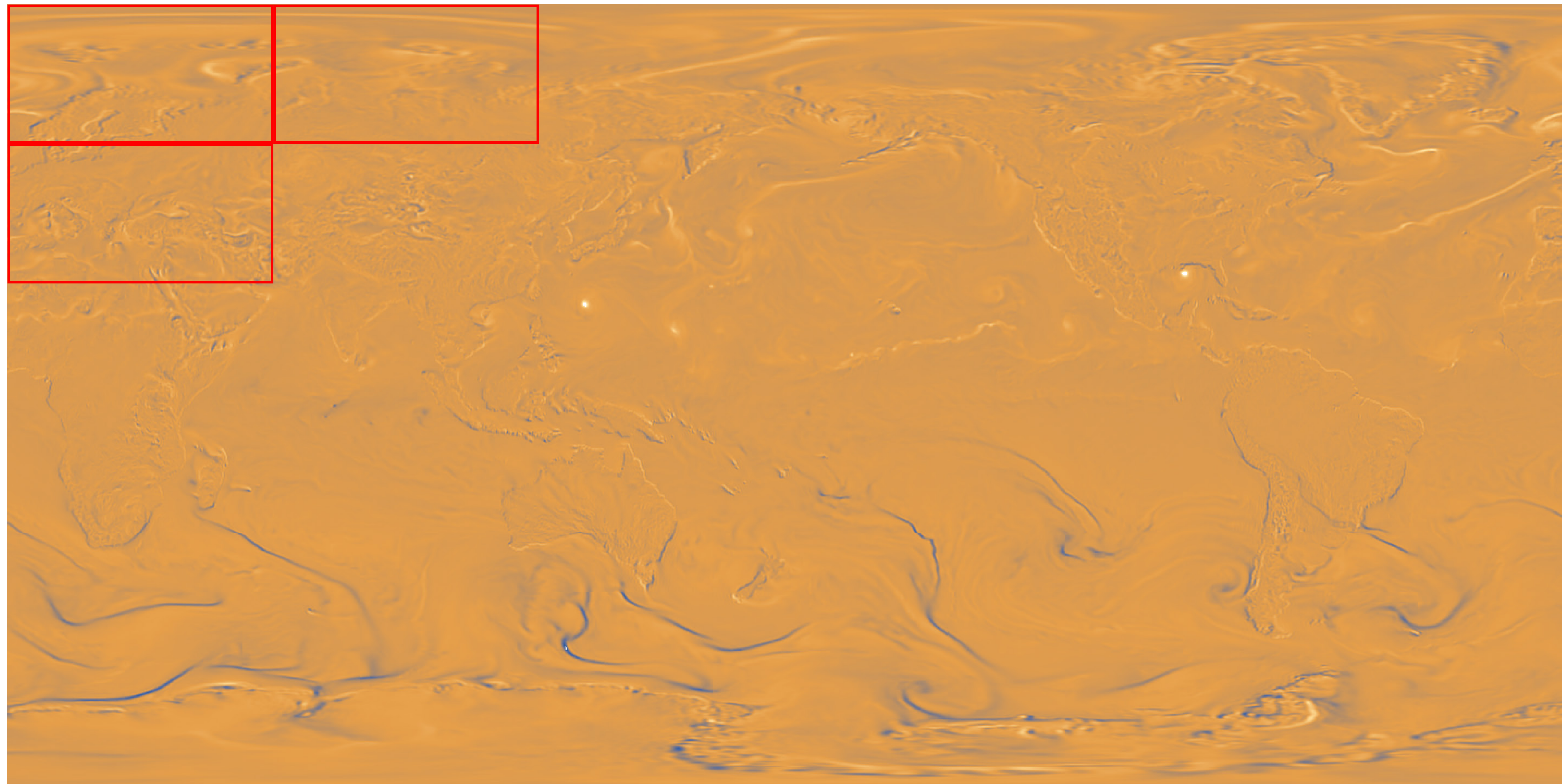
Medium range forecasting

- How to do global forecasts with a local model?



