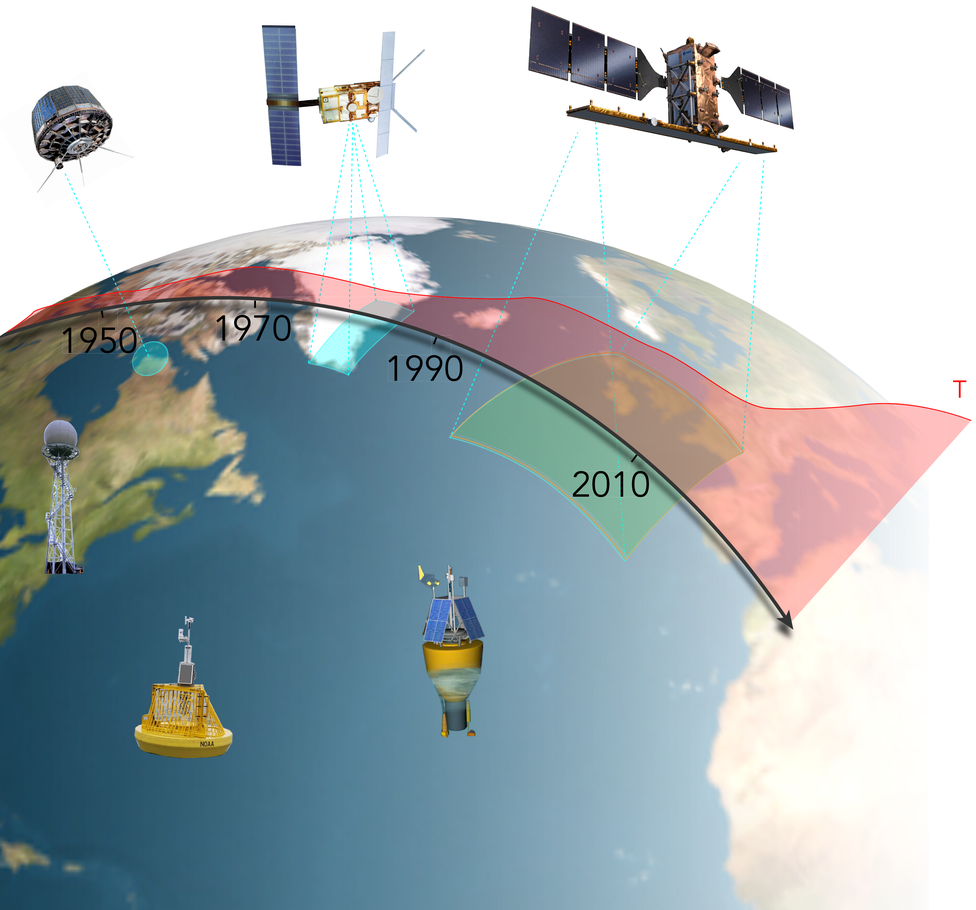
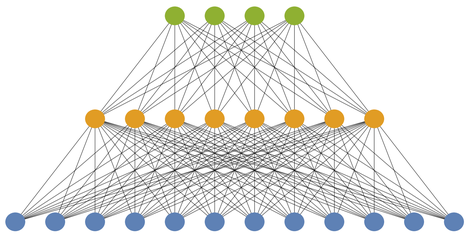


AtmoRep

Large scale representation learning of atmospheric dynamics

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



applications

large scale representation learning

scientific insight

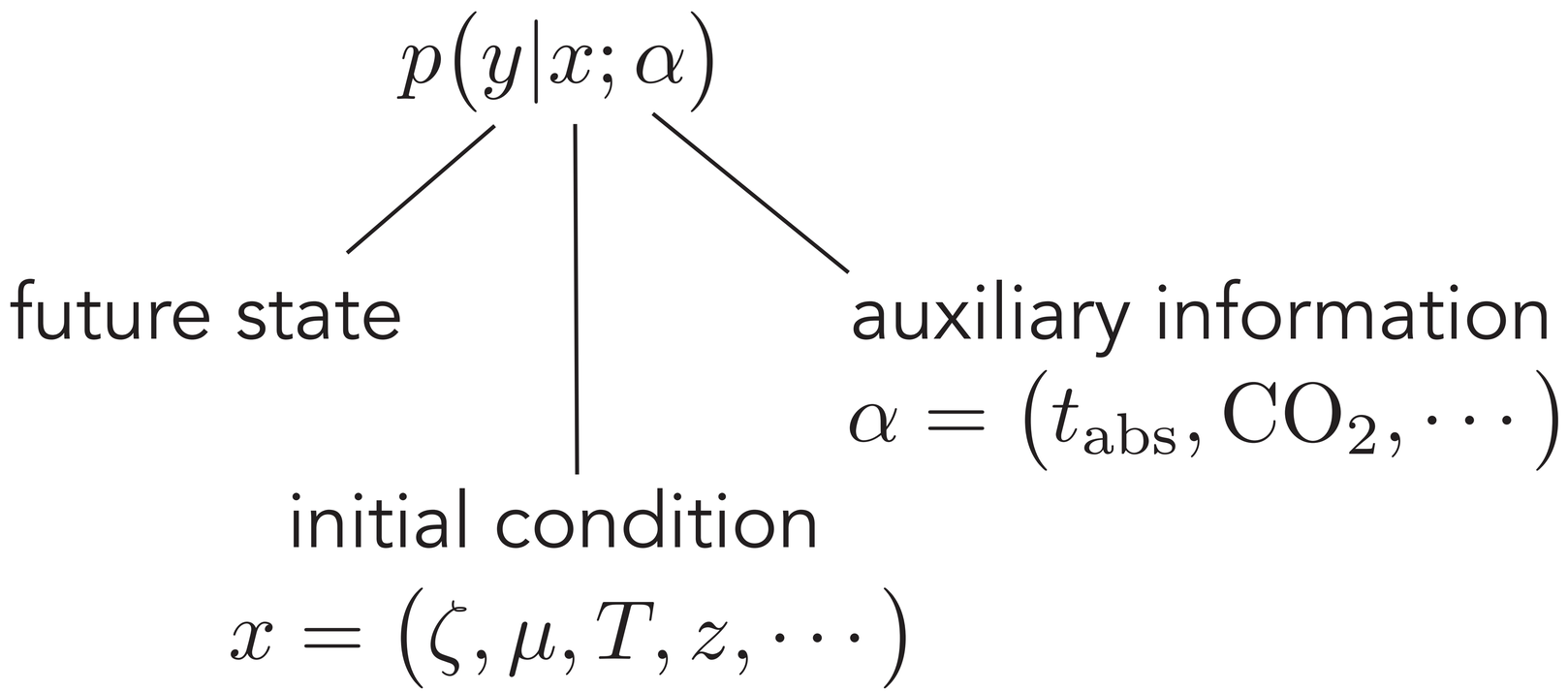
Model formulation

- Atmosphere as stochastic dynamical system:

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

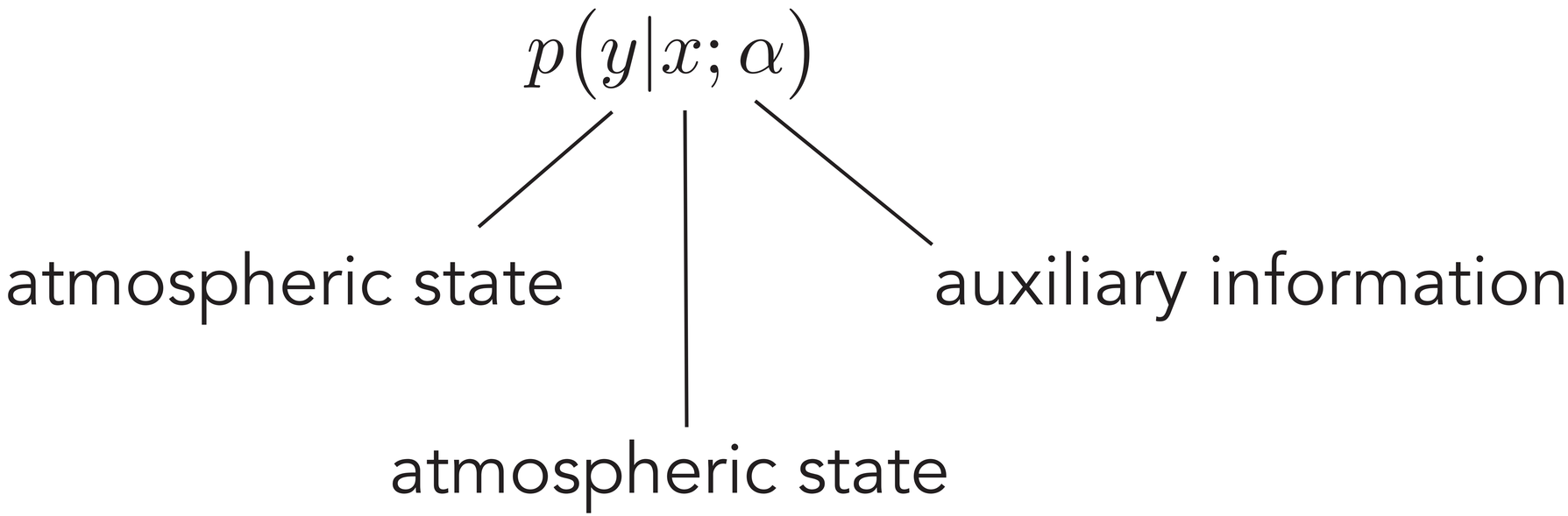
Model formulation

- Atmosphere as stochastic dynamical system:



Model formulation

- Atmosphere as stochastic dynamical system:



Model formulation

- Atmosphere as stochastic dynamical system:

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

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

Model formulation

- Atmosphere as stochastic dynamical system:

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

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

Model formulation

- Atmosphere as stochastic dynamical system:

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

  - Forecasting, downscaling, interpolation: $p\big(y|x;\alpha\big)$
  - Counterfactuals: $p\big(y|x;\hat{\alpha}\big)$
  - Climate: $p_\alpha(y) = \int p\big(y|x;\alpha\big)\, p(x)\, \mathrm{d}x$

Model formulation

- Atmosphere as stochastic dynamical system:

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

  - Forecasting, downscaling, interpolation: $p(y|x;\alpha)$
  - Counterfactuals: $p(y|x;\hat{\alpha})$
  - Climate: $p_\alpha(y) = \int p(y|x;\alpha)\, p(x)\, \mathrm{d}x$

Model formulation

- Atmosphere as stochastic dynamical system:

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

- Numerical statistical atmospheric model:

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

Model formulation

- Atmosphere as stochastic dynamical system:

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

- Numerical statistical atmospheric model:

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

Model formulation

- Atmosphere as stochastic dynamical system:

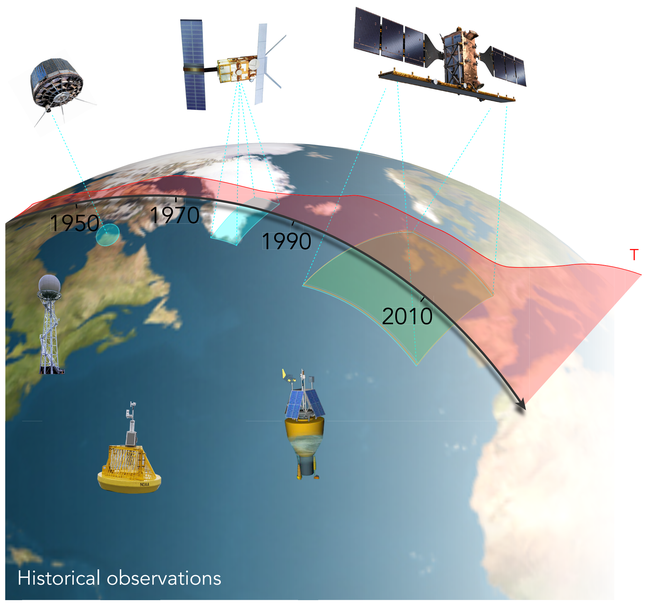
$$p\big(y|x;\alpha\big)$$

- Numerical statistical atmospheric model:

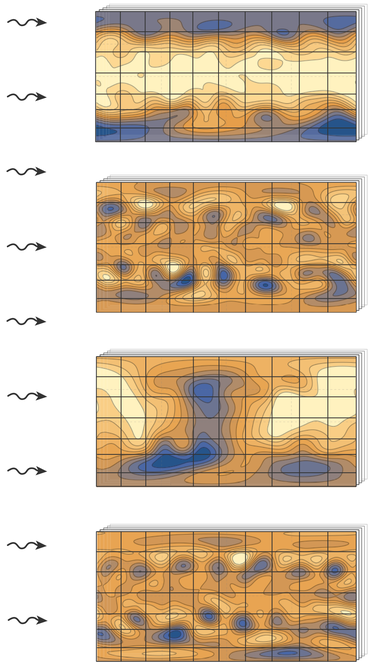
very large neural network

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

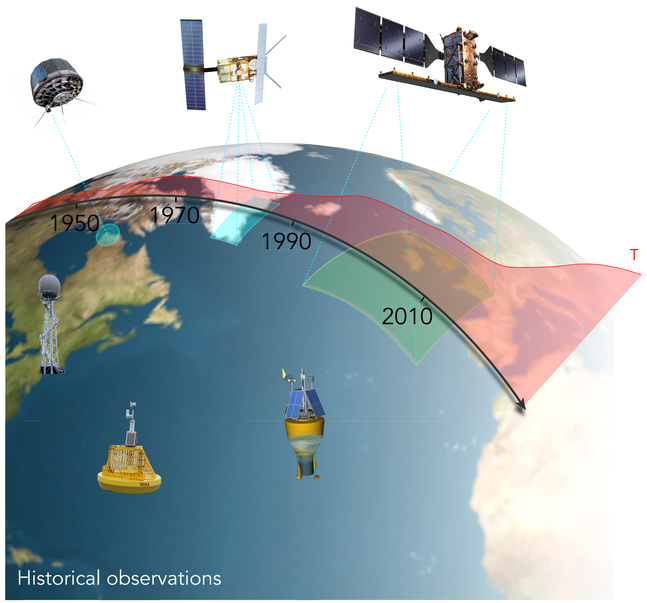
AtmoRep



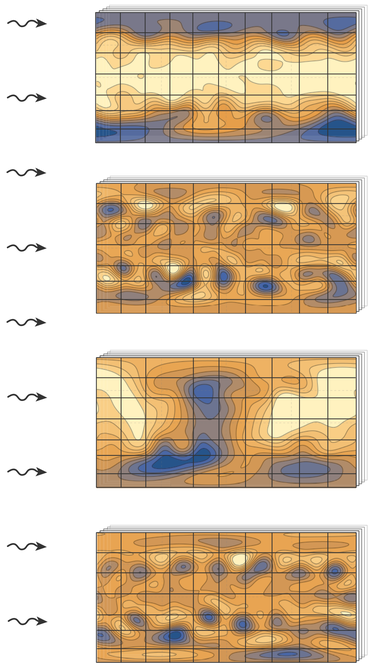
ERA5 reanalysis



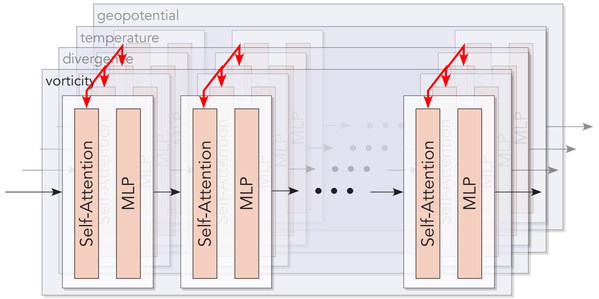
AtmoRep



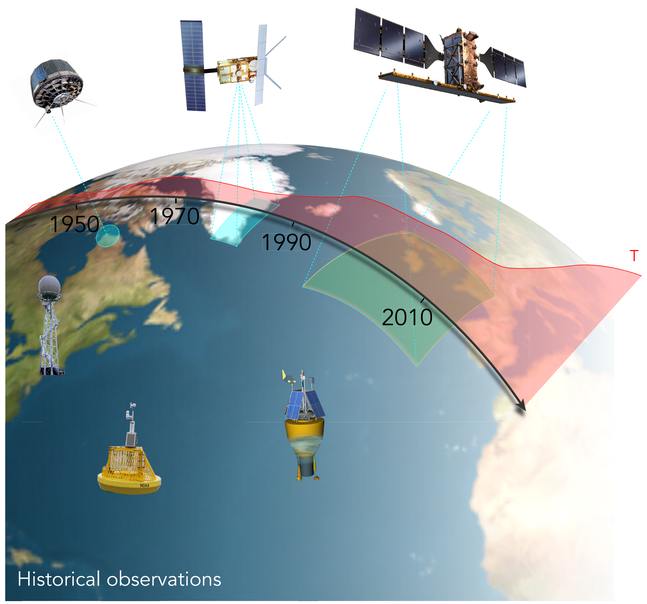
ERA5 reanalysis



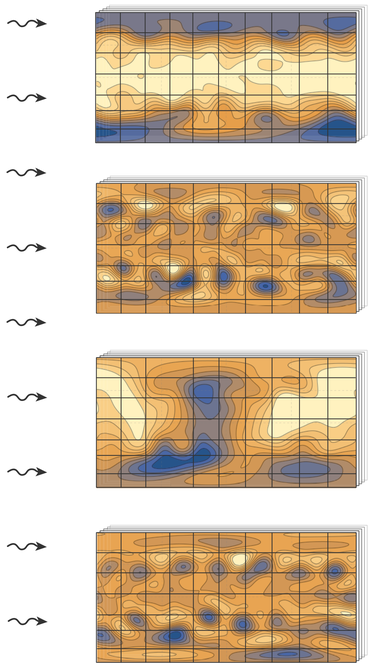
$$p_\theta(y|x)$$



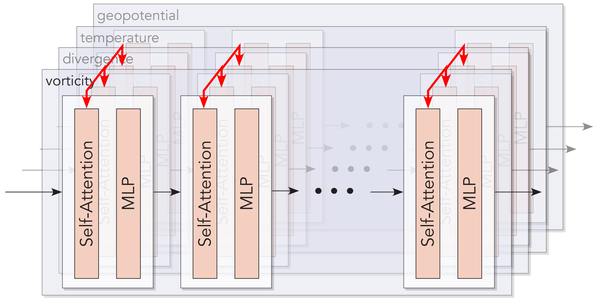
AtmoRep



ERA5 reanalysis

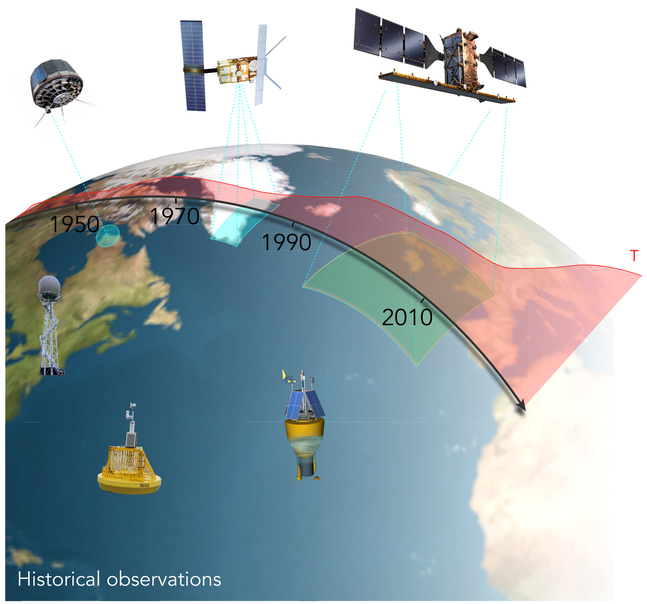


$$p_\theta(y|x)$$

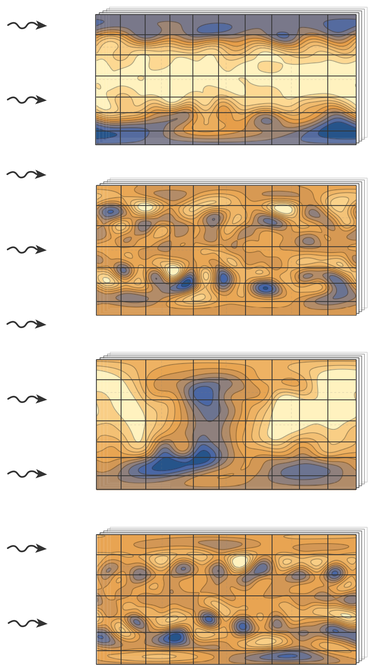


large transformer
with 3.5 x 10^9 parameters

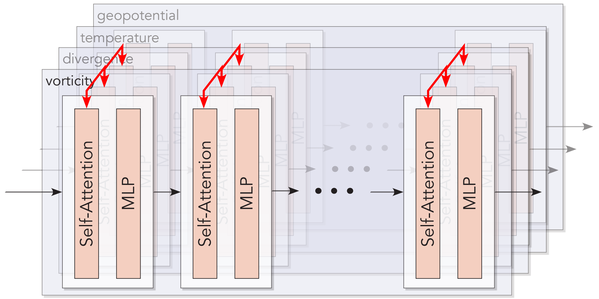
AtmoRep



ERA5 reanalysis

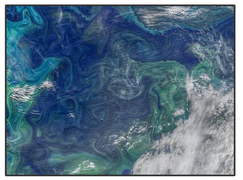


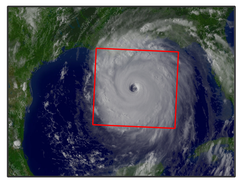
$$p_\theta(y|x)$$

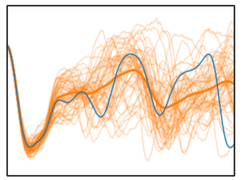


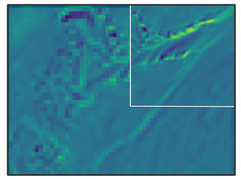
large transformer
with 3.5 x 10^9 parameters

applications

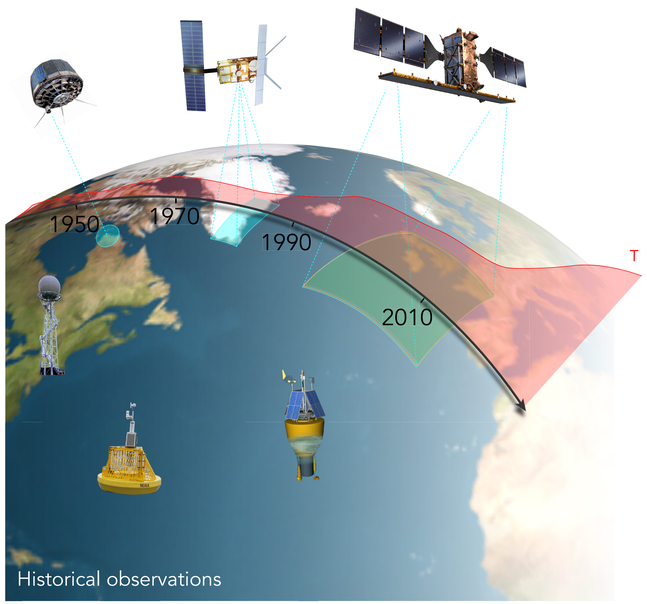

Forecasting


Impact analysis

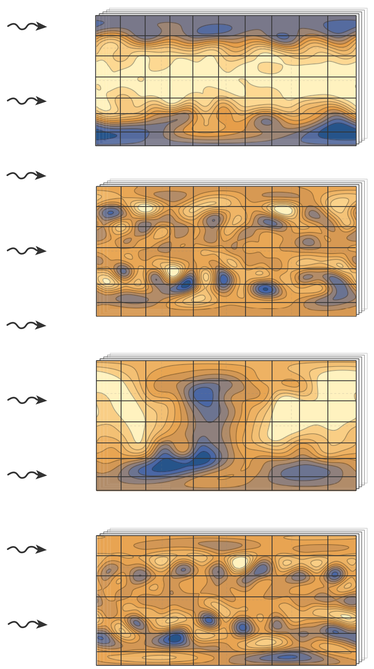

Statistical analysis


Downscaling

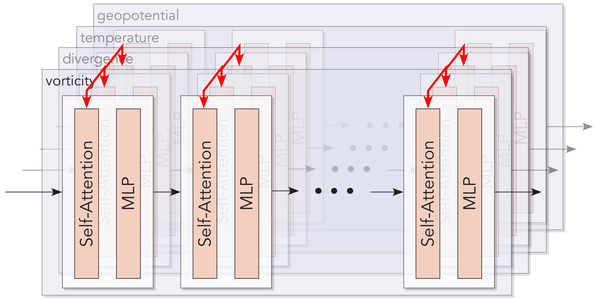
AtmoRep



ERA5 reanalysis

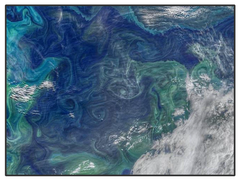


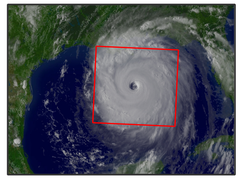
$p_\theta(y|x)$

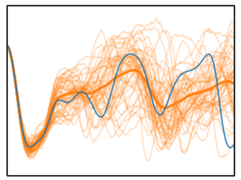


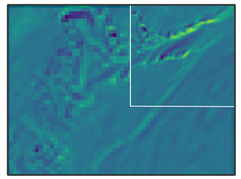
large transformer
with 3.5 x 10^9 parameters

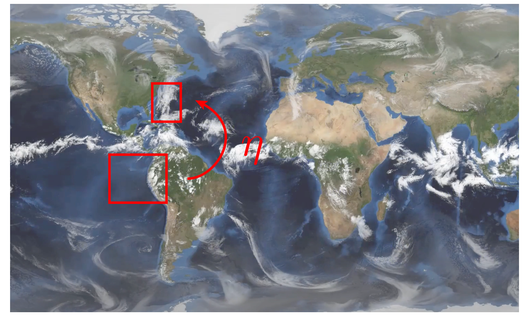
applications


Forecasting


Impact analysis


Statistical analysis

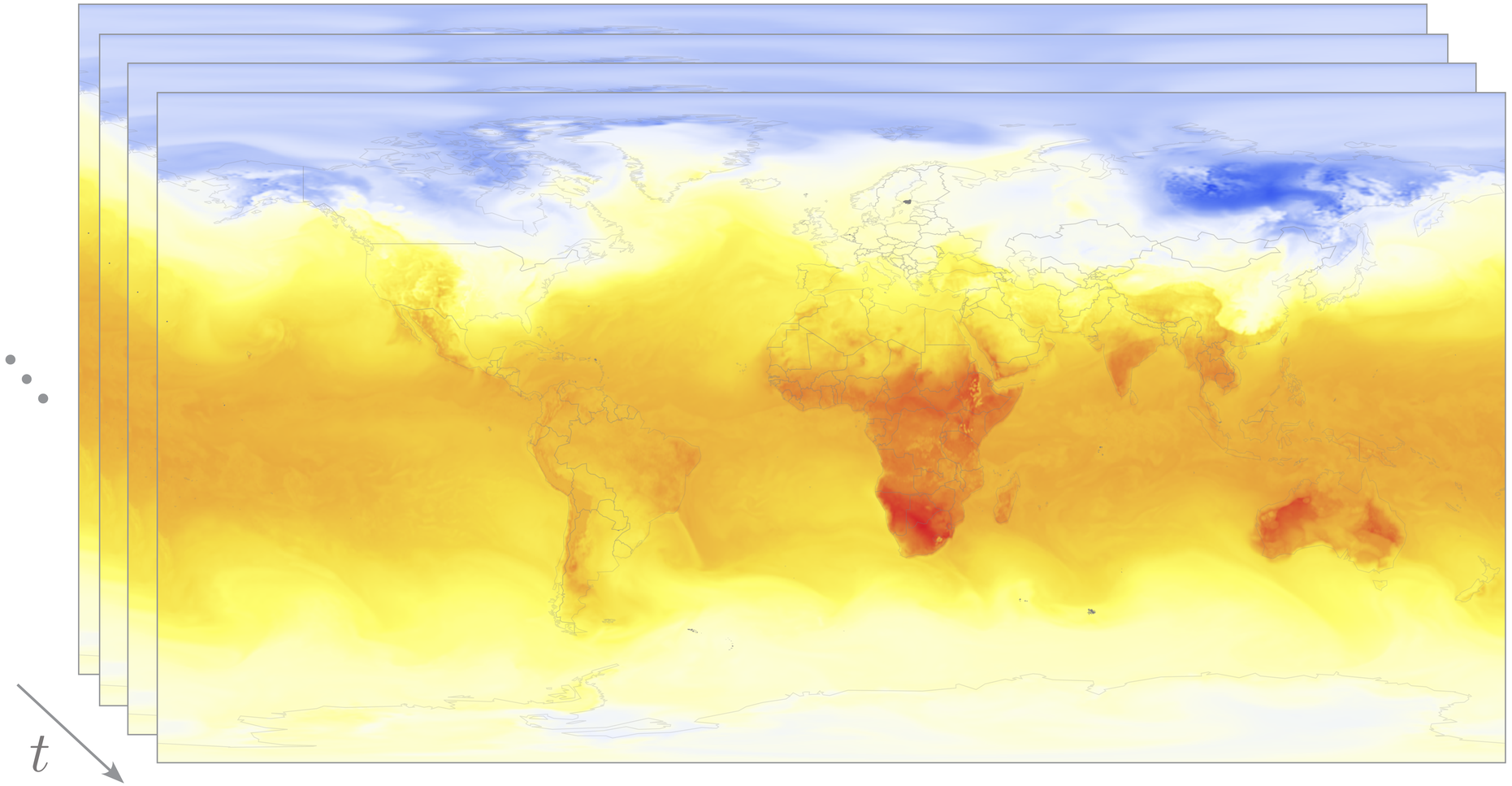

Downscaling



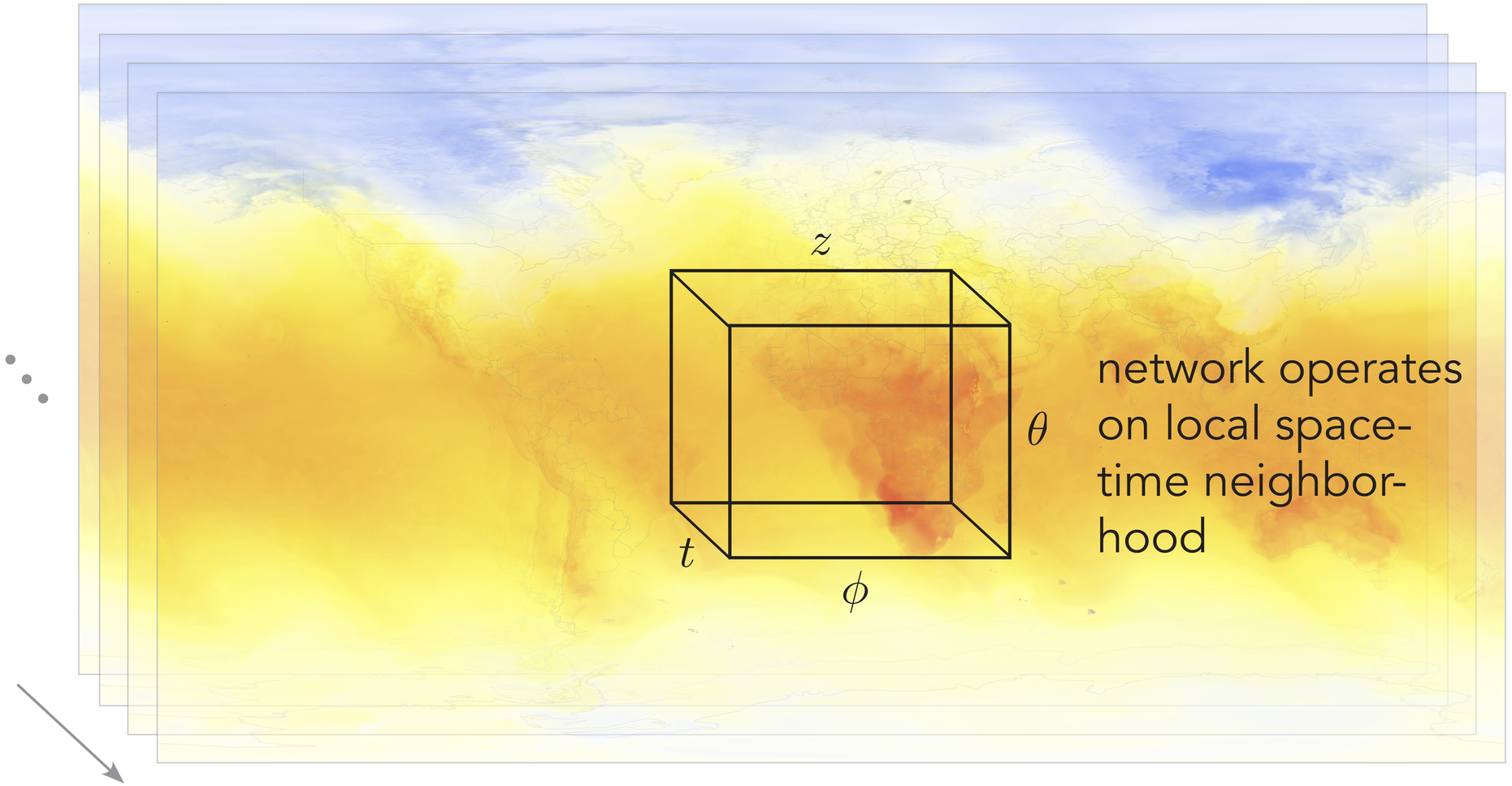
scientific insight

Network architecture

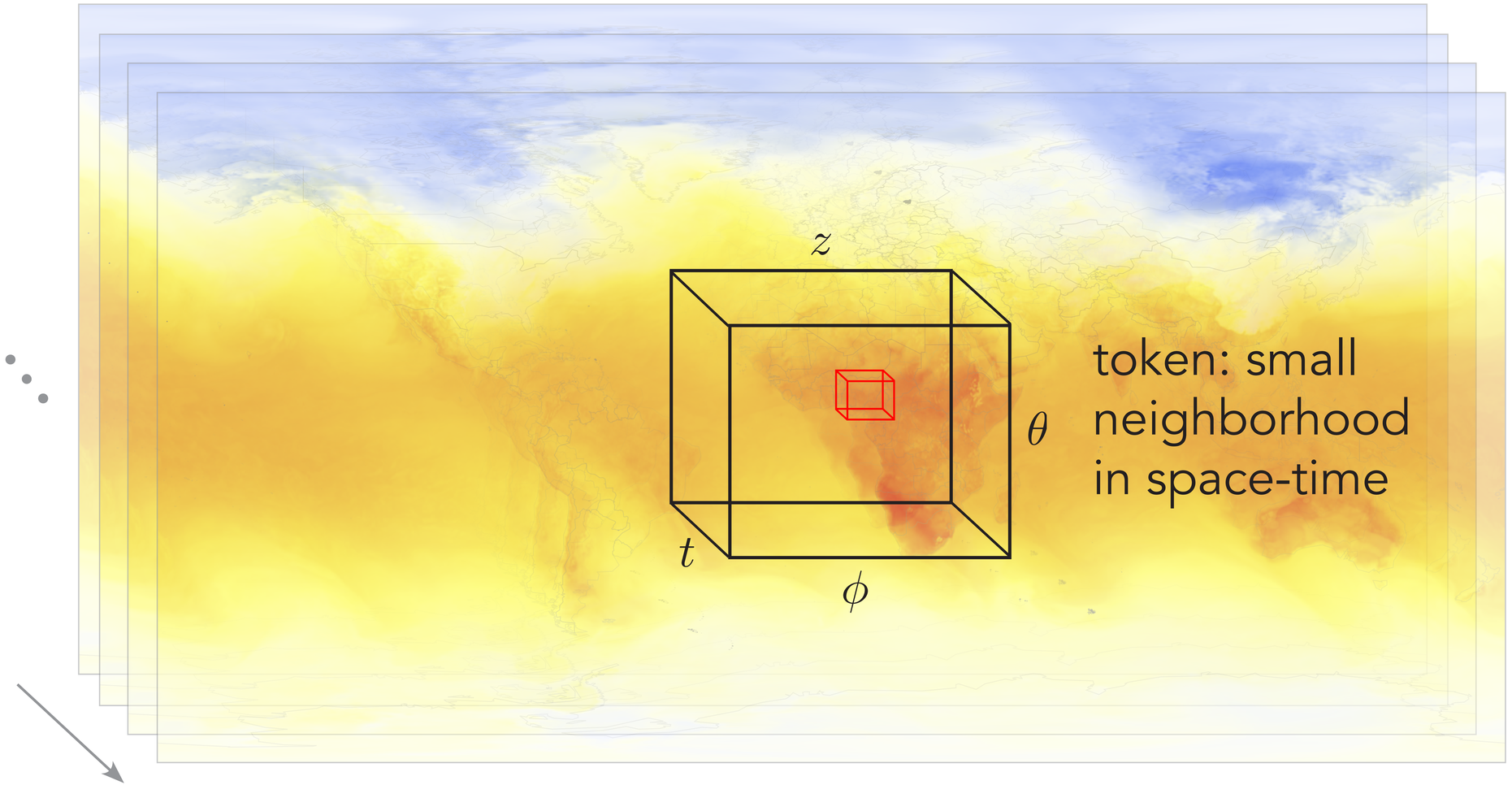
AtmoRep network architecture



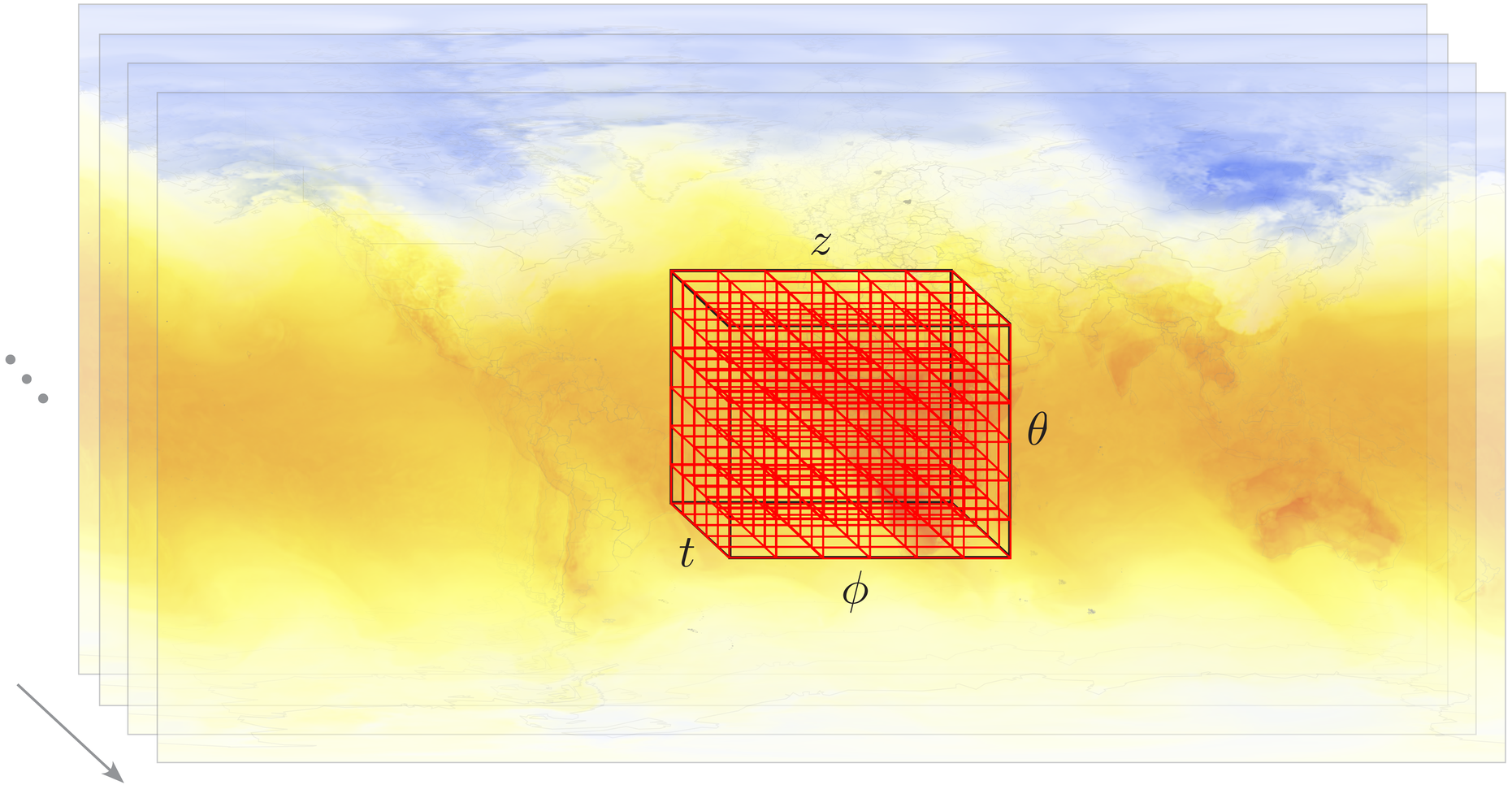
AtmoRep network architecture



What is a token?

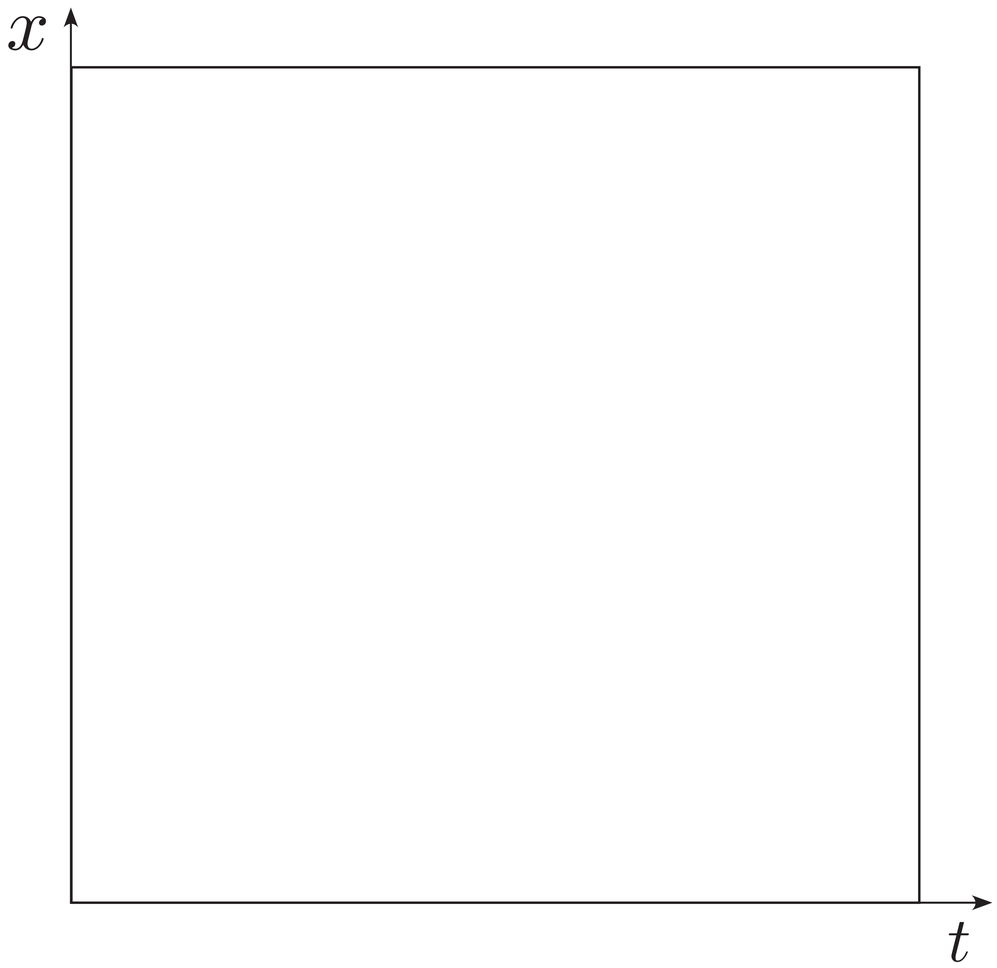


What is a token?



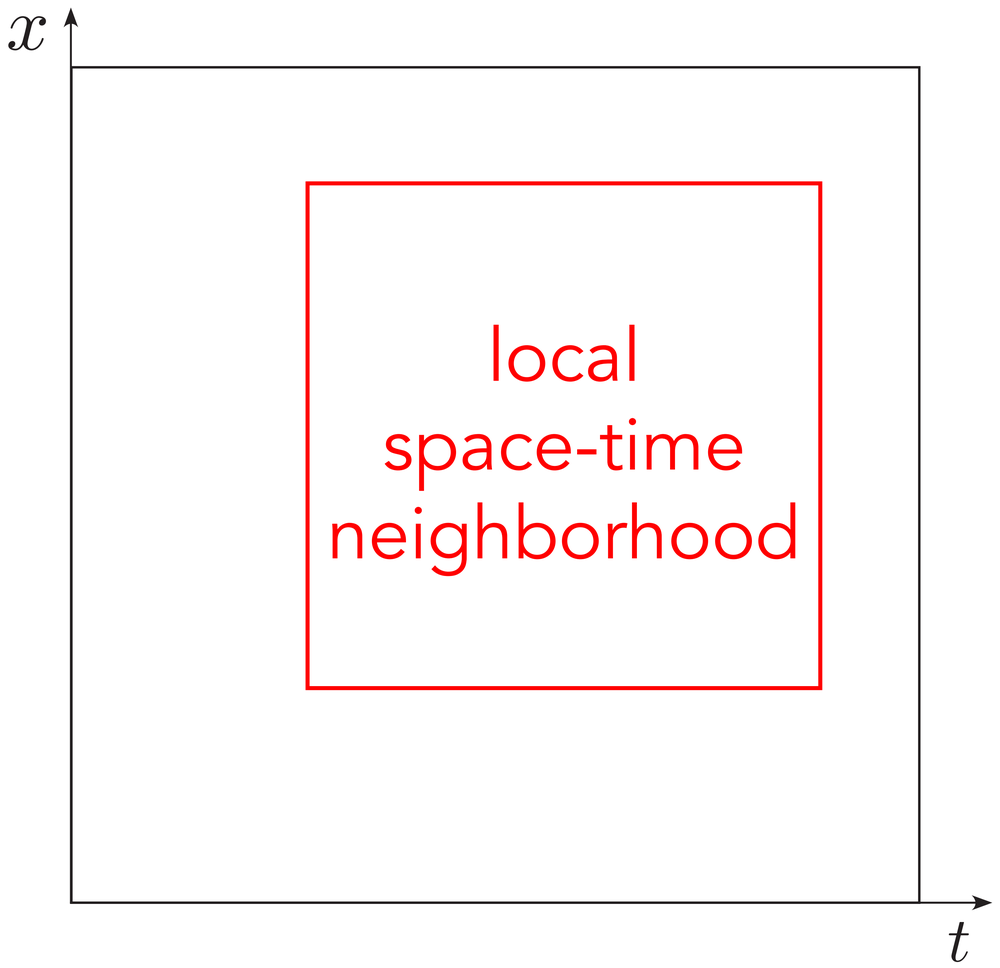
What is a token?

- Flatland view:



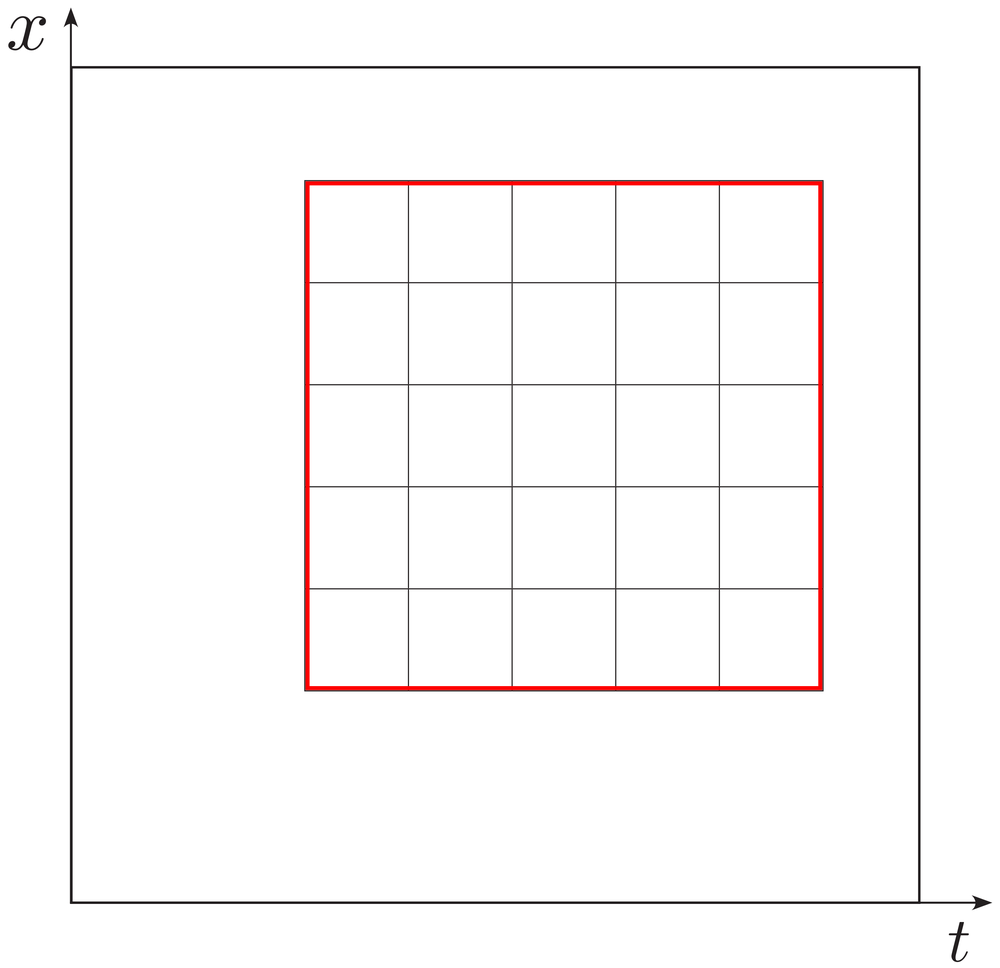
What is a token?

- Flatland view:



What is a token?

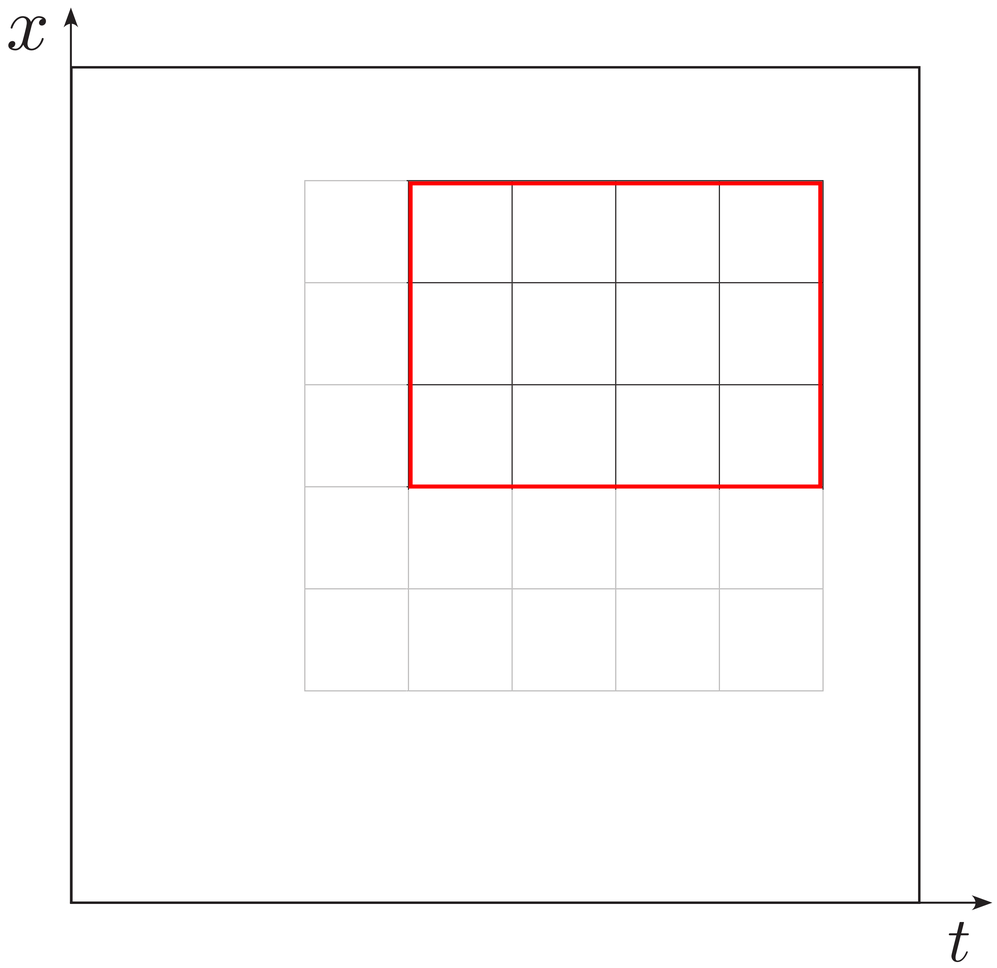
- Flatland view:



What is a token?

- Flatland view:

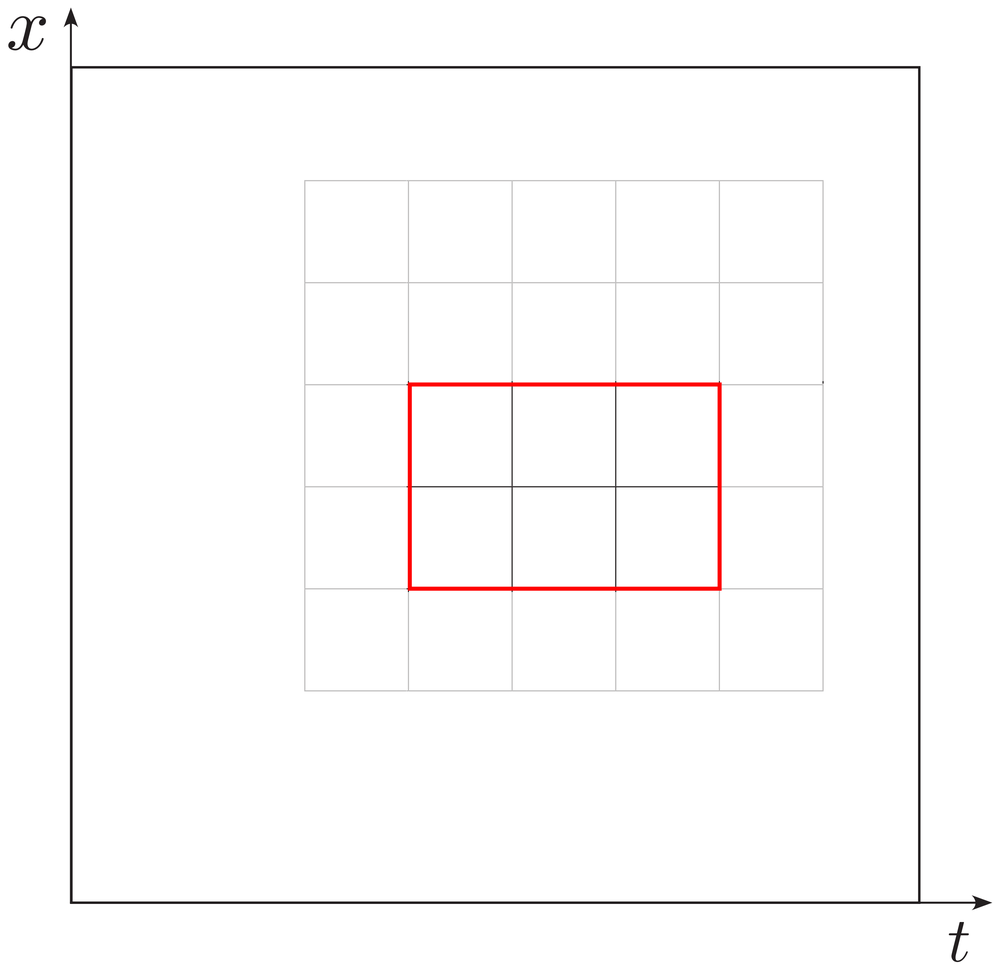
local window
size is flexible



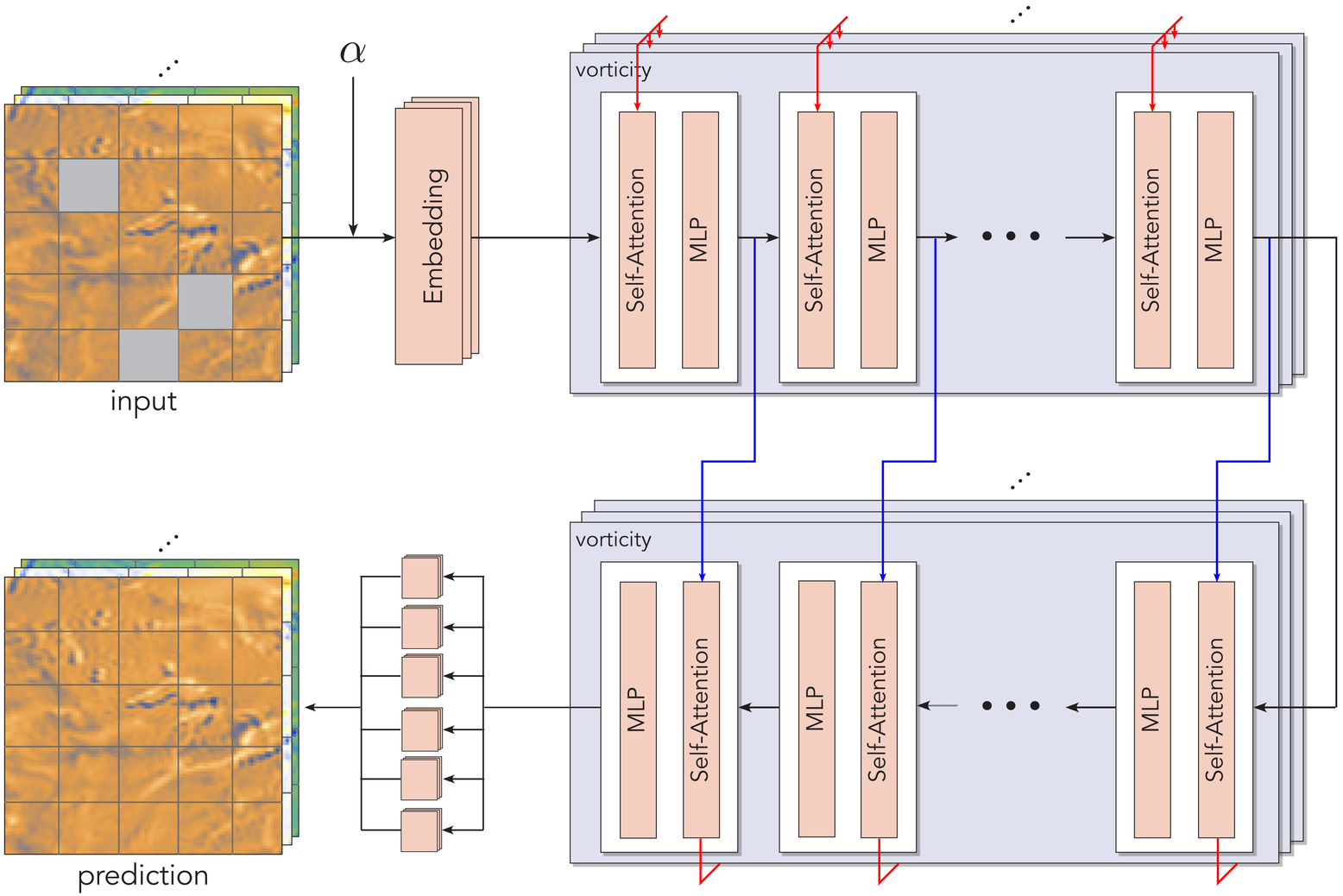
What is a token?

- Flatland view:

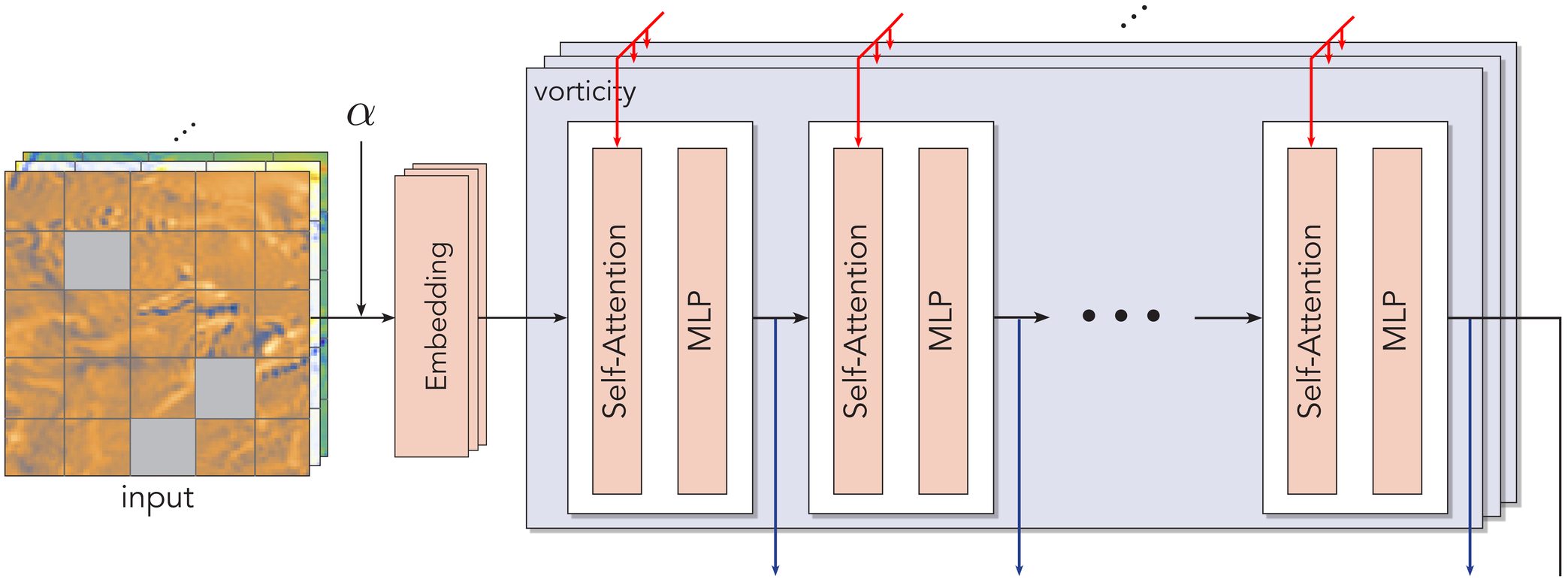
local window
size is flexible



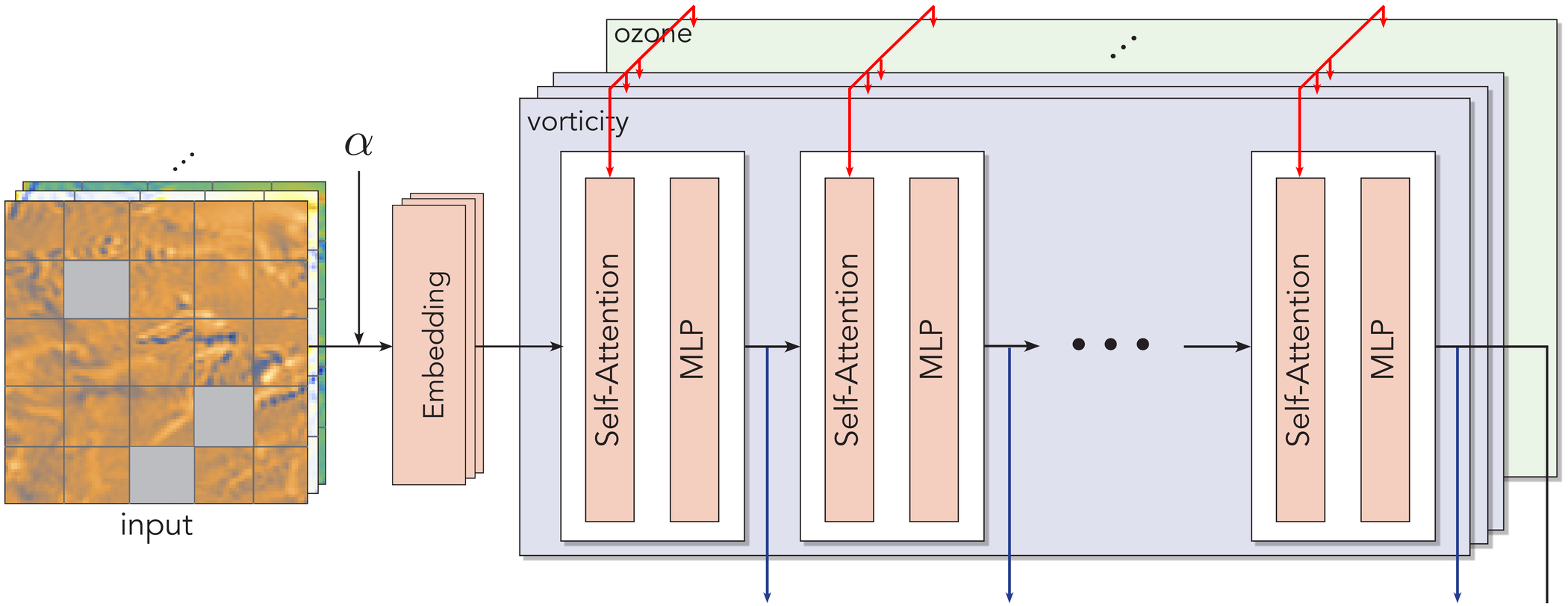
Multiformer



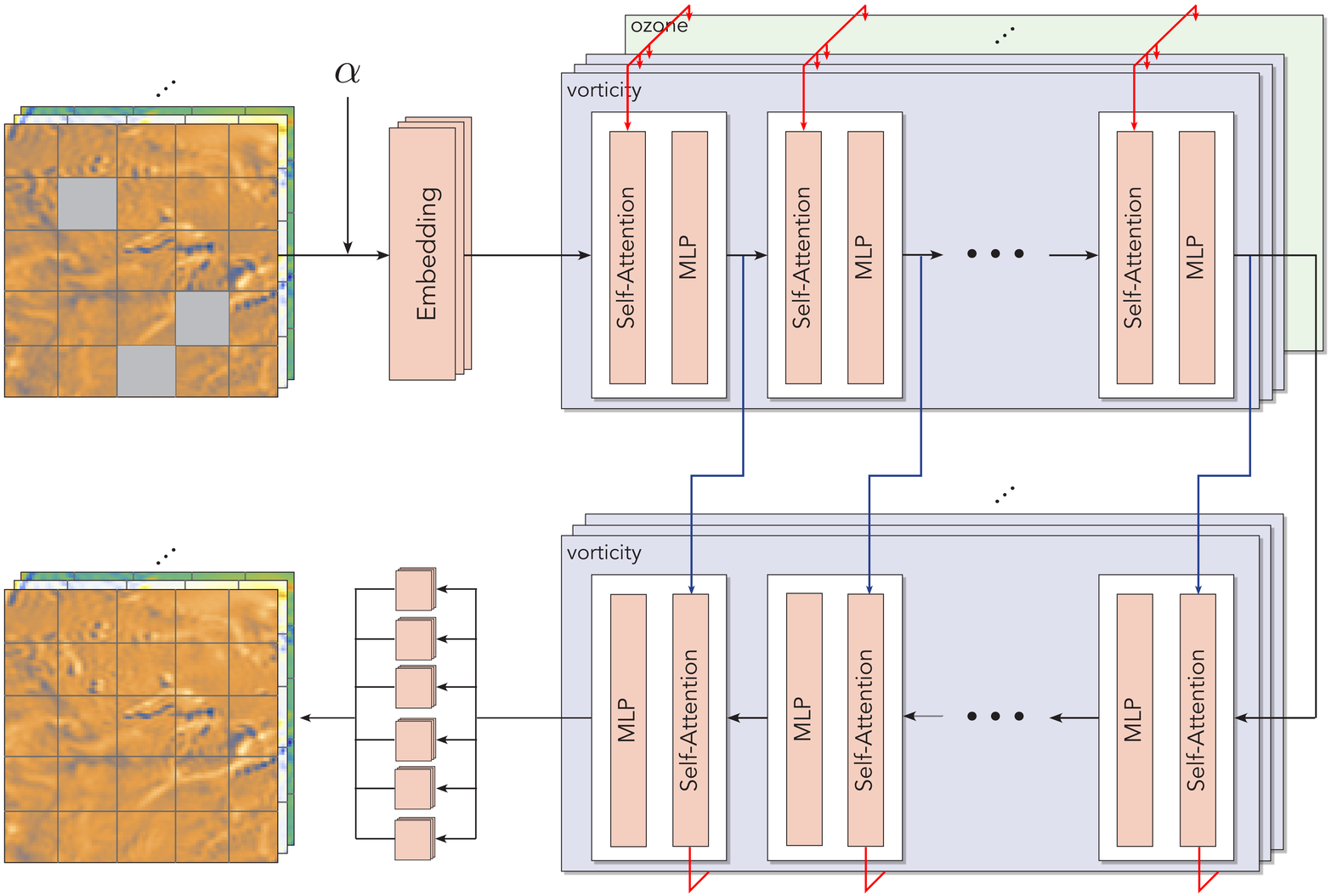
Multiformer



Multiformer



Multiformer



Multiformer: standard configuration

- Neighhorhood: 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

- Neighhorhood: 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, velocity, vertical velocity, temperature, specific humidity, total precipitation

Multiformer: standard configuration

- Neighhorhood: 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, velocity, vertical velocity, temperature, specific humidity, total precipitation
- Vertical model levels: 96, 105, 114, 123, 137

Multiformer: standard configuration

- Neighhorhood: 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, velocity, vertical velocity, temperature, specific humidity, total precipitation
- Vertical model levels: 96, 105, 114, 123, 137
- Depth=20 (encoder: 10, decoder: 10) x 2048 embedding

Multiformer: standard configuration

- Neighhorhood: 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, velocity, vertical velocity, temperature, specific humidity, total precipitation
- Vertical model levels: 96, 105, 114, 123, 137
- Depth=20 (encoder: 10, decoder: 10) x 2048 embedding
- Total number of parameters: 3.5 billion

Loss and Pre-training

Training objective

- Numerical statistical atmospheric model

$$p_\theta\left(y|x,\alpha\right)$$

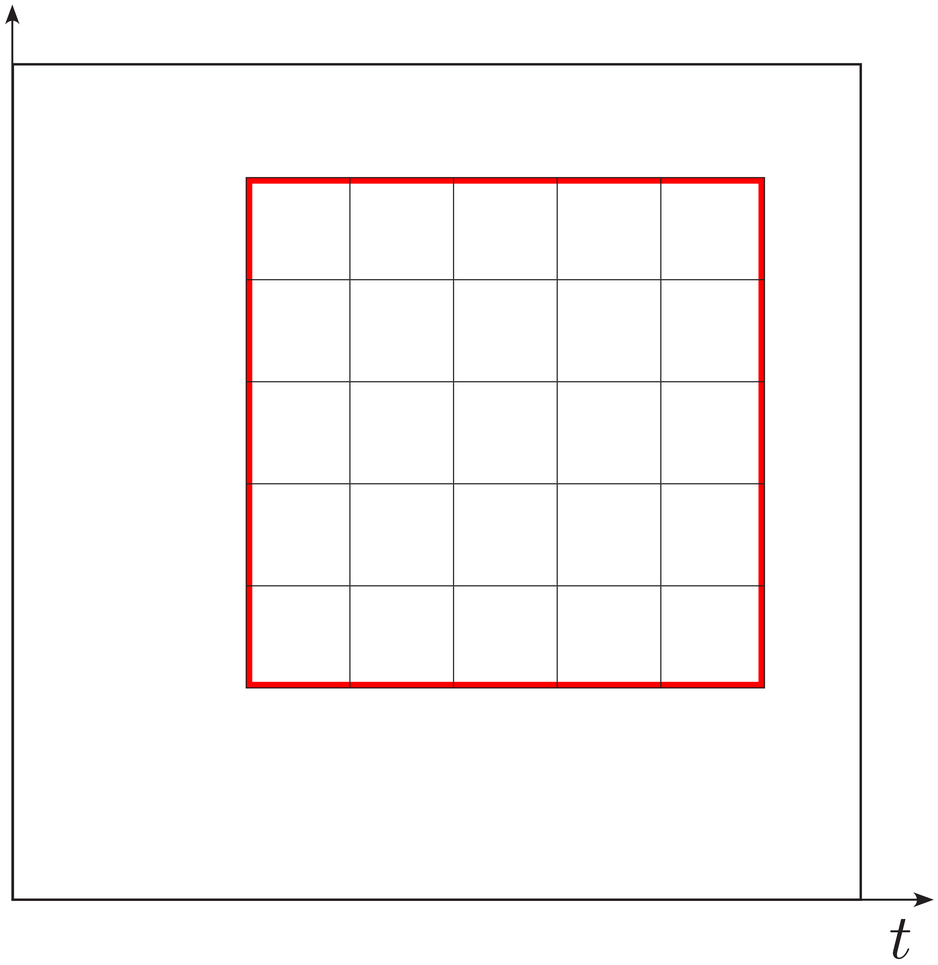
Training objective

- Numerical statistical atmospheric model

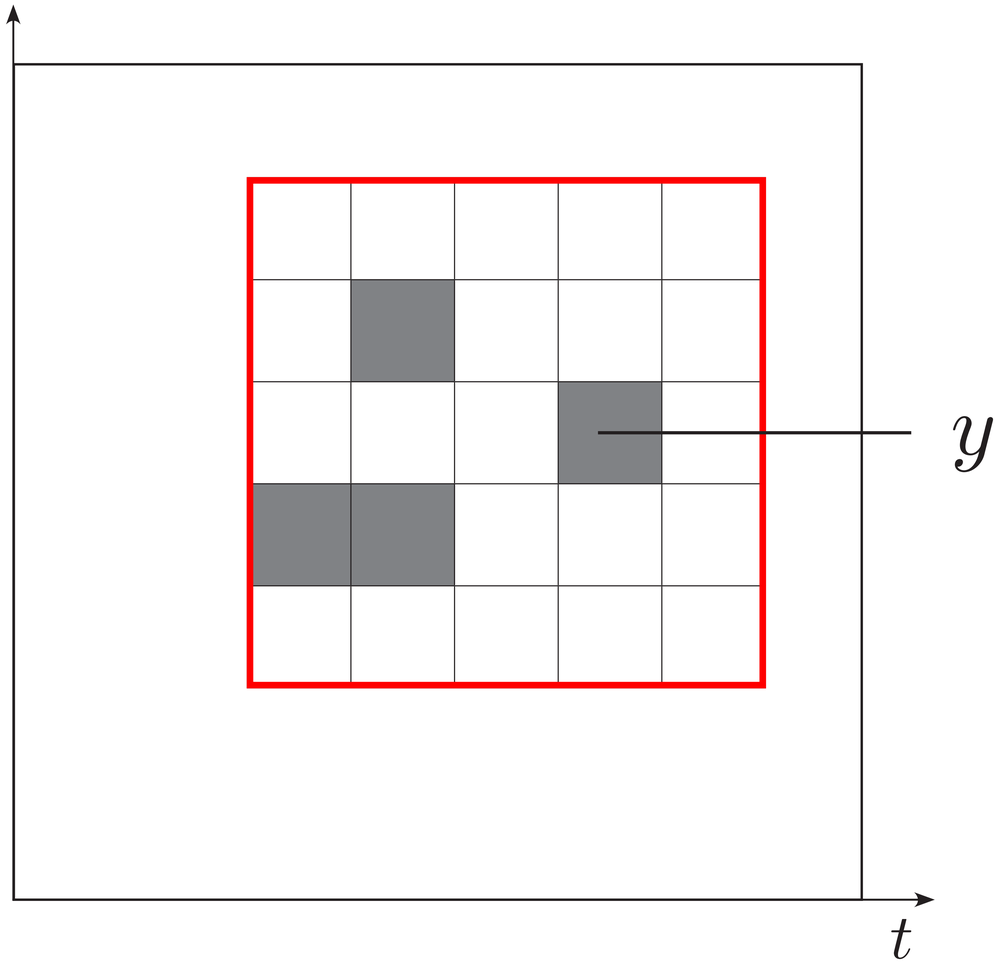
$$p_\theta\left(y|x,\alpha\right)$$

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

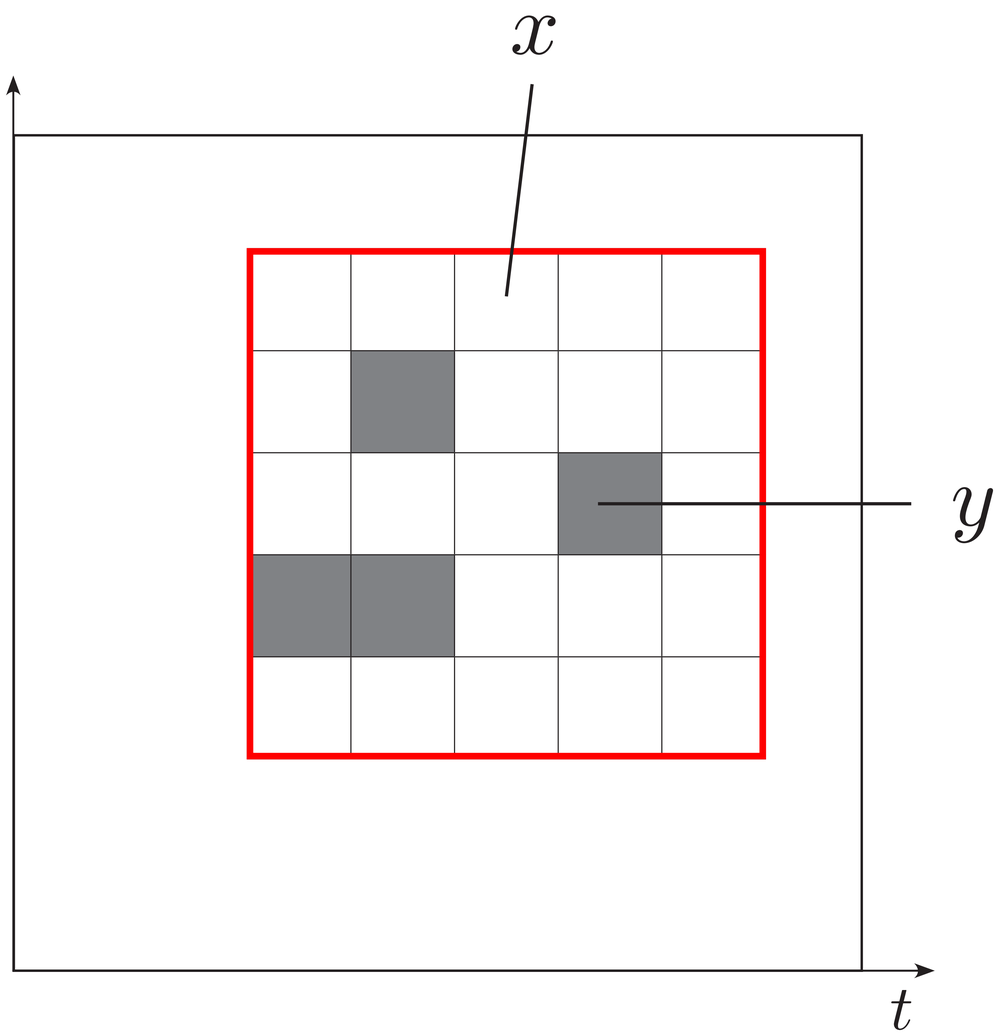
Training objective



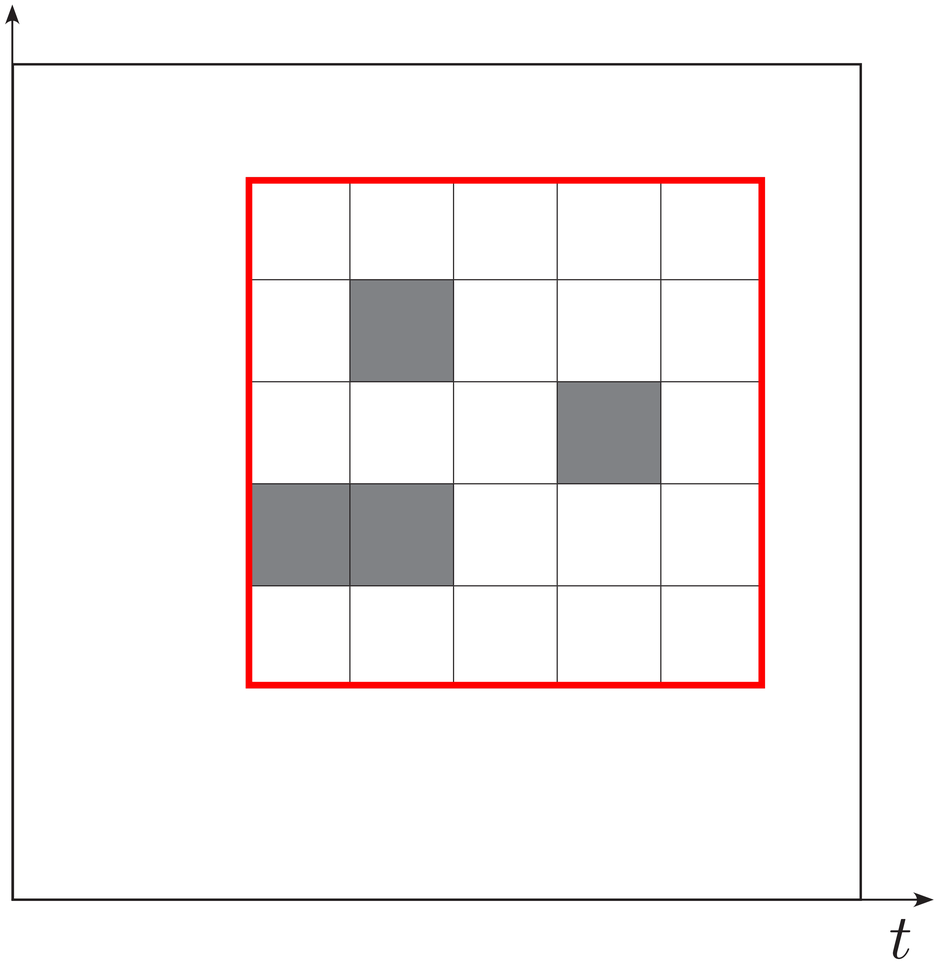
Training objective



Training objective



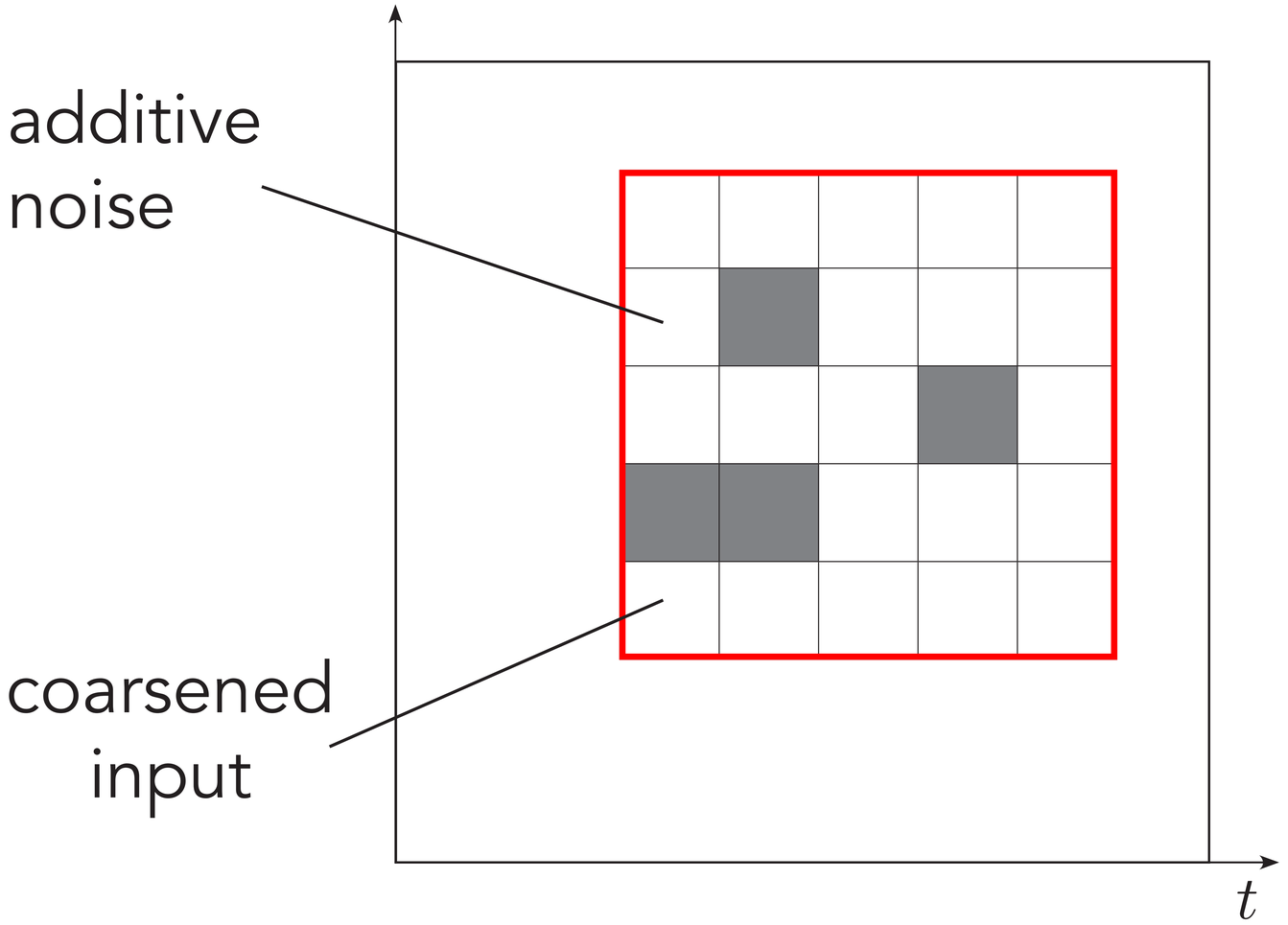
Training objective



masked
token model:

training to predict
randomly masked
information

Training objective

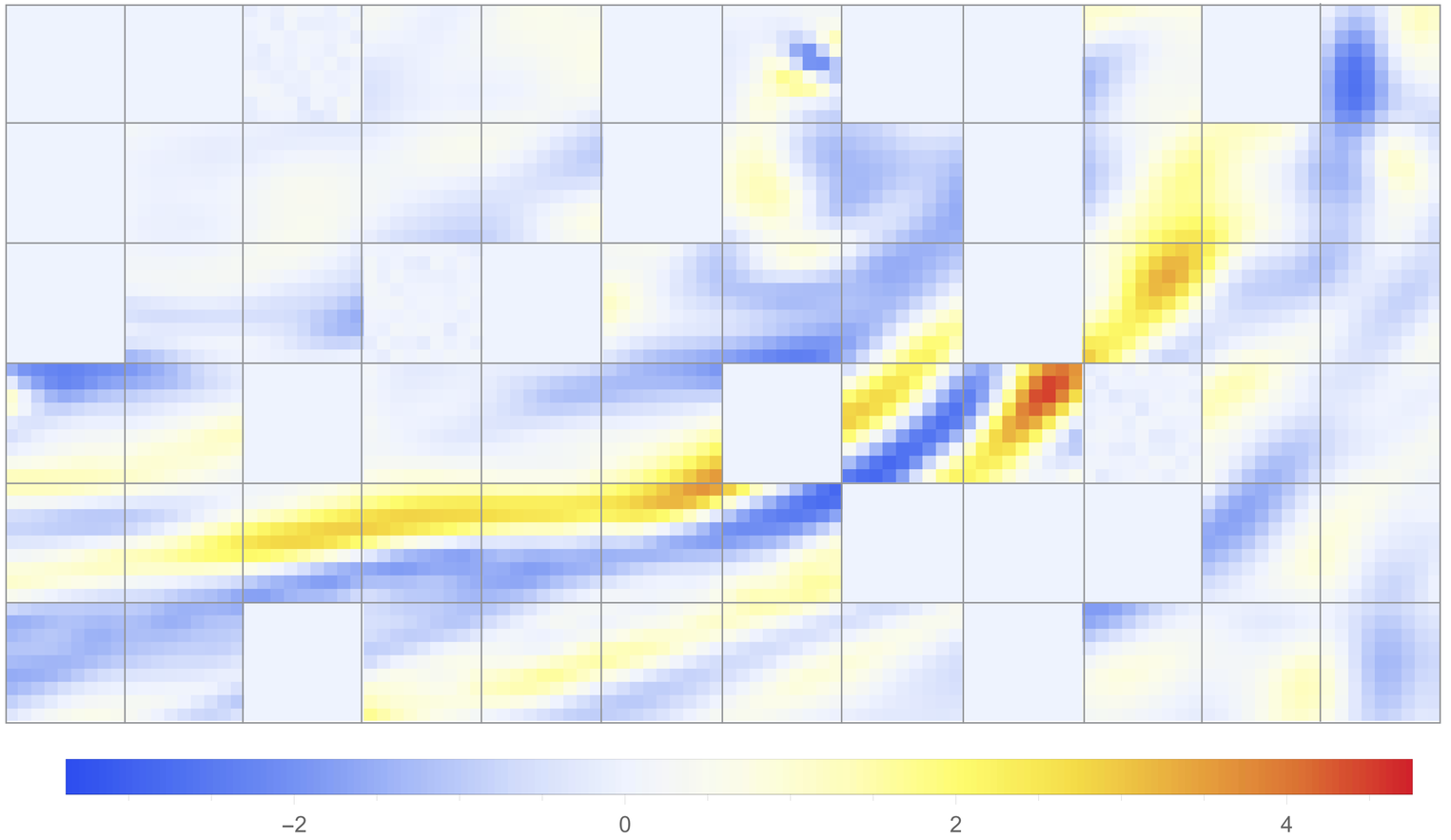


masked
token model:

training to predict
randomly masked
information

Training objective

divergence, ml=96



Training objective

- BERT-style masked token model
  - Mixture of masking, noise-pertubation and coarsening to learn robust, probabilistic representation over x, y
  - Masking ratio is sampled up to, e.g., 0.5, 0.75

Training objective

- BERT-style masked token model
  - Mixture of masking, noise-pertubation and coarsening to learn robust, probabilistic representation over x, y
  - Masking ratio is sampled up to, e.g., 0.5, 0.75
- Monte Carlo sampling of spatio-temporal neighborhood
  - Random sampling of space-time neighborhood by first sampling month (=1 file) and then spatial patches
  - Robust sampling from data and embarassingly parallel

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?

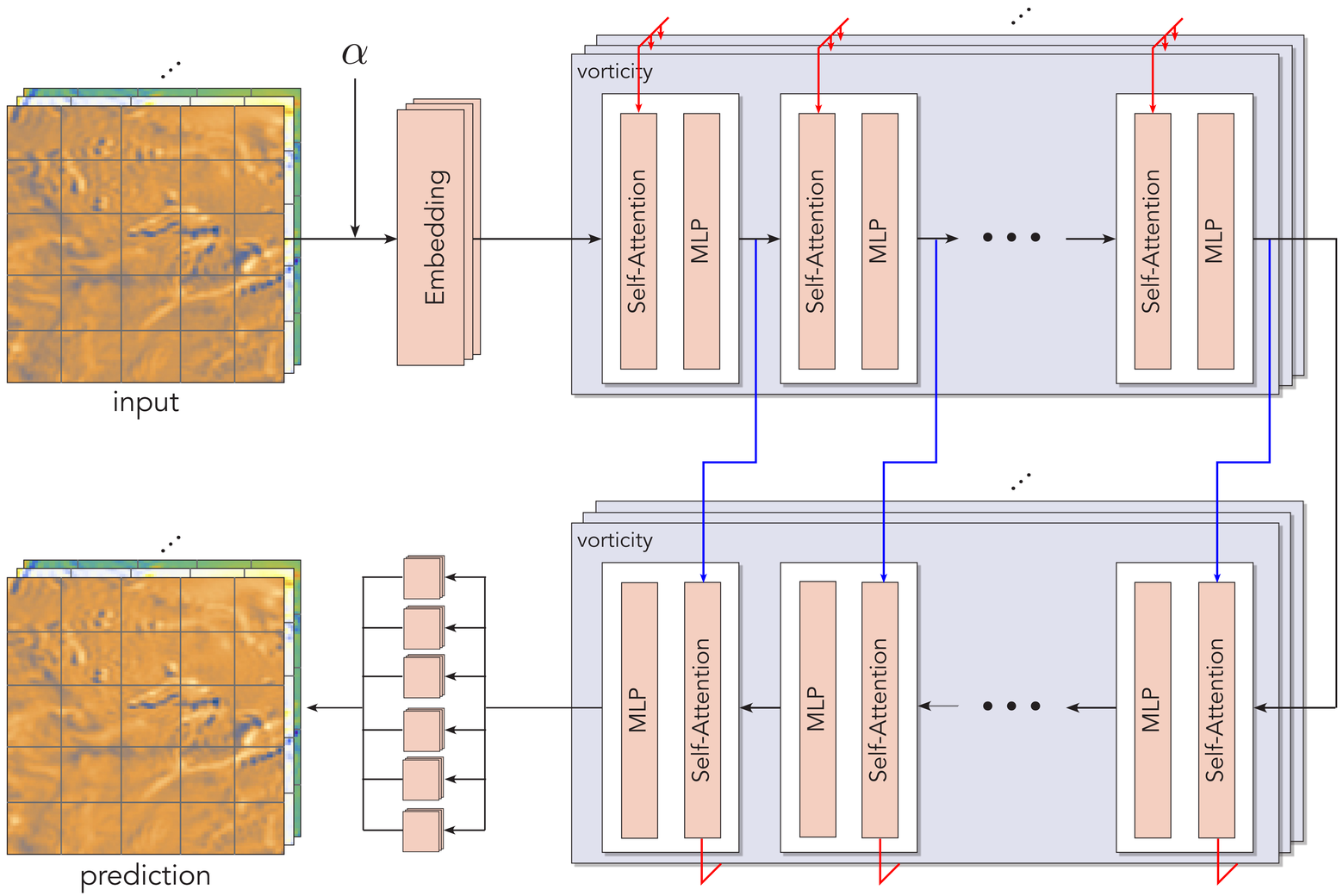
Loss

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

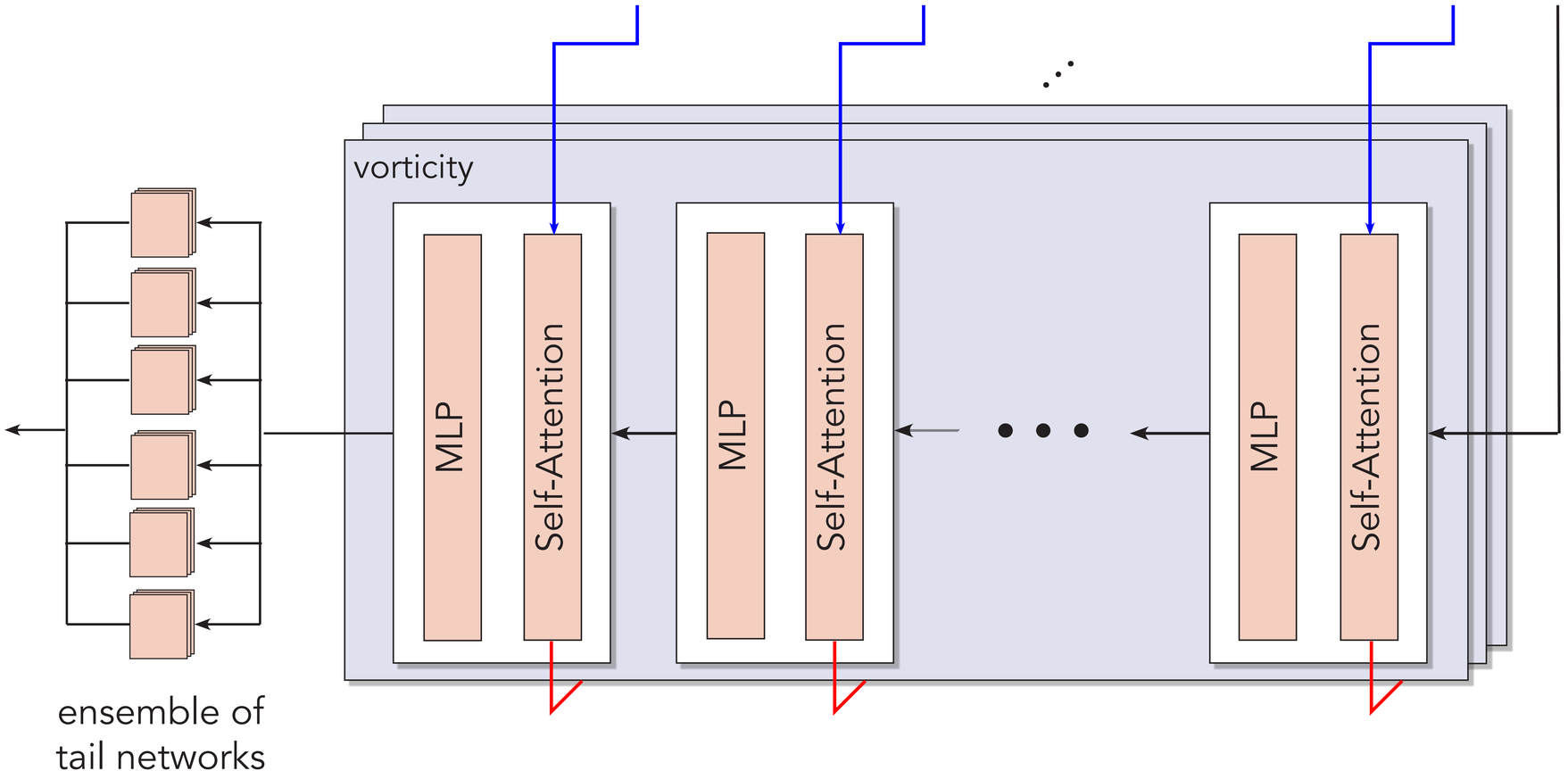
=> Analog of cross-entropy loss for regression problems?