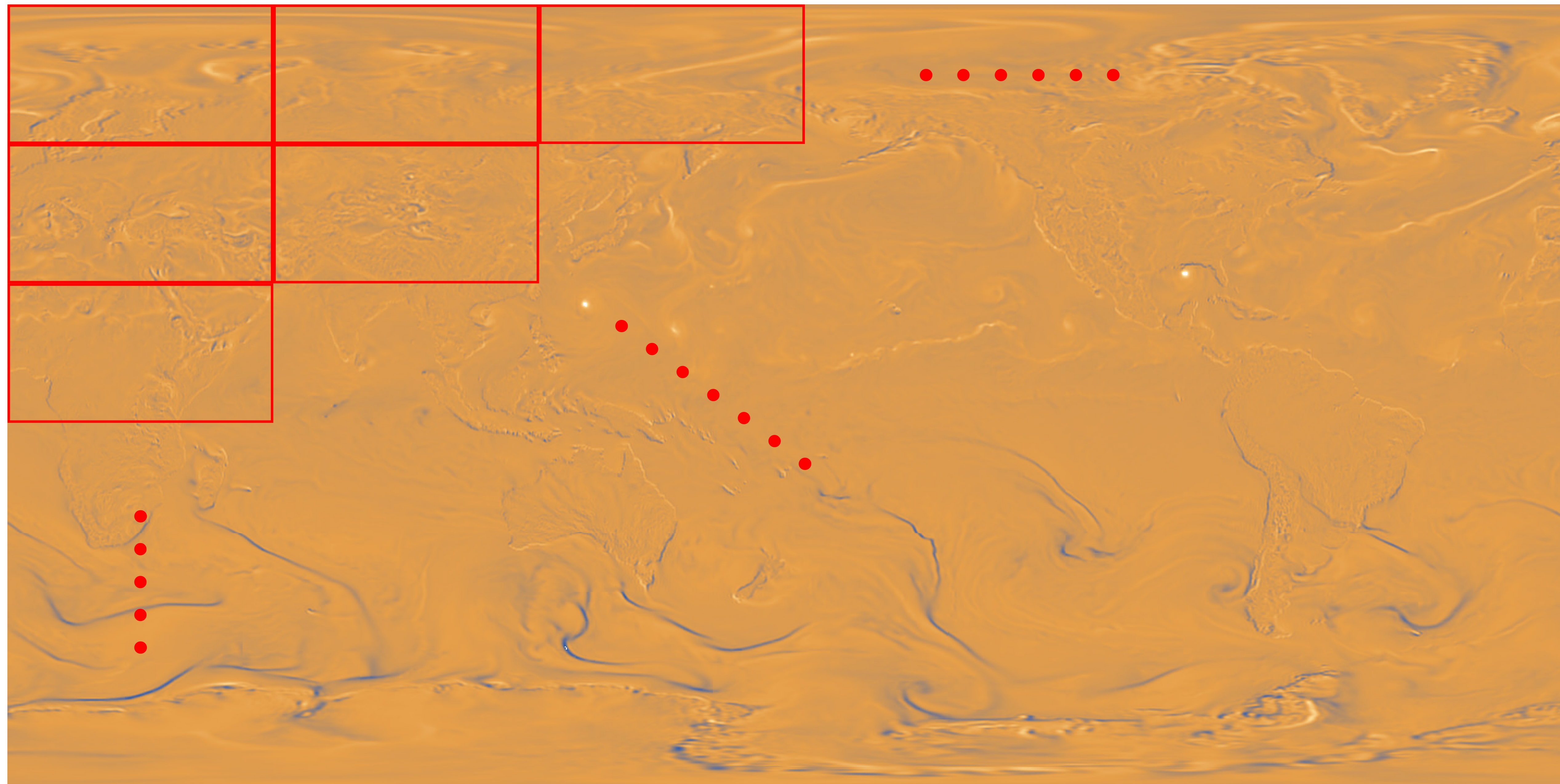
Medium range forecasting

- How to do global forecasts with a local model?



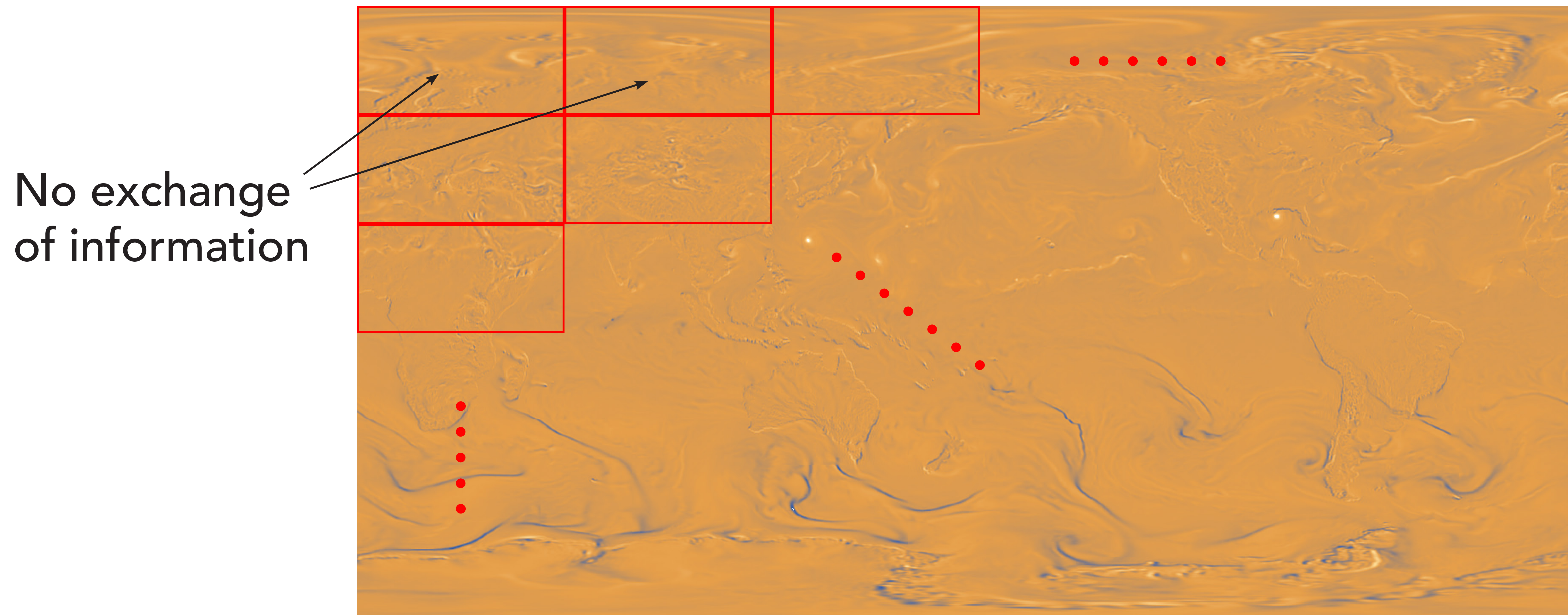
Medium range forecasting

- How to do global forecasts with a local model?