  - Use unparametric distribution over possible outputs
  - Require only one correct label (Kronecker distribution)

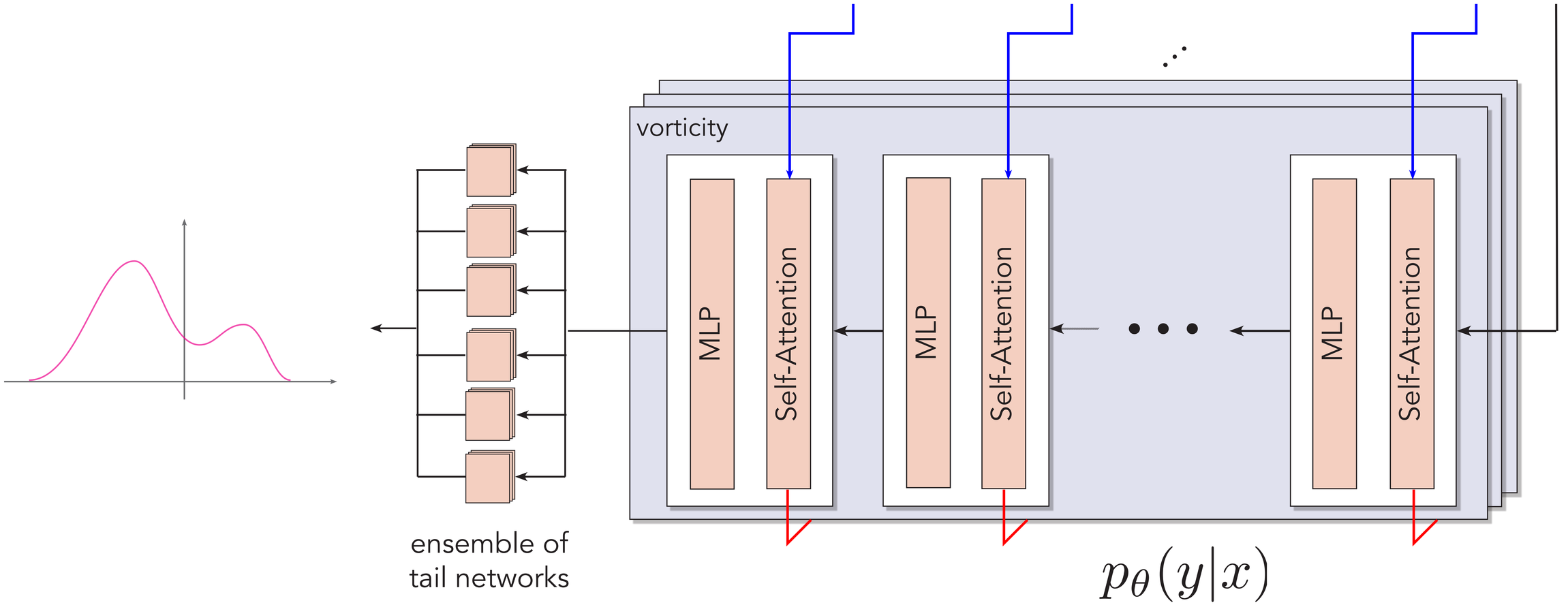
Statistical loss



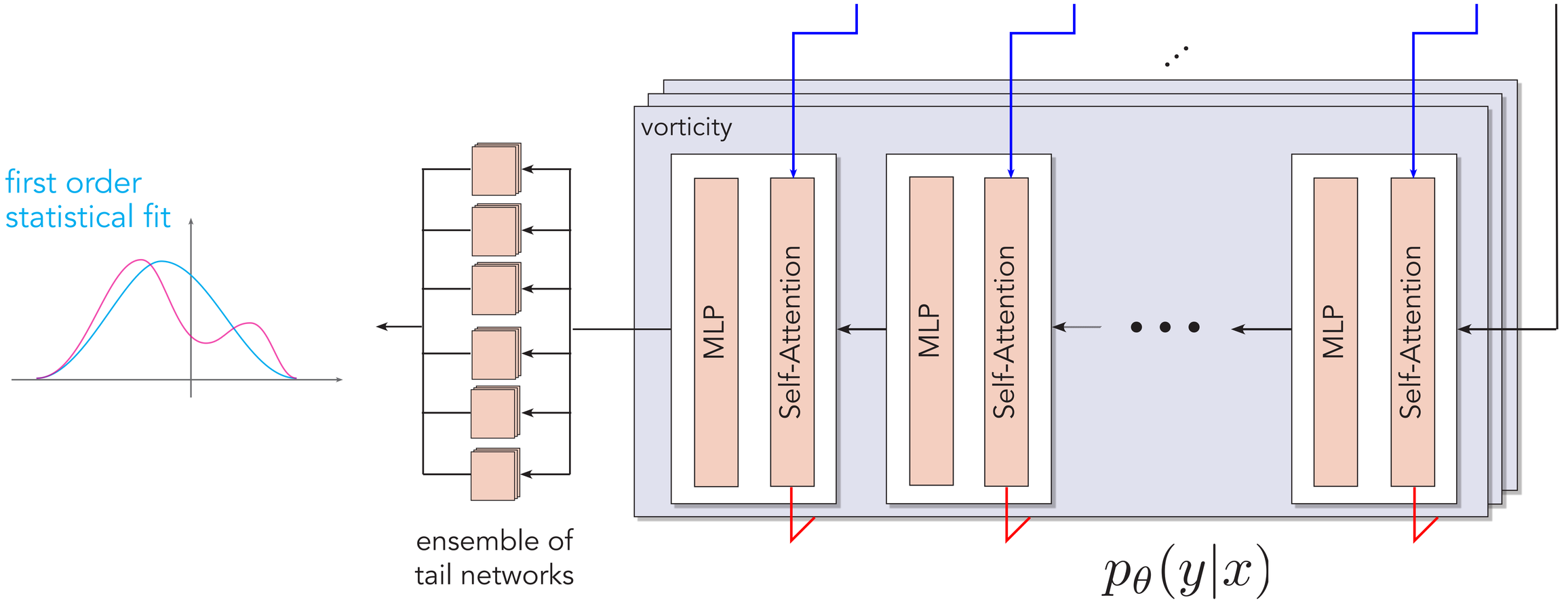
Statistical loss



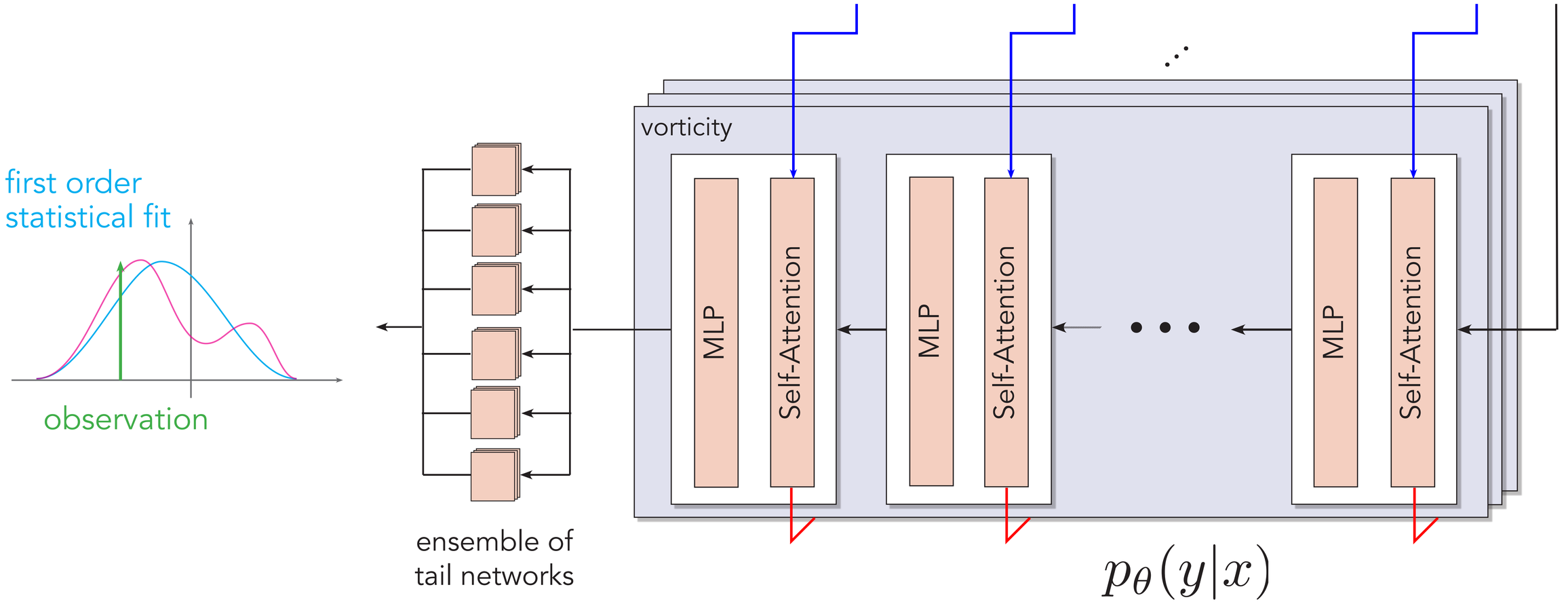
Statistical loss



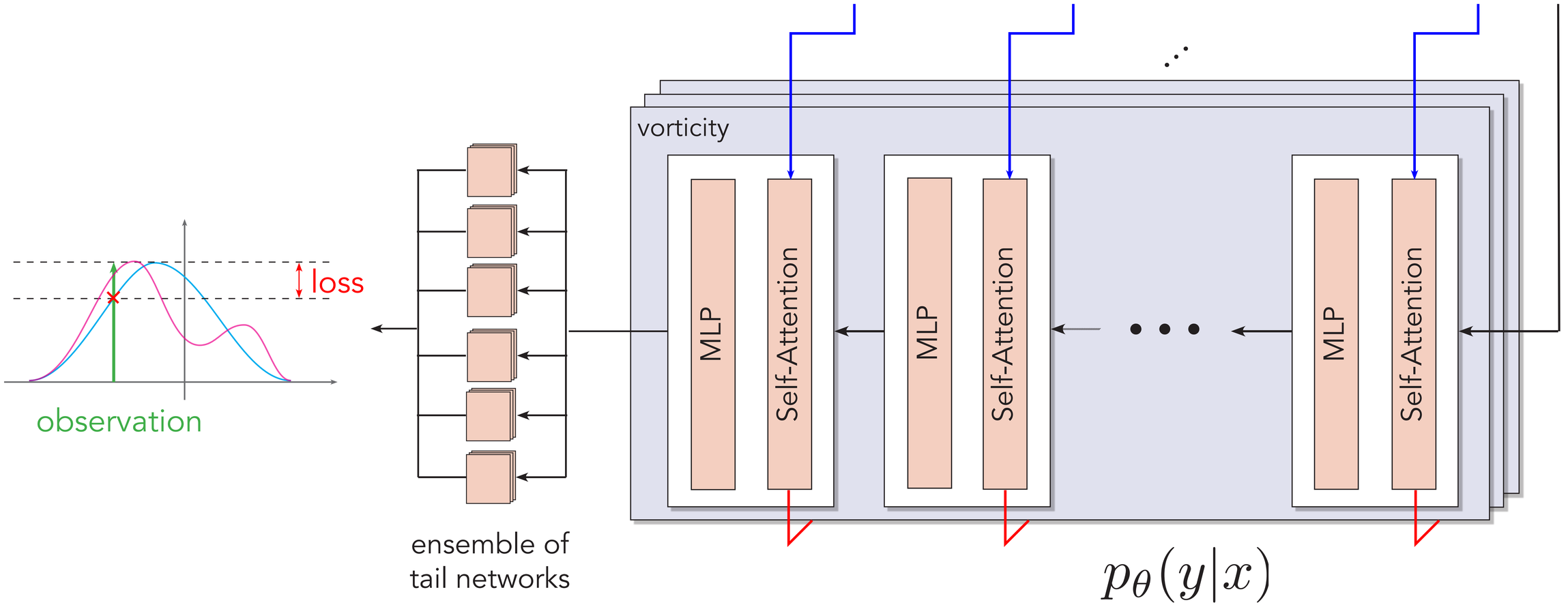
Statistical loss



Statistical loss

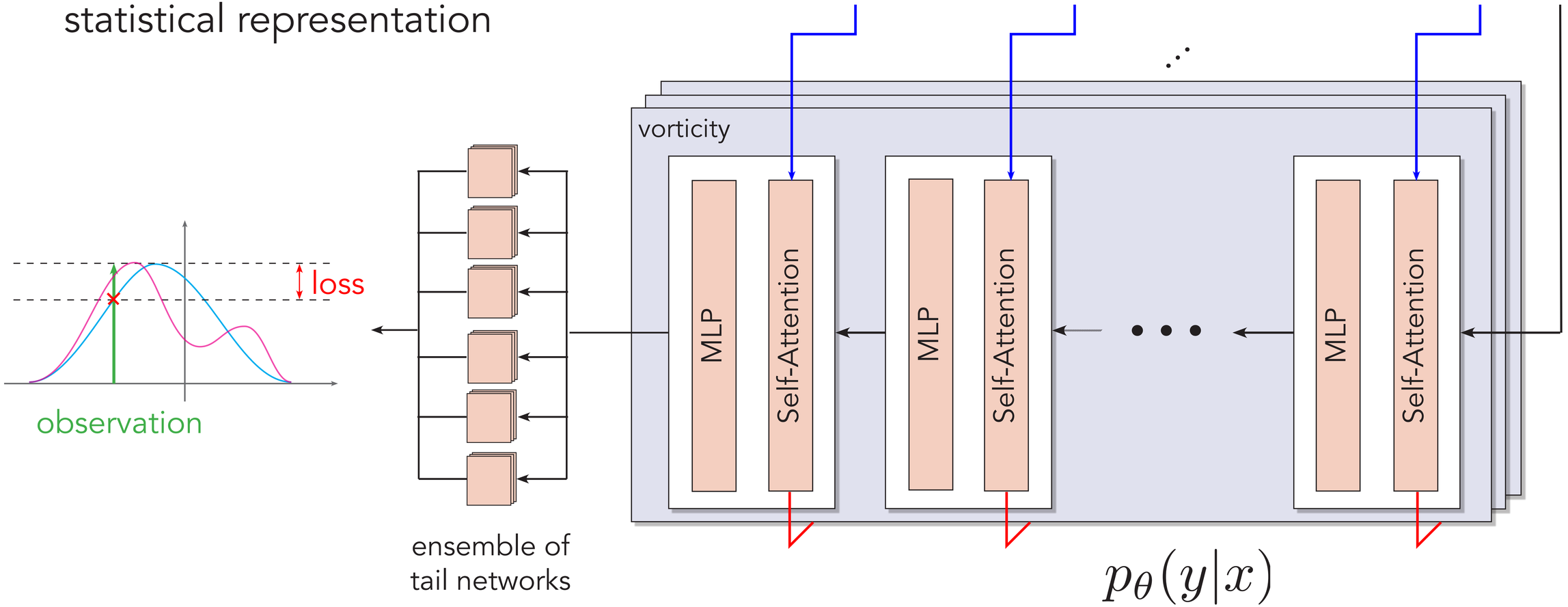


Statistical loss

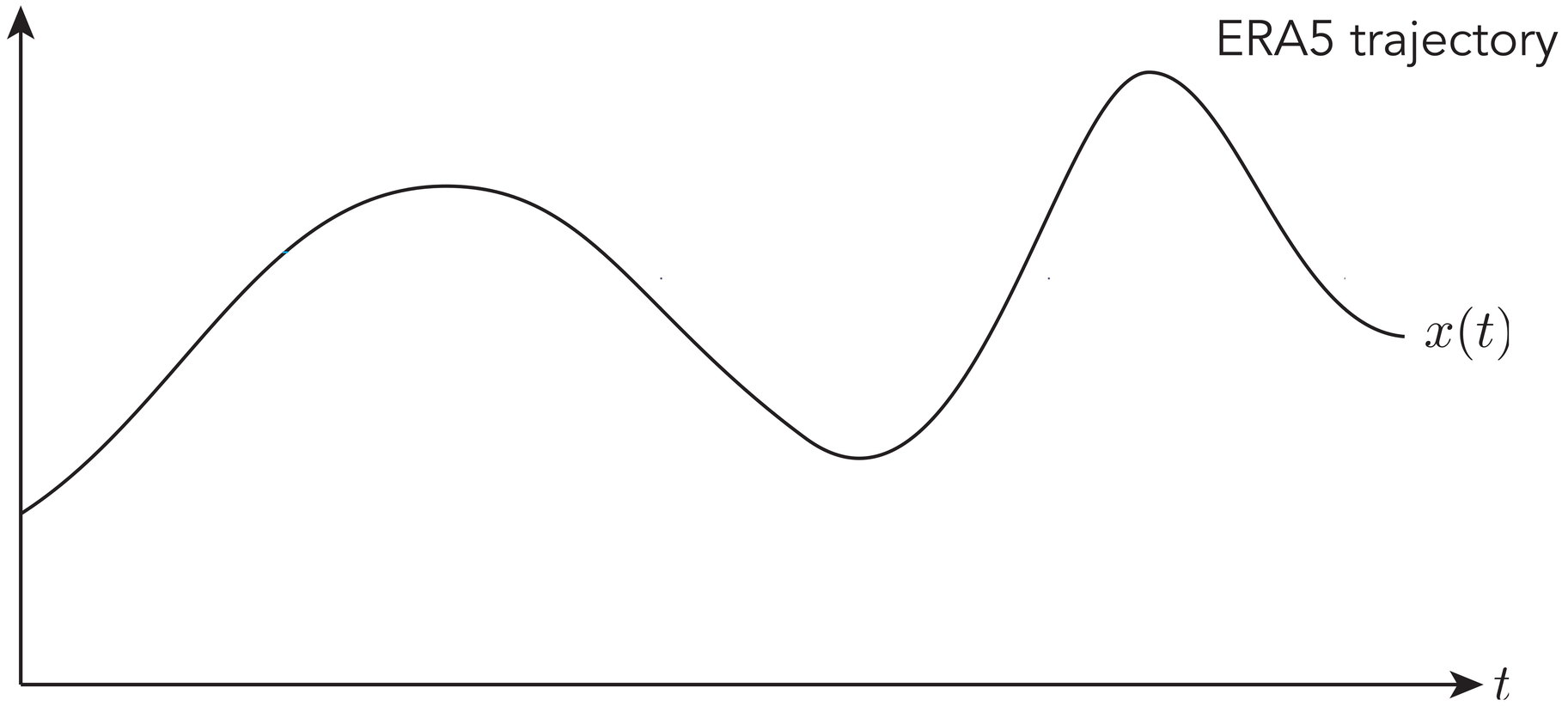


Statistical loss

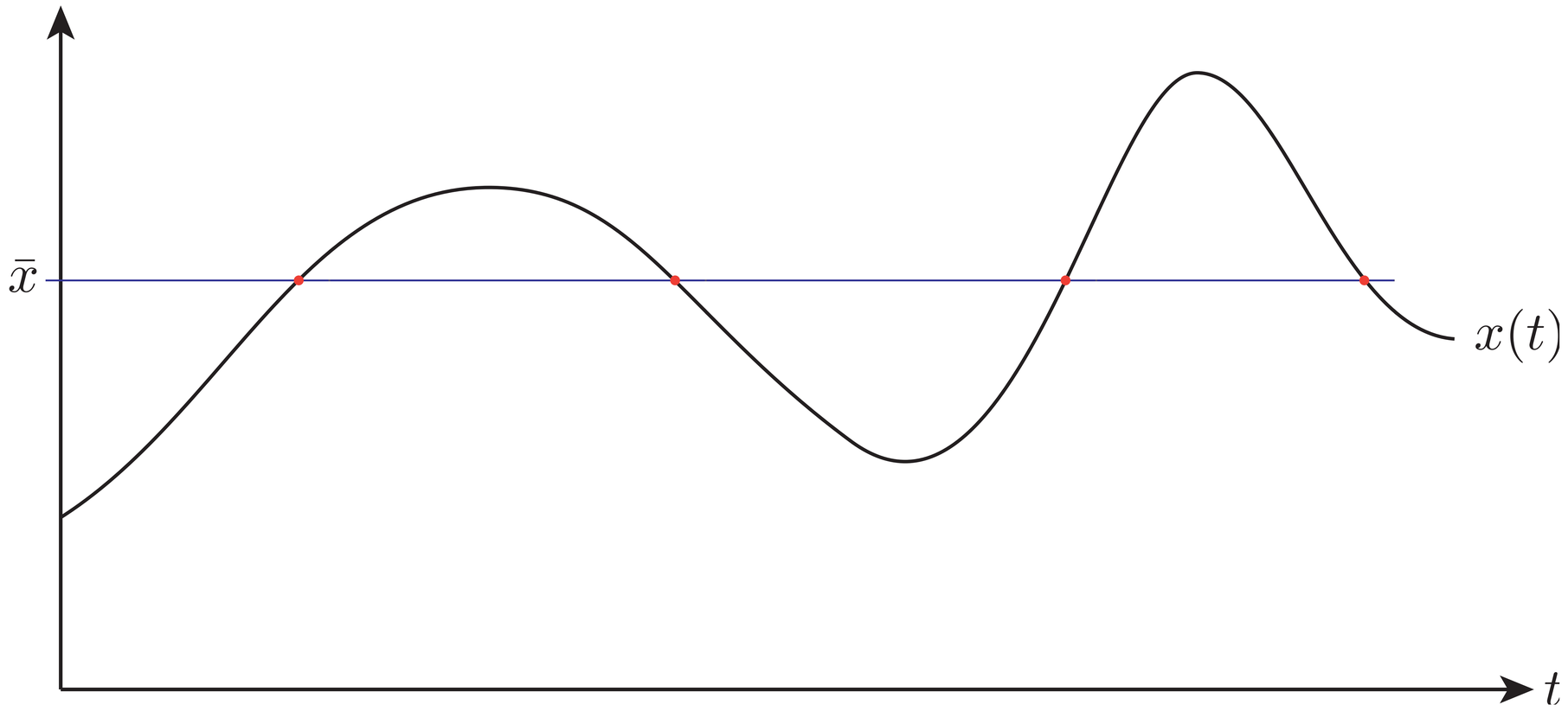
end-to-end training encourages
statistical representation



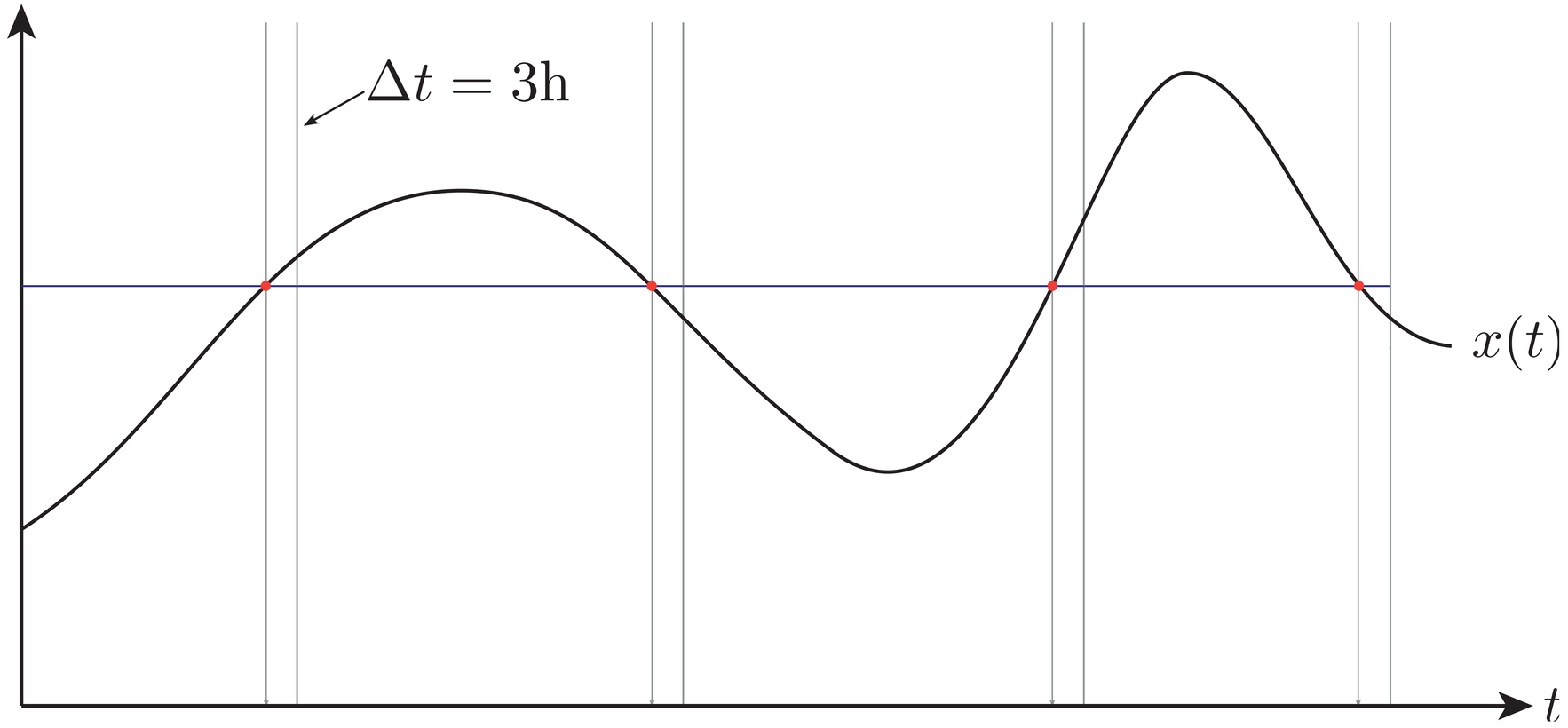
Statistical loss



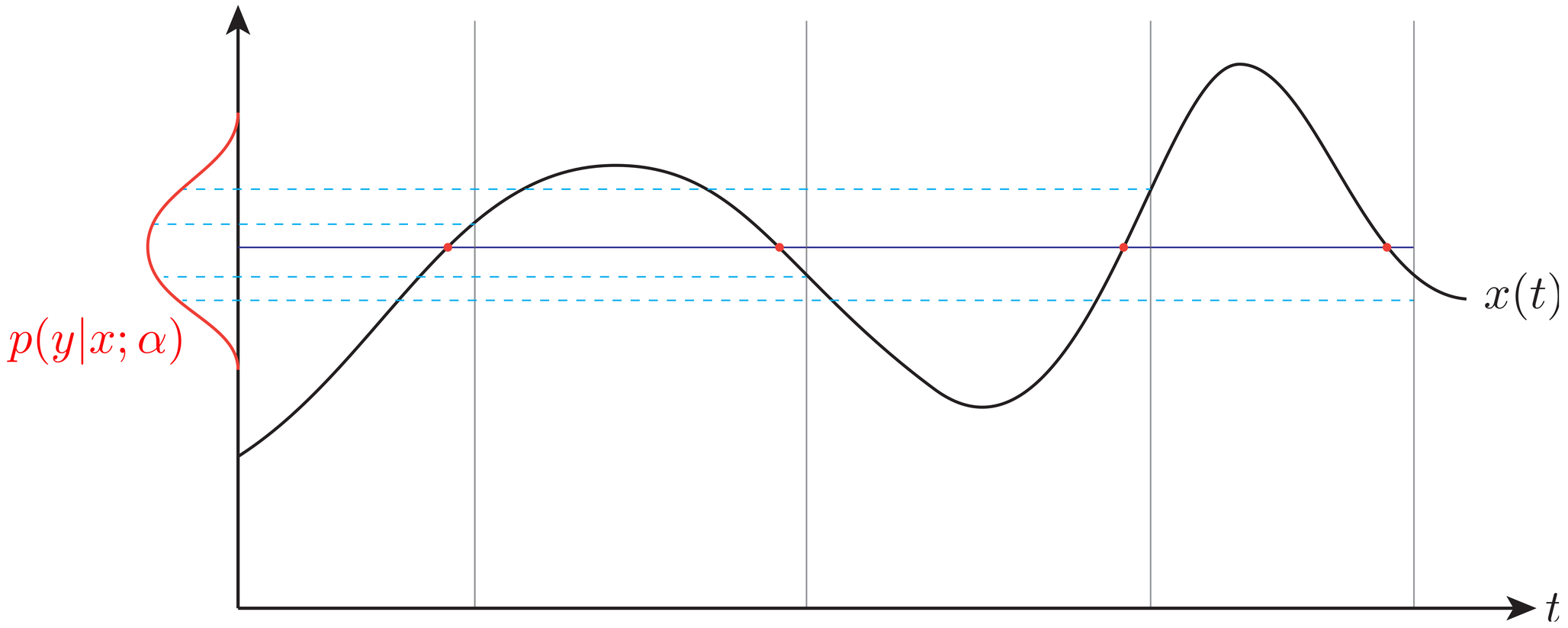
Statistical loss



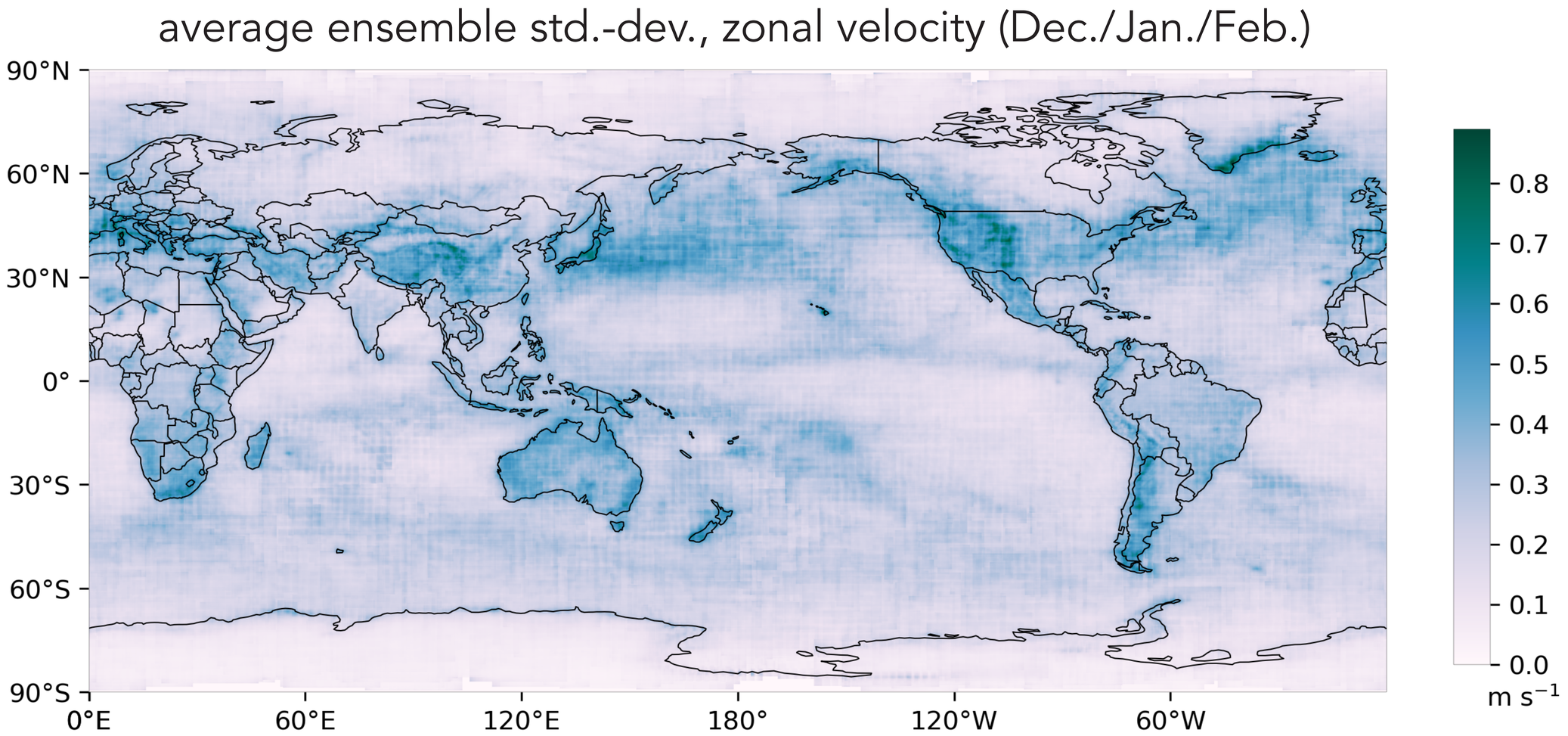
Statistical loss



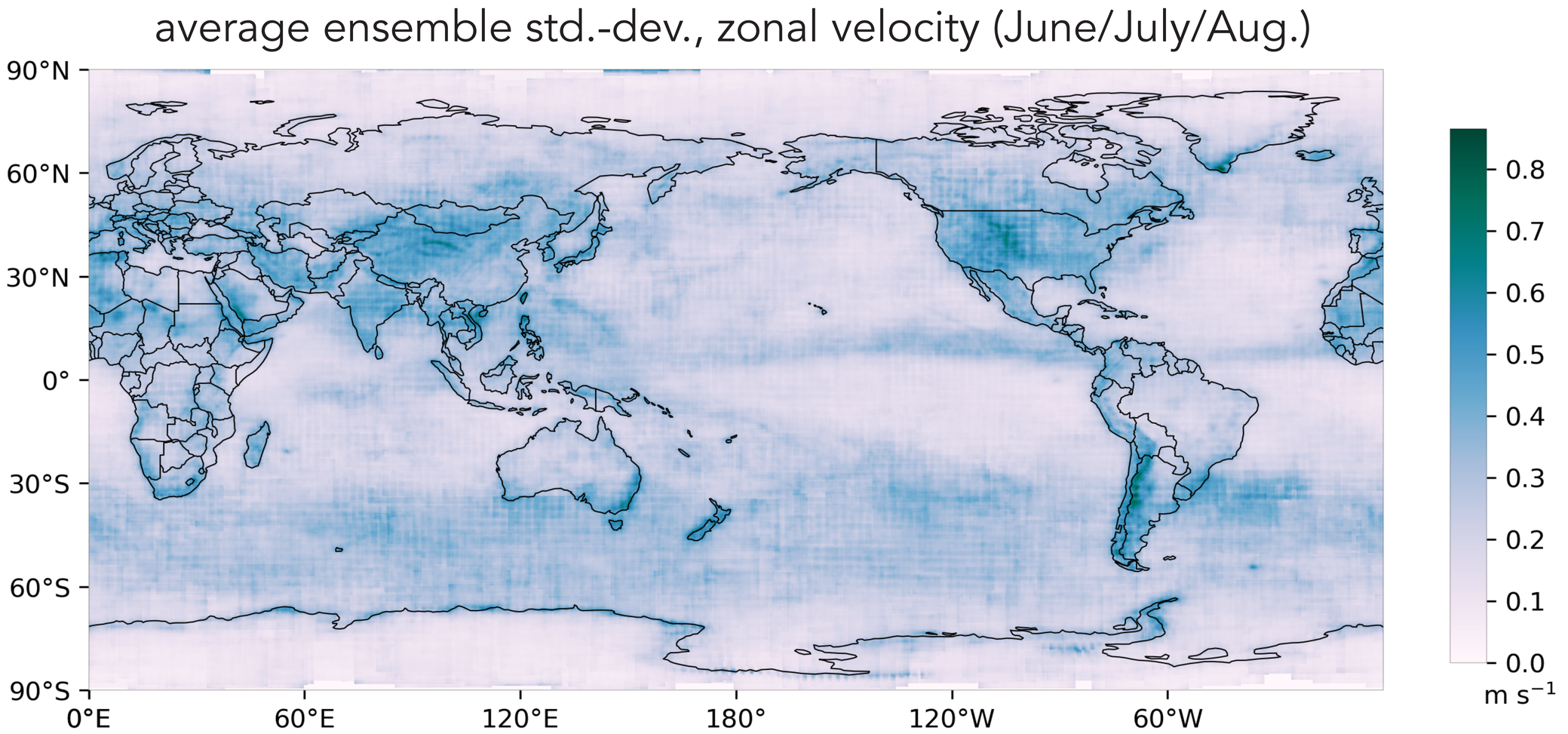
Statistical loss



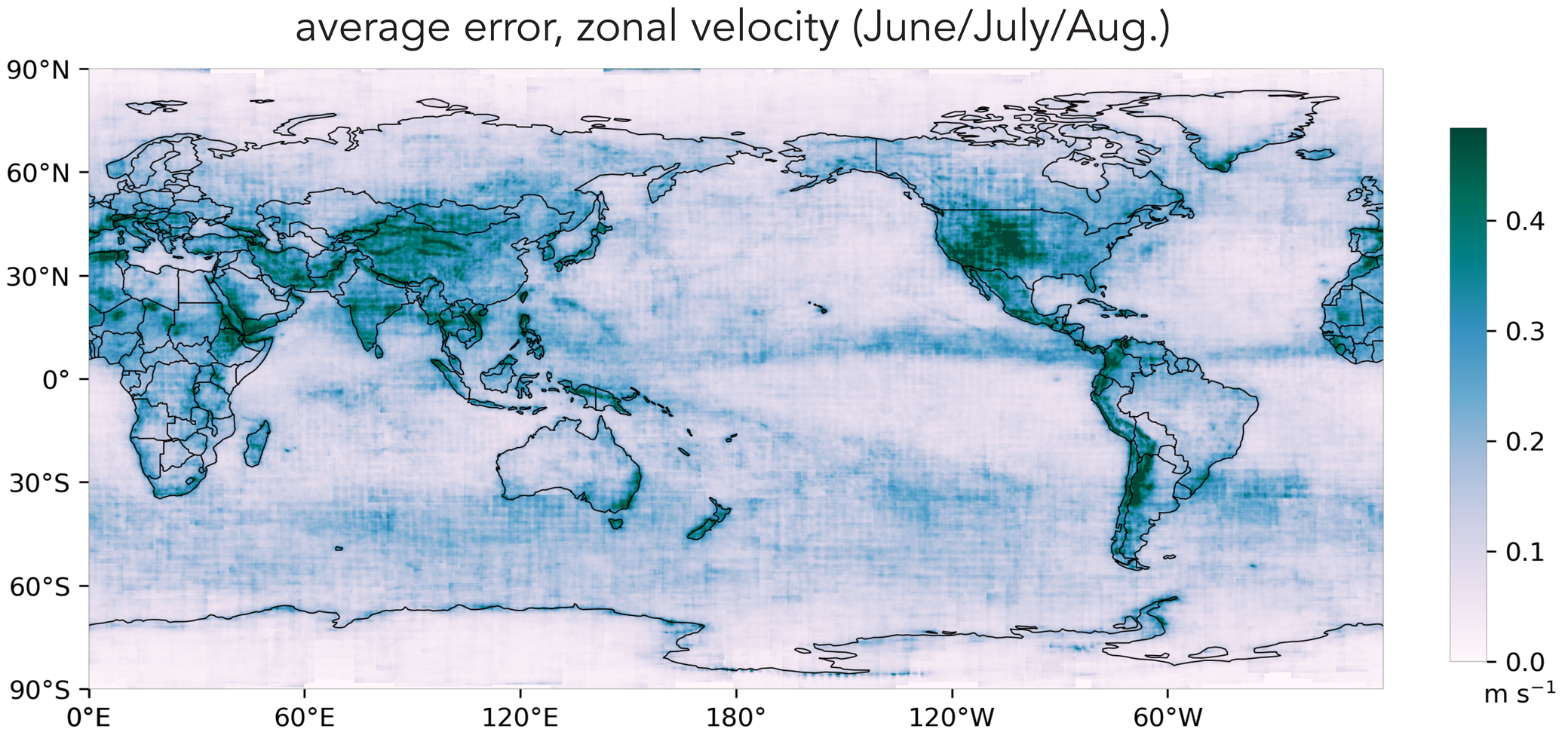
Statistical loss



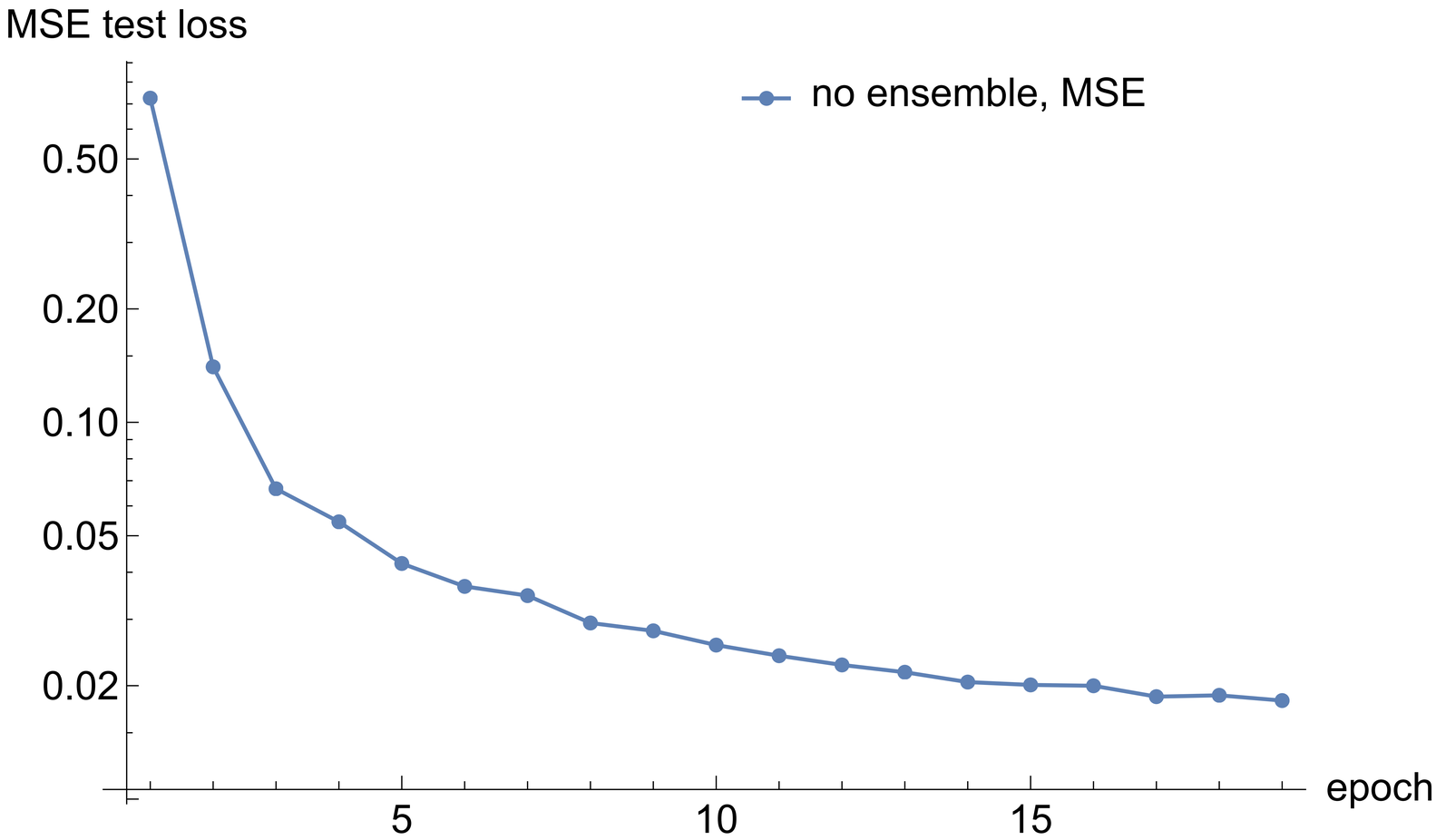
Statistical loss



Statistical loss

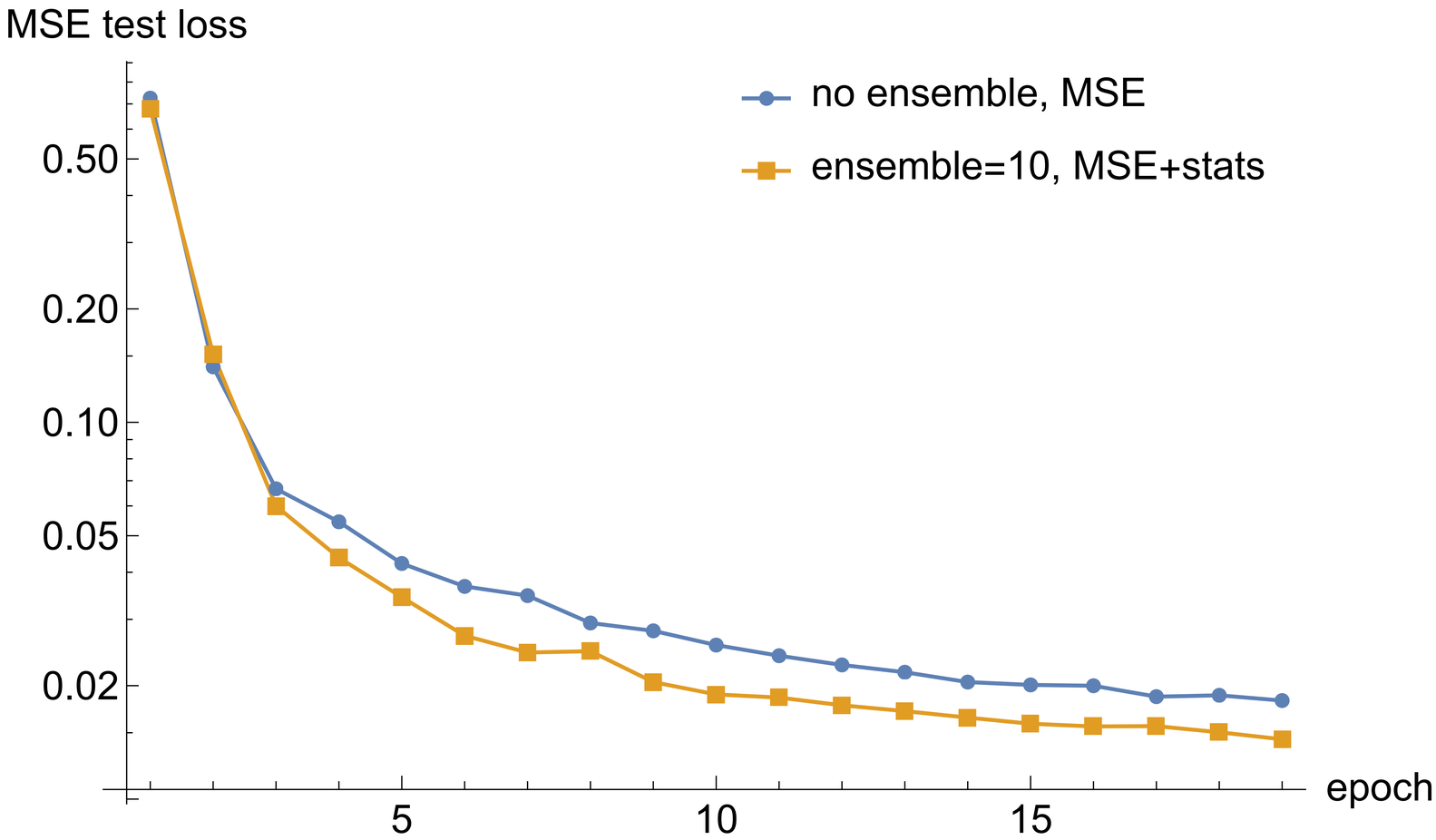


Statistical loss



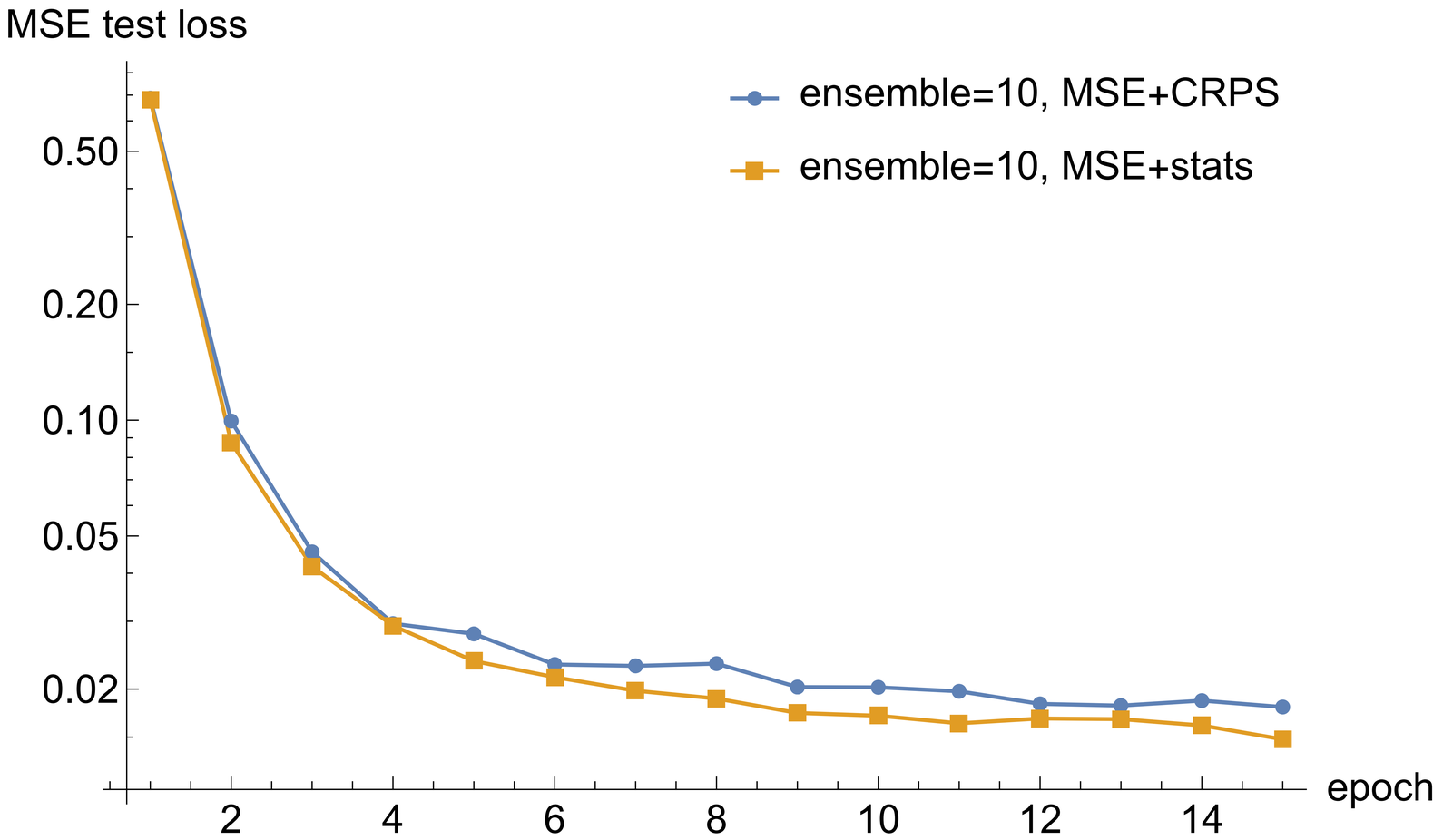
vorticity,
ml=137

Statistical loss



vorticity,
ml=123

Statistical loss



vorticity,
ml=123

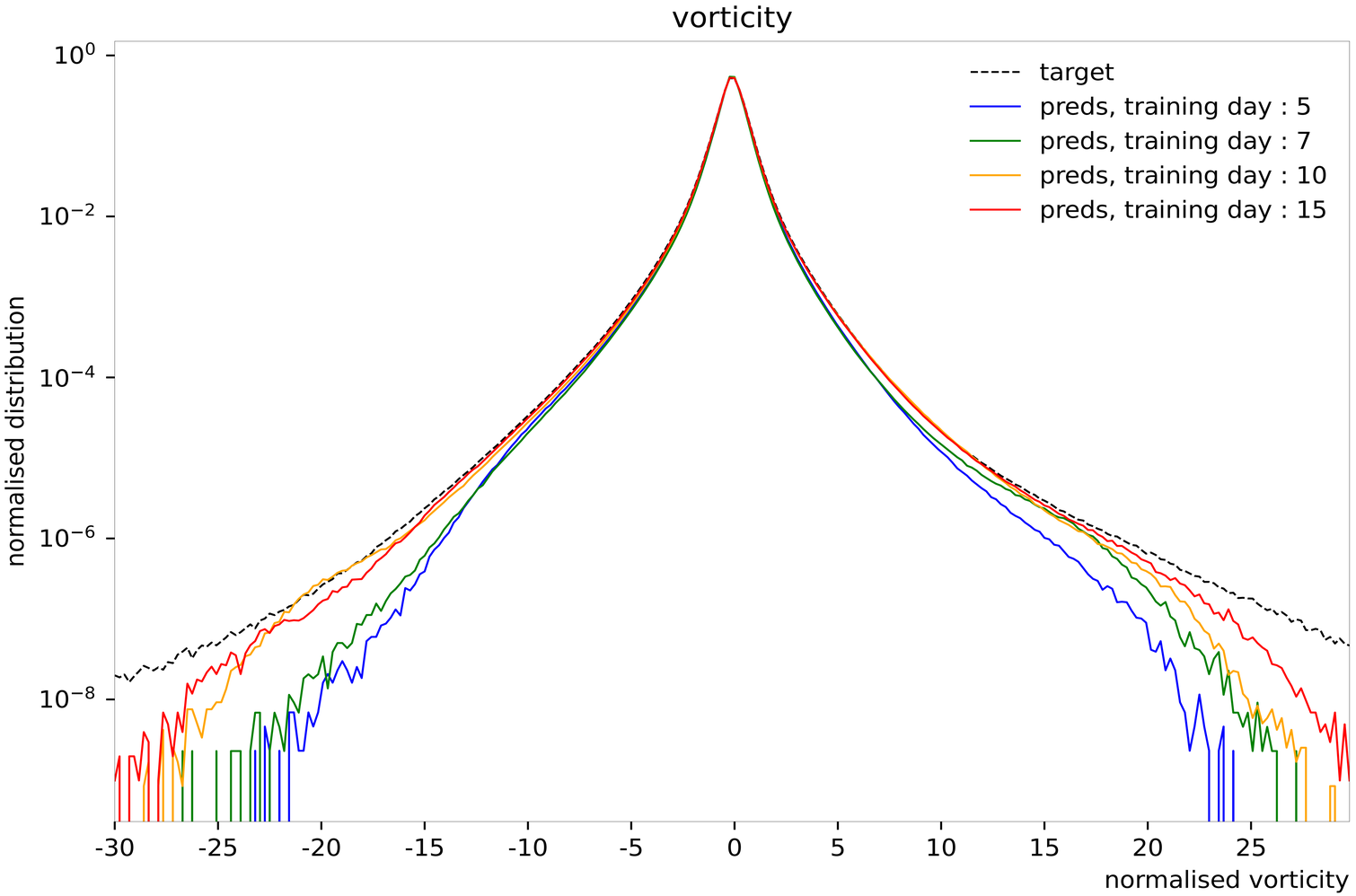
Pre-training results

- 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



Intrinsic Capabilities

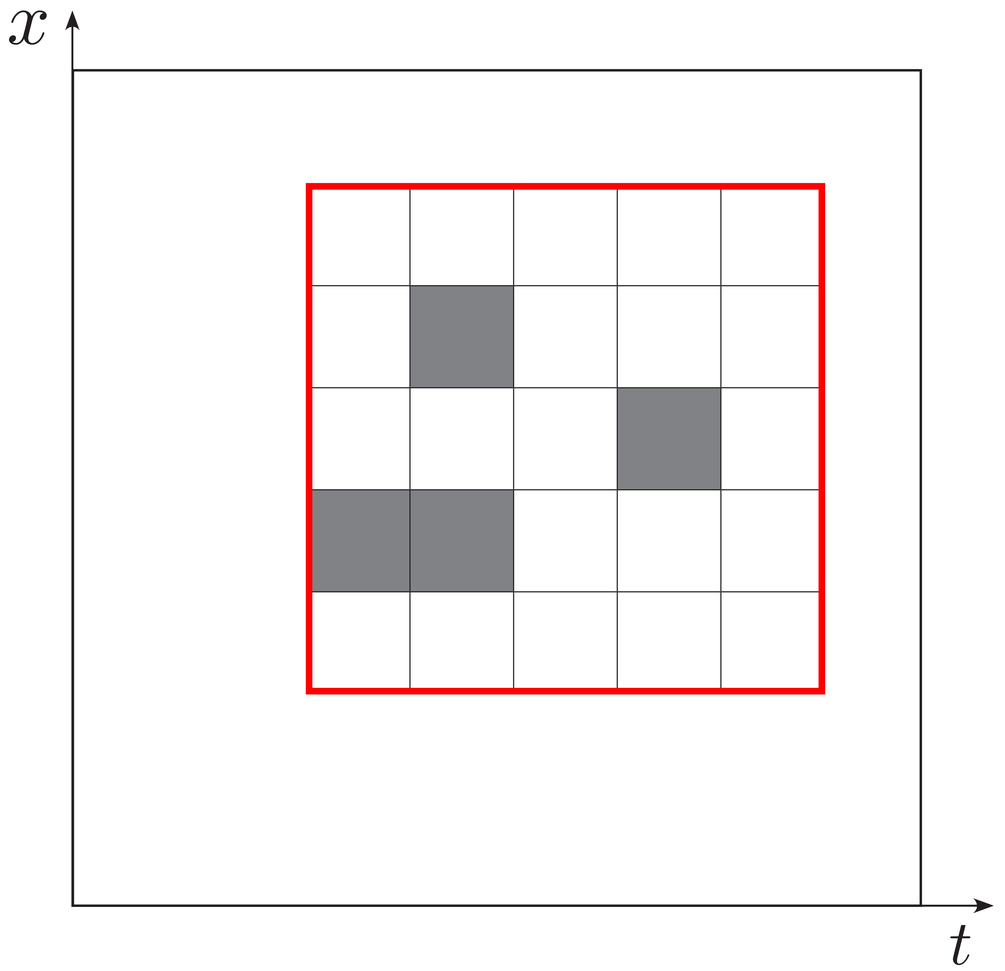
Intrinsic capabilities

- Numerical statistical atmospheric model:

$$p_\theta\left(y|x,\alpha\right)$$

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

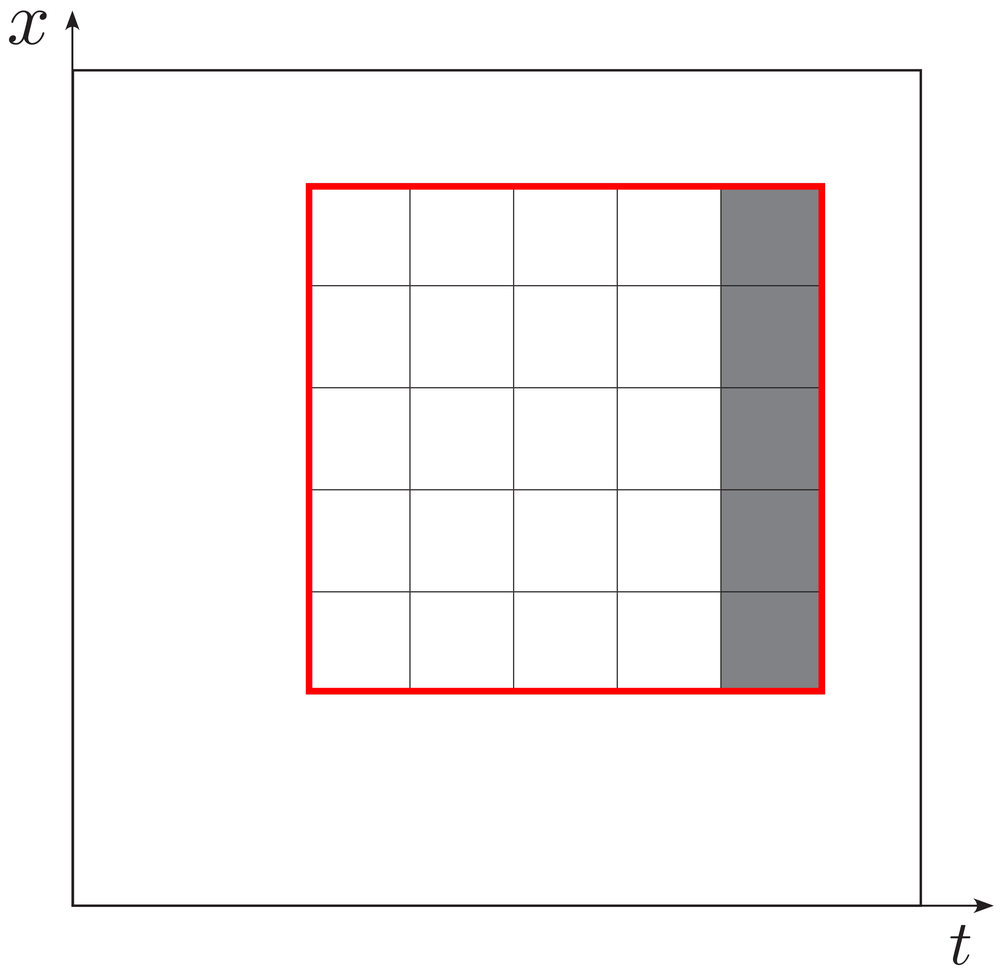
Intrinsic capabilities



Training task:

predict randomly masked neighbor-hoods in space-time

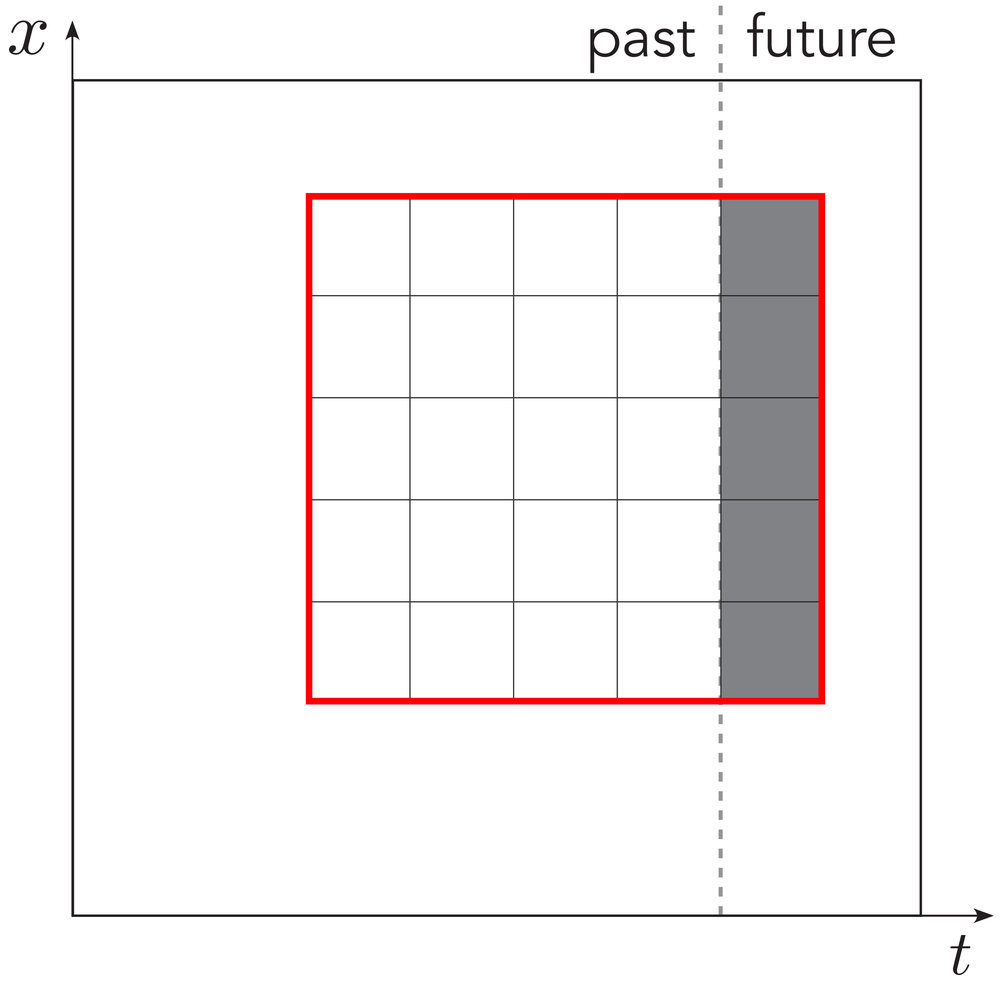
Intrinsic capabilities



Training task:

predict randomly masked neighbor-hoods in space-time

Intrinsic capabilities

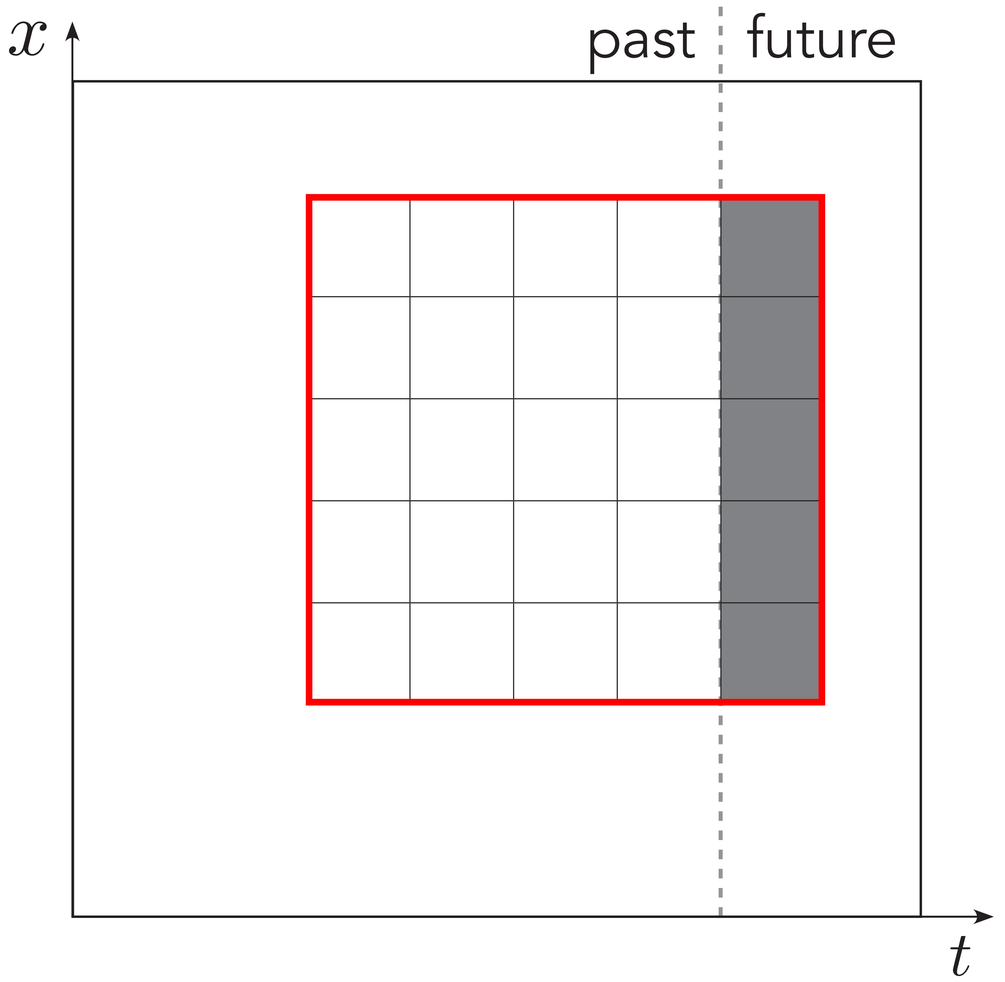


Training task:

predict randomly masked neighbor-hoods in space-time

Intrinsic capabilities

Forecasting



Training task:

predict randomly masked neighbor-hoods in space-time

Intrinsic capabilities

Large language model: $p_\theta(x, y)$

- x, y are text sequences

Intrinsic capabilities

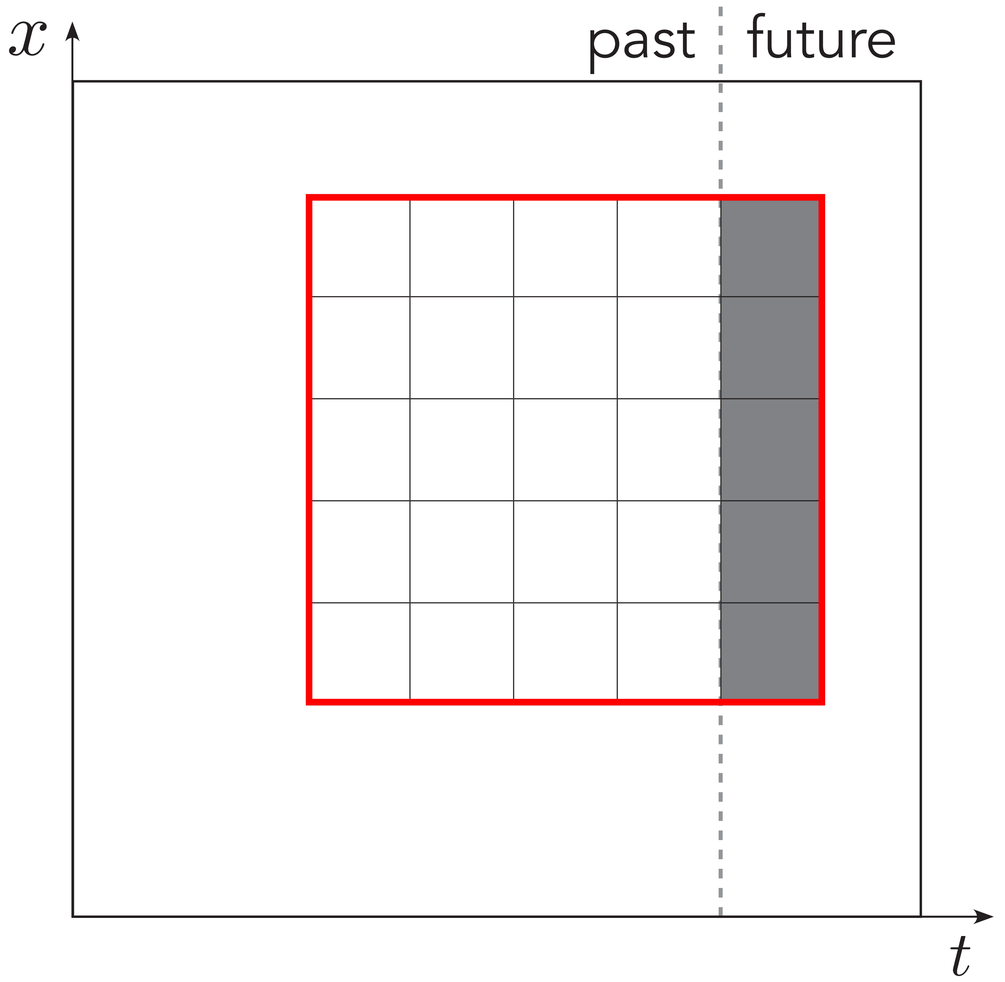
Large language model: $p_\theta(x, y)$

- x, y are text sequences
- Learning probability distribution over x, y includes many relevant applications as special cases, e.g.
  - auto-completion, spell/grammar correction, translation, ...
- Known as **zero-shot capabilities** in the literature[1]

[1] 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.

Intrinsic capabilities

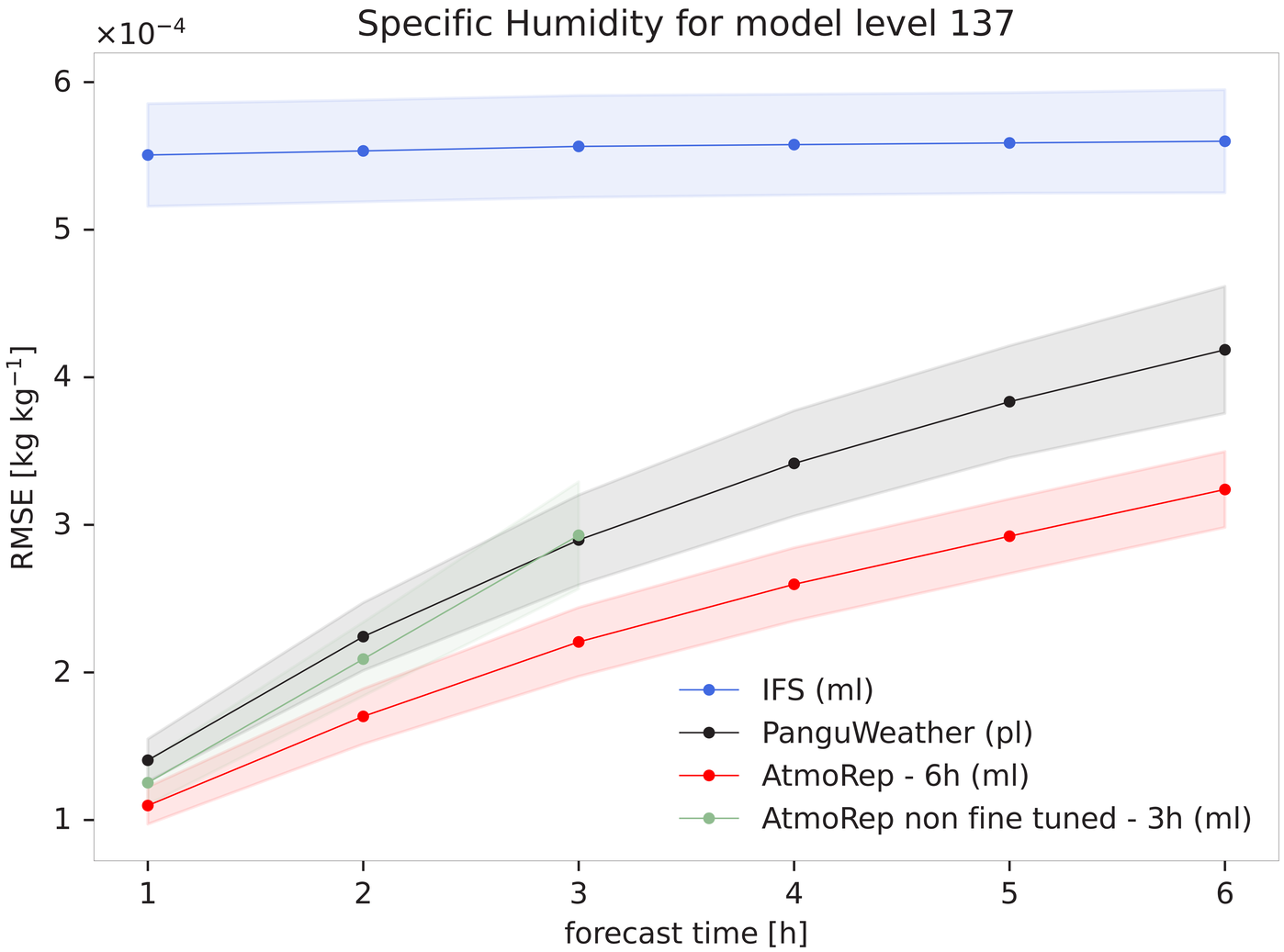
Forecasting



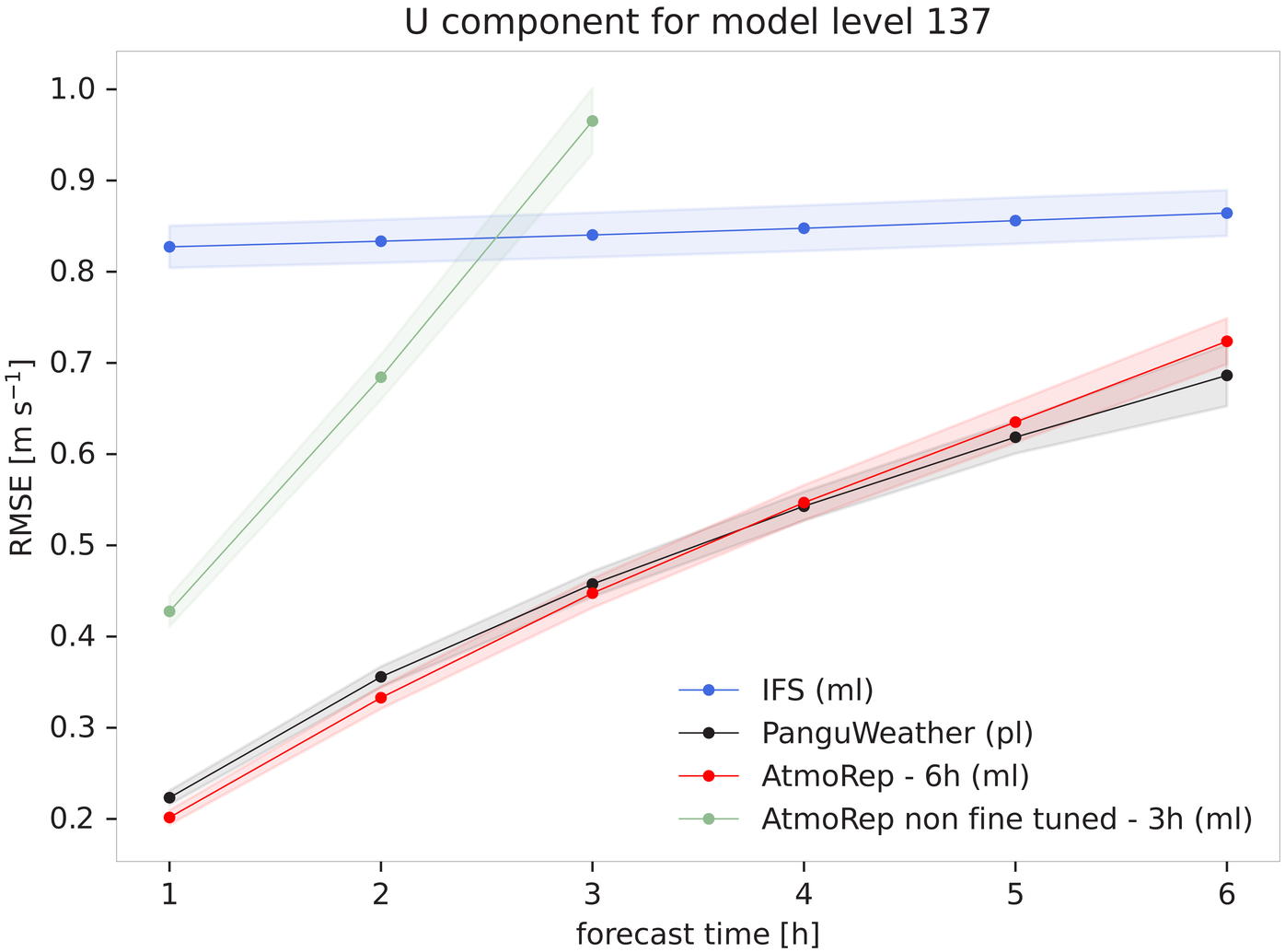
Training task:

predict randomly masked neighbor-hoods in space-time

Intrinsic capabilities

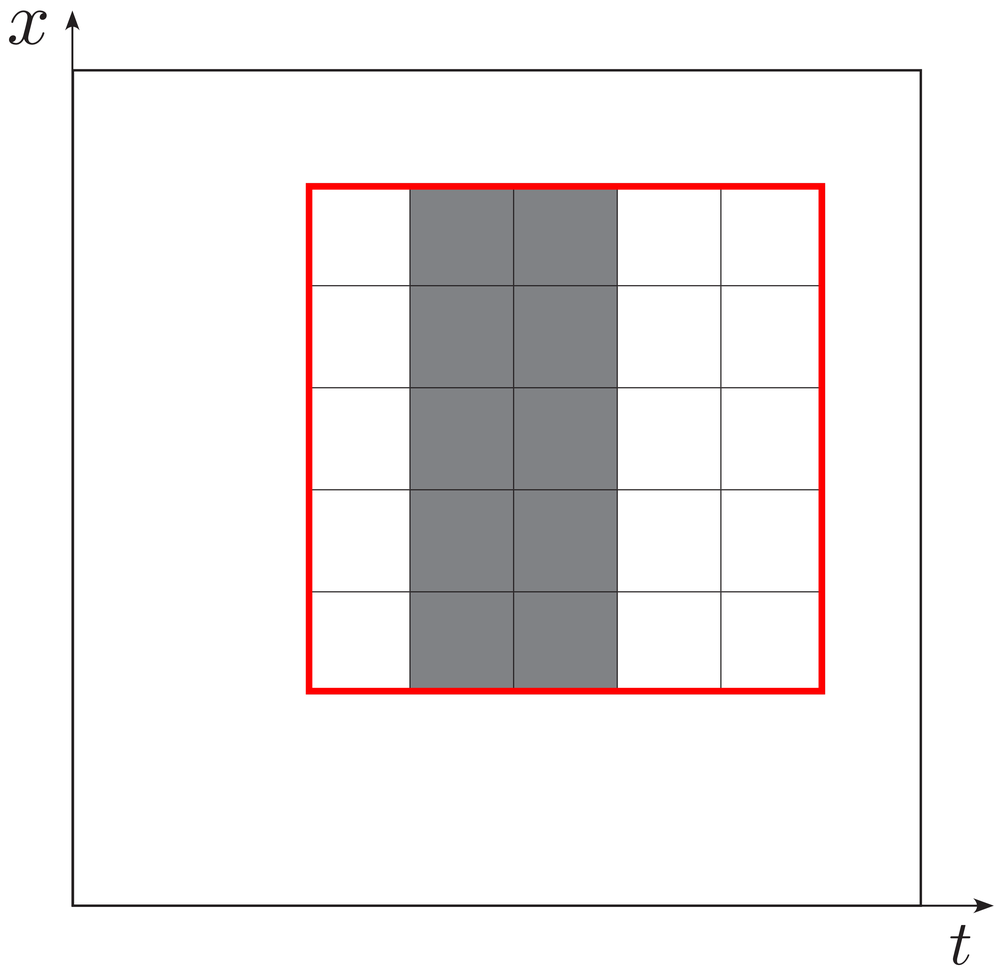


Intrinsic capabilities



Intrinsic capabilities

Temporal interpolation



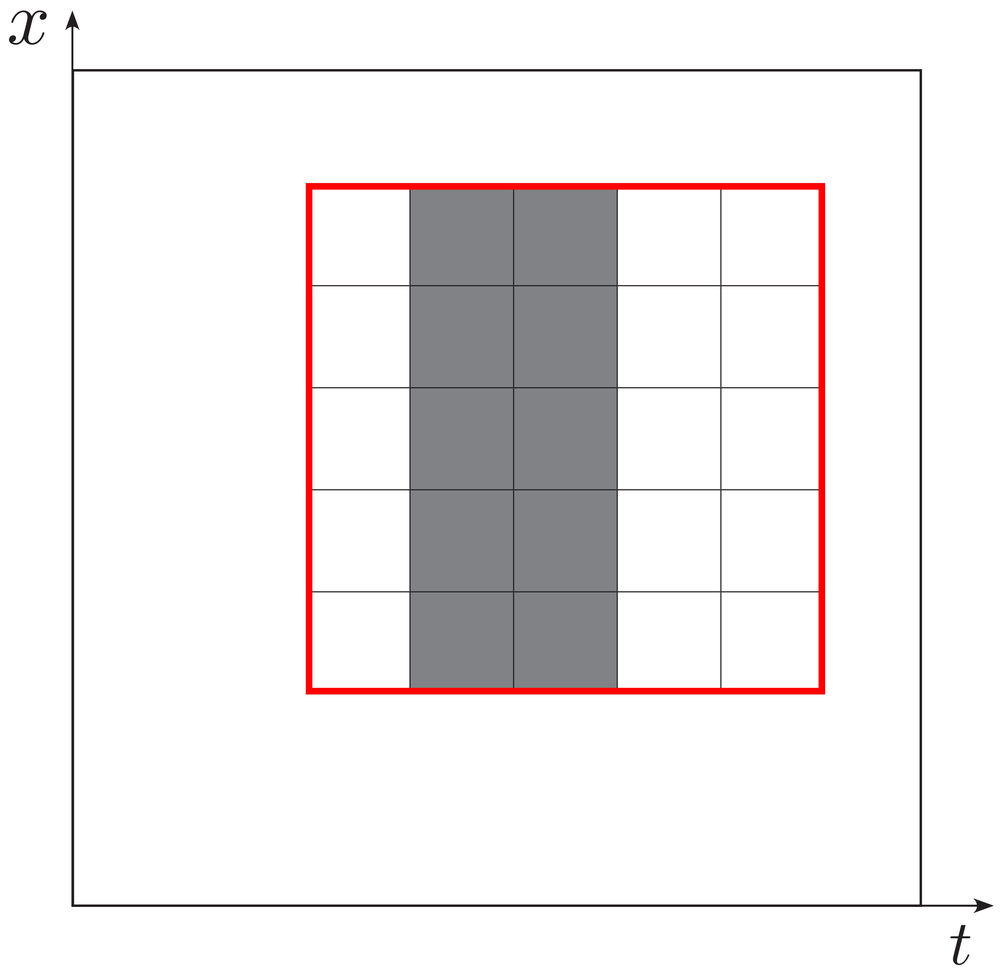
Training task:

predict randomly masked neighbor-hoods in space-time

Intrinsic capabilities

Also:

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



Training task:

predict randomly masked neighborhoods in space-time

Model correction

- Numerical statistical atmospheric model:

$$p_\theta\left(y|x,\alpha\right)$$

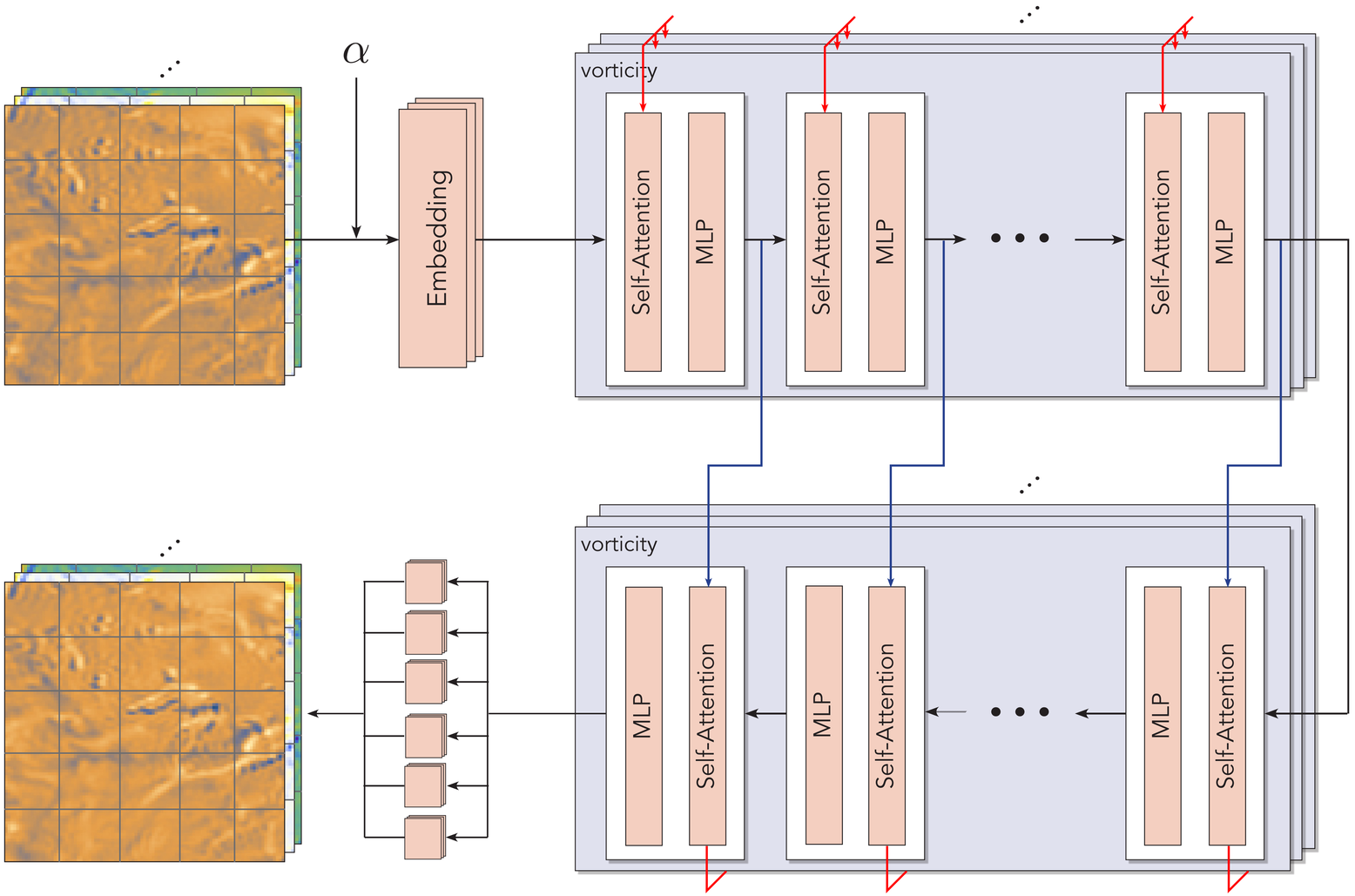
Model correction

- Numerical statistical atmospheric model:

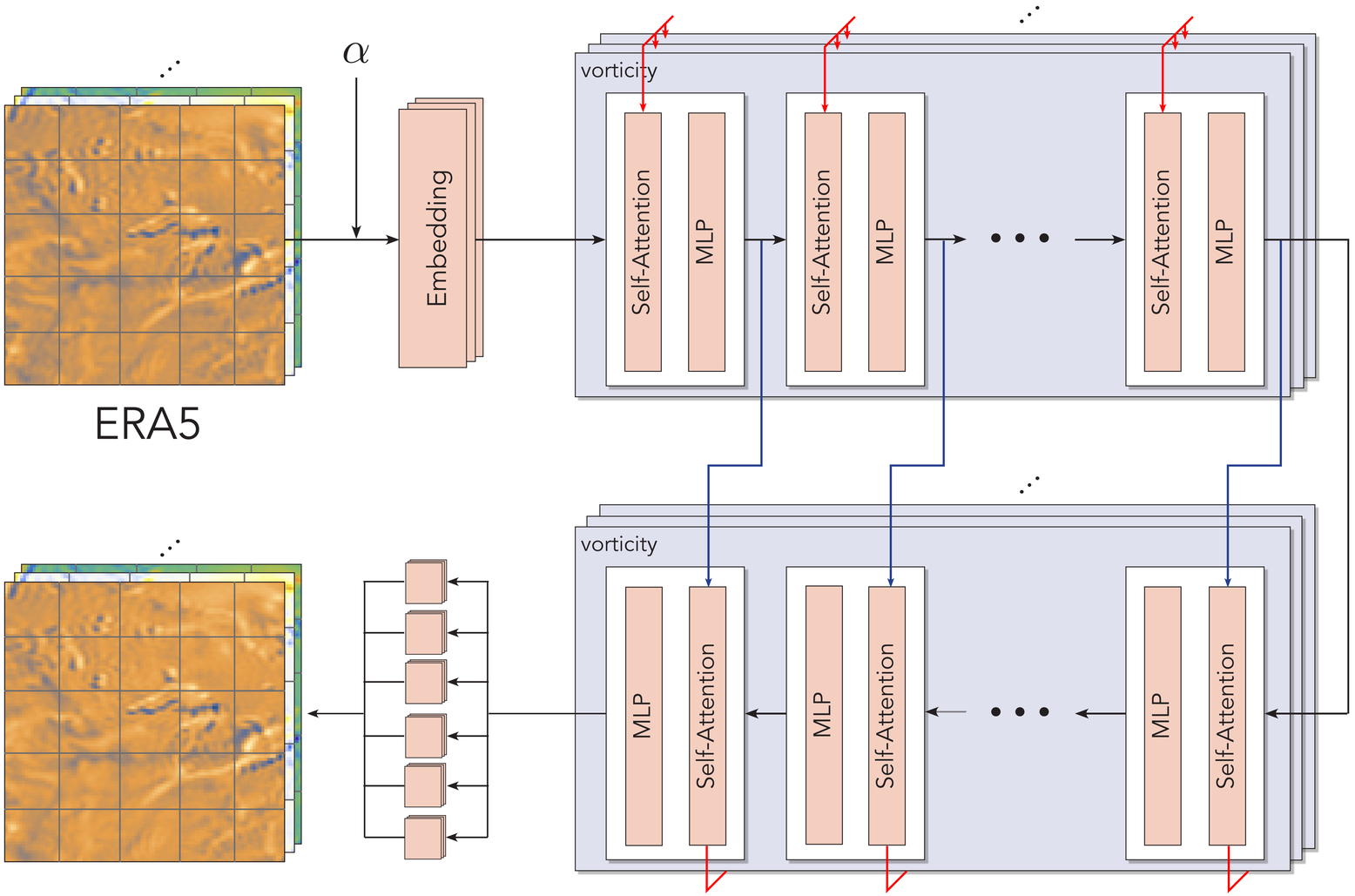
$$p_\theta\left(y|x,\alpha\right)$$

approximate initial state

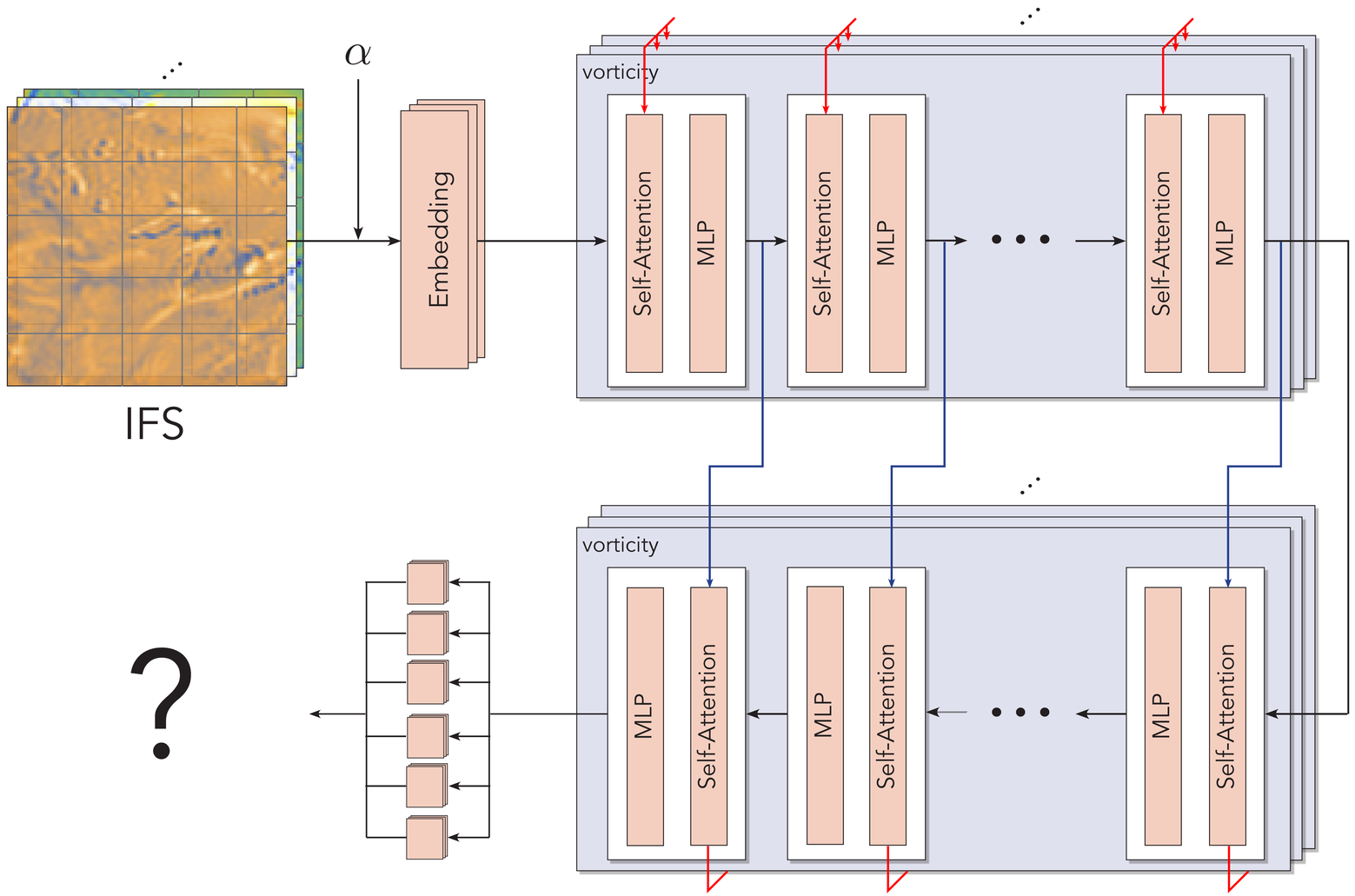
Model correction



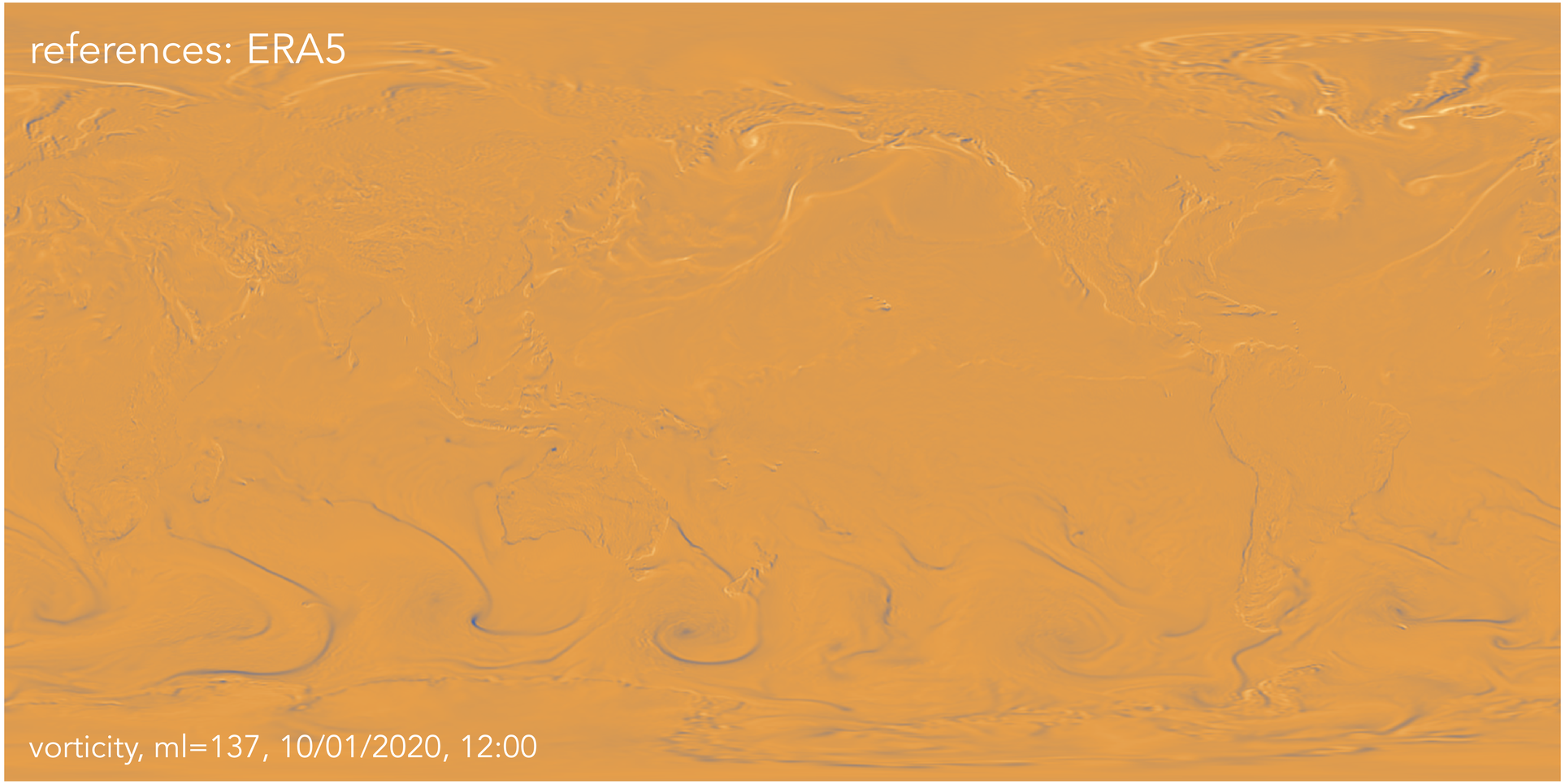
Model correction



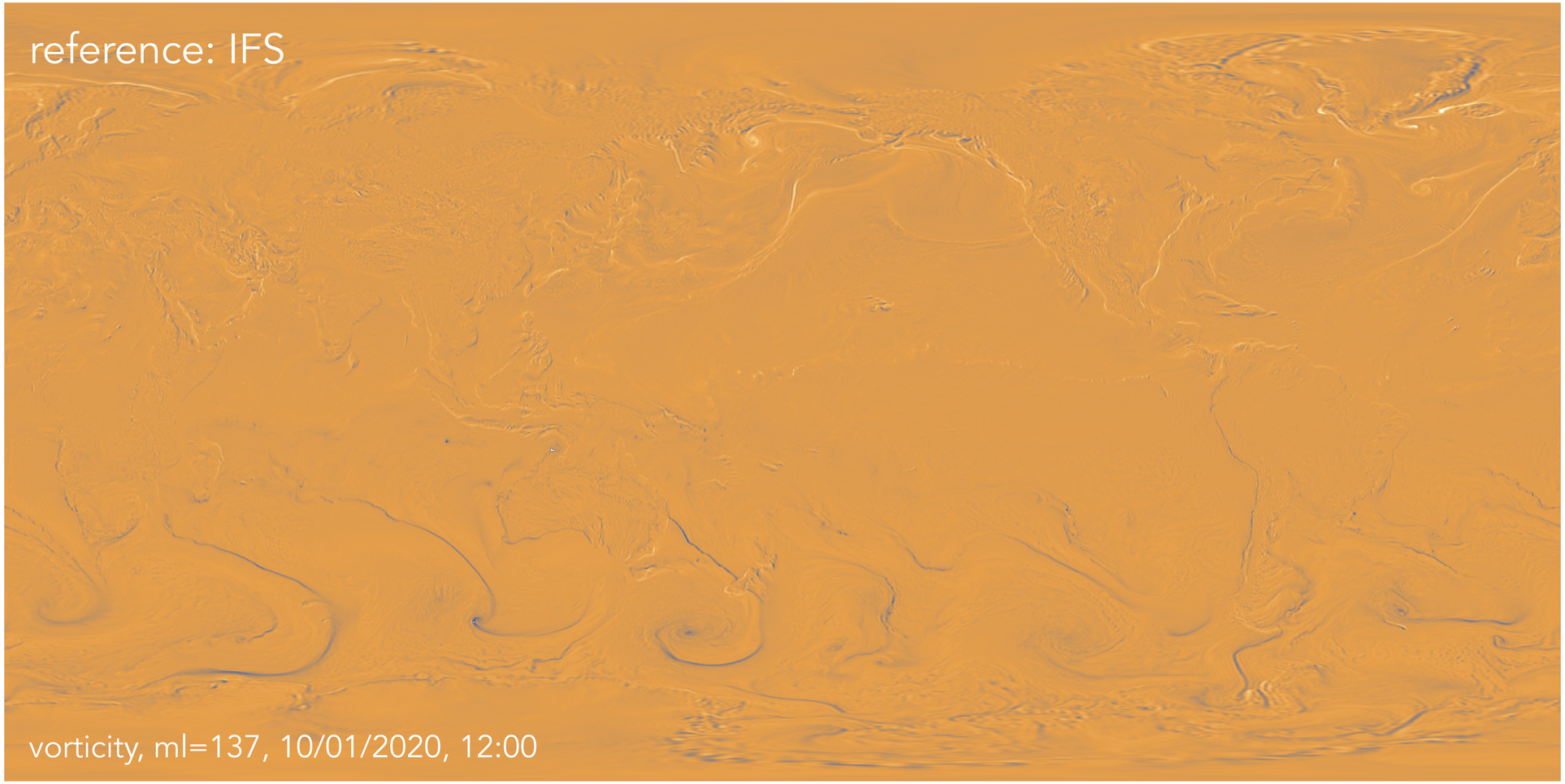
Model correction



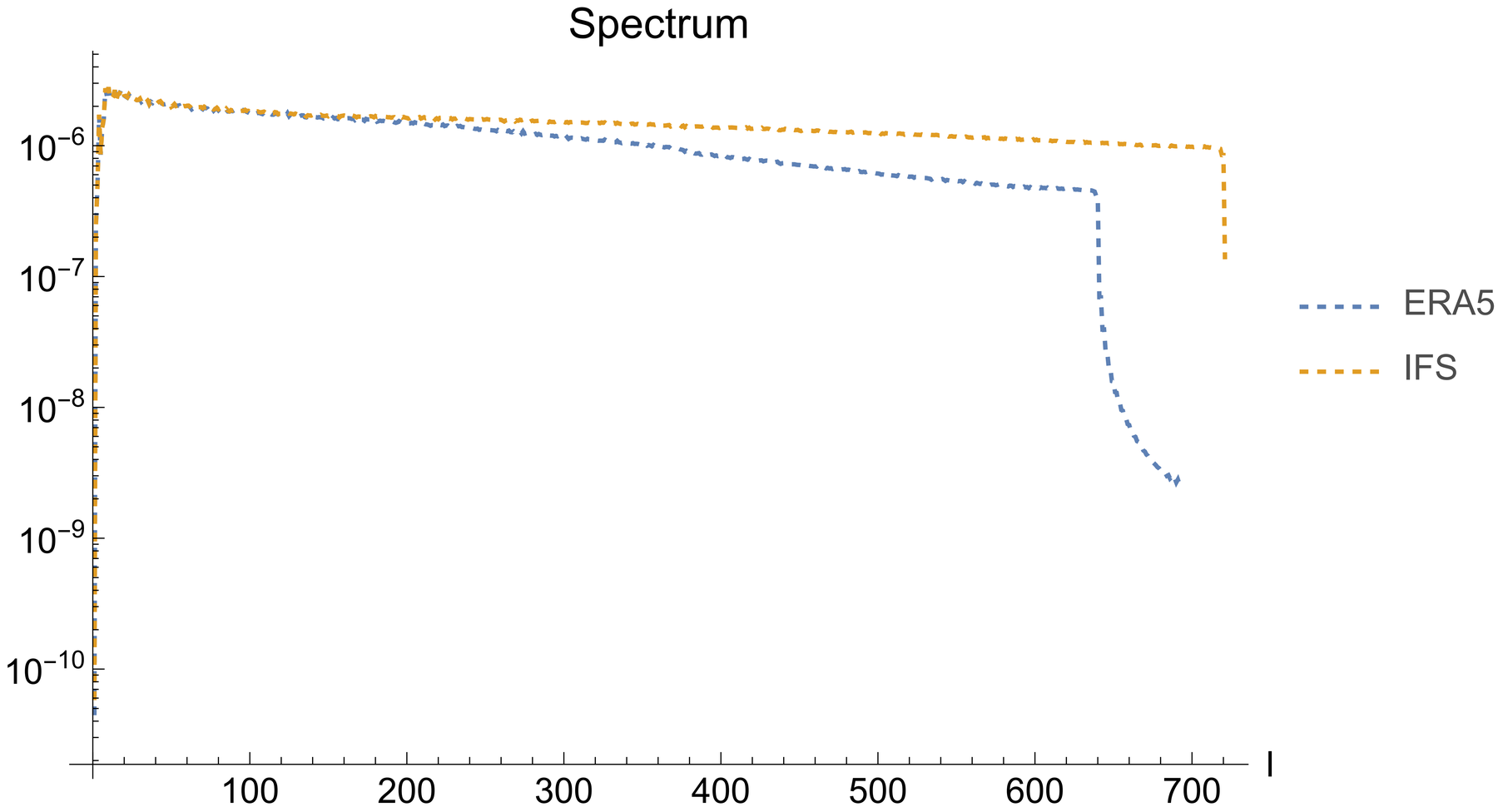
Model correction



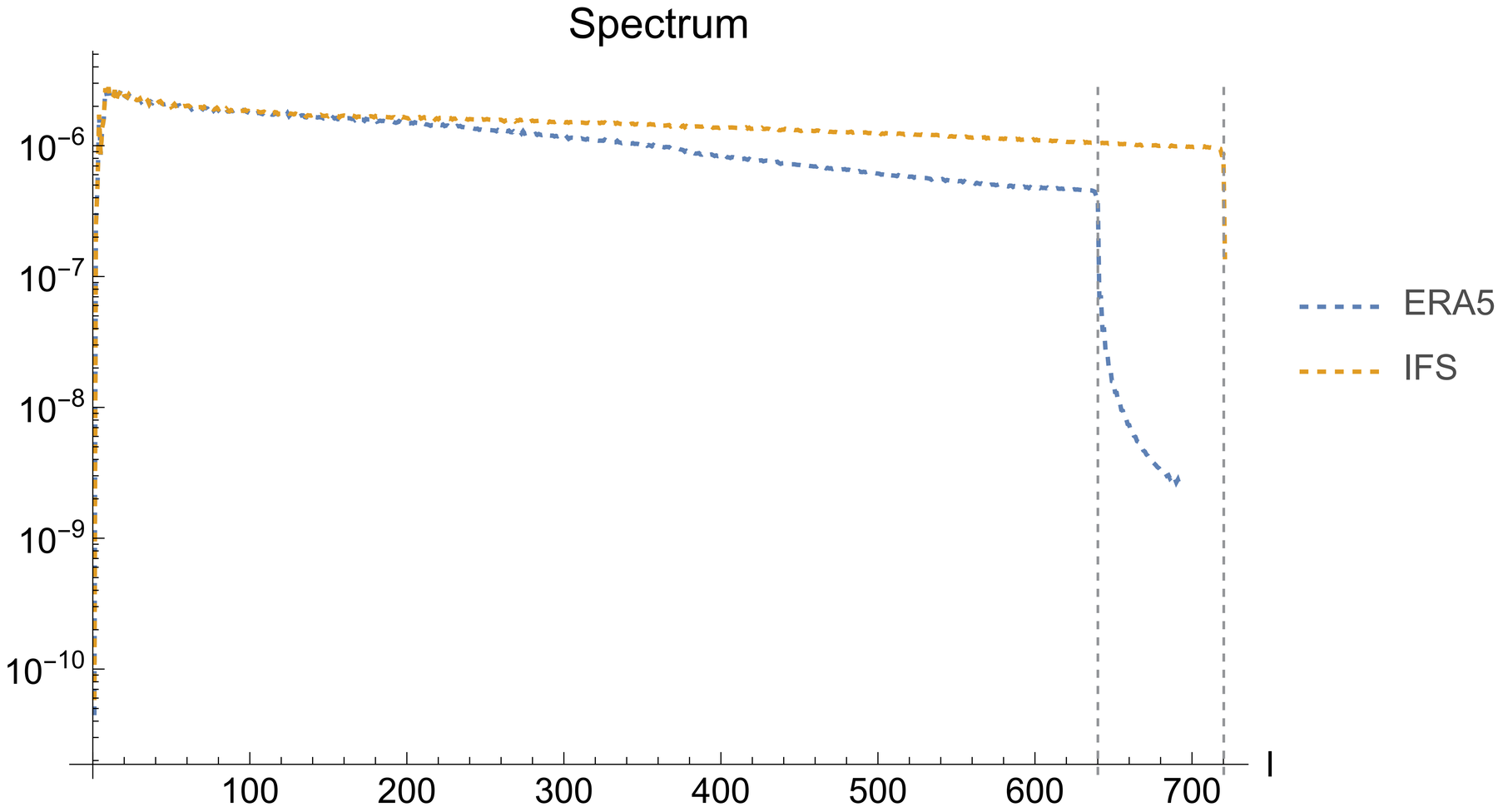
Model correction



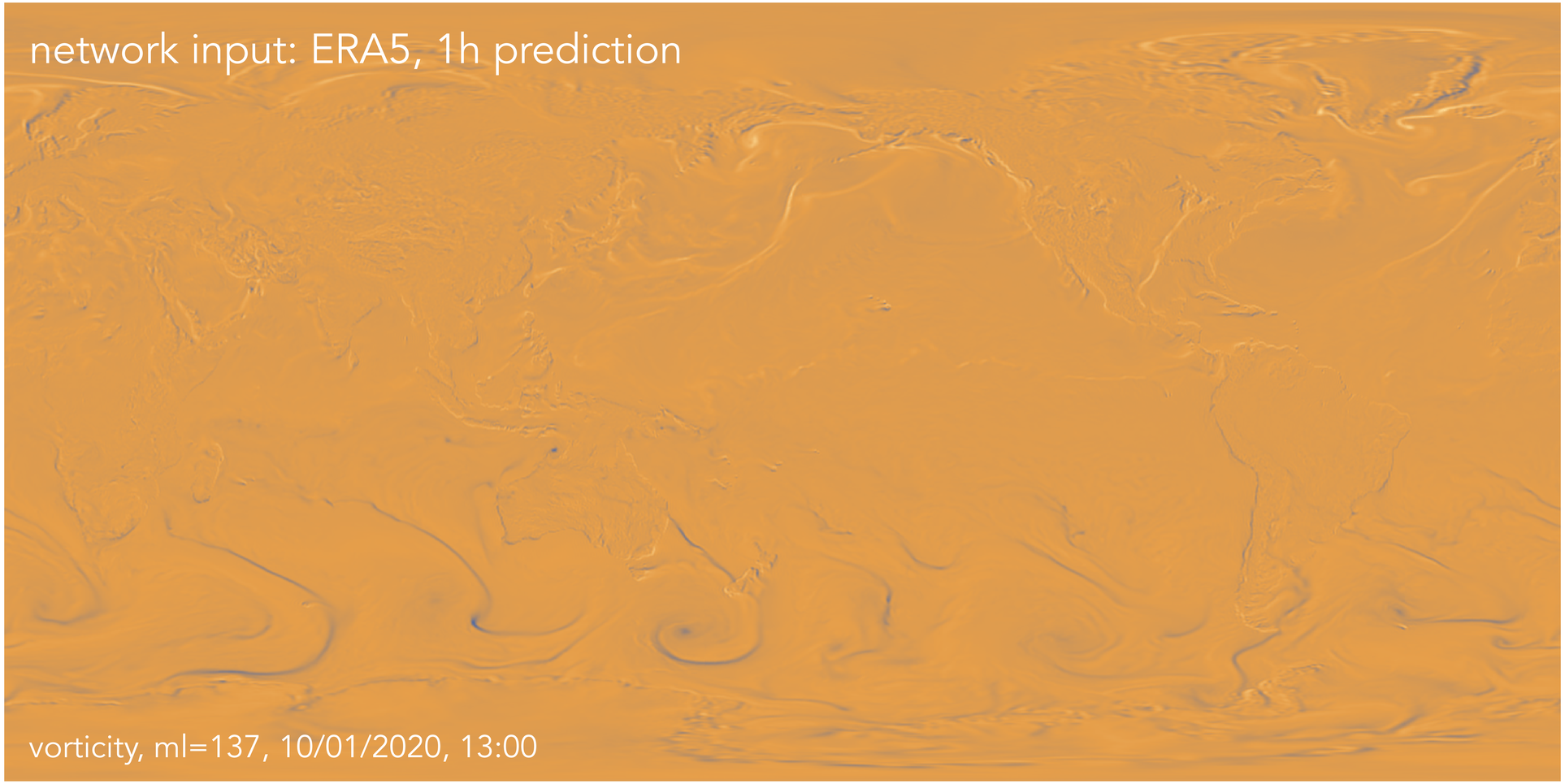
Model correction



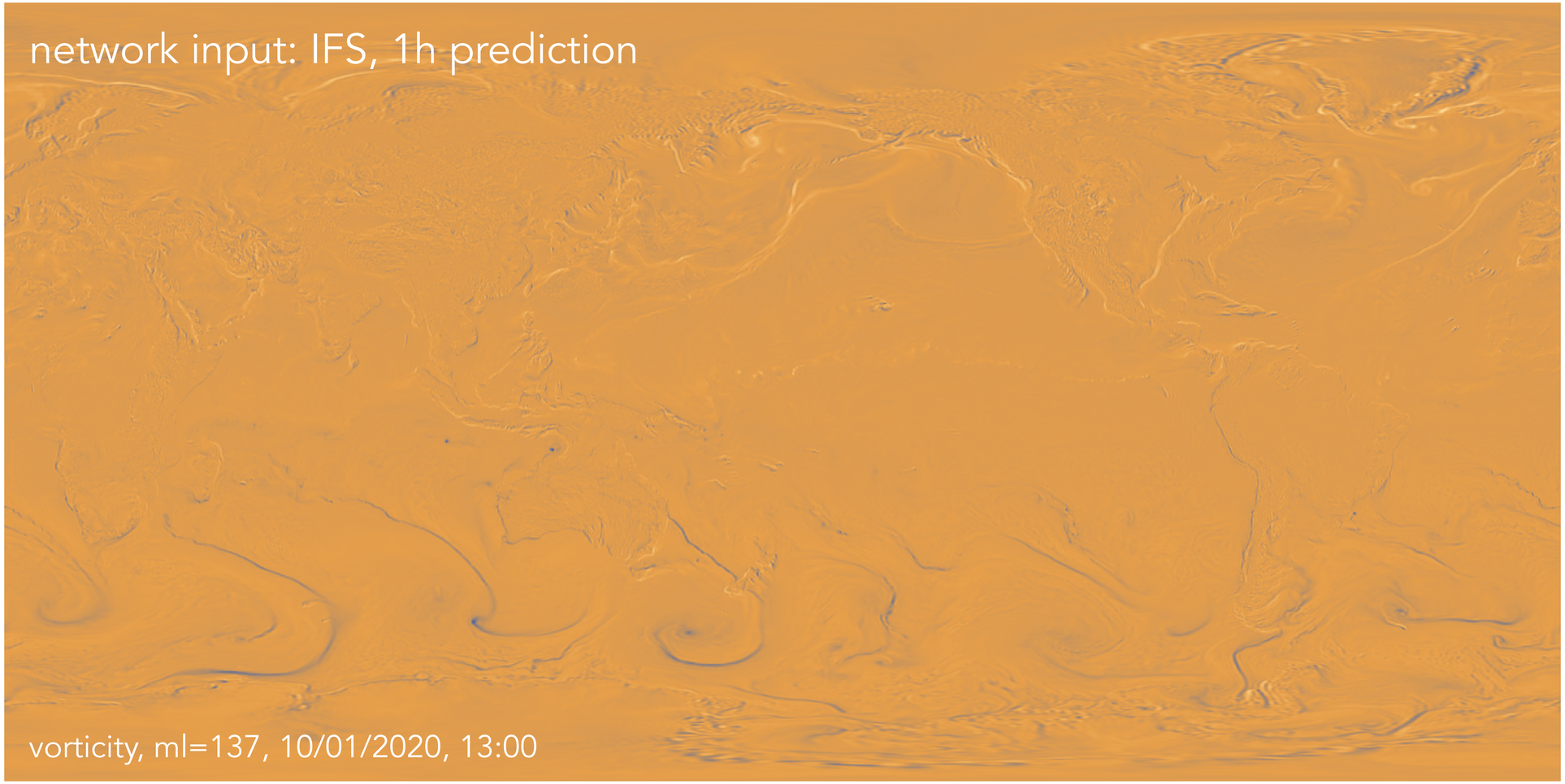
Model correction



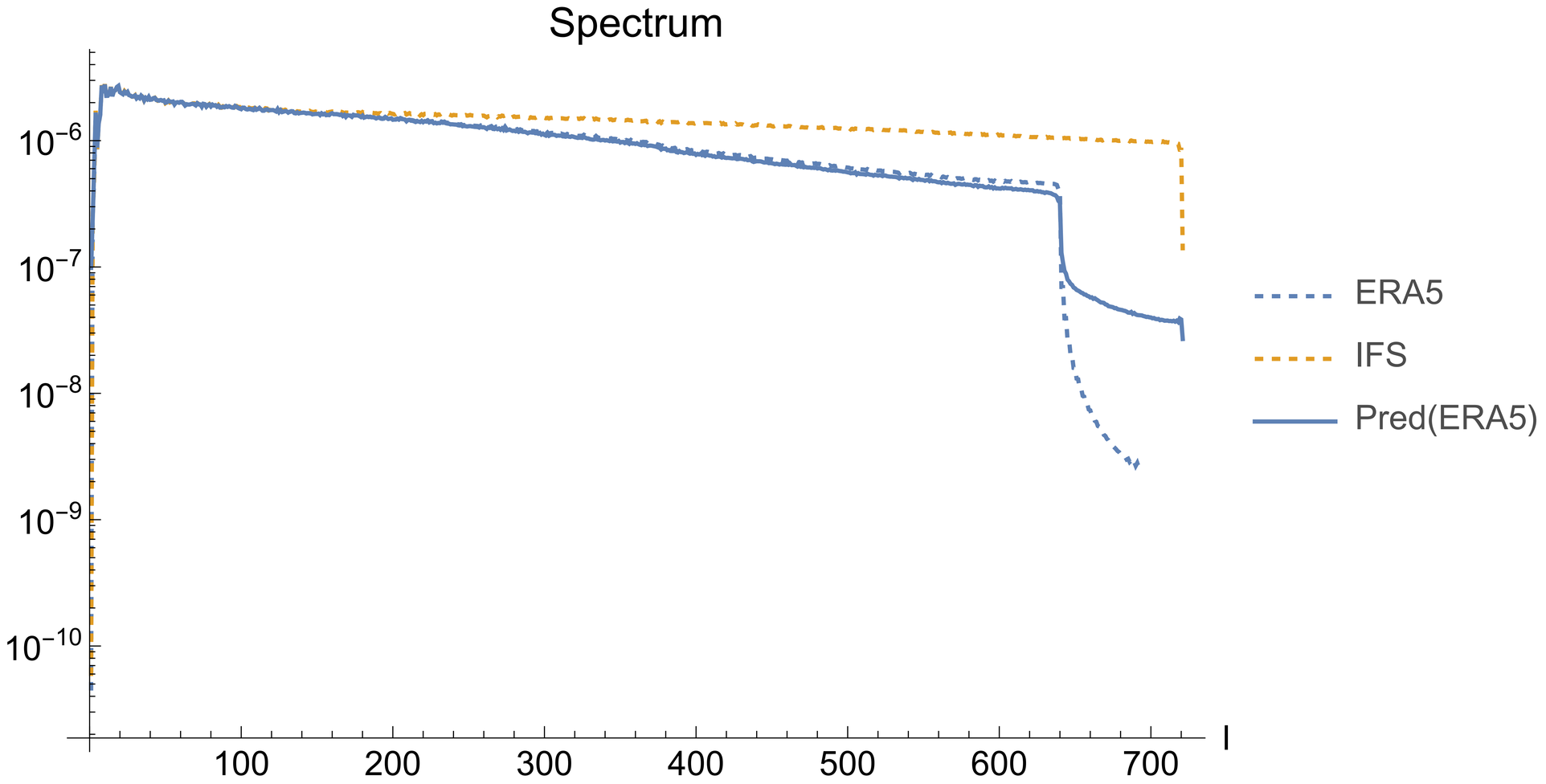
Model correction



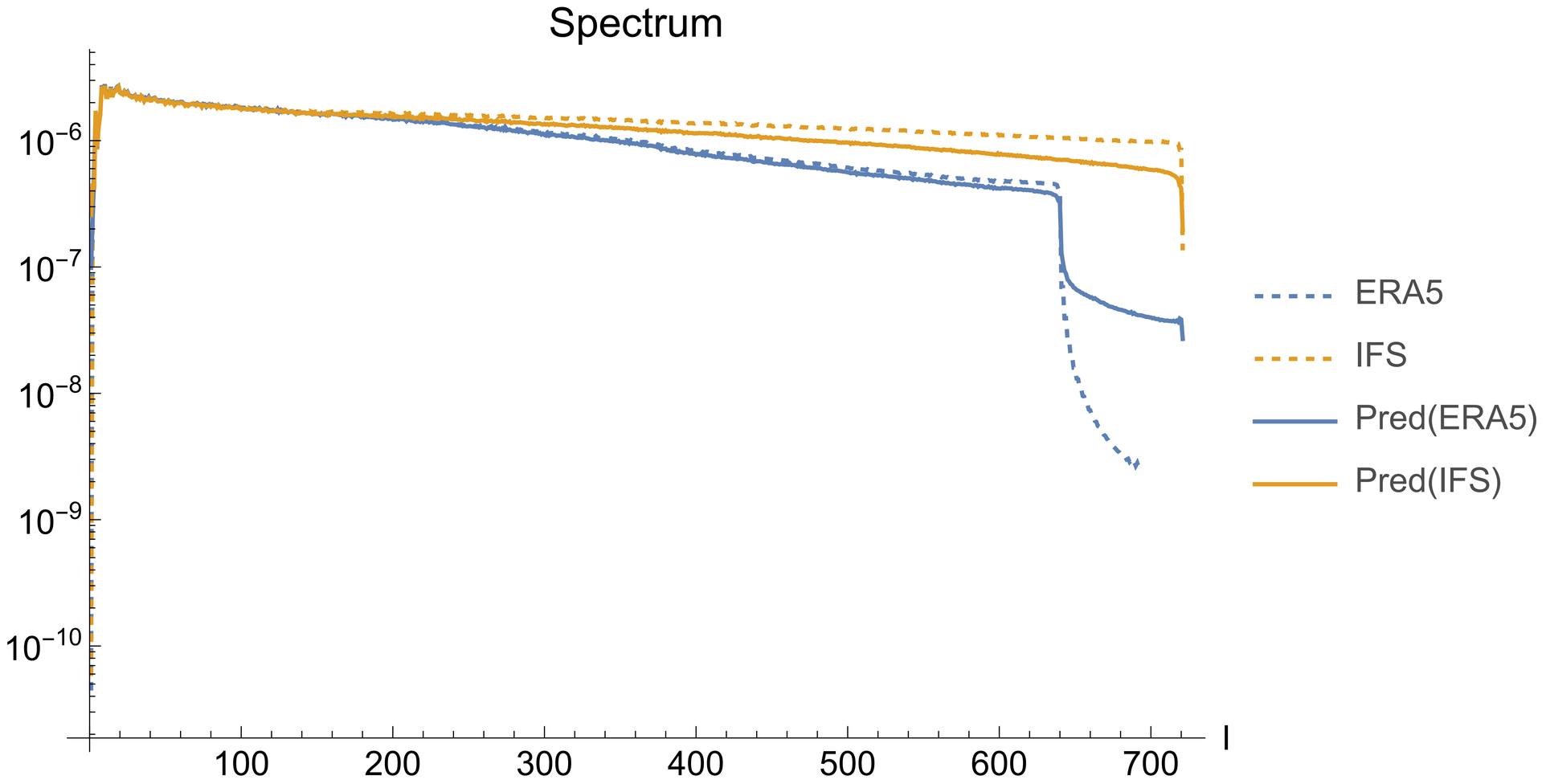
Model correction



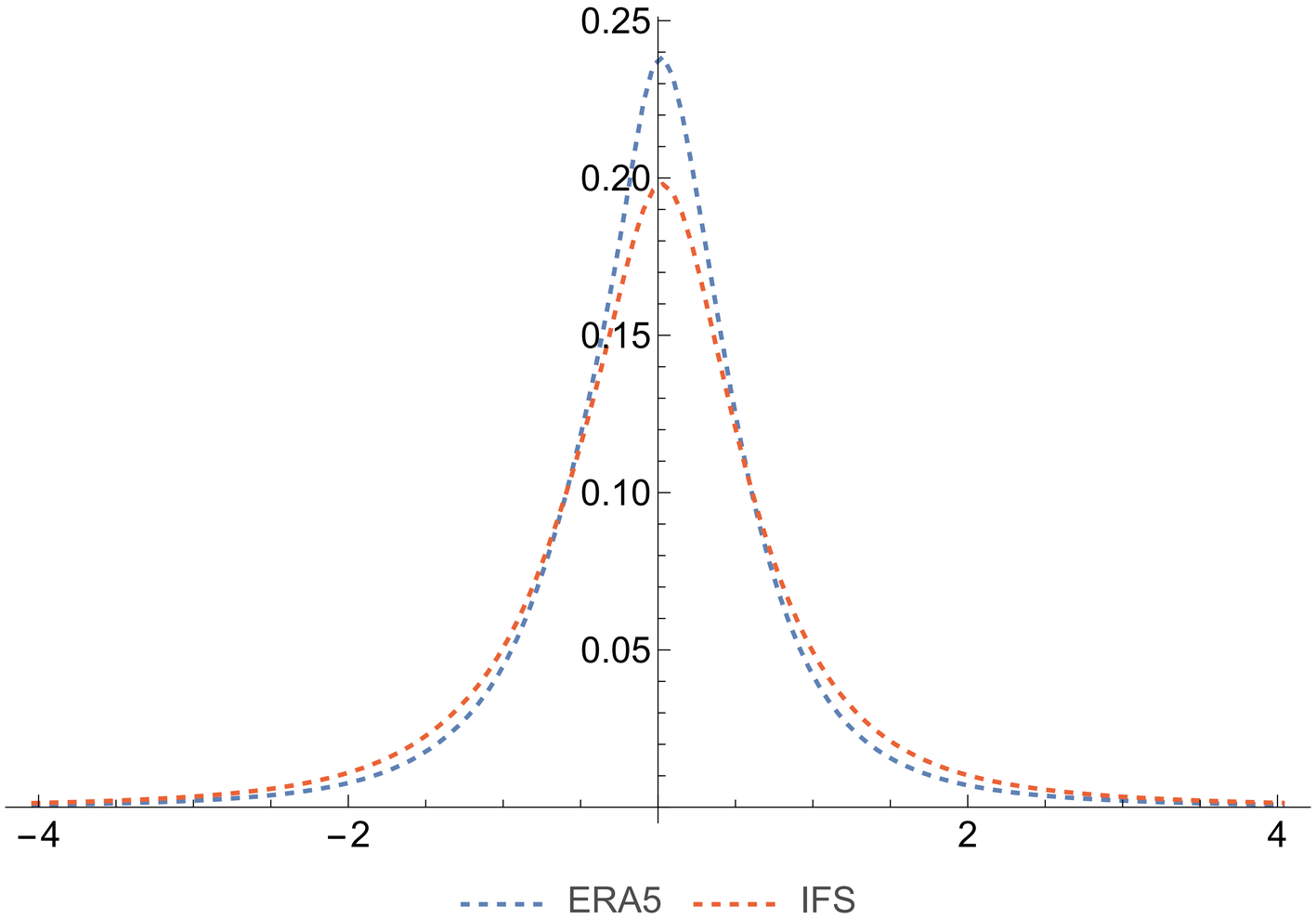
Model correction



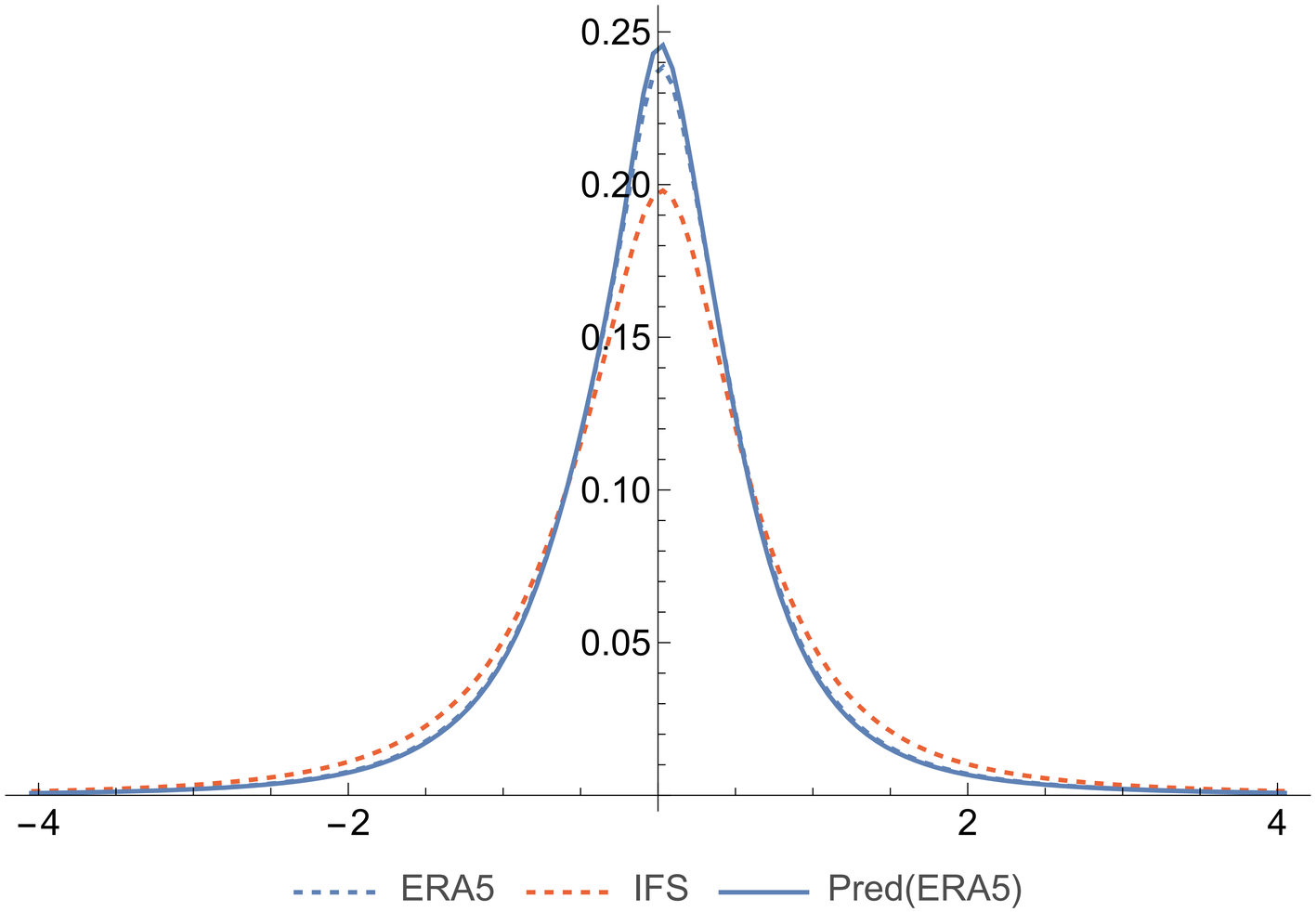
Model correction



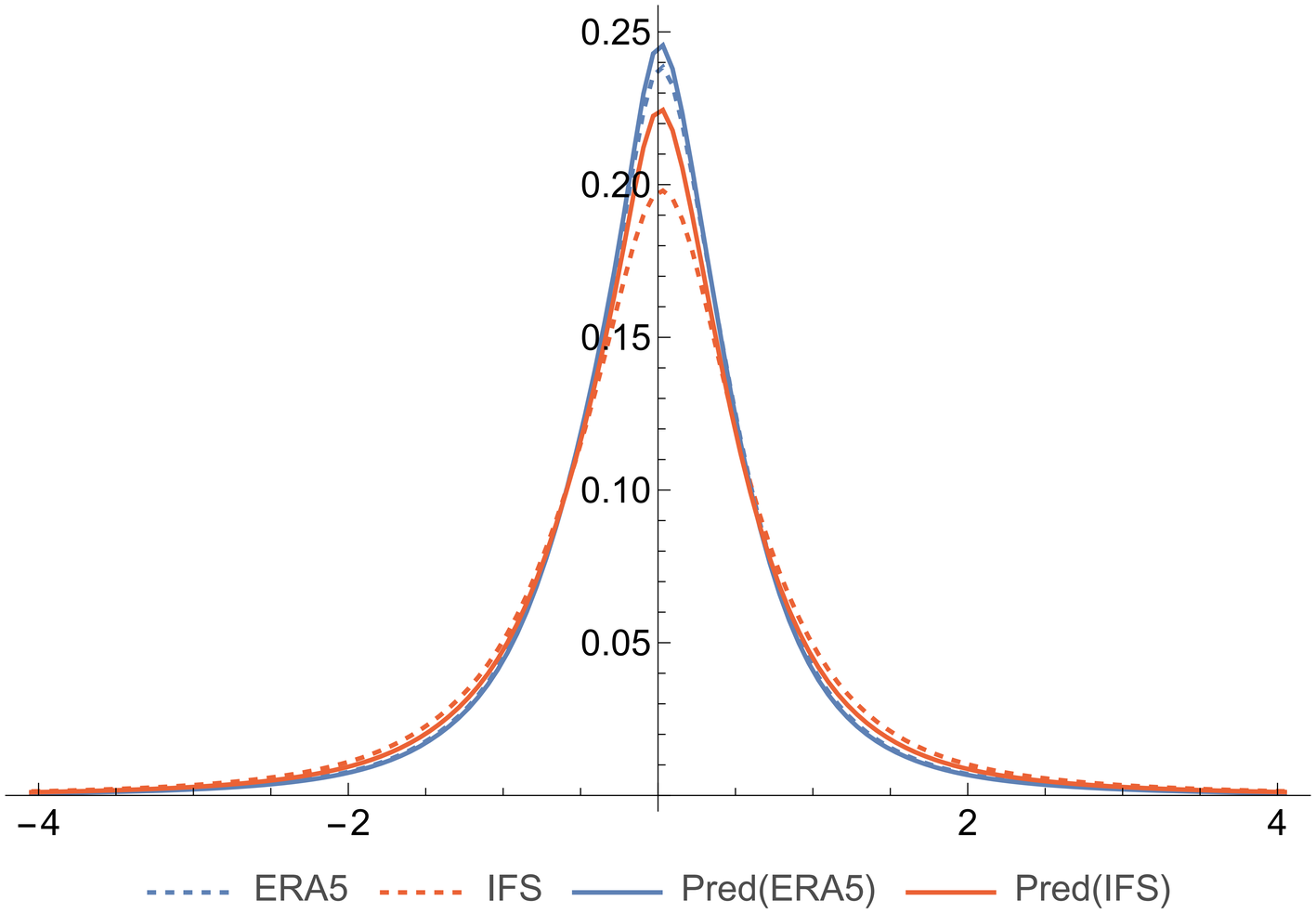
Model correction



Model correction



Model correction



Counterfactuals

- Numerical statistical atmospheric model:

$$p_\theta\left(y|x,\alpha\right)$$

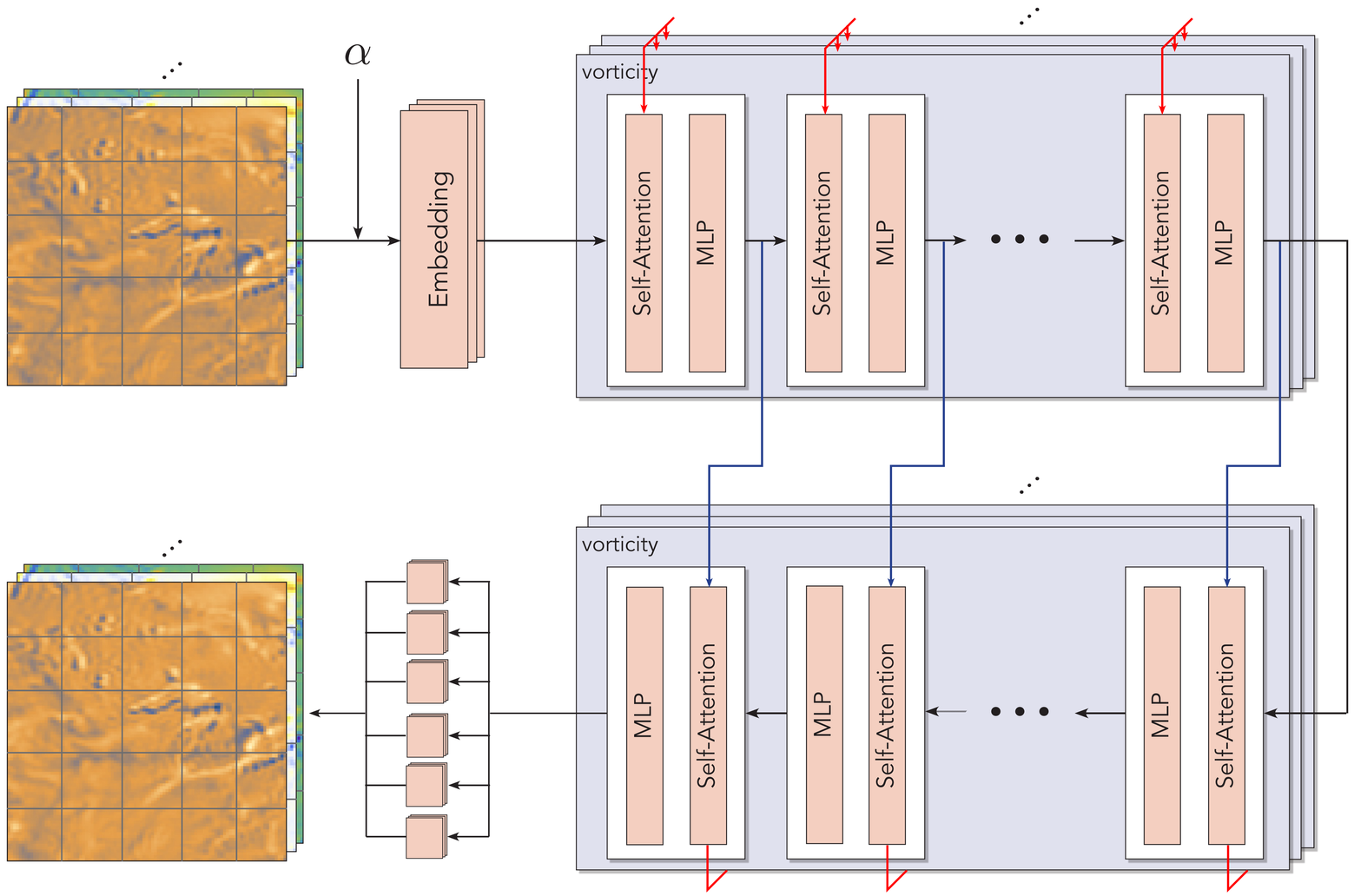
Counterfactuals

- Numerical statistical atmospheric model:

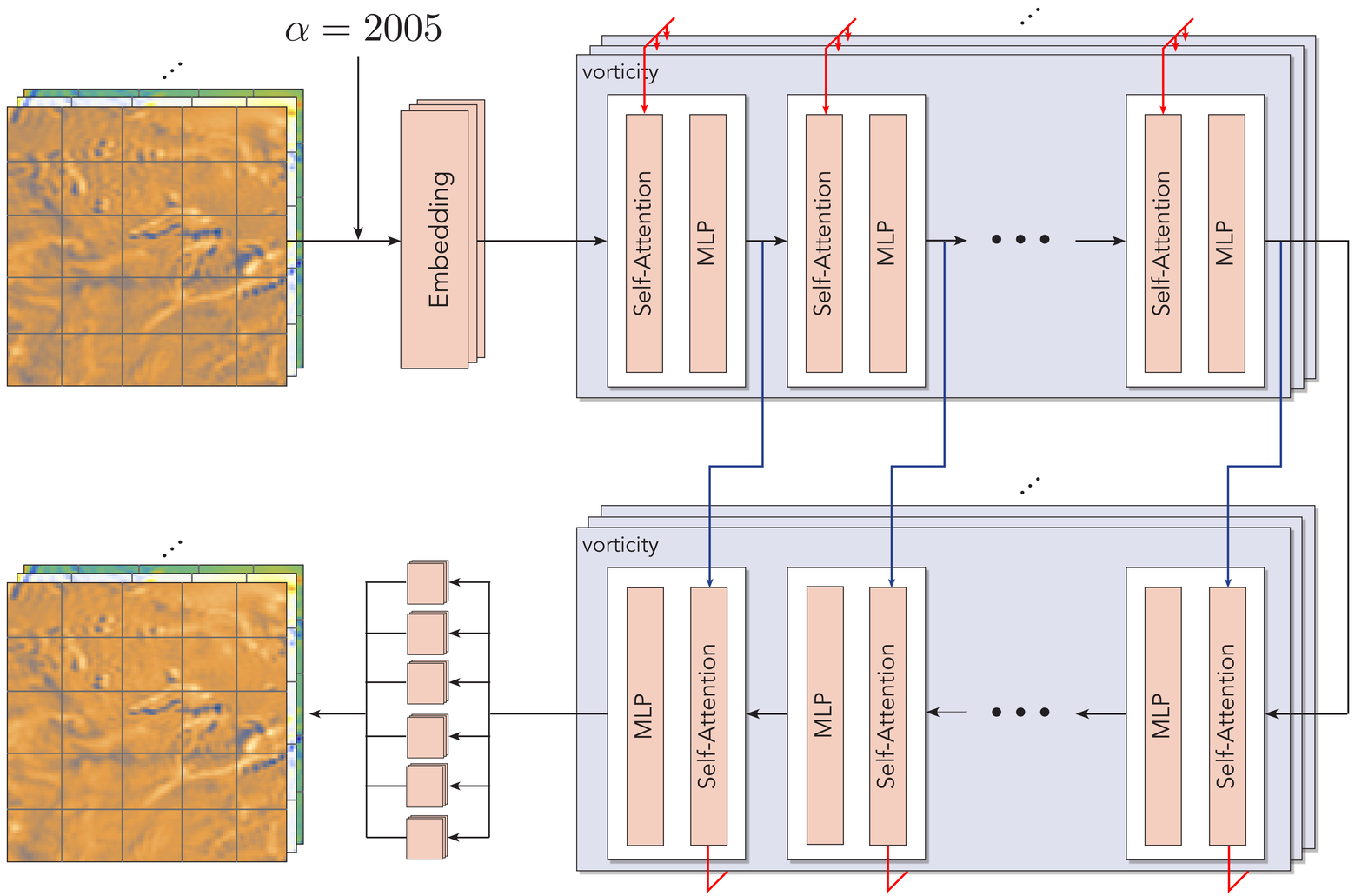
$$p_\theta\left(y|x,\alpha\right)$$

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

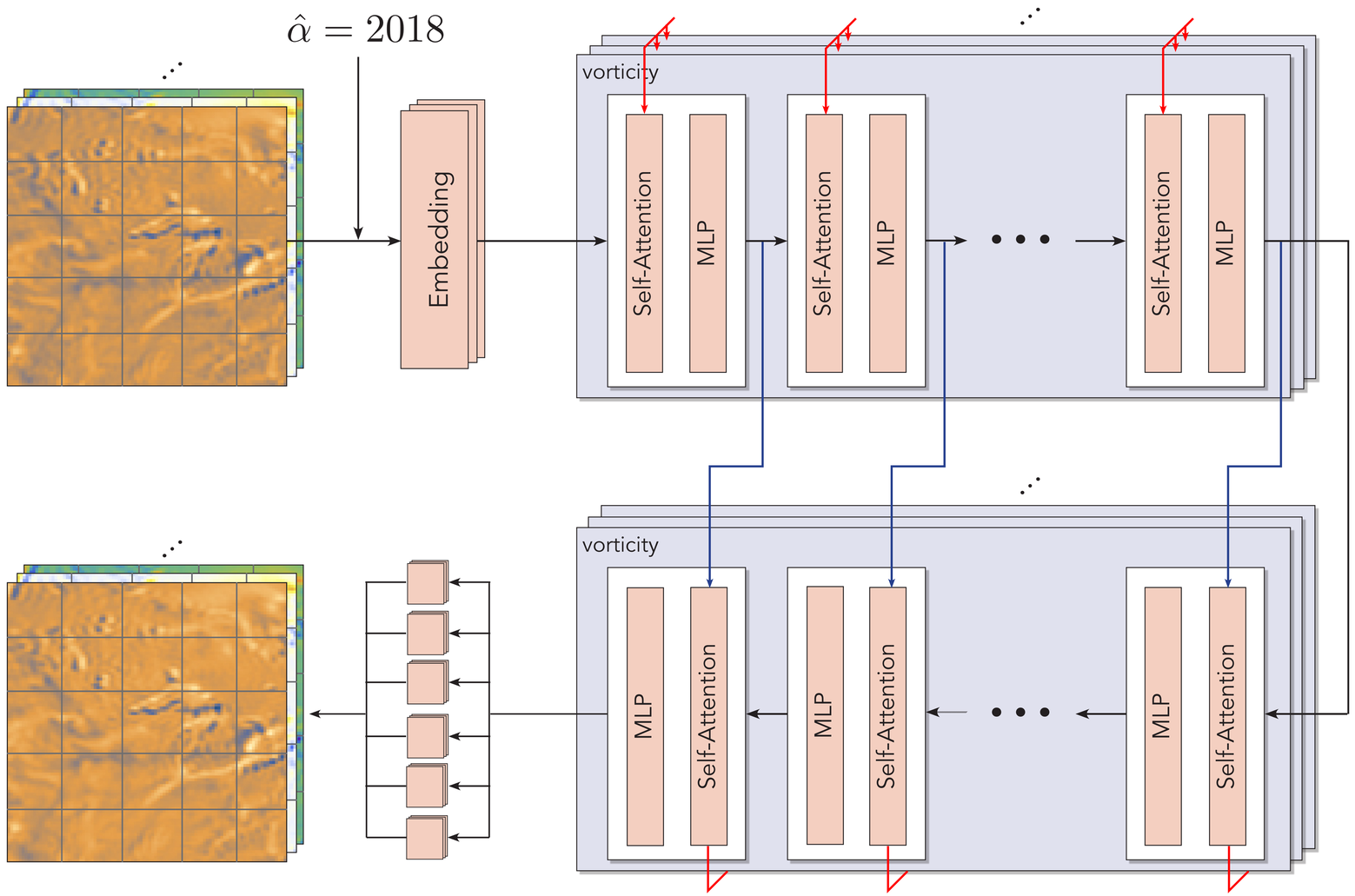
Counterfactuals



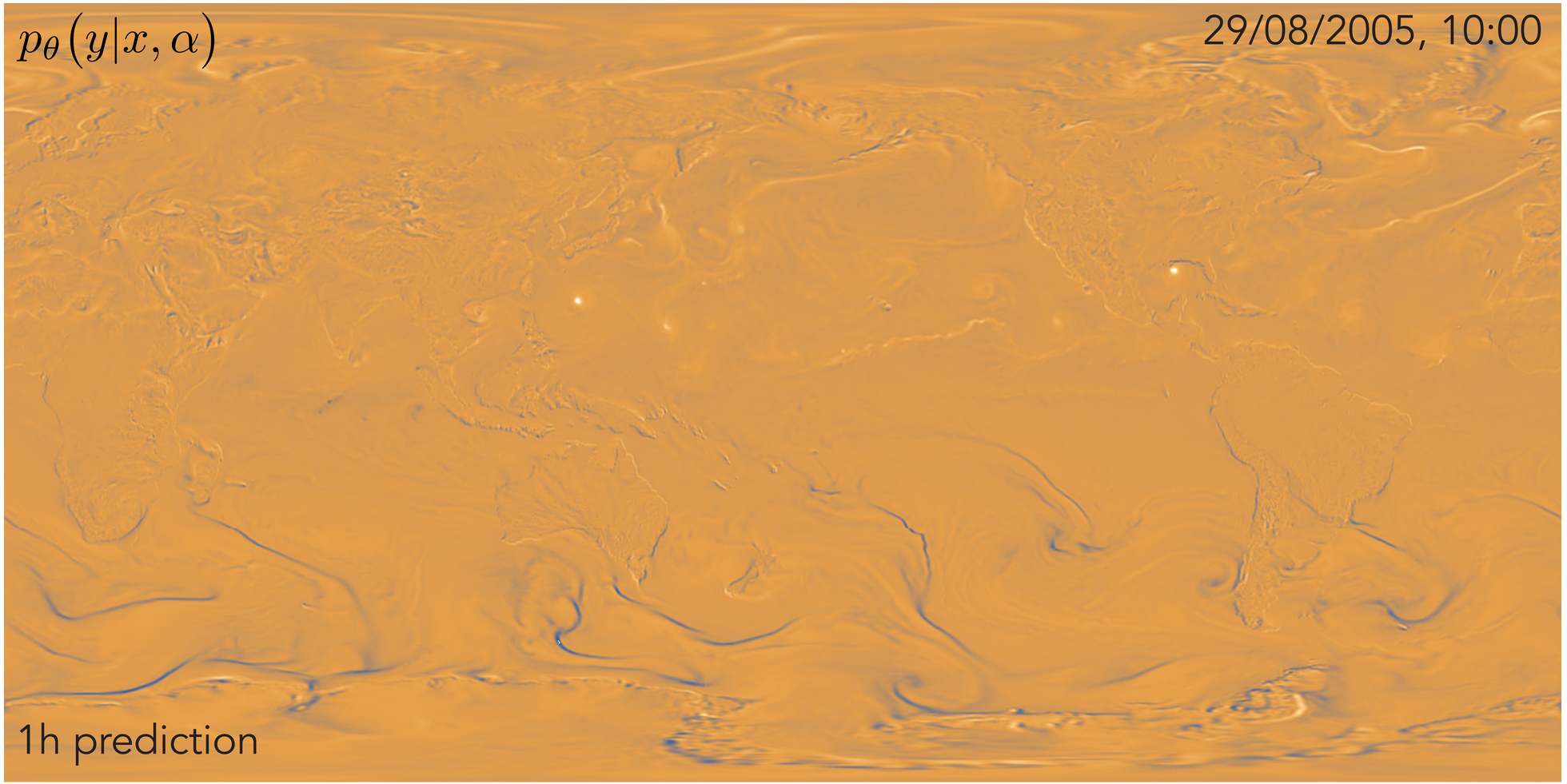
Counterfactuals



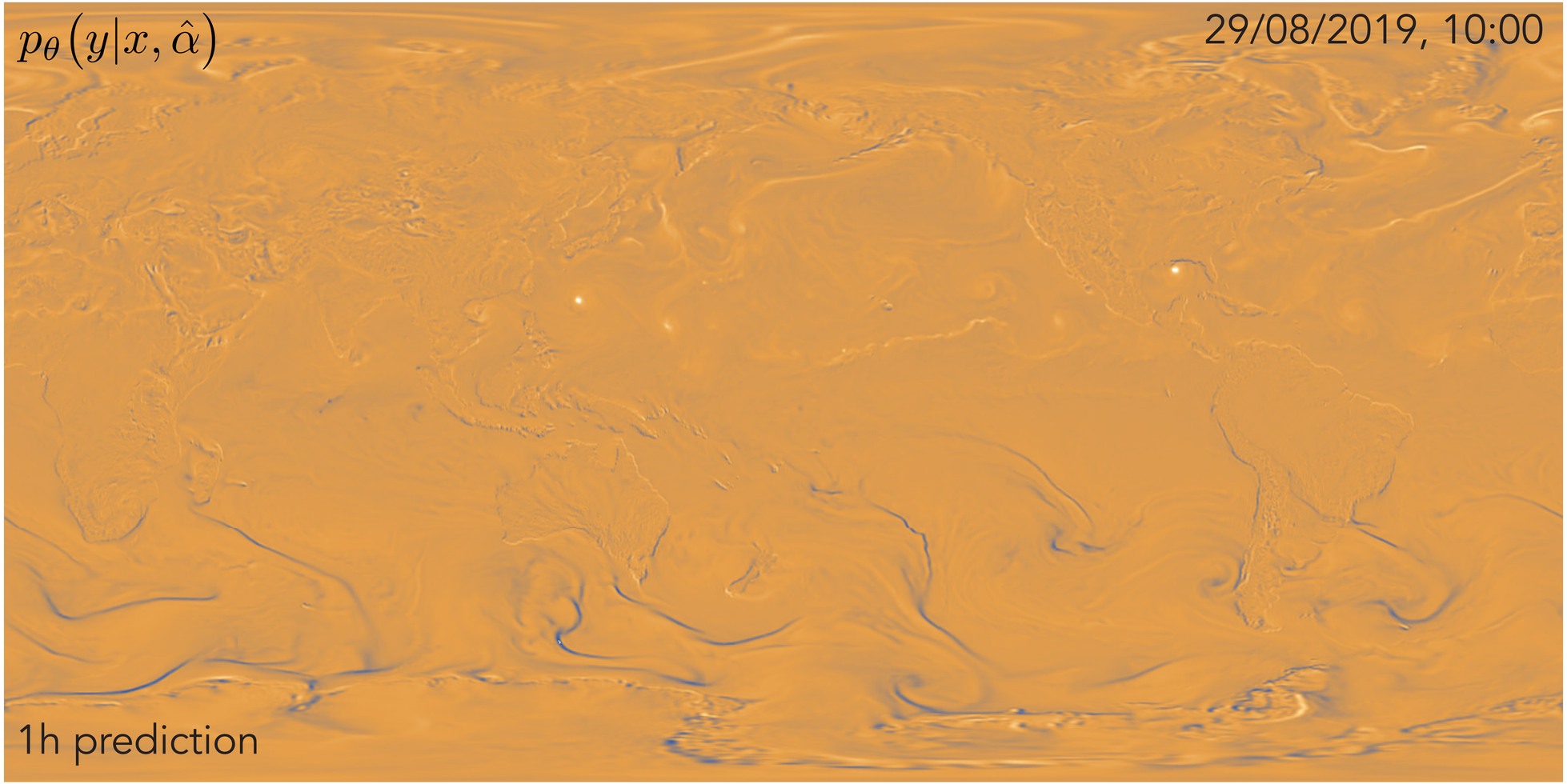
Counterfactuals



Counterfactuals



Counterfactuals

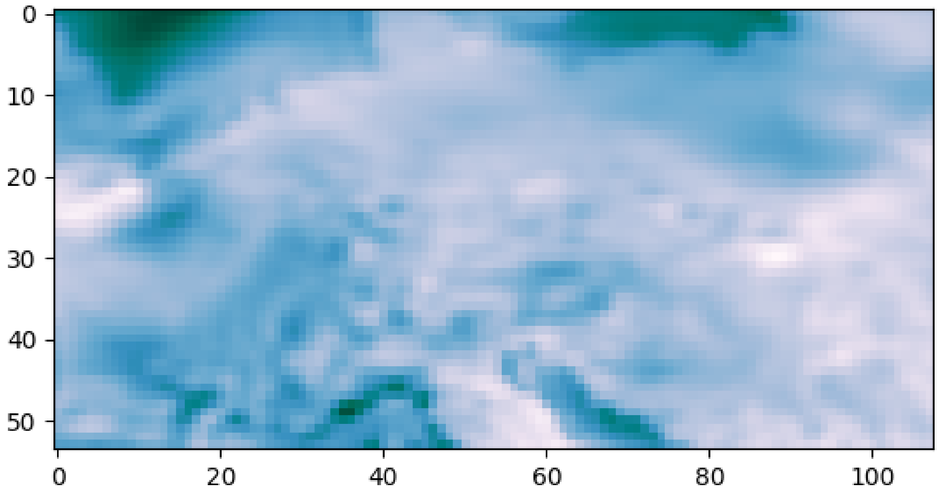


Counterfactuals

Other Applications

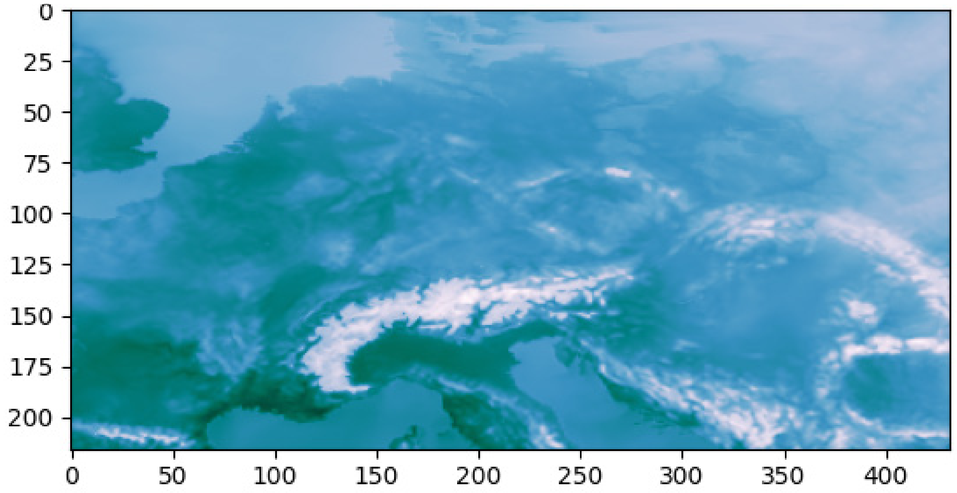
Downscaling

ERA5



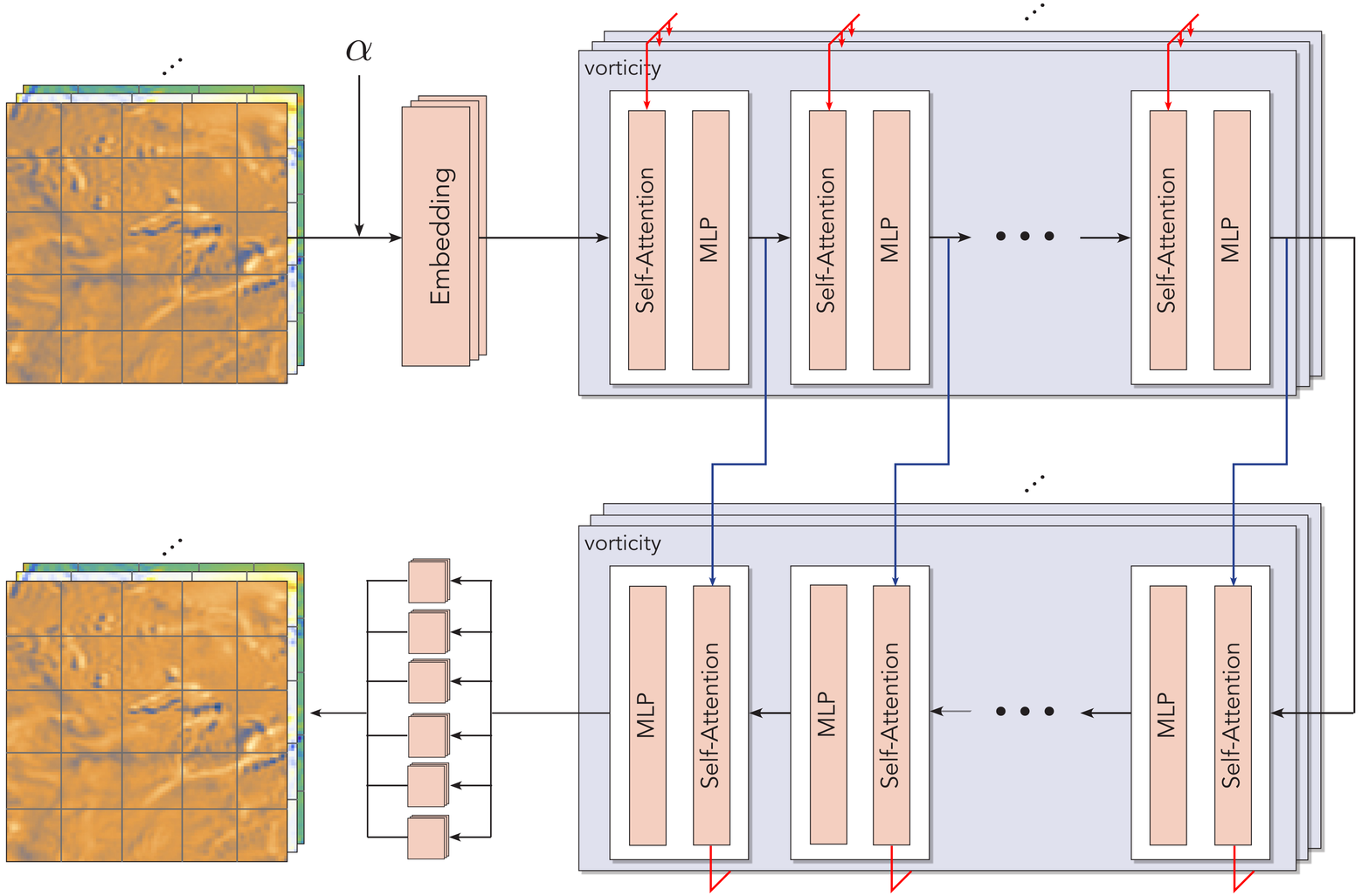
temperature, ml=137

COSMO-REA6

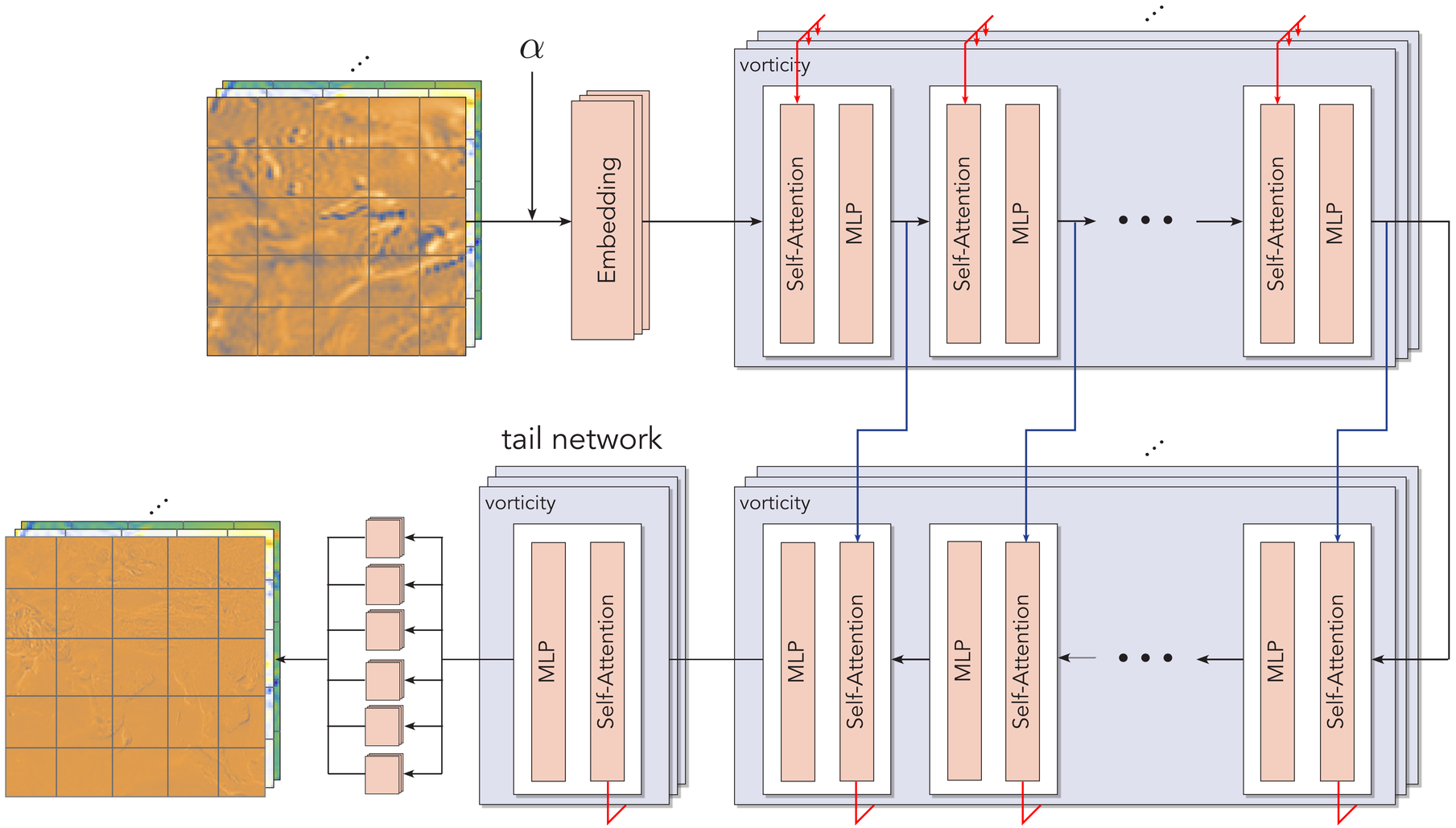


temperature, 2 m

Downscaling



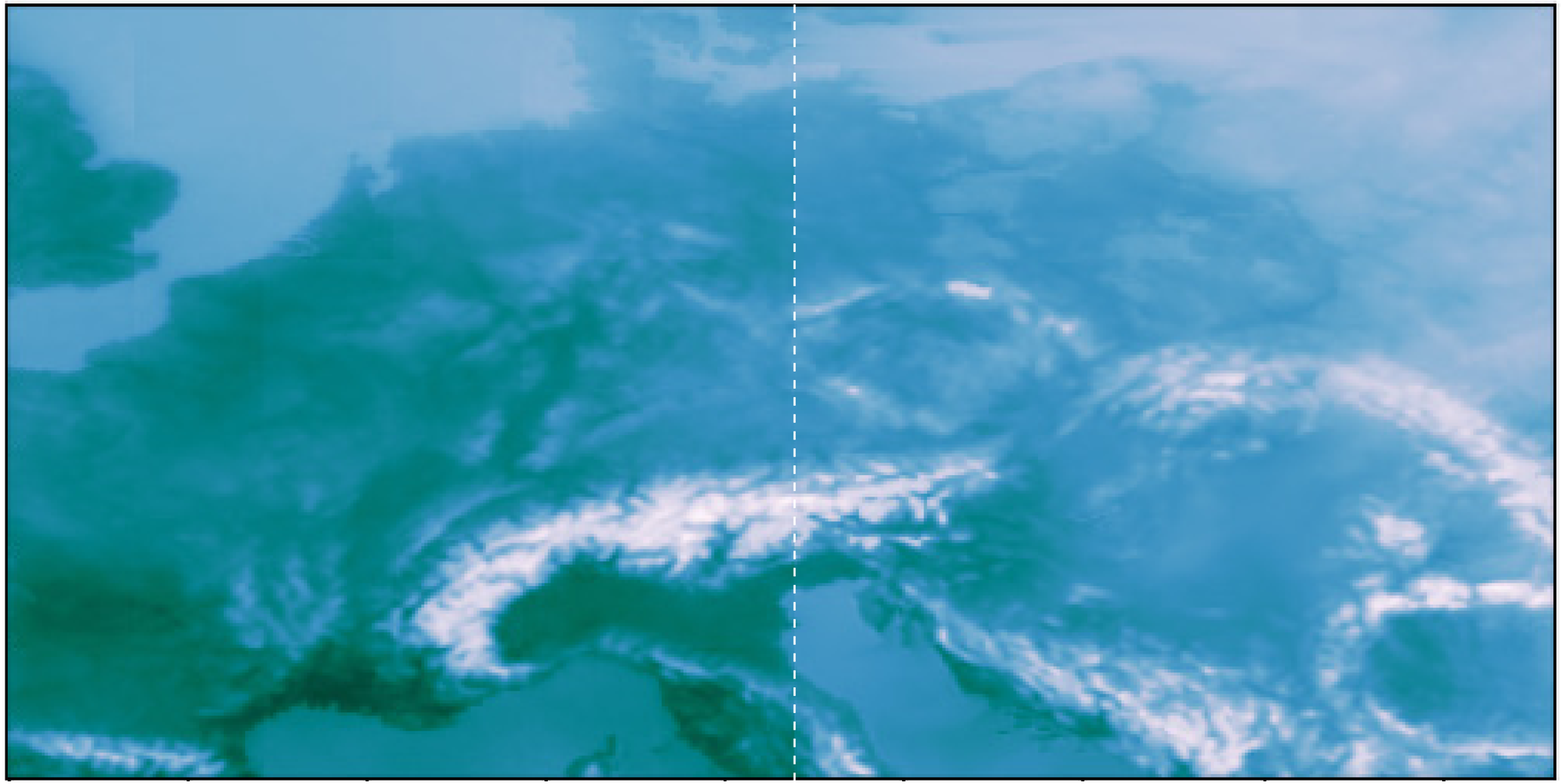
Downscaling



Downscaling

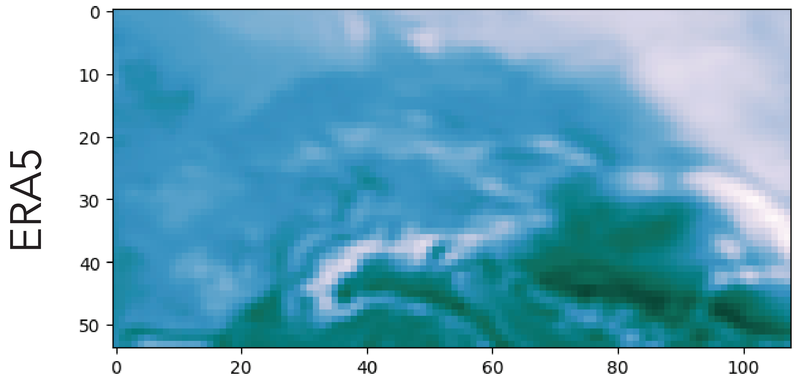
AtmoRep prediction

COSMO-REA6

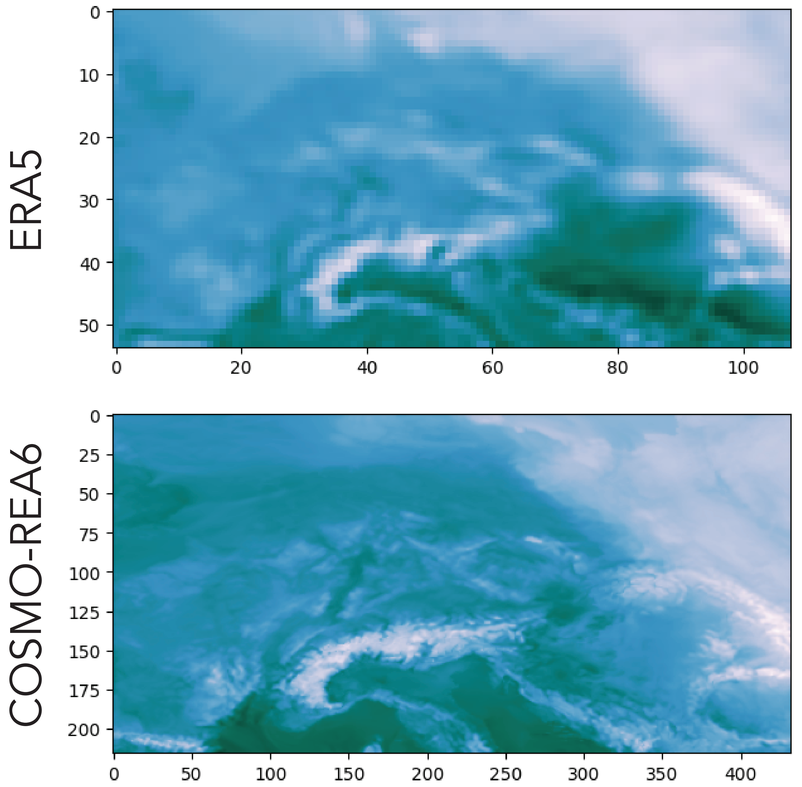


temperature, 2 m

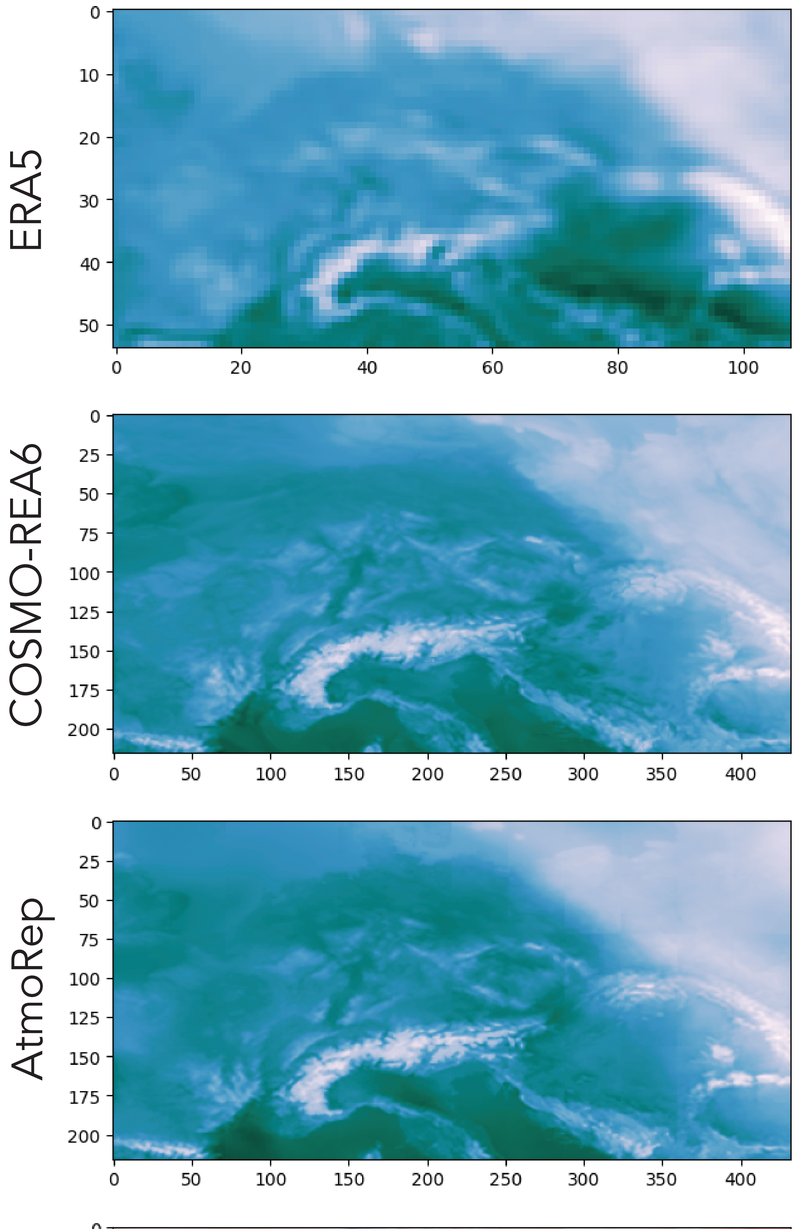
Downscaling



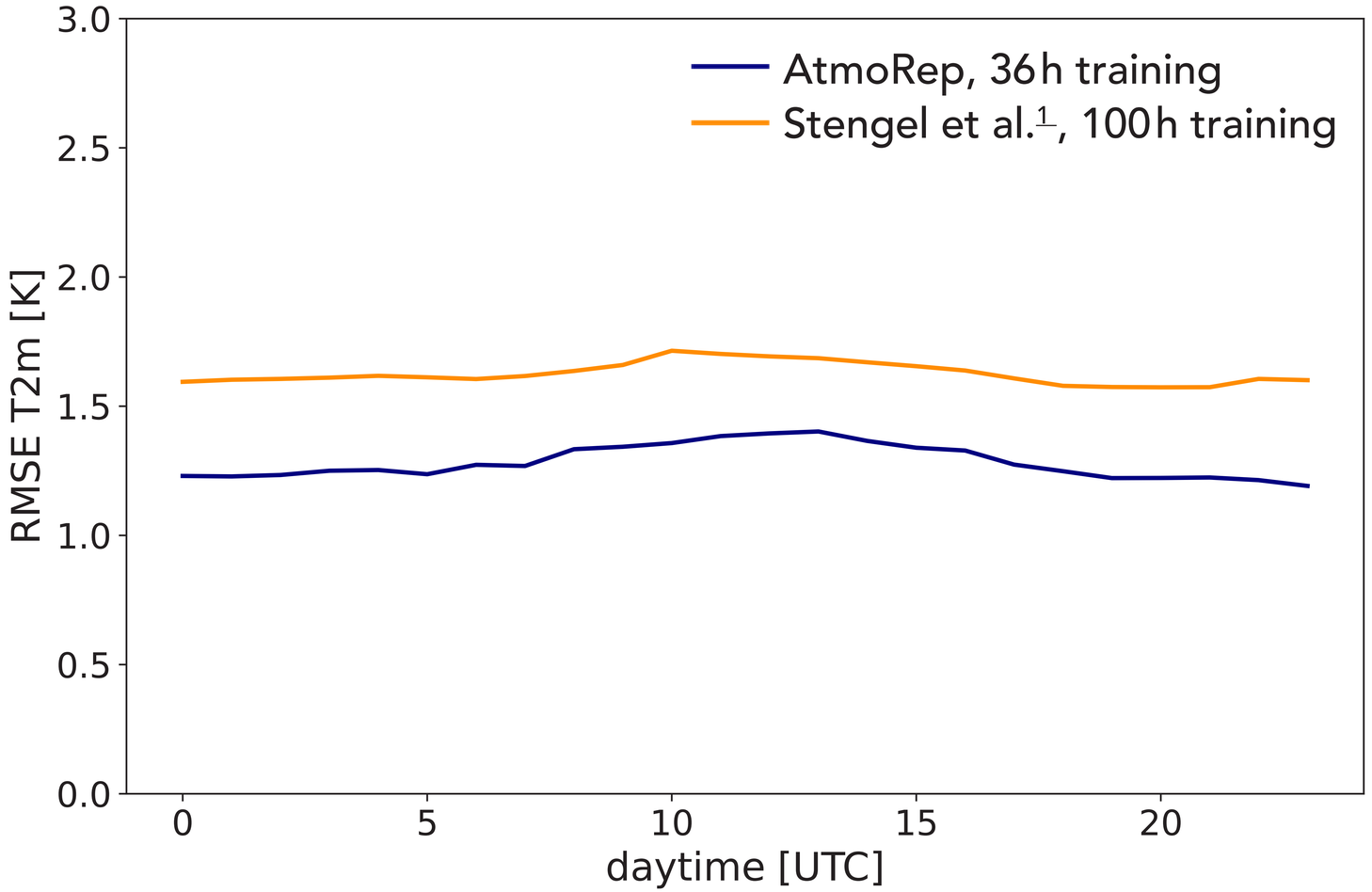
Downscaling



Downscaling

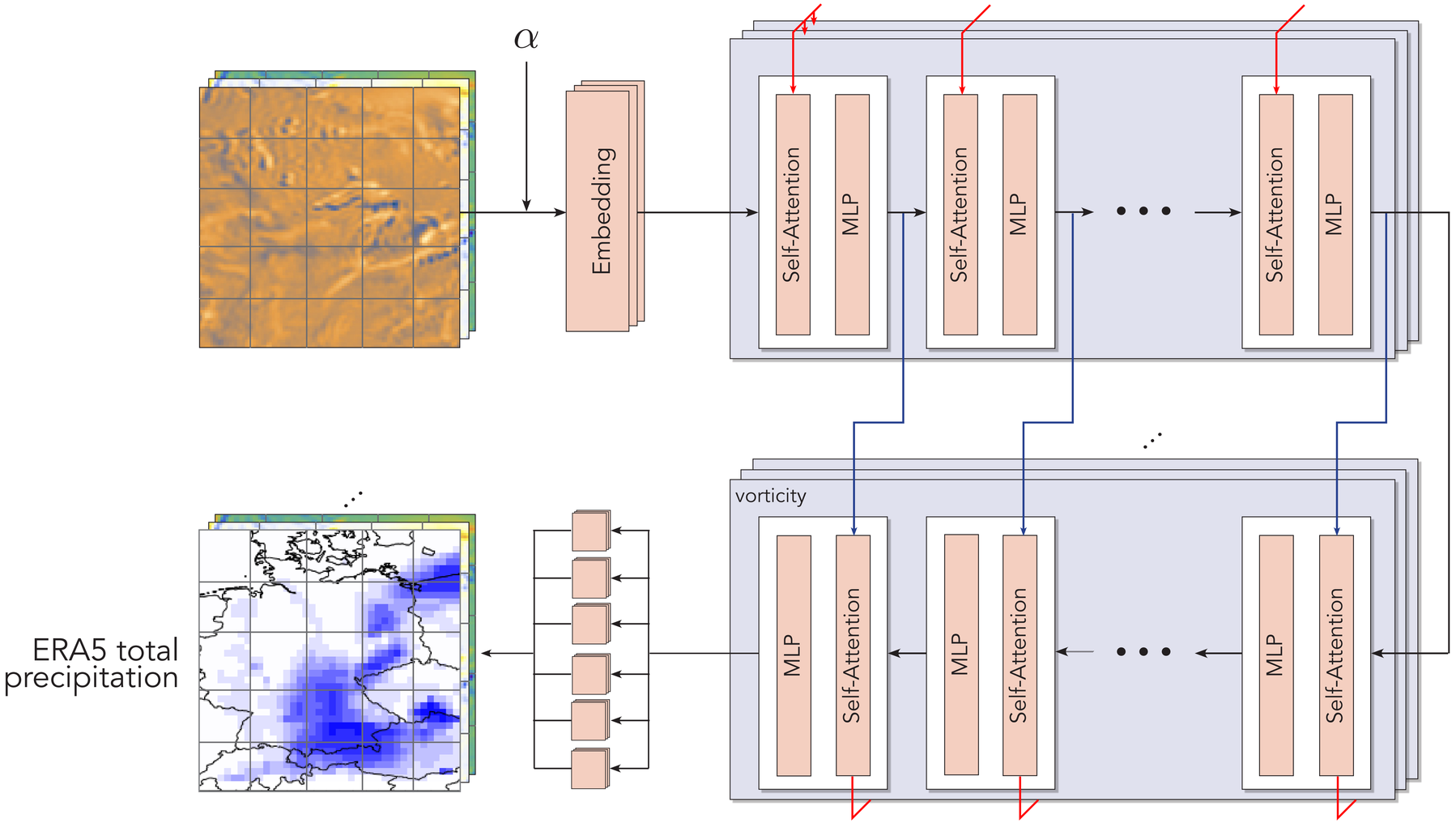


Downscaling

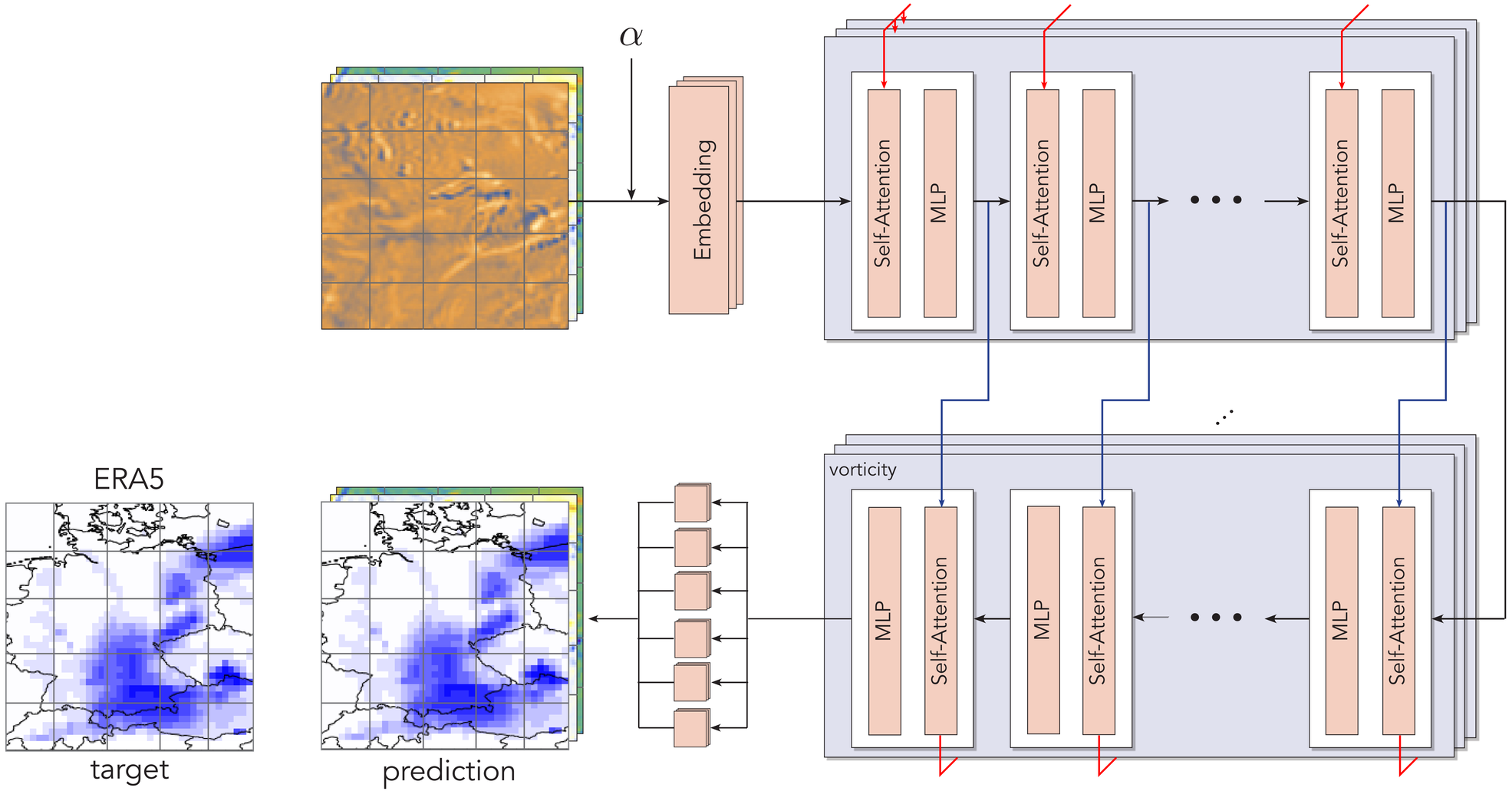


[1] K. Stengel, A. Glaws, D. Hettinger, and R. N. King. Adversarial super-resolution of climatological wind and solar data. Proceedings of the National Academy of Sciences, 117(29):16805–16815, 2020.

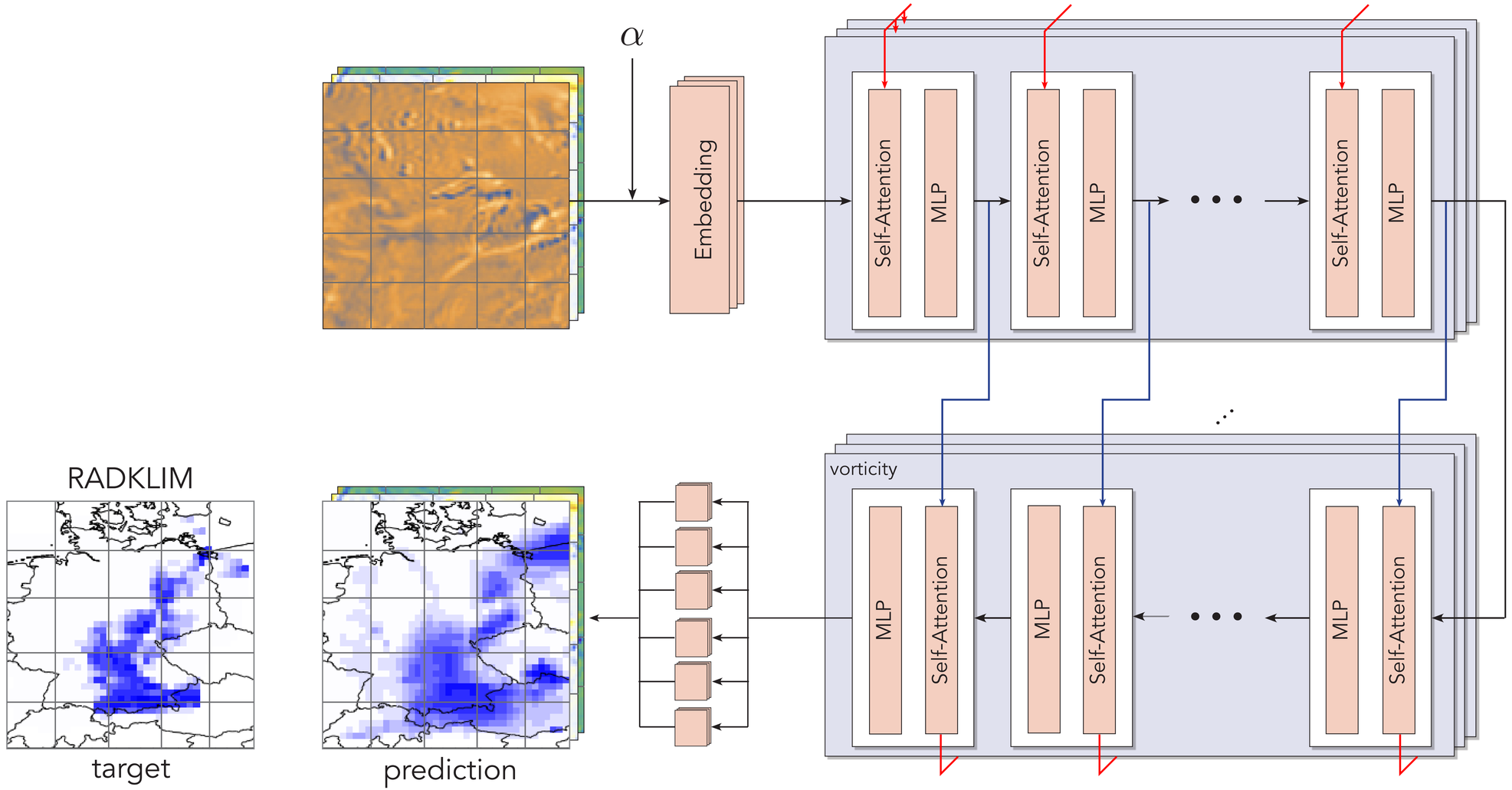
Bias correction



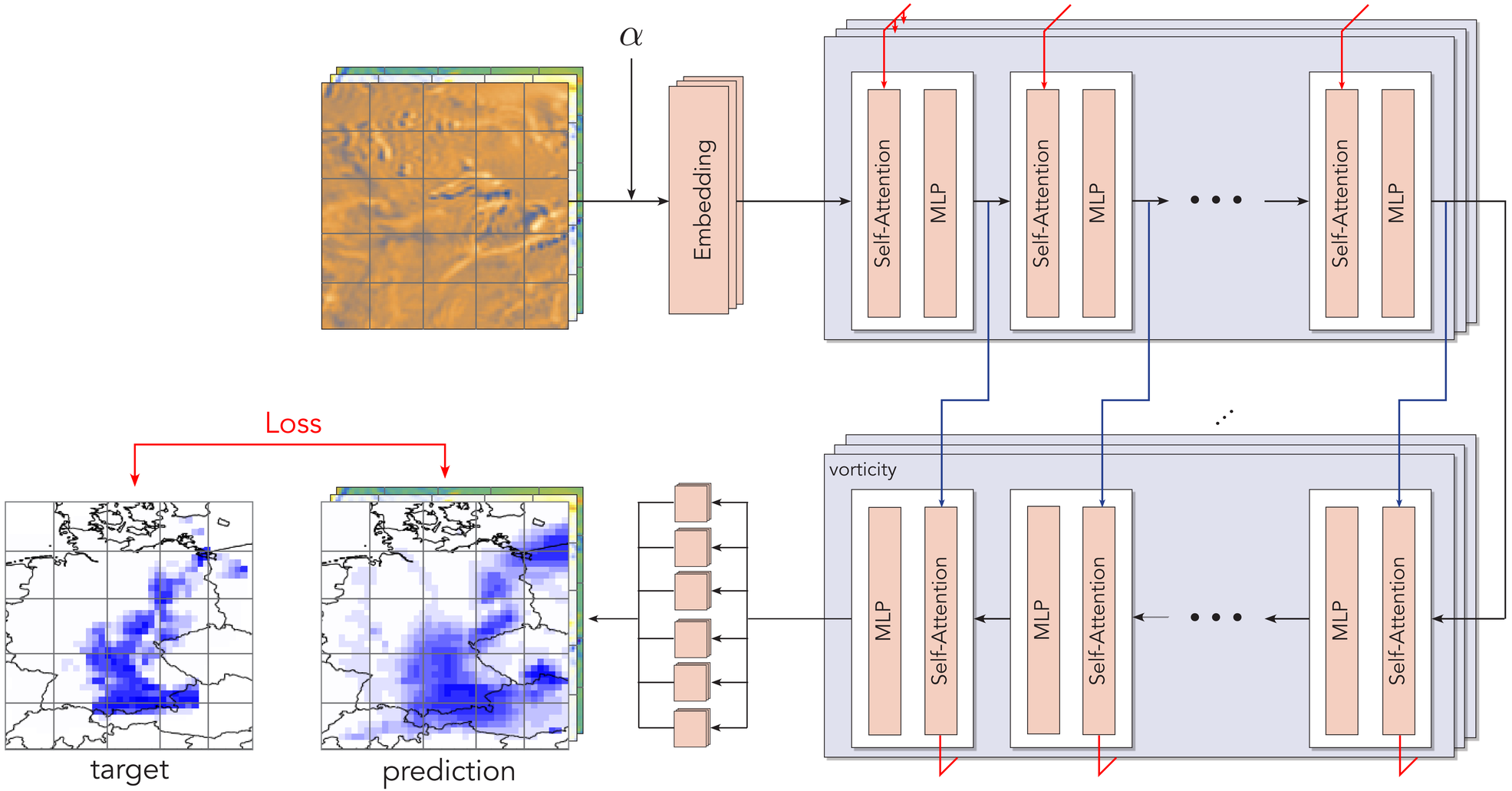
Bias correction



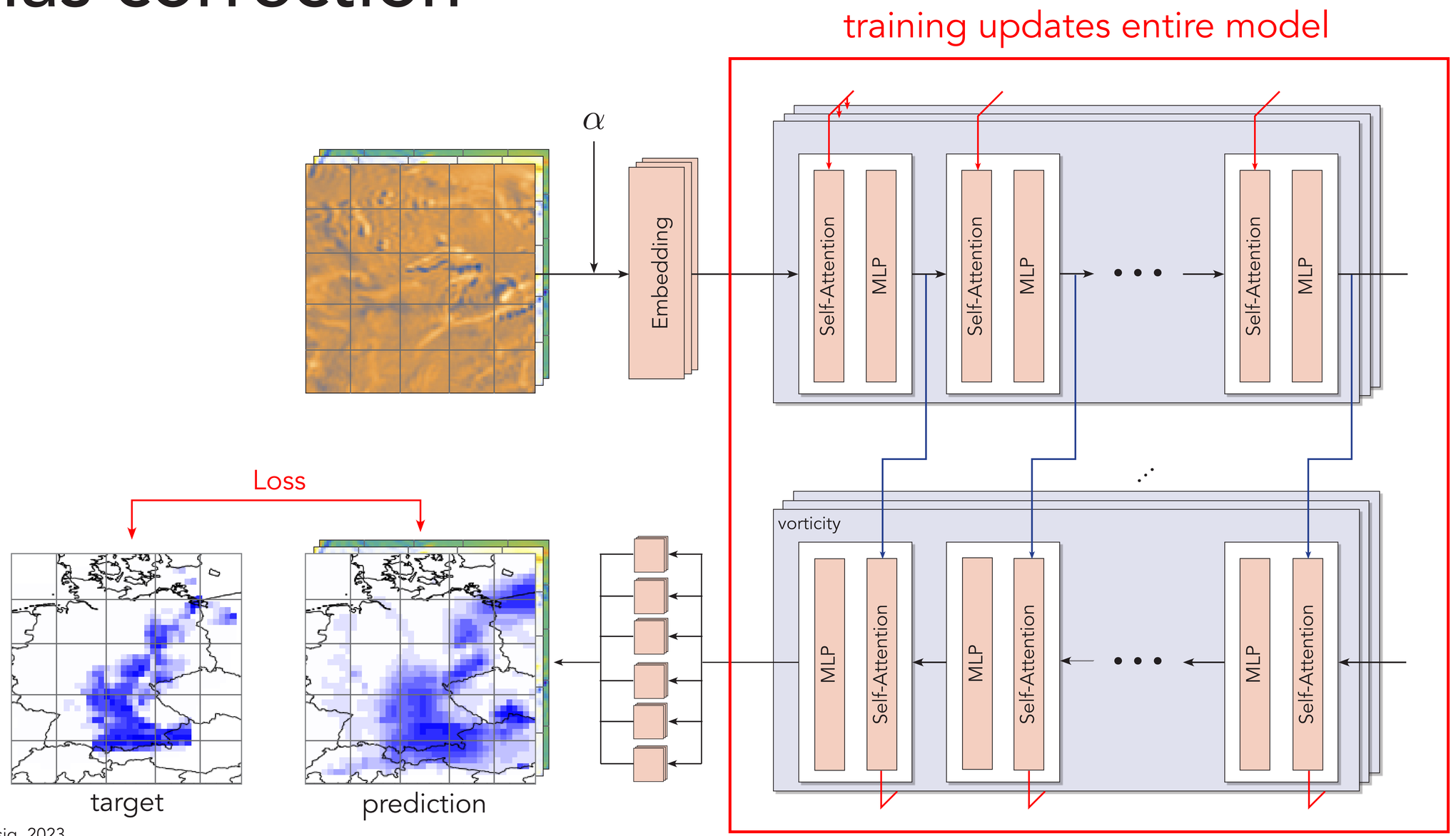
Bias correction



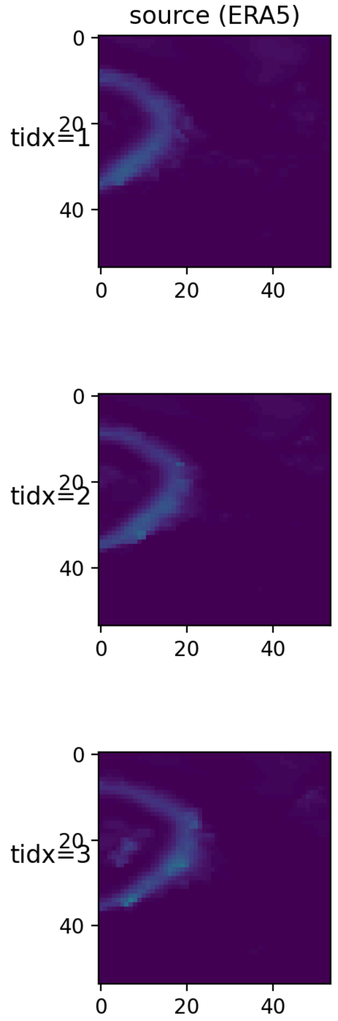
Bias correction



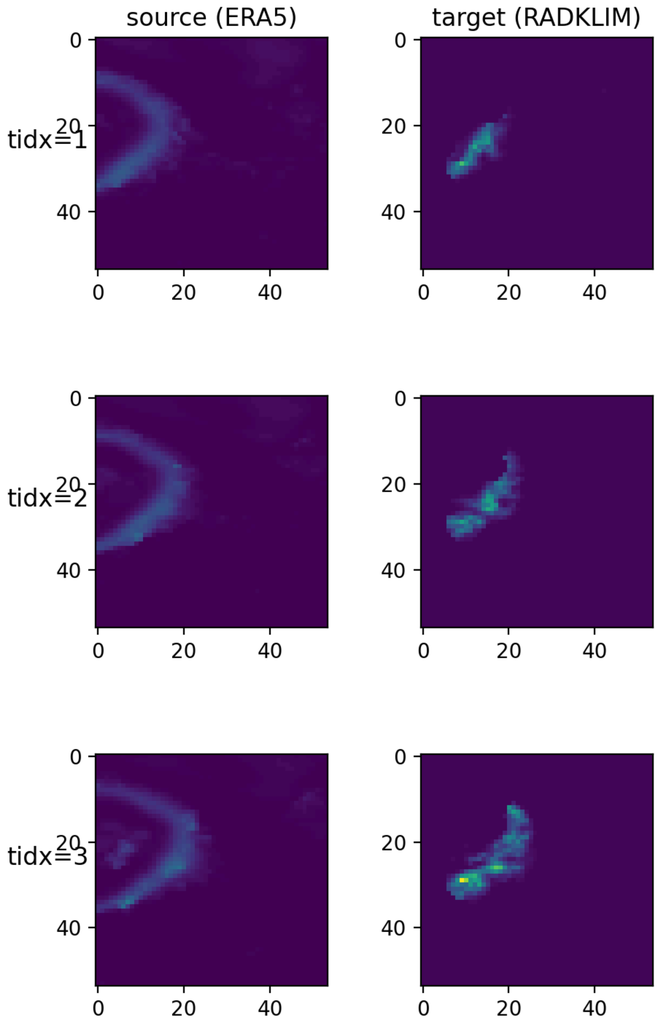
Bias correction



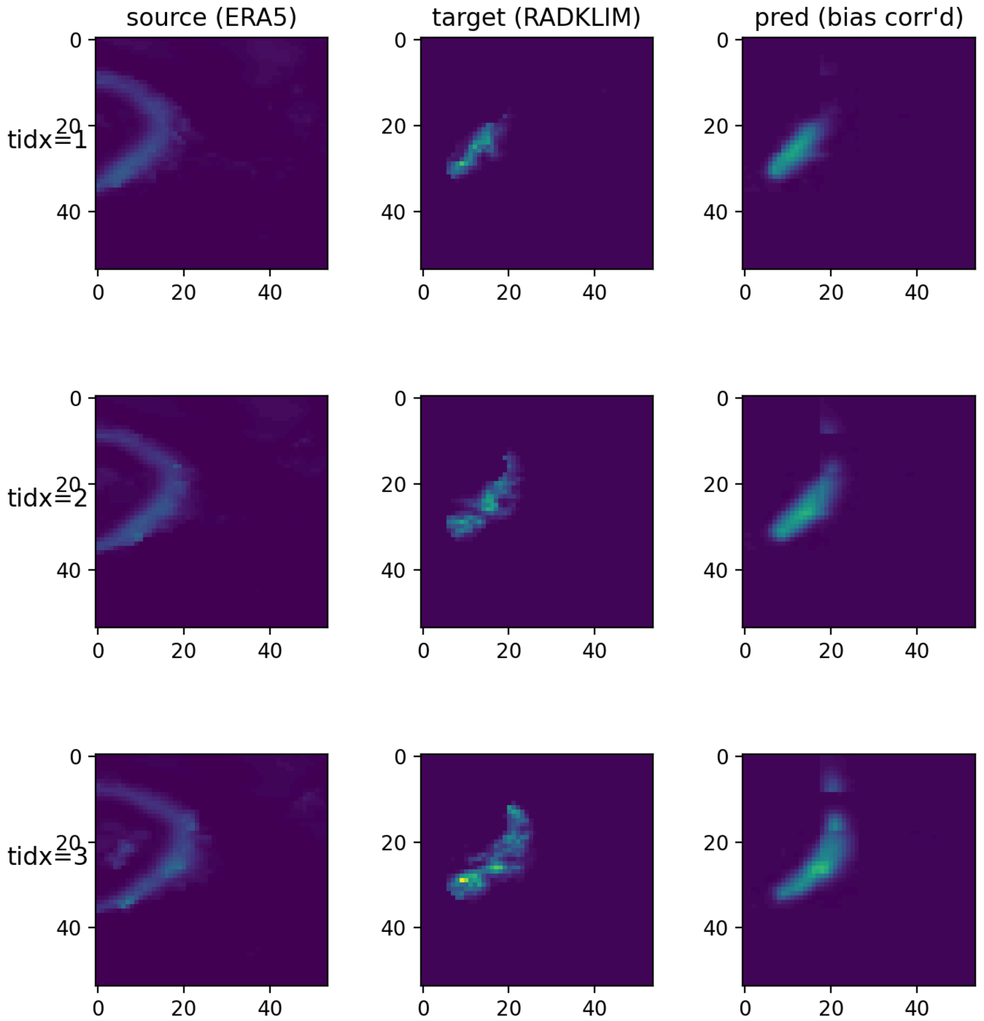
Bias correction



Bias correction



Bias correction



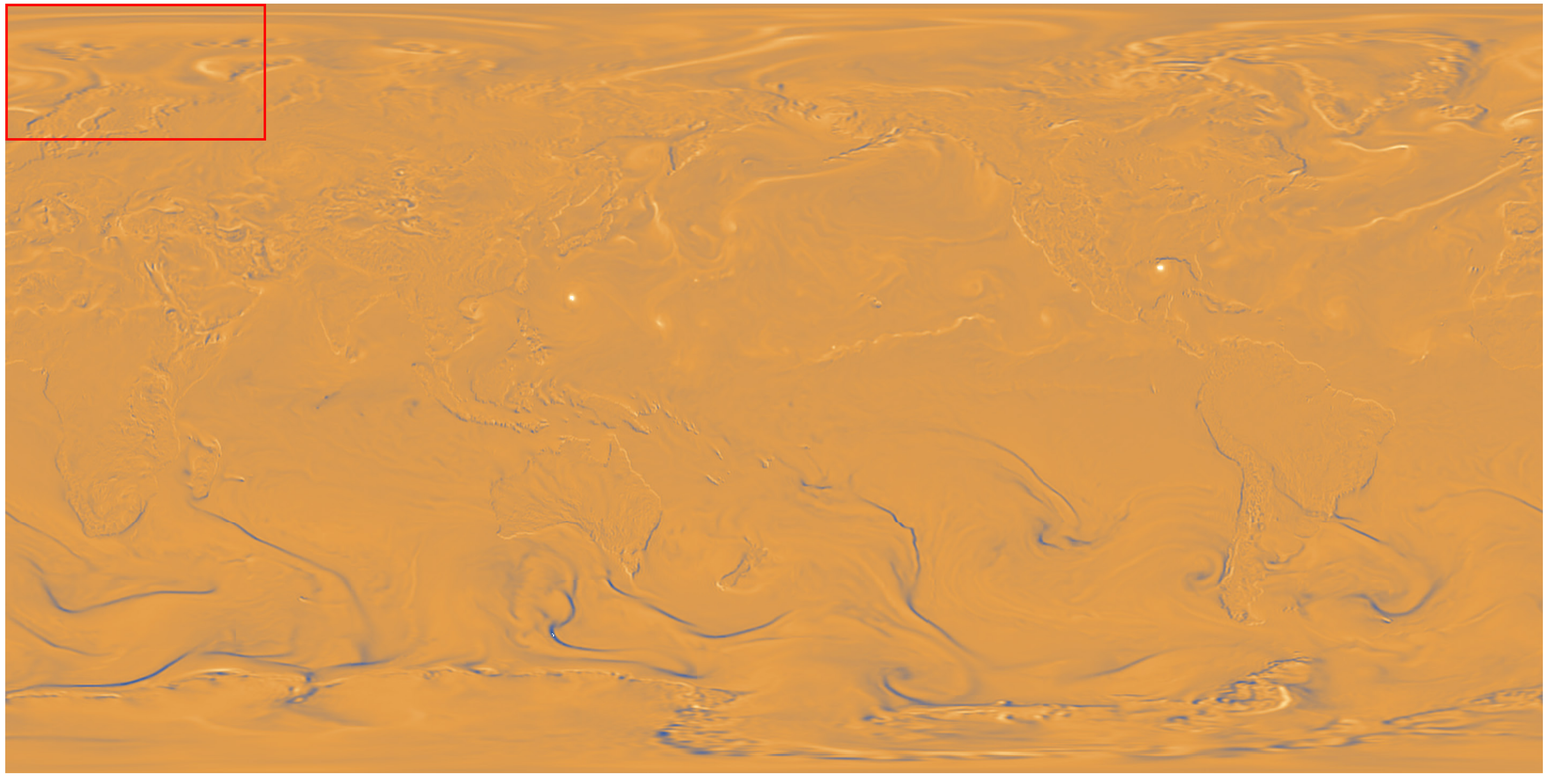
Outlook

Outlook

- Medium range forecasting
  - Patch local predictions into a global forecast, iterate
    - Avoid artifacts due to patching and ensure robust exchange of information