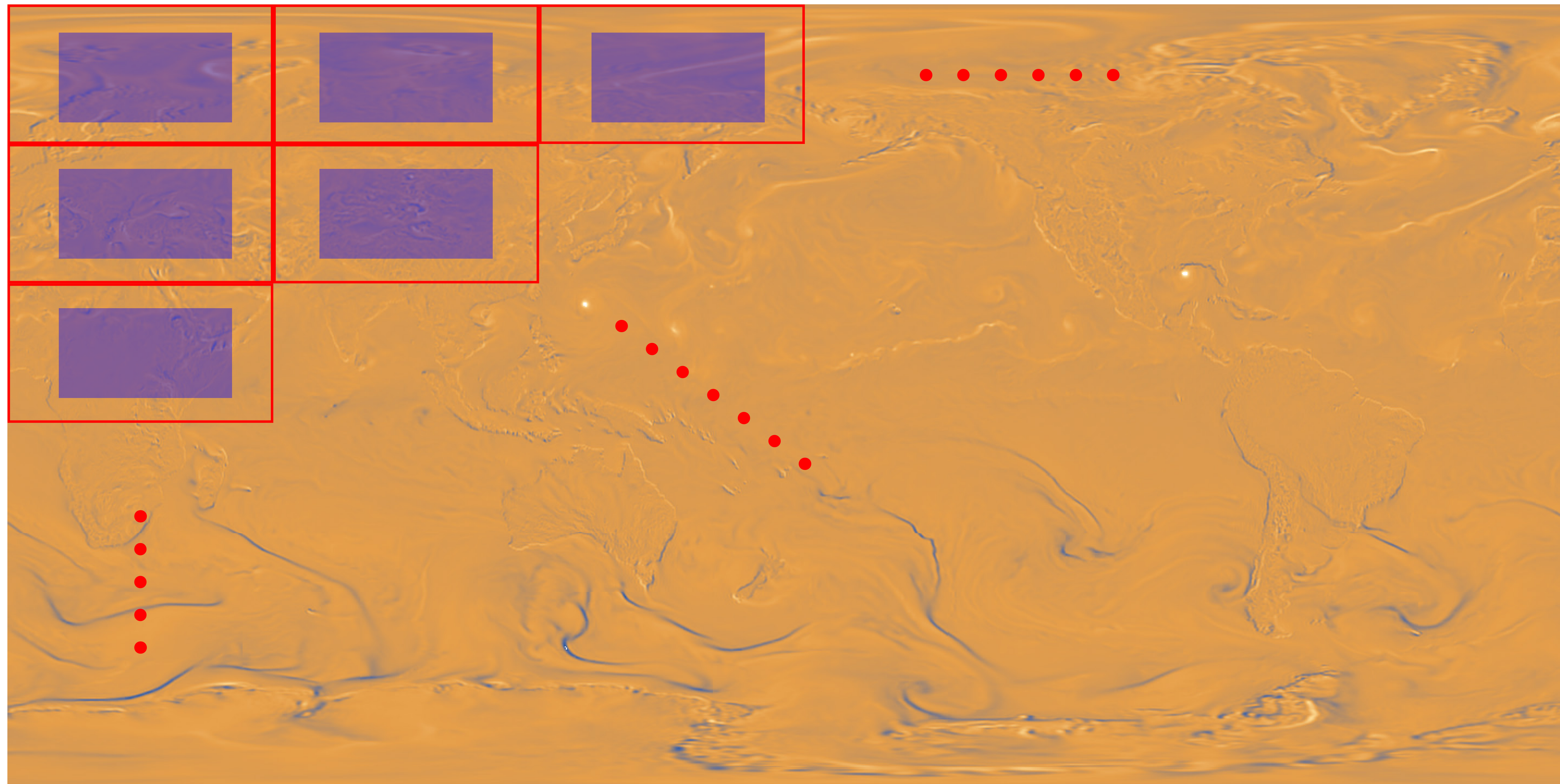
Medium range forecasting

- How to do global forecasts with a local model?



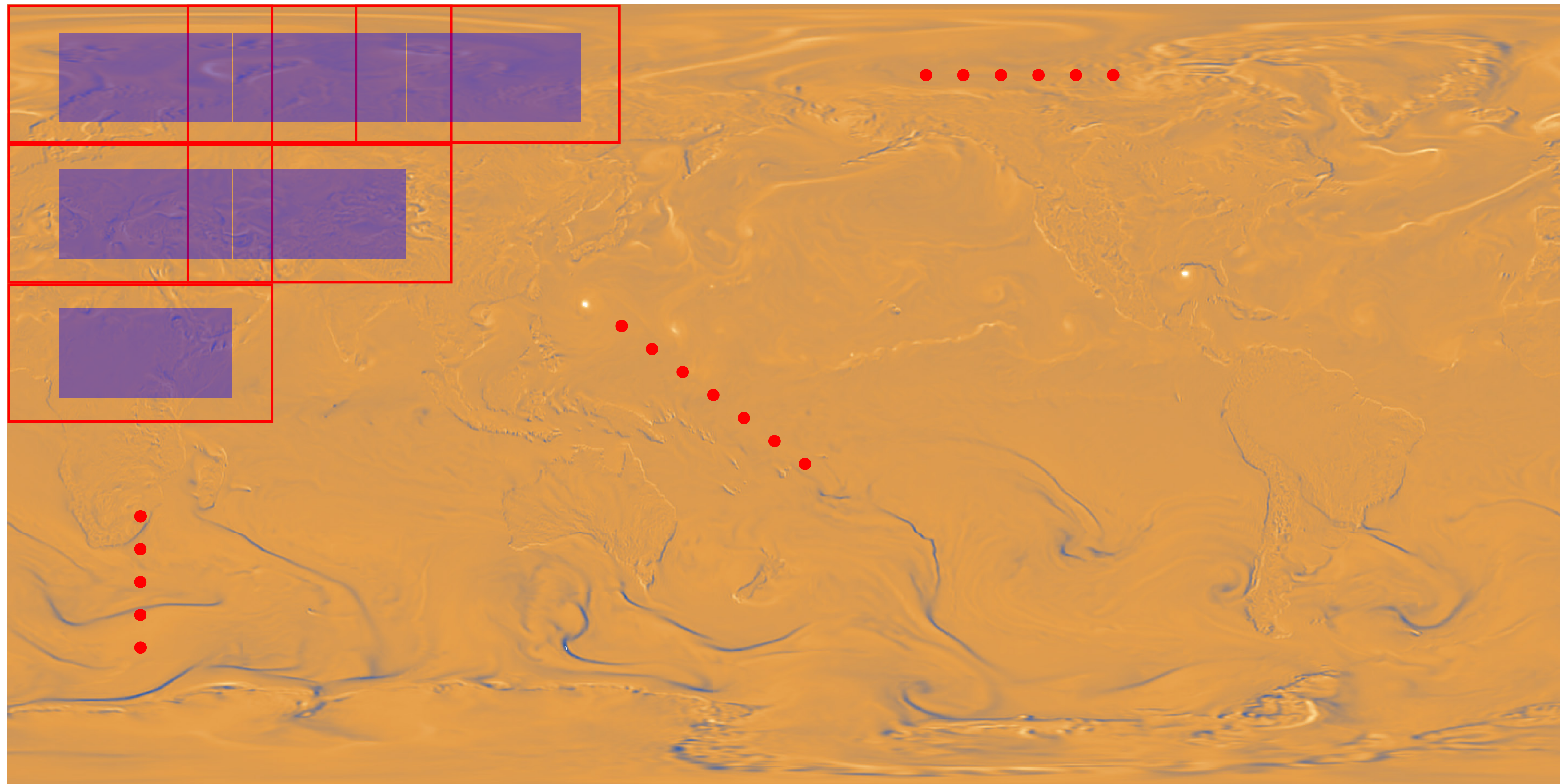
Medium range forecasting

- How to do global forecasts with a local model?