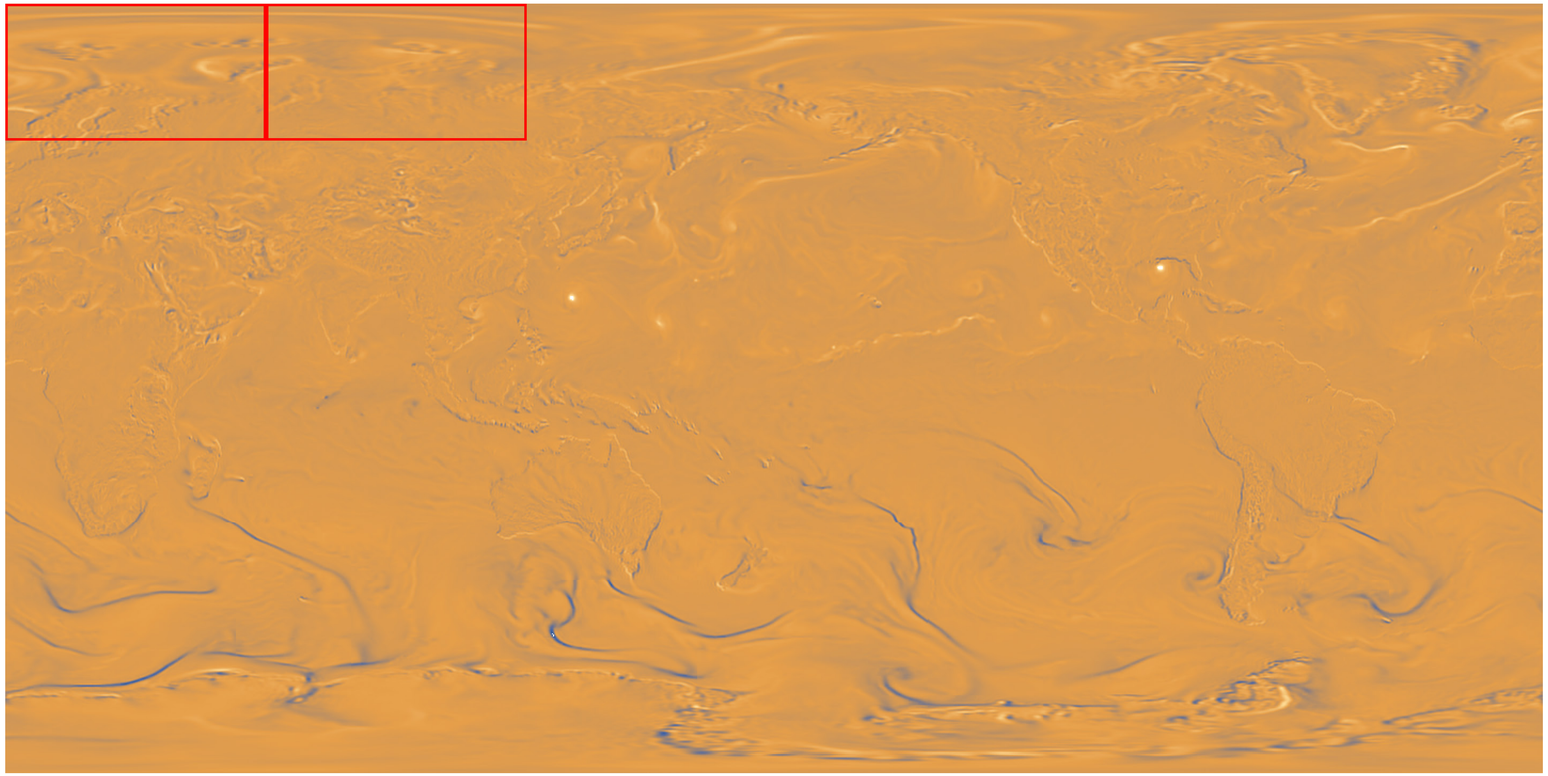
Outlook

- How to do global forecasts with a local model?



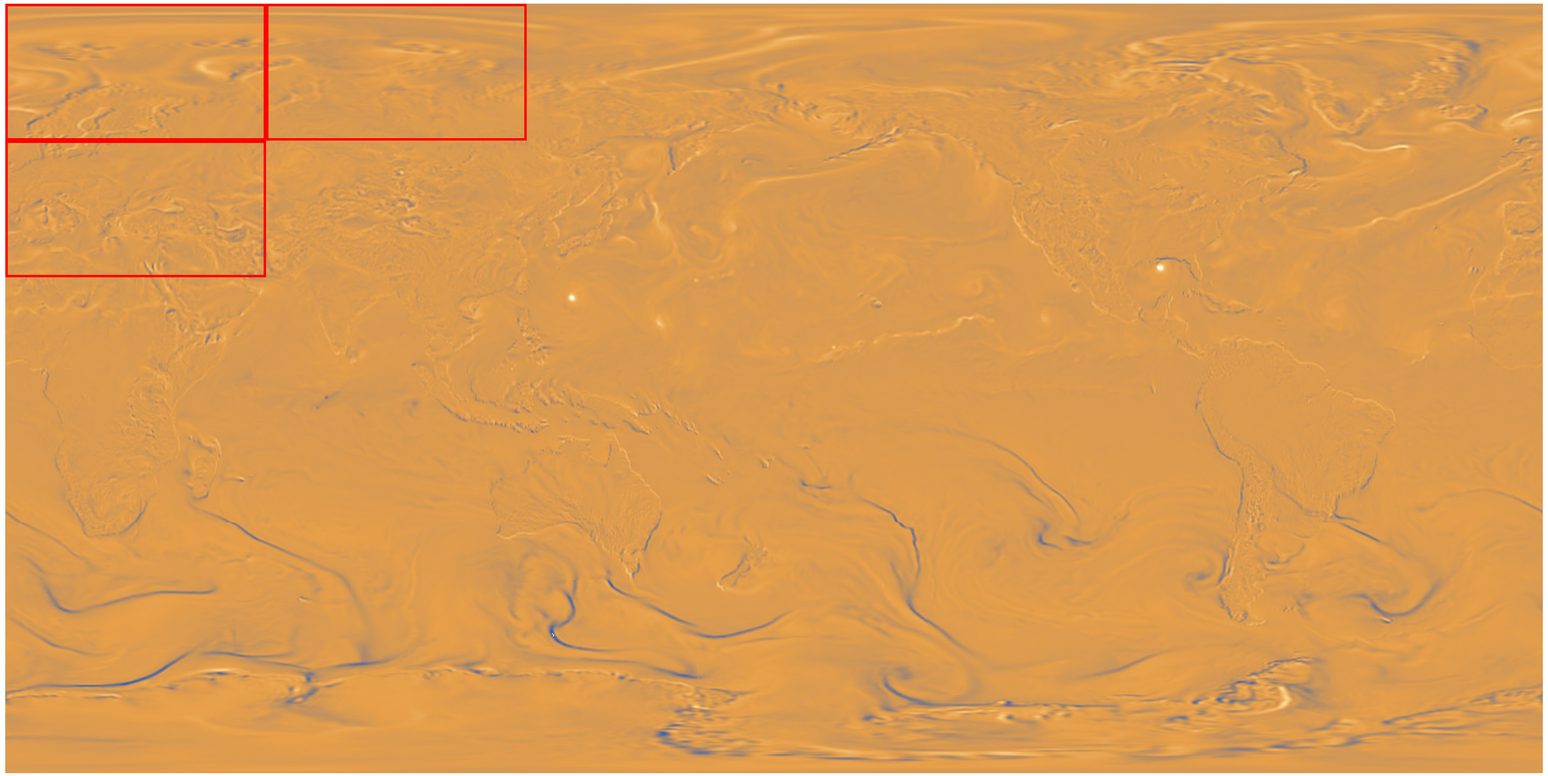
Outlook

- How to do global forecasts with a local model?



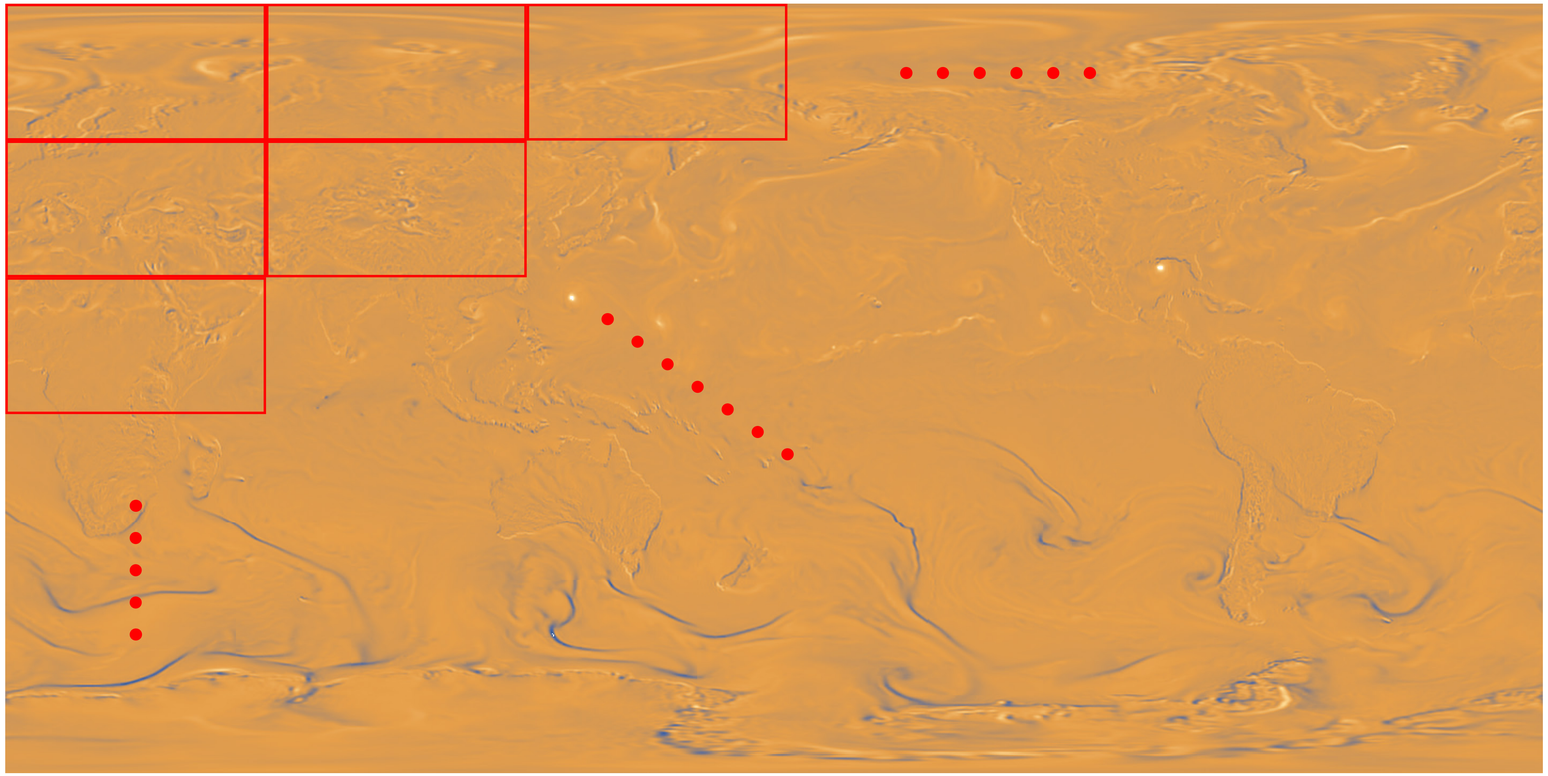
Outlook

- How to do global forecasts with a local model?



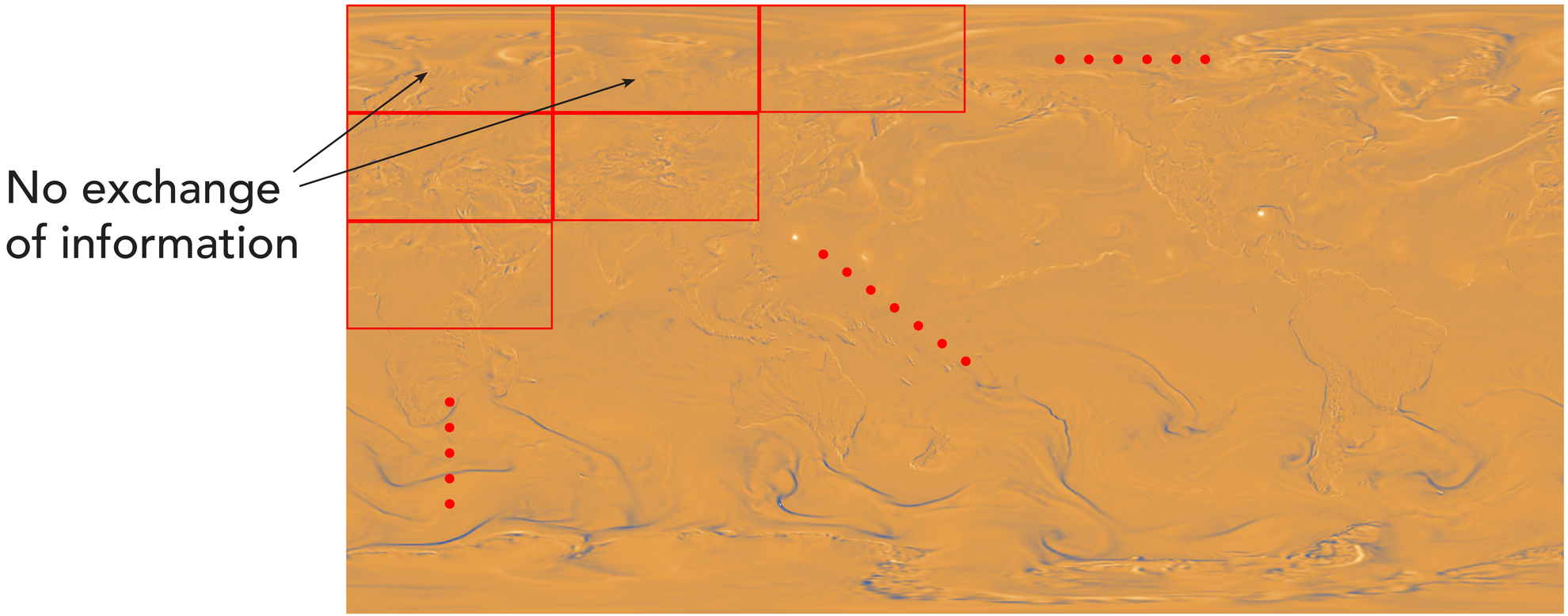
Outlook

- How to do global forecasts with a local model?



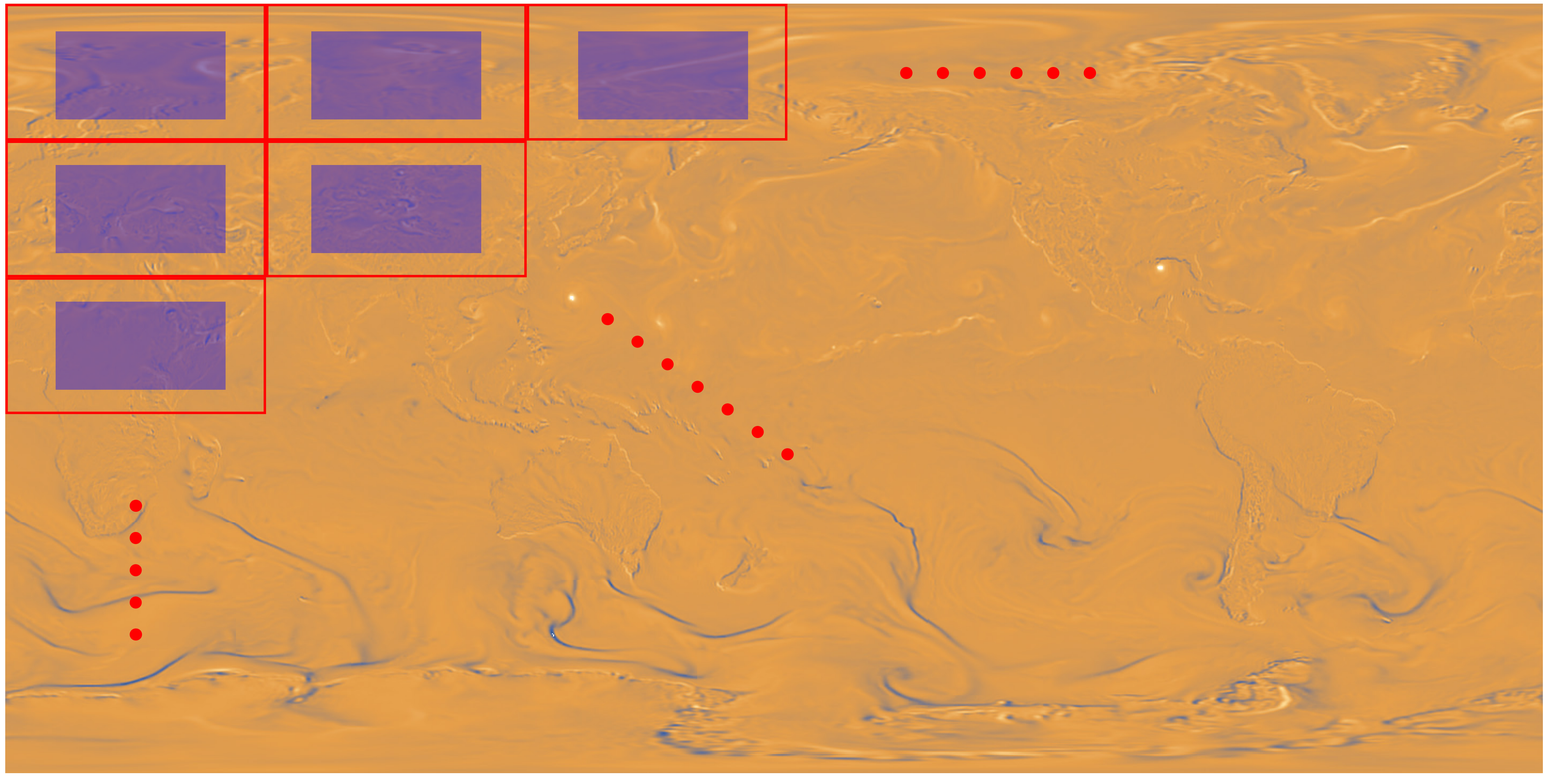
Outlook

- How to do global forecasts with a local model?



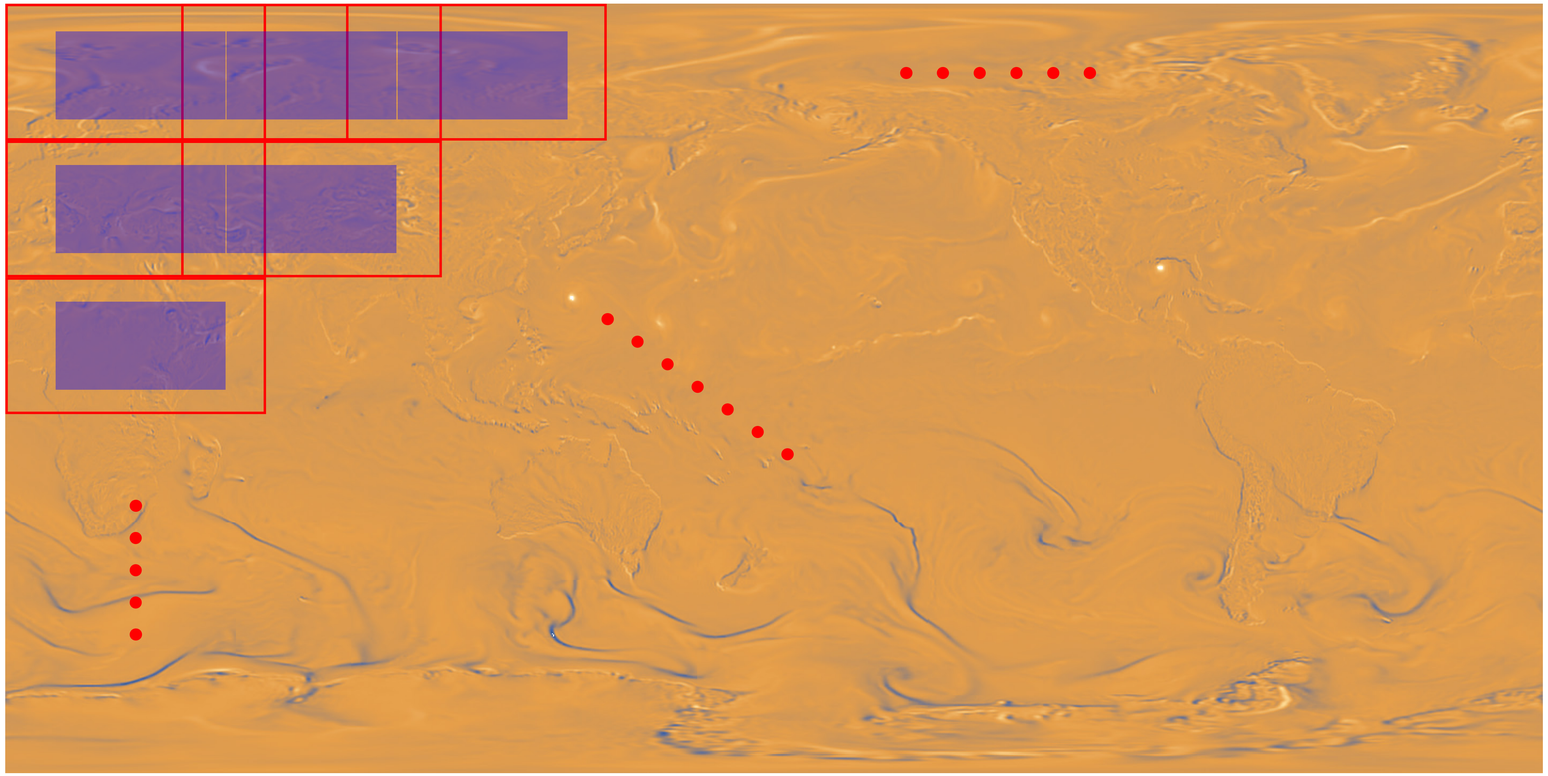
Outlook

- How to do global forecasts with a local model?



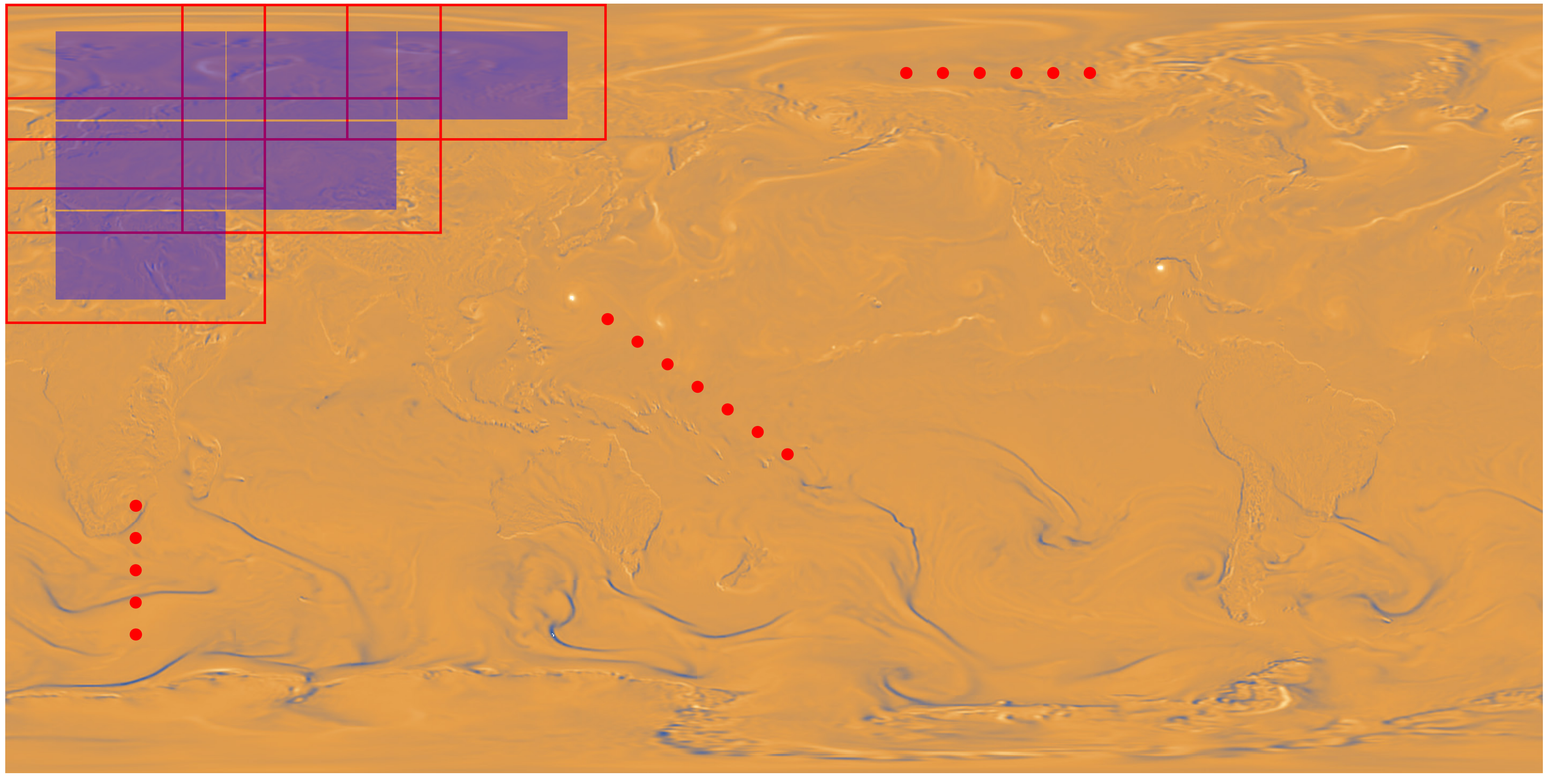
Outlook

- How to do global forecasts with a local model?



Outlook

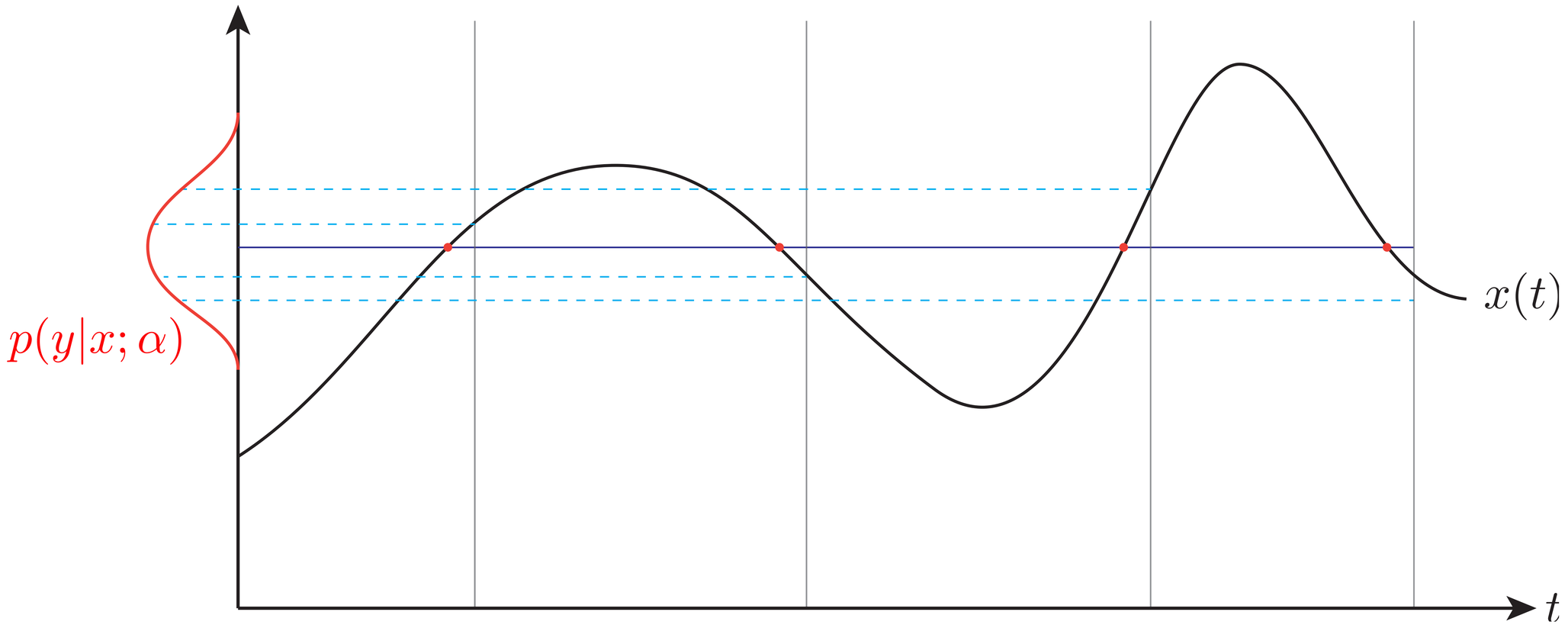
- How to do global forecasts with a local model?



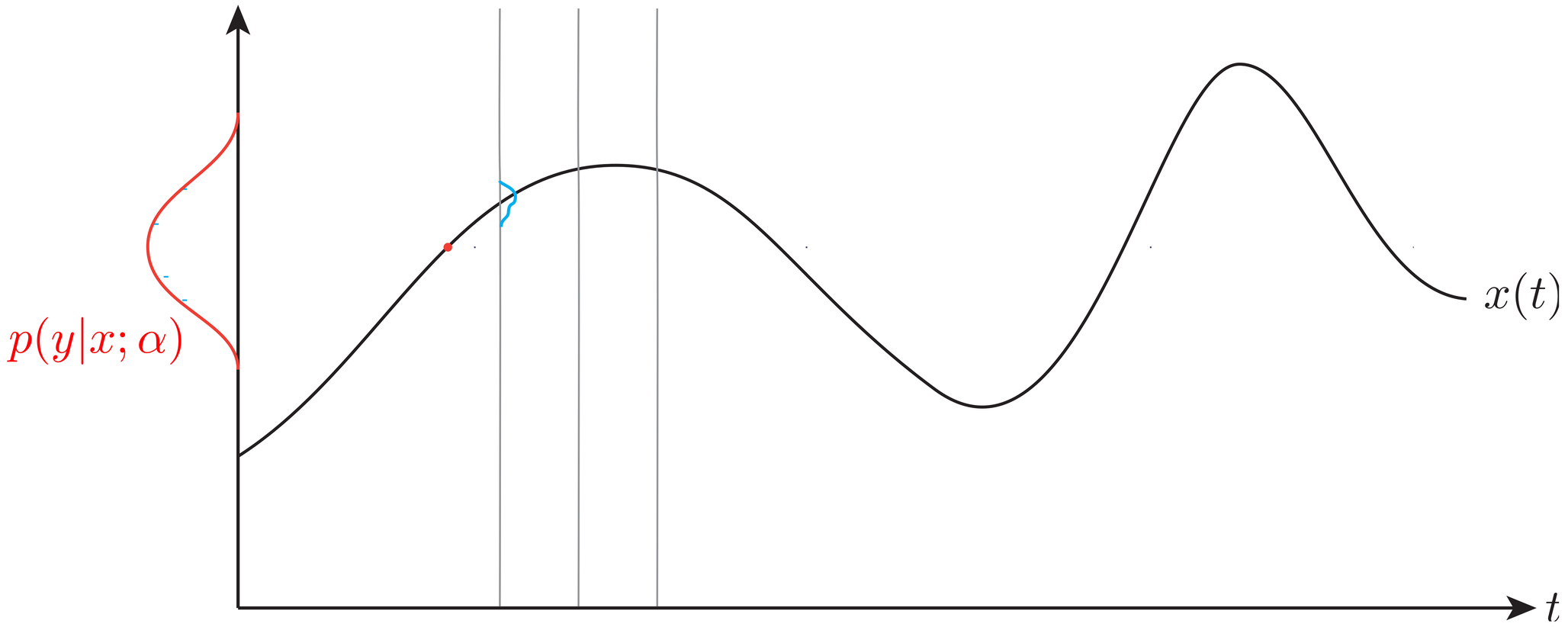
Outlook

- Medium range forecasting
  - Patch local predictions into a global forecast, iterate
    - Avoid artifacts due to patching and ensure robust exchange of information
  - How to sample reasonable number of ensemble trajectories from from current 1-step ensemble?
    - Similar to text generation for LLMs--what tricks can we borrow (e.g. reinforcement learning)?

Outlook



Outlook



Outlook

- Multiformer trained on local neighborhoods
  - Flexibility to use as global or local model and for different region sizes and with different vertical levels
  - Easily assembled into application-specific configurations and fine-tuned to consistent model
    - Also allows to control computational costs of applications
  - Pre-train more fields and tracers?
  - Ocean, land multi-formers?

Outlook

- Training from/with observations
  - Motivation: subsume observational record into coherent form amenable for applications with less biases compared to existing ones and small-scale effects

Outlook

- Training from/with observations
  - Motivation: subsume observational record into coherent form amenable for applications with less biases compared to existing ones and small-scale effects
  - How to train with diverse set of observations?
  - Use reanalysis for pre-training, then bias correct?
  - Can one continuously update a model with observation?
  - How to handle and propagate uncertainties?

Outlook

- Statistical model
  - Reconstruct *statistically consistent* states from just (data, time, location)
  - Reconstruct missing fields from other ones
  - Further develop concept of AI-based counter-factuals

Outlook

- How to exploit a model like AtmoRep for climate?
  - Observations contain, in an aggregated, statistical form, small scale effects that are difficult to model classically (closure problem)
  - Use pre-trained AtmoRep as parametrization for conventional climate model to improve long-term biases compared to existing models?

Summary

AtmoRep

- Numerical statistical atmospheric model $p_\theta(x|y,\alpha)$

AtmoRep

- Numerical statistical atmospheric model $p_\theta(x|y,\alpha)$
  - Makes the observational record (in the form of ERA5) available for applications
  - Complementary to classical GCMs and ESMs
  - Long training leads to continuous improvement

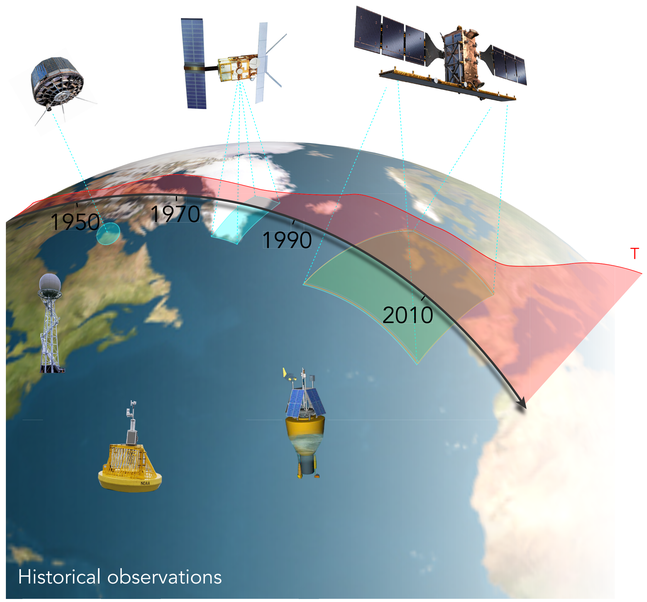
AtmoRep

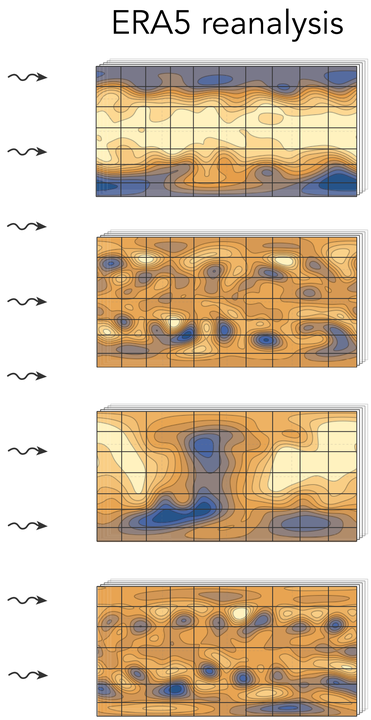
- Numerical statistical atmospheric model $p_\theta(x|y,\alpha)$
  - Makes the observational record (in the form of ERA5) available for applications
  - Complementary to classical GCMs and ESMs
  - Long training leads to continuous improvement
- Versatile capabilites
  - Intrinsic capabilities through BERT-type training
  - Extension to various applications, e.g. with tail network

AtmoRep

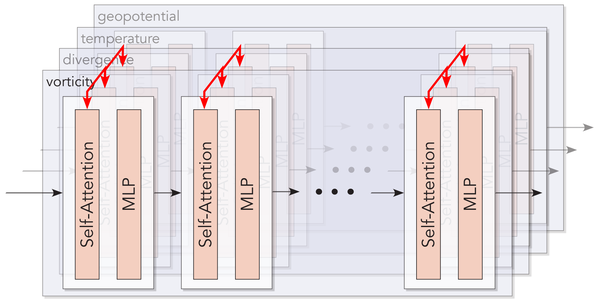
Code and pre-trained models coming soon.