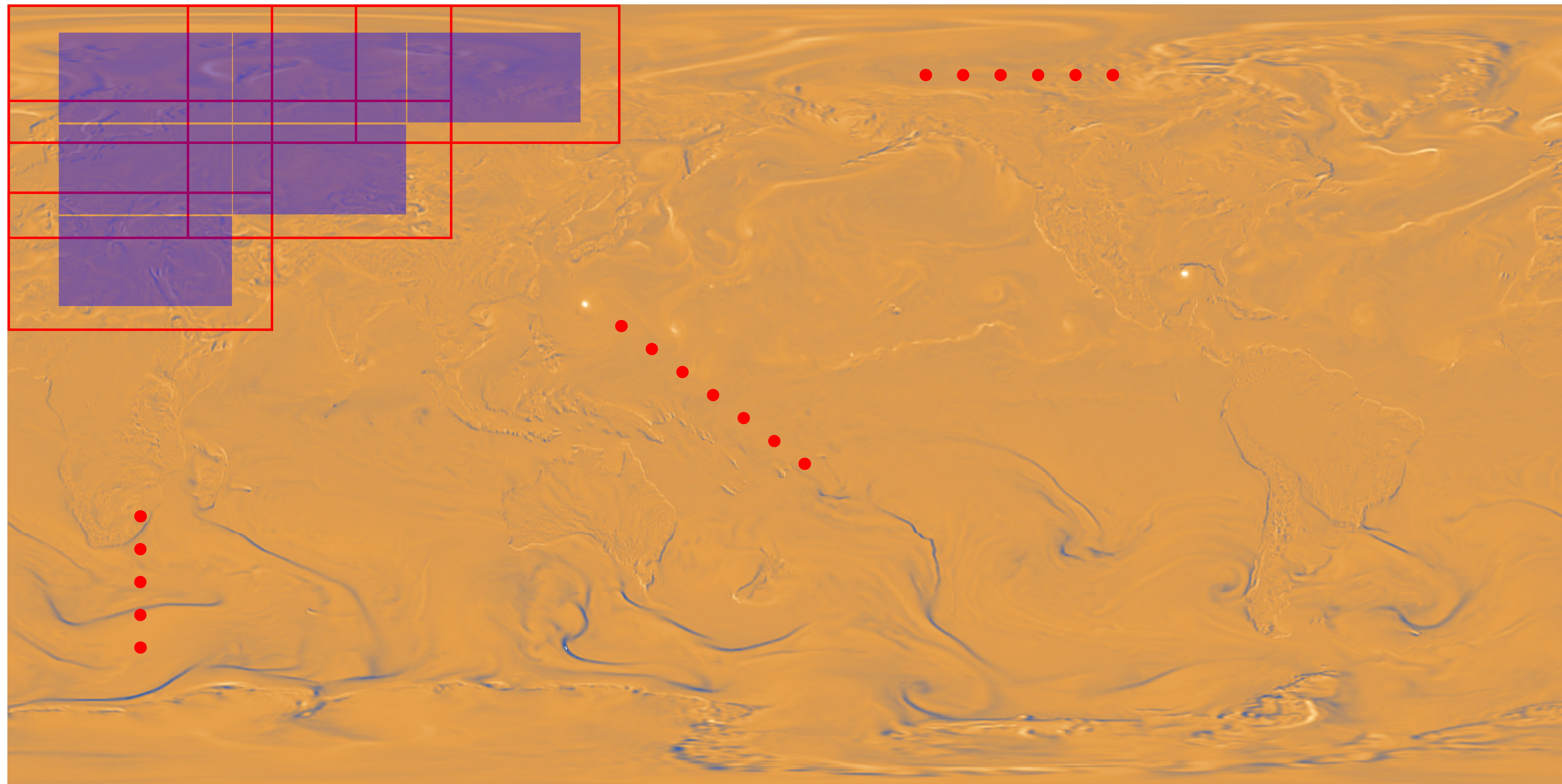
Medium range forecasting

- How to do global forecasts with a local model?



Medium range forecasting

- How to do global forecasts with a local model?



Multiformer: standard configuration

- Per token: $\alpha = (\text{year}, \text{day}, \text{hour}, \text{ml}, \theta, \phi, \text{res})$

Multiformer: standard configuration

- Per token: $\alpha = (\text{year}, \text{day}, \text{hour}, \text{ml}, \theta, \phi, \text{res})$
- Neighborhood: 36 h x 1350 km x 2700 km
 - › 12 x 6 x 12 tokens with 3 x 9 x 9 grid points

Multiformer: standard configuration

- Per token: $\alpha = (\text{year}, \text{day}, \text{hour}, \text{ml}, \theta, \phi, \text{res})$
- Neighborhood: 36 h x 1350 km x 2700 km
 - 12 x 6 x 12 tokens with 3 x 9 x 9 grid points
- Fields: vorticity, divergence, vertical velocity, temperature, specific humidity, total precipitation, orography
- Vertical model levels: 96, 105, 114, 123, 137

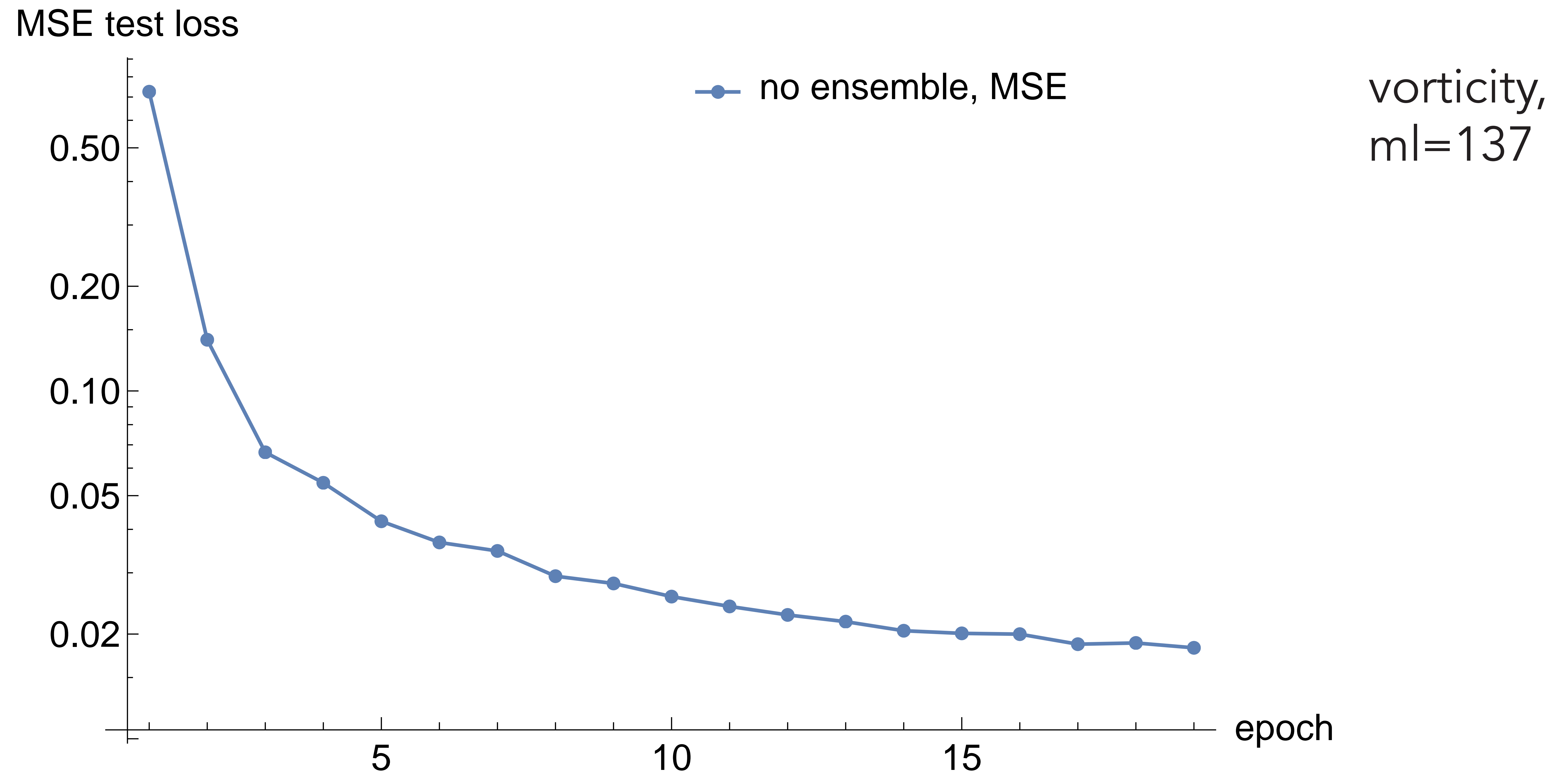
Multiformer: standard configuration

- Per token: $\alpha = (\text{year}, \text{day}, \text{hour}, \text{ml}, \theta, \phi, \text{res})$
- Neighborhood: 36 h x 1350 km x 2700 km
 - › 12 x 6 x 12 tokens with 3 x 9 x 9 grid points
- Fields: vorticity, divergence, vertical velocity, temperature, specific humidity, total precipitation, orography
- Vertical model levels: 96, 105, 114, 123, 137
- Depth=20 (encoder: 10, decoder: 10) x 2048 embedding

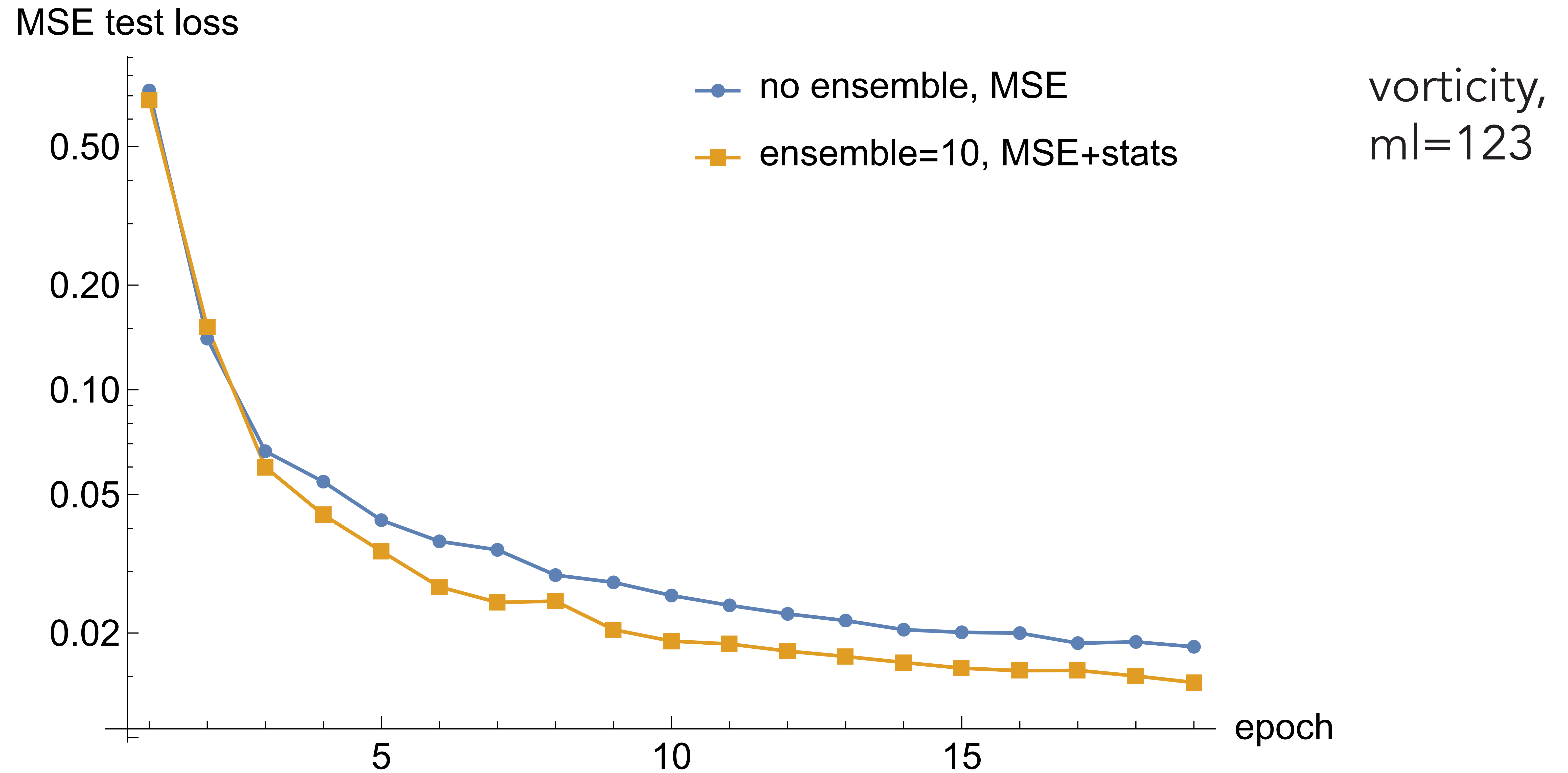
Multiformer: standard configuration

- Per token: $\alpha = (\text{year}, \text{day}, \text{hour}, \text{ml}, \theta, \phi, \text{res})$
- Neighborhood: 36 h x 1350 km x 2700 km
 - › 12 x 6 x 12 tokens with 3 x 9 x 9 grid points
- Fields: vorticity, divergence, vertical velocity, temperature, specific humidity, total precipitation, orography
- Vertical model levels: 96, 105, 114, 123, 137
- Depth=20 (encoder: 10, decoder: 10) x 2048 embedding
- Total number of parameters: 3.5 billion

Statistical loss

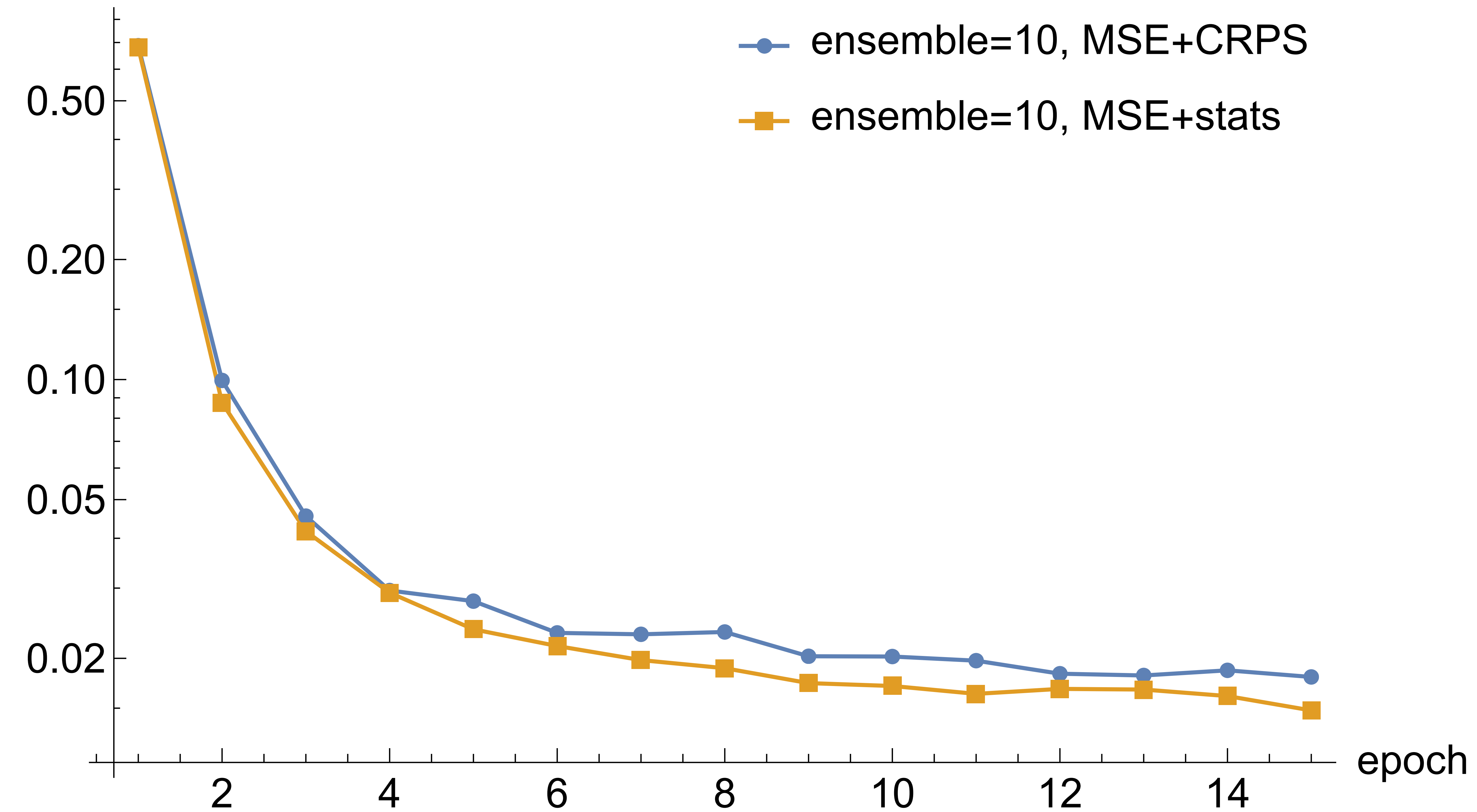


Statistical loss



Statistical loss

MSE test loss



vorticity,
ml=123

Zero-shot capabilities

Large language model: $p_{\theta}(y|x)$

The three settings we explore for in-context learning

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

1 Translate English to French: ← task description
2 cheese => ← prompt

Traditional fine-tuning (not used for GPT-3)

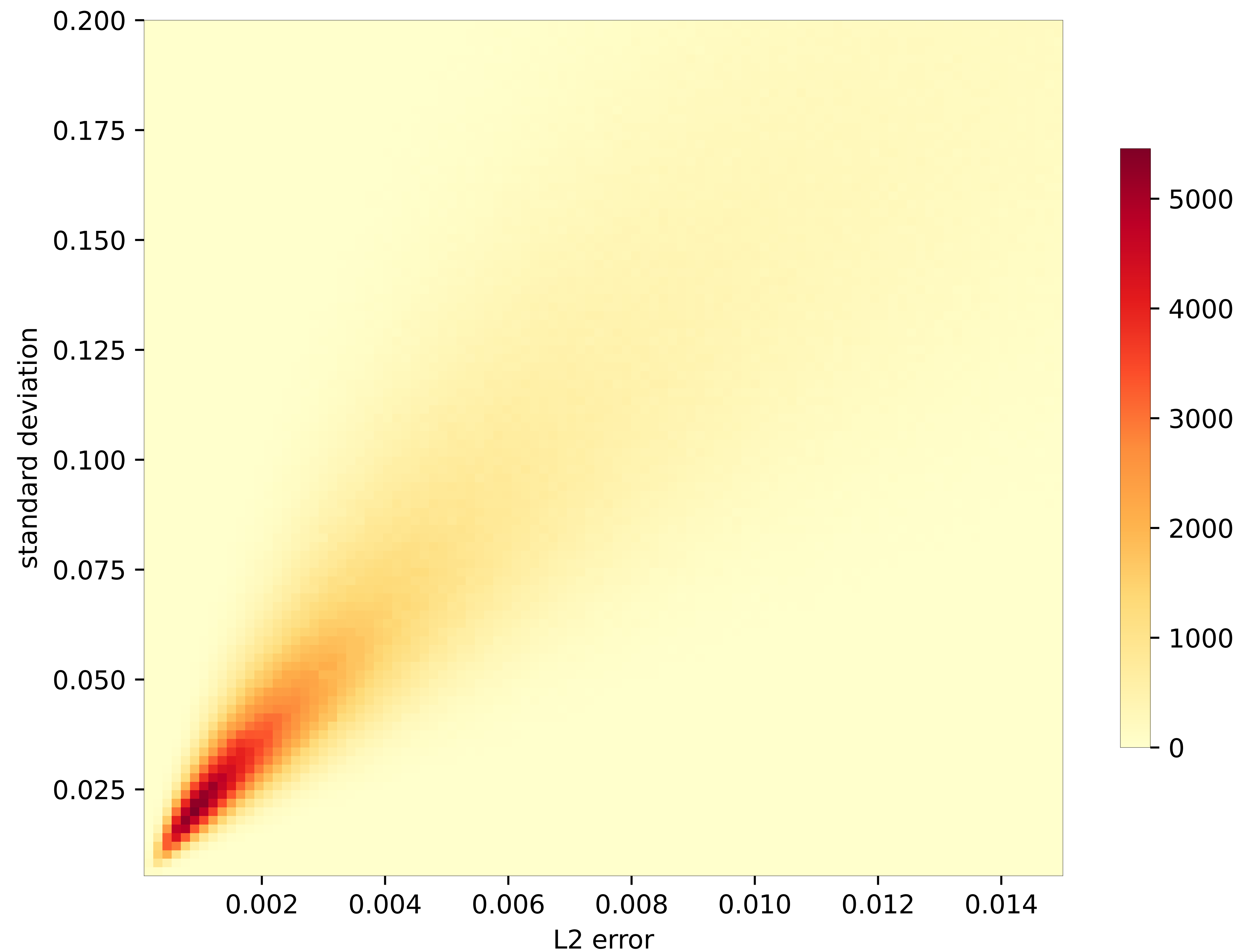
Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.

1 sea otter => loutre de mer ← example #1
↓
gradient update
↓

T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. *Language models are few-shot learners*. In Advances in Neural Information Processing Systems, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

Statistical loss



2D Histogram
of L_2 error vs.
std.-dev.
(temperature)

Pre-training results

- Pre-training of individual fields
 - › More compute-efficient

Pre-training results

- Pre-training of individual fields
 - › More compute-efficient
- Assembly of multiformer from pre-trained fields
 - › Fields can be assembled as needed for application
 - › Very little training time needed to “synchronize” pre-trained fields in assembled multiformer

Pre-training results