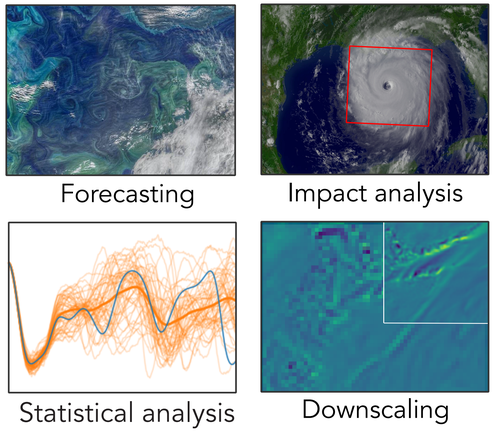
pre-print

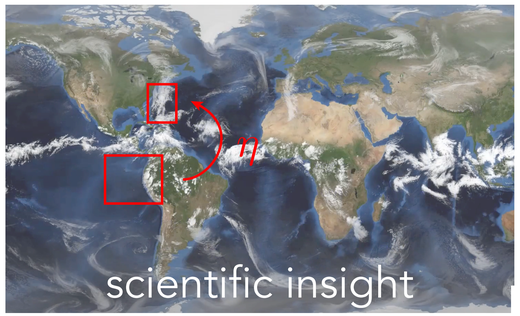




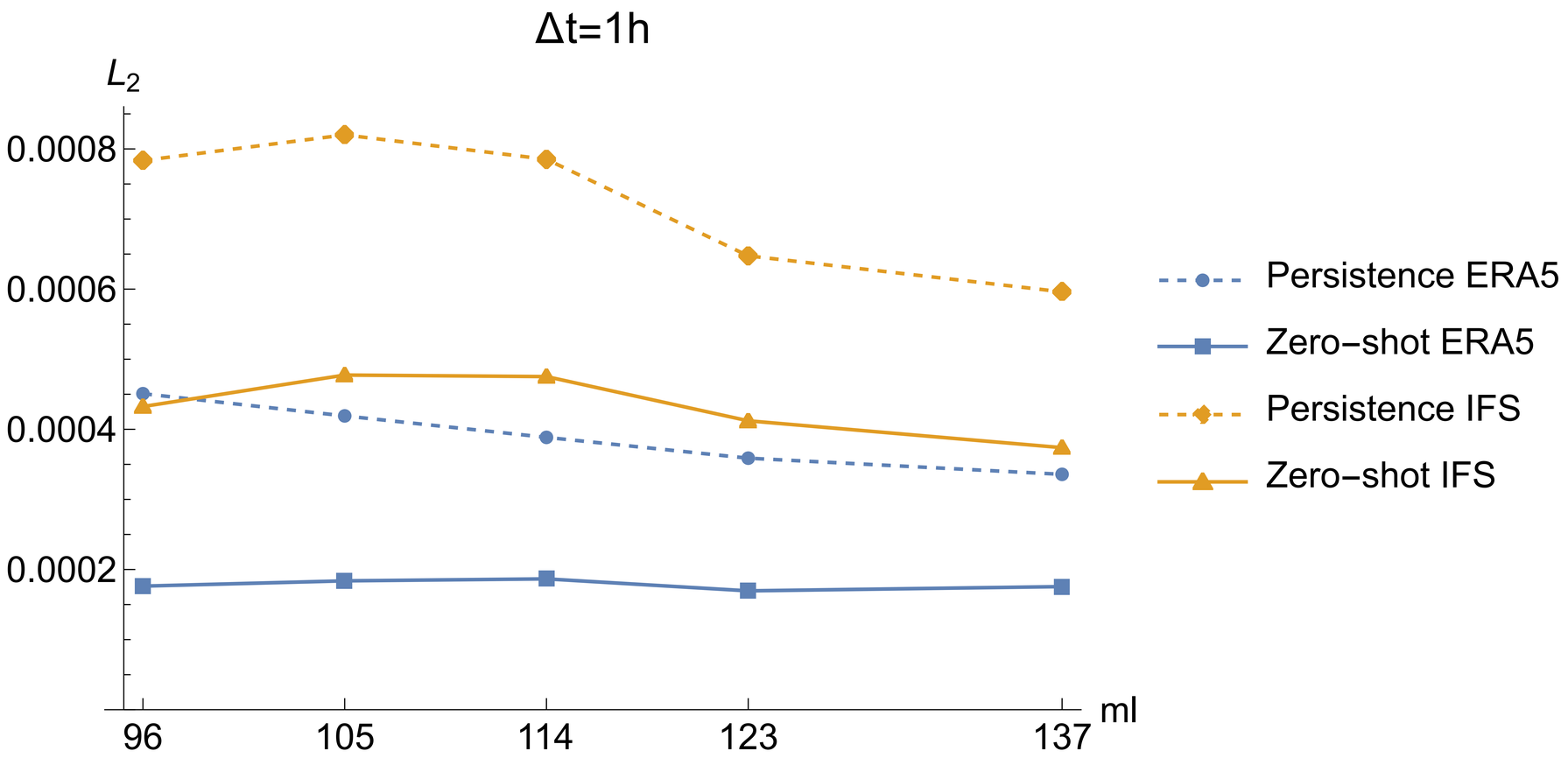
$$p_\theta(y|x)$$



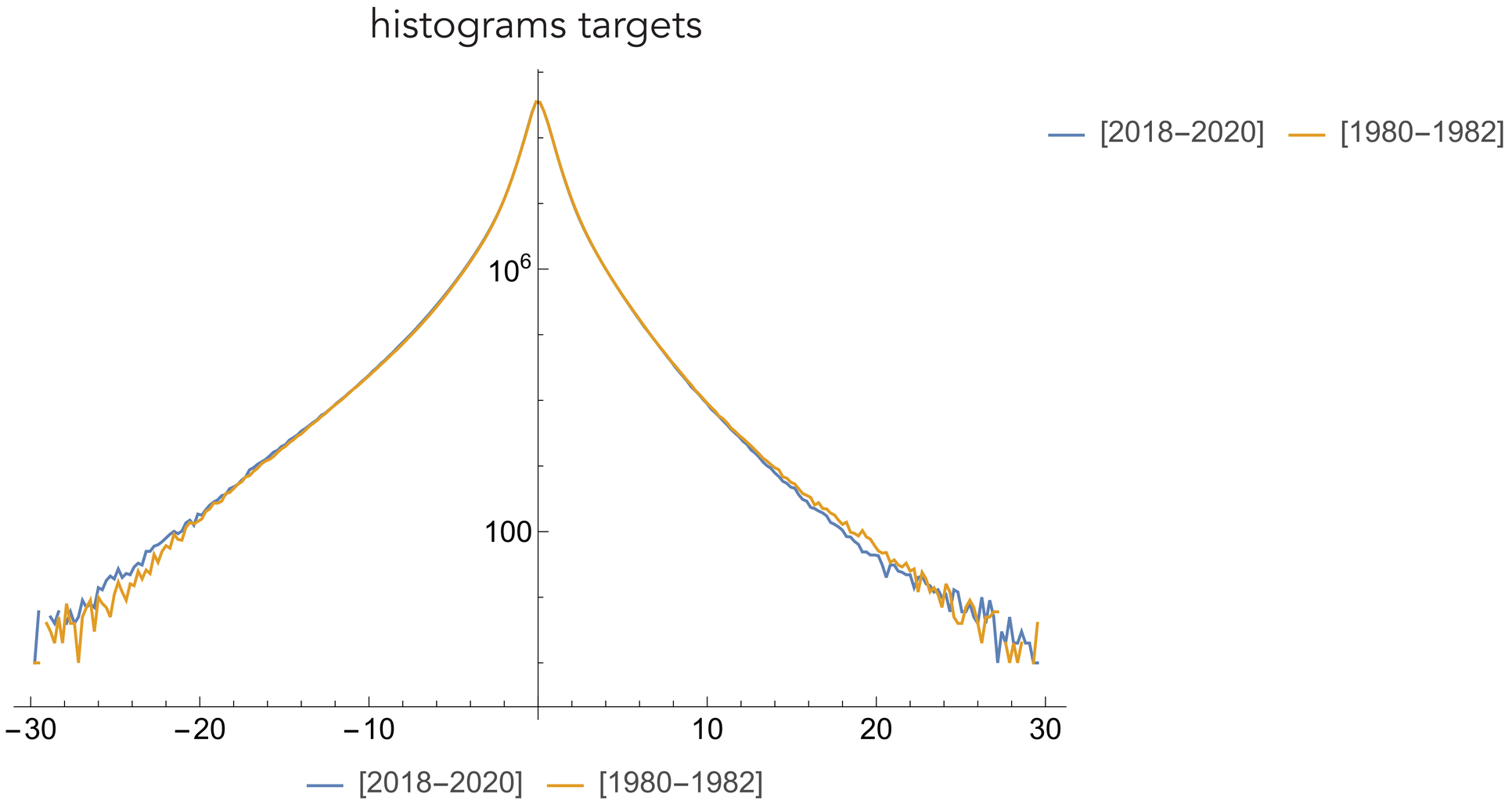




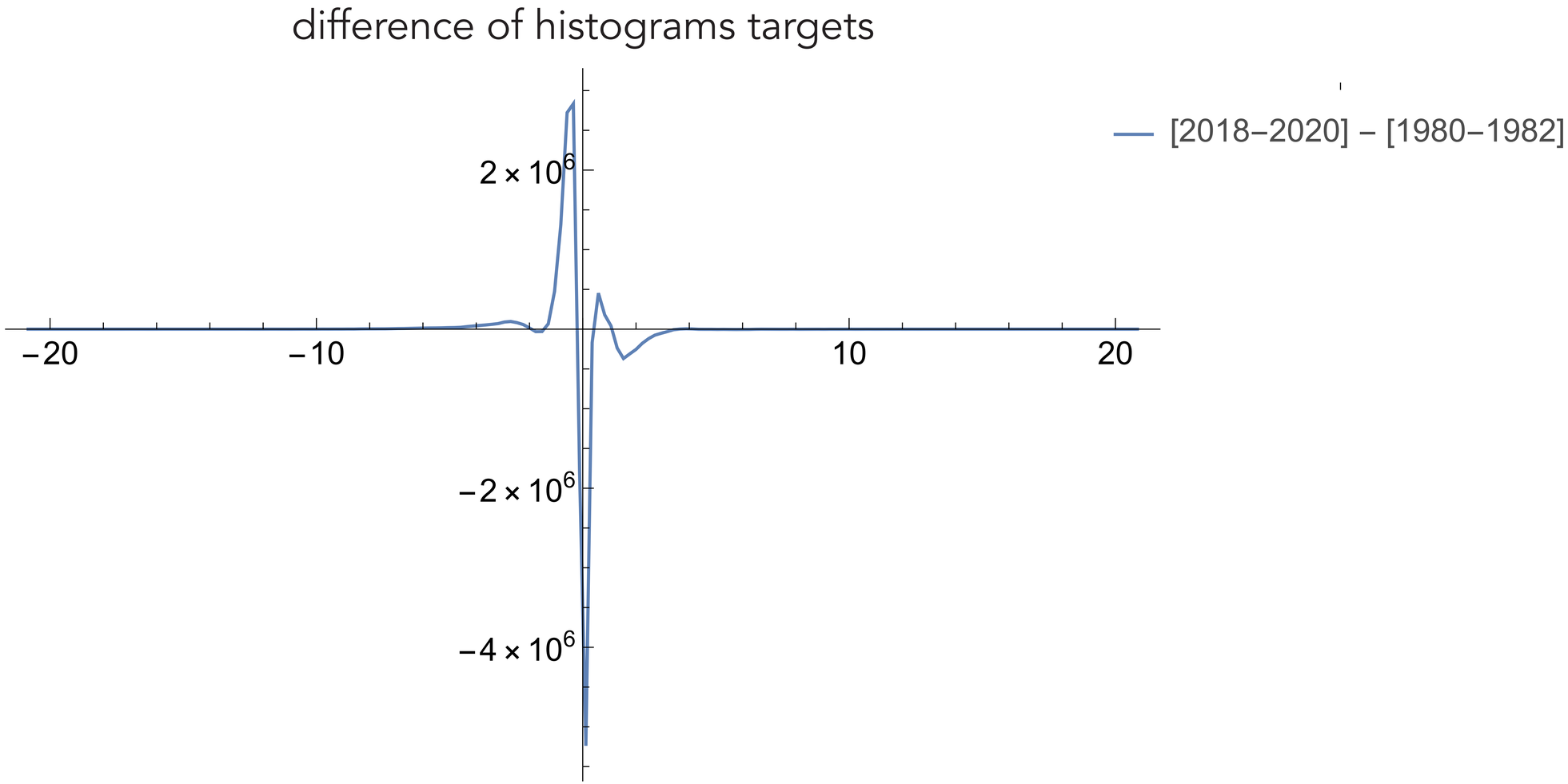
Model correction



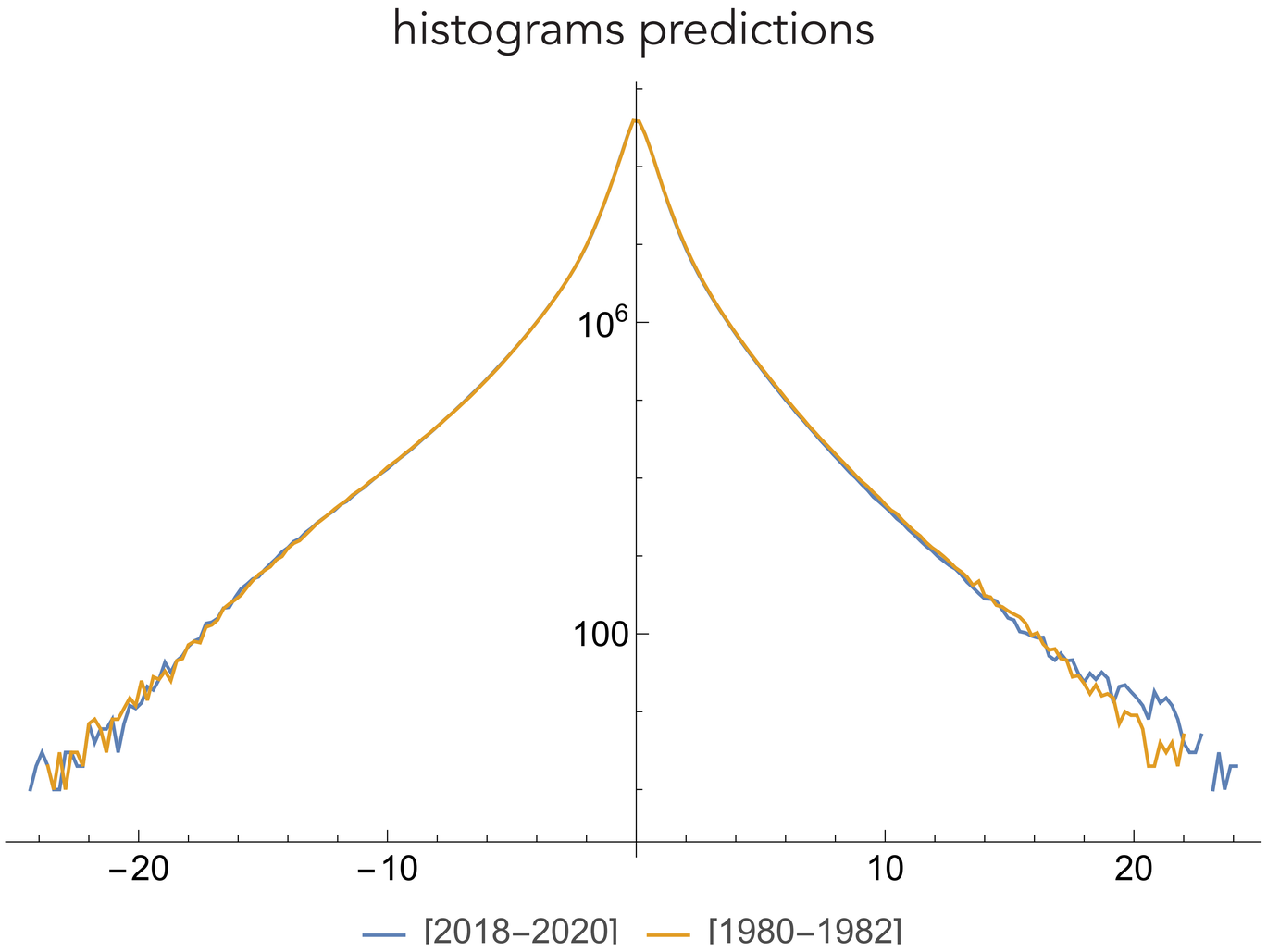
Counterfactuals



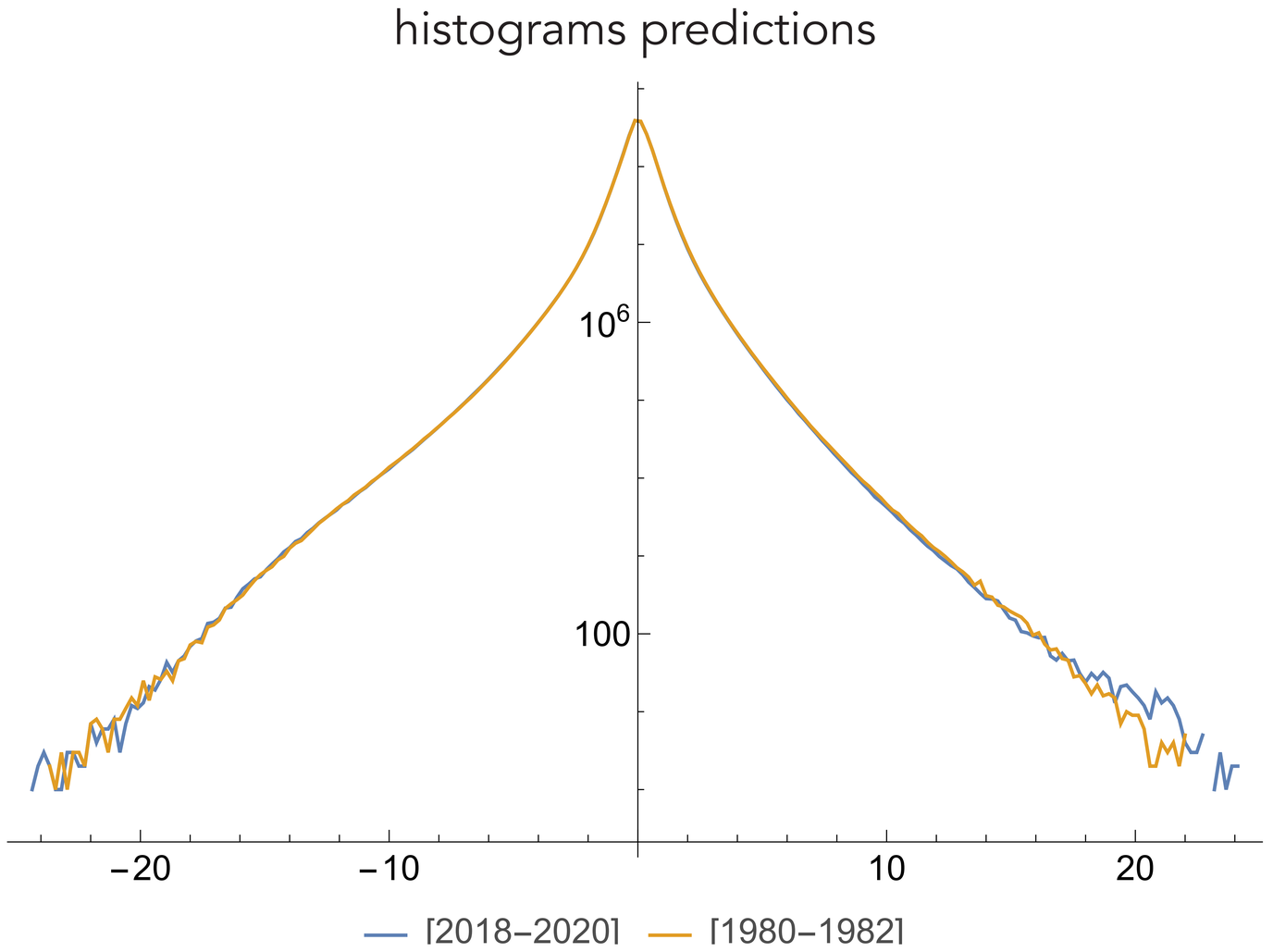
Counterfactuals



Counterfactuals



Counterfactuals

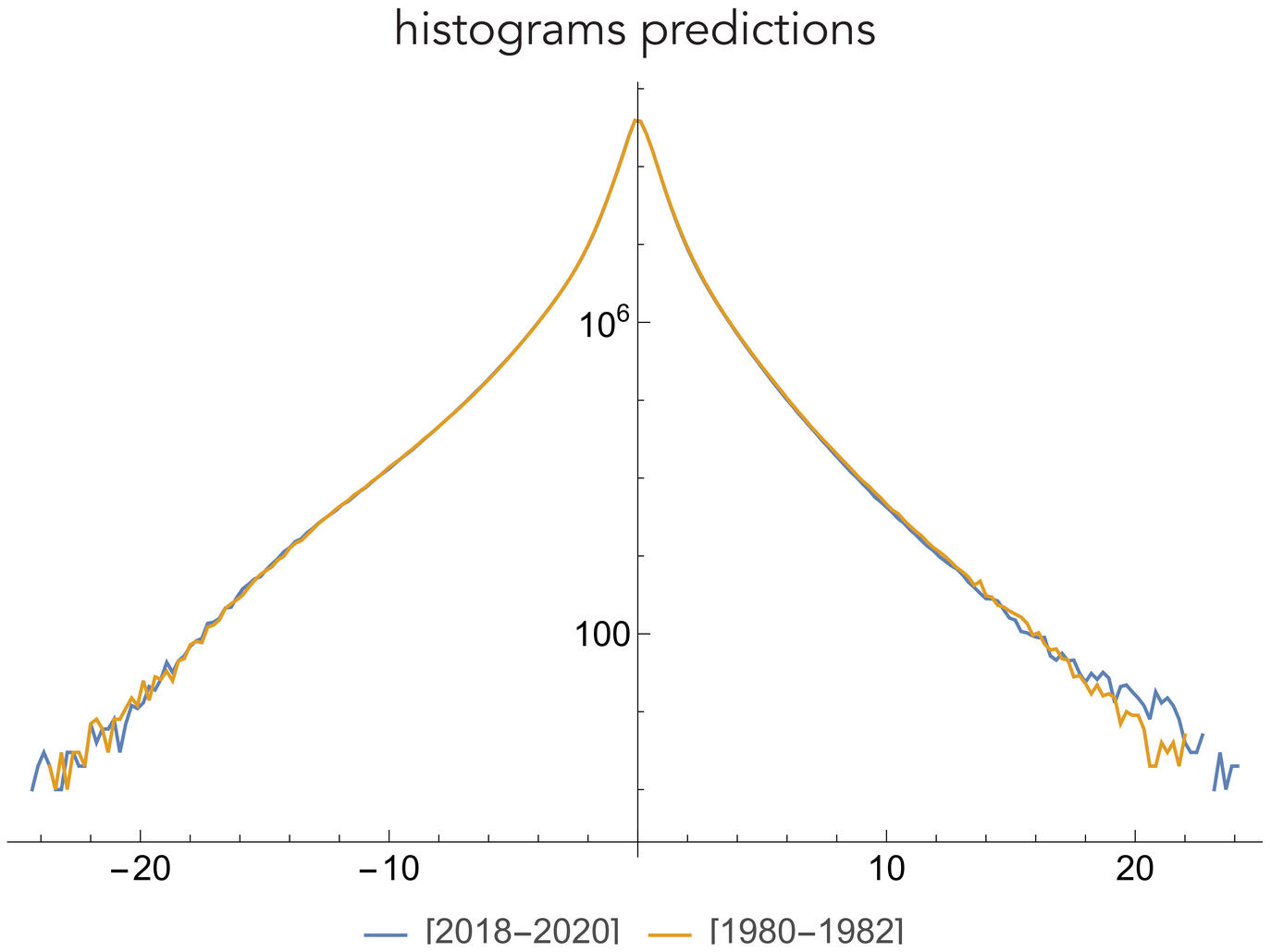


[2018–2020]

initial conditions *but*

$\alpha_{\text{year}} = [1980, 1982]$

Counterfactuals



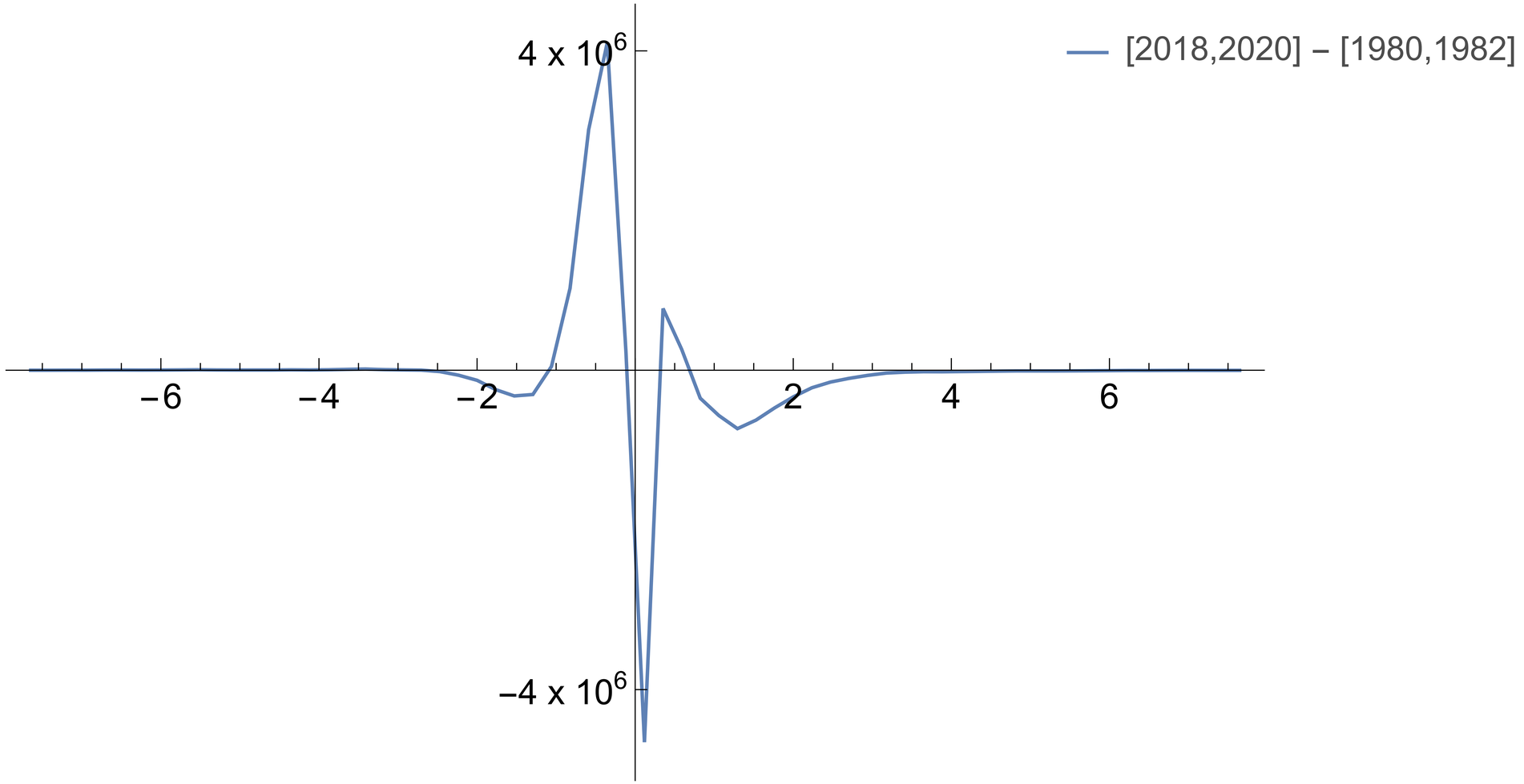
[2018–2020]

initial conditions *but* $\alpha_{\text{year}} = [1980, 1982]$

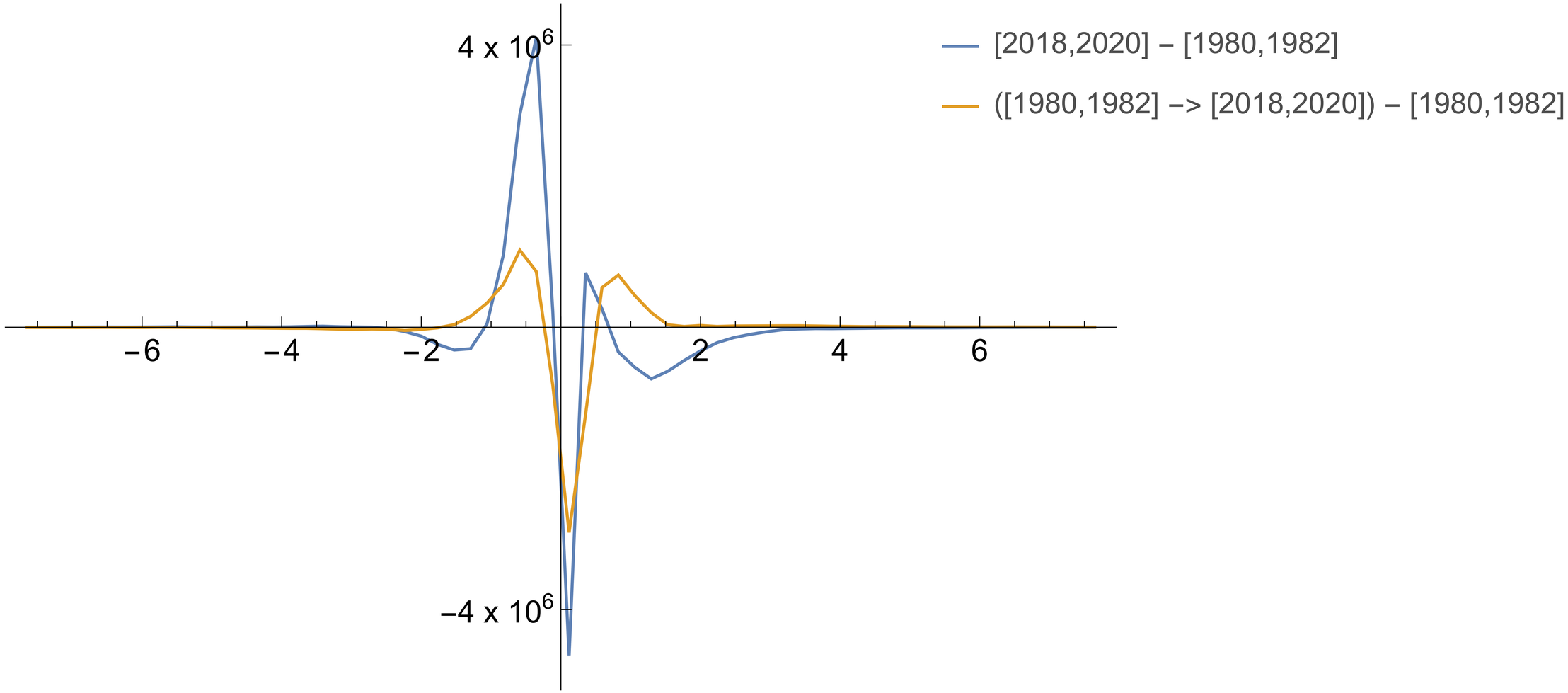
[1980–1982]

initial conditions *but* $\alpha_{\text{year}} = [2018, 2020]$

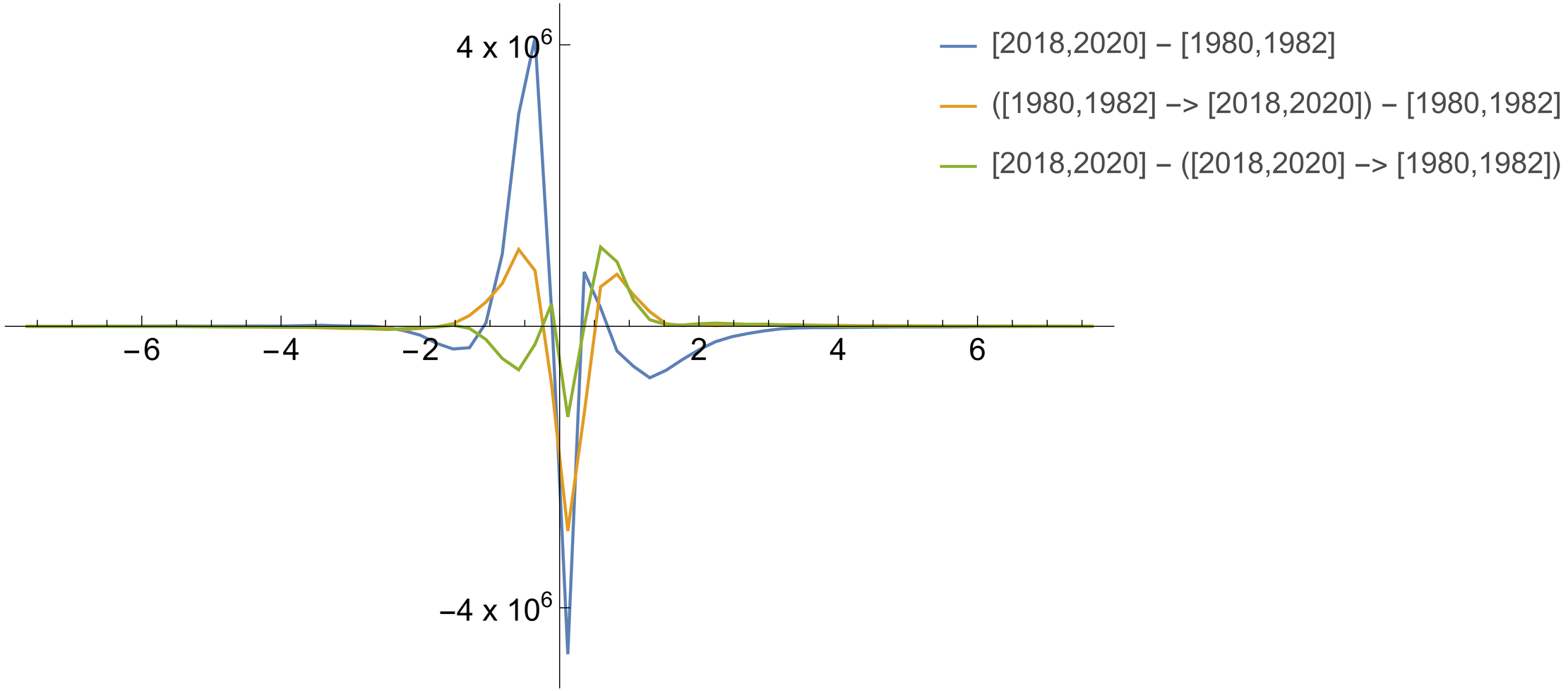
Counterfactuals



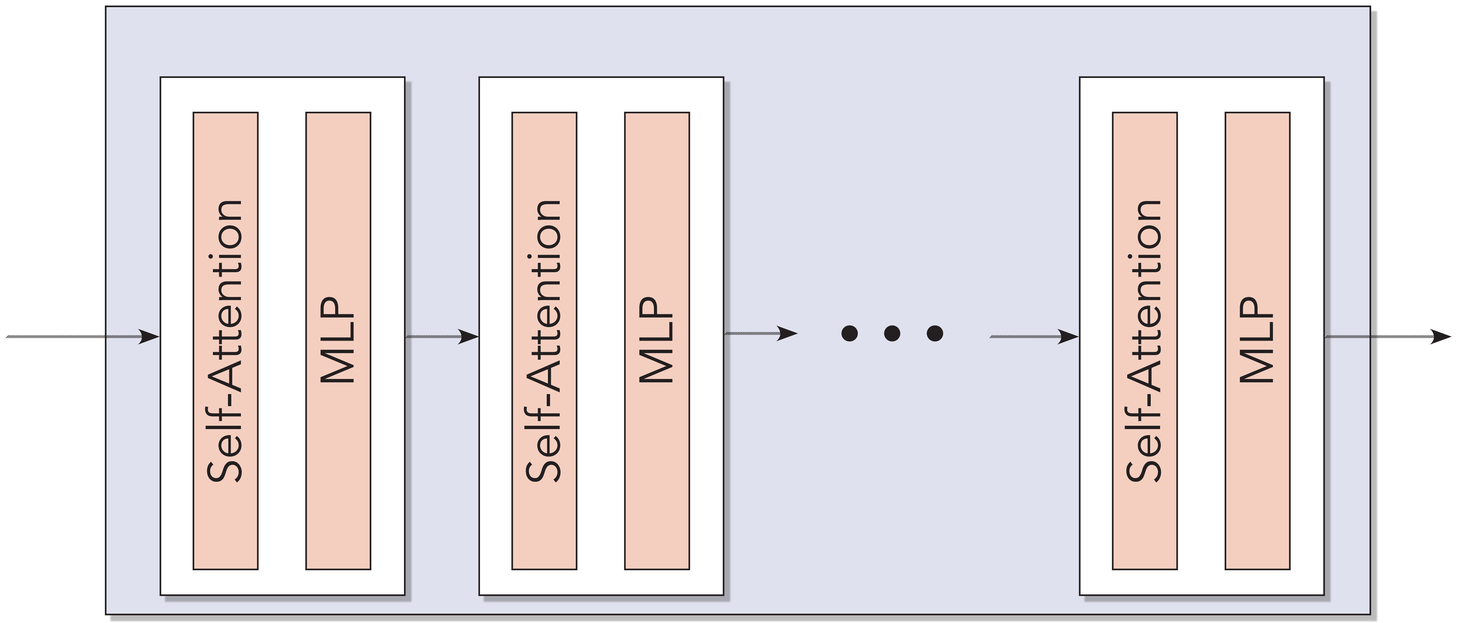
Counterfactuals



Counterfactuals



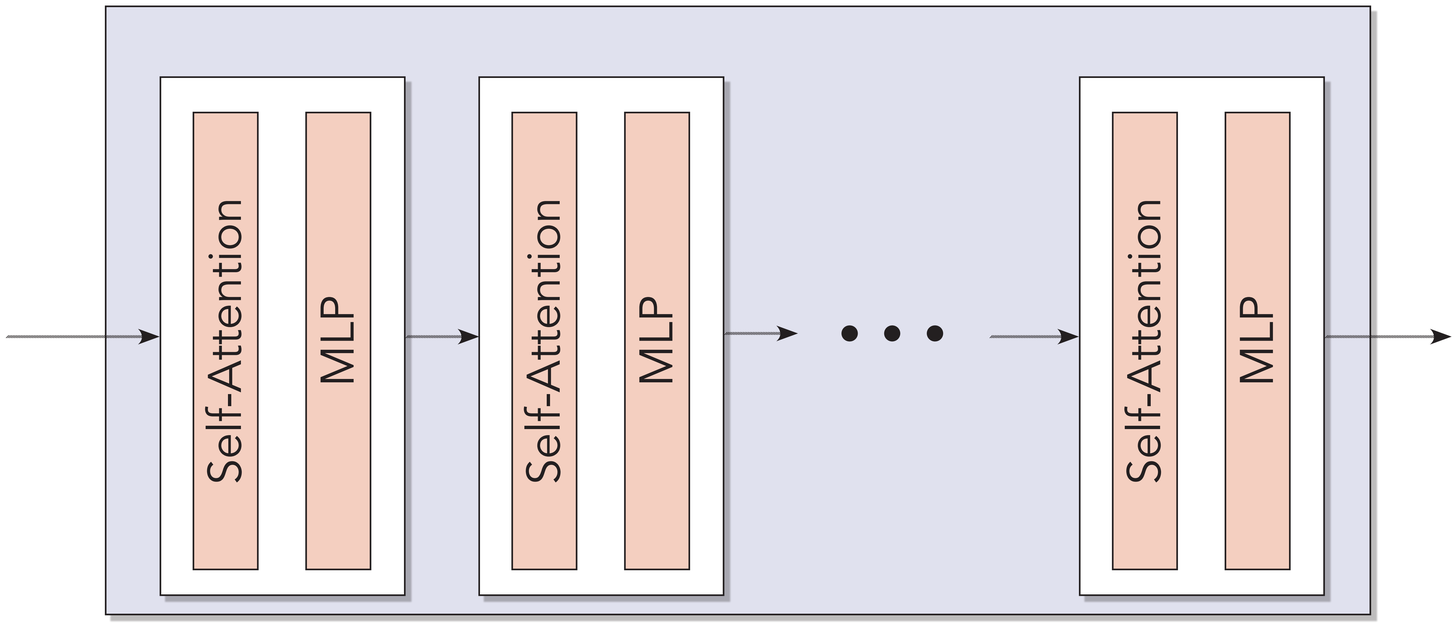
Multiformer



Multiformer

Self attention

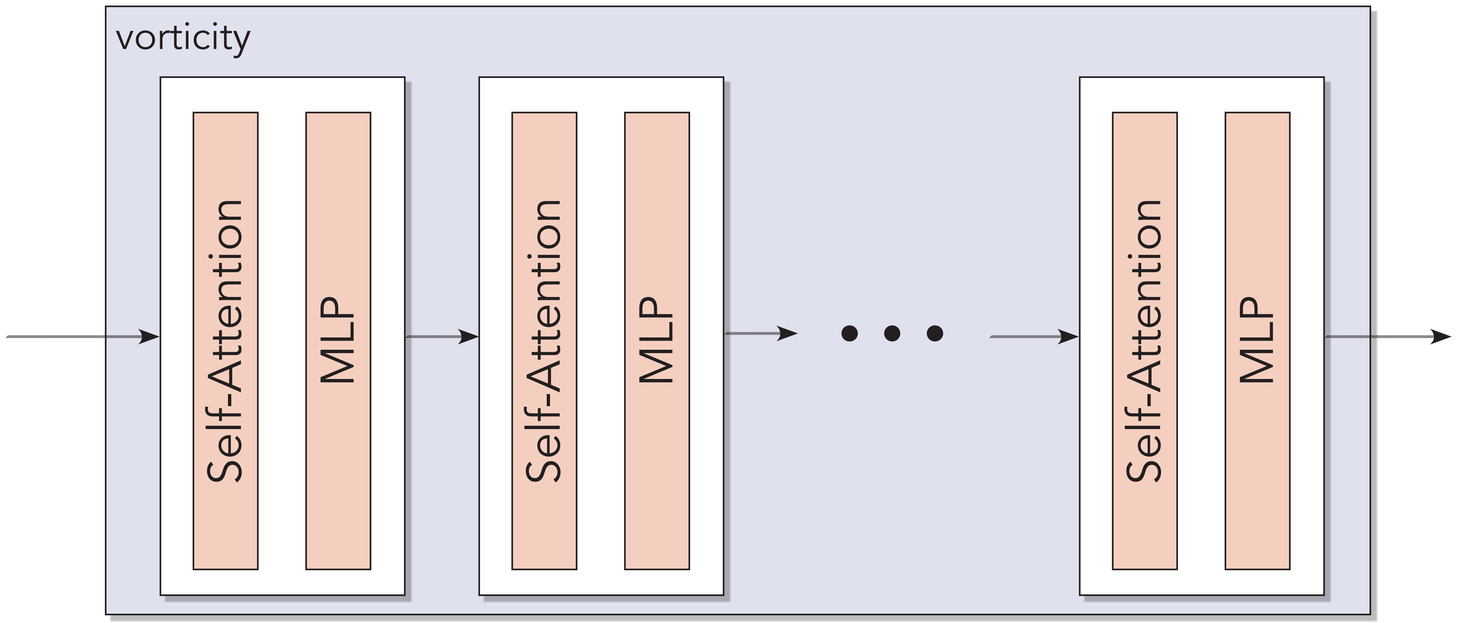
$$\sigma\left(Q\,K^T\right)V$$



Multiformer

Self
attention

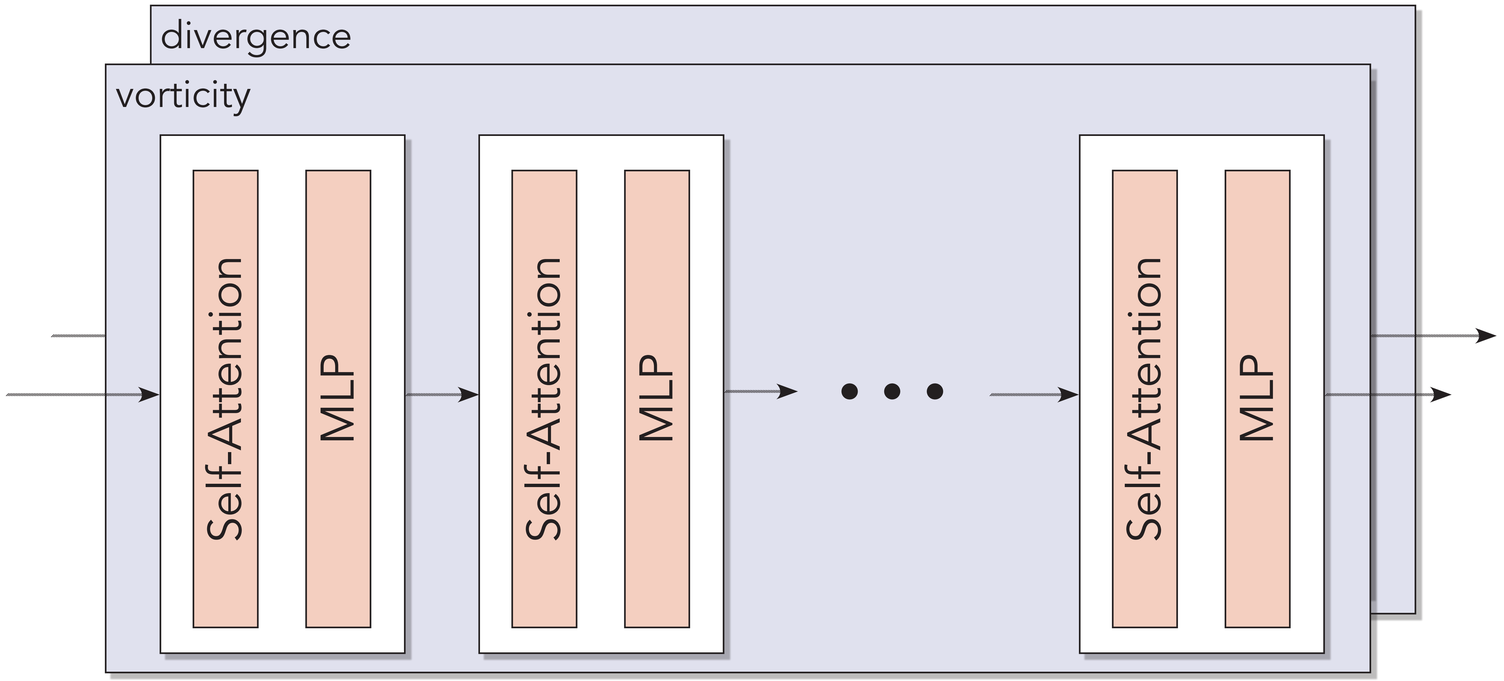
$$\sigma\left(Q\,K^T\right)V$$



Multiformer

Self
attention

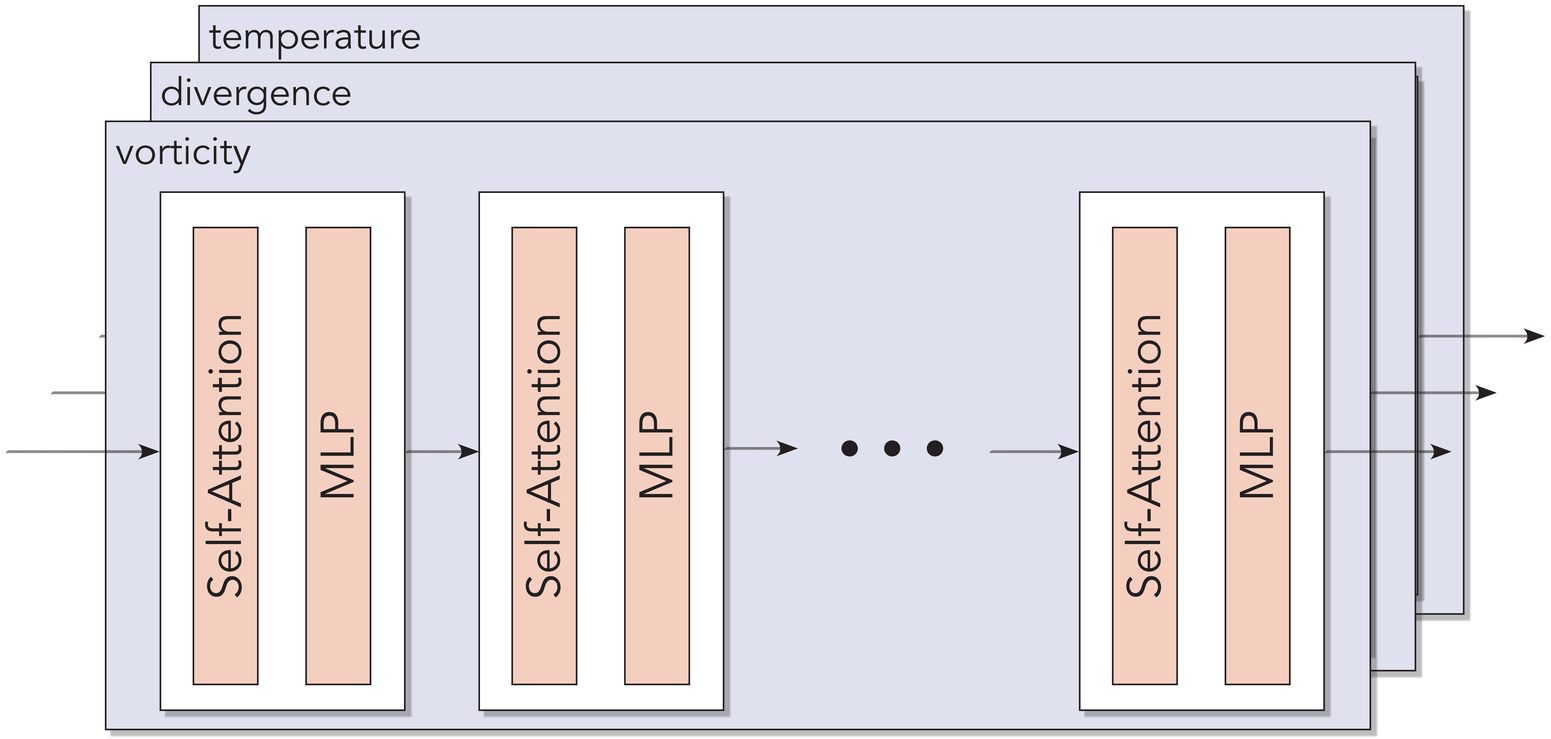
$$\sigma\left(Q\,K^T\right)V$$



Multiformer

Self
attention

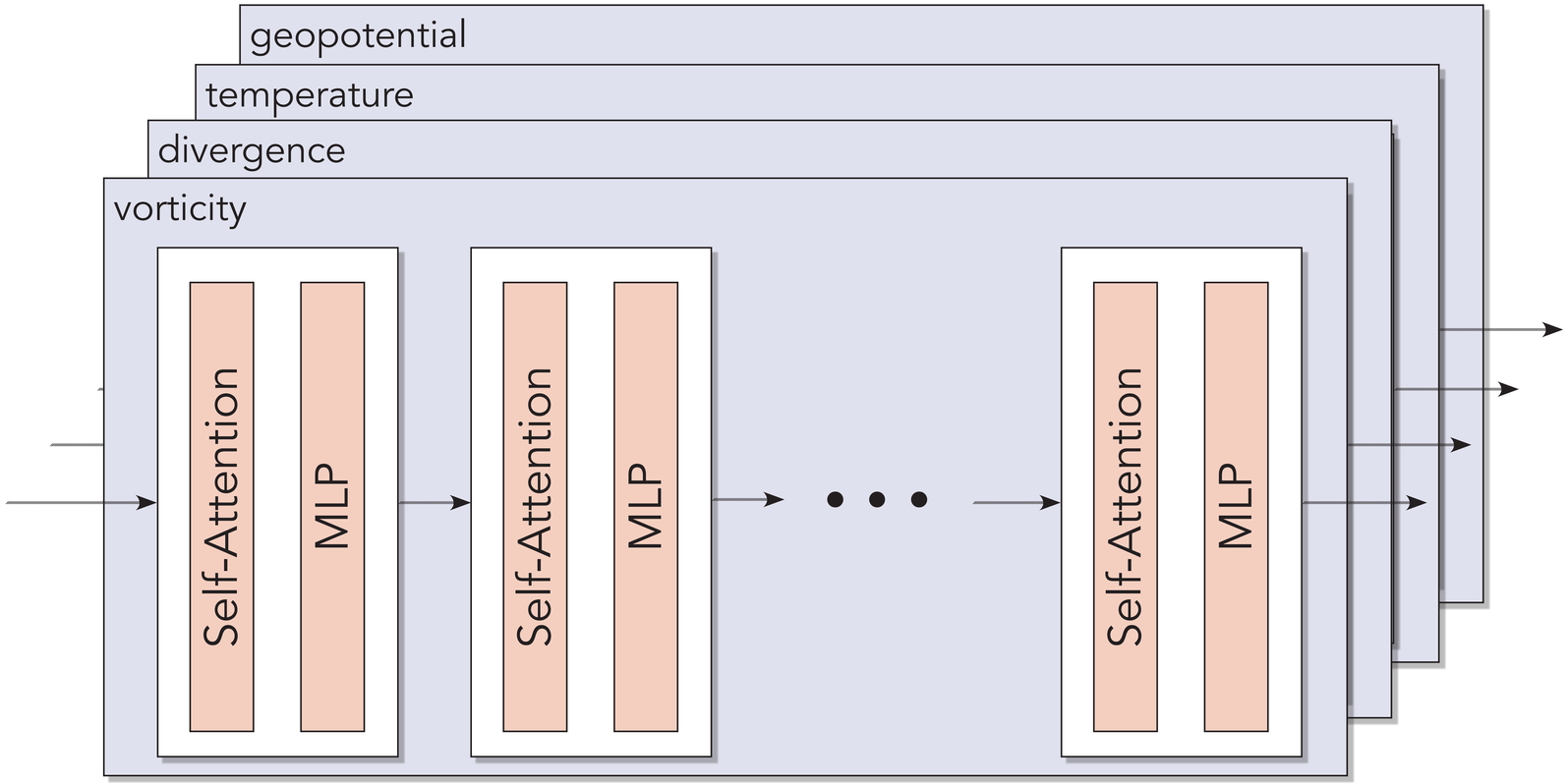
$$\sigma\left(Q\,K^T\right)V$$



Multiformer

Self
attention

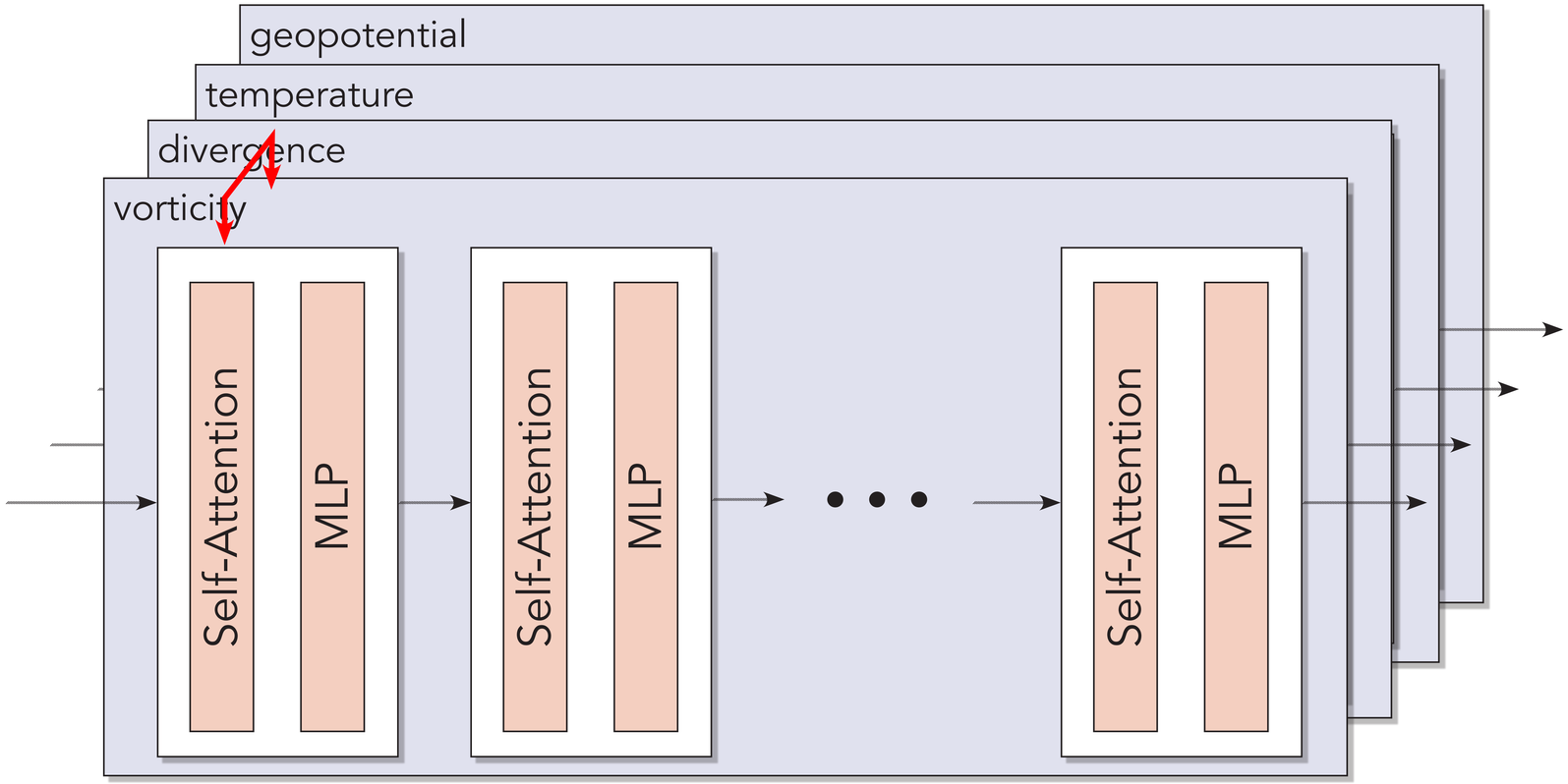
$$\sigma\left(Q\,K^T\right)V$$



Multiformer

Self
attention

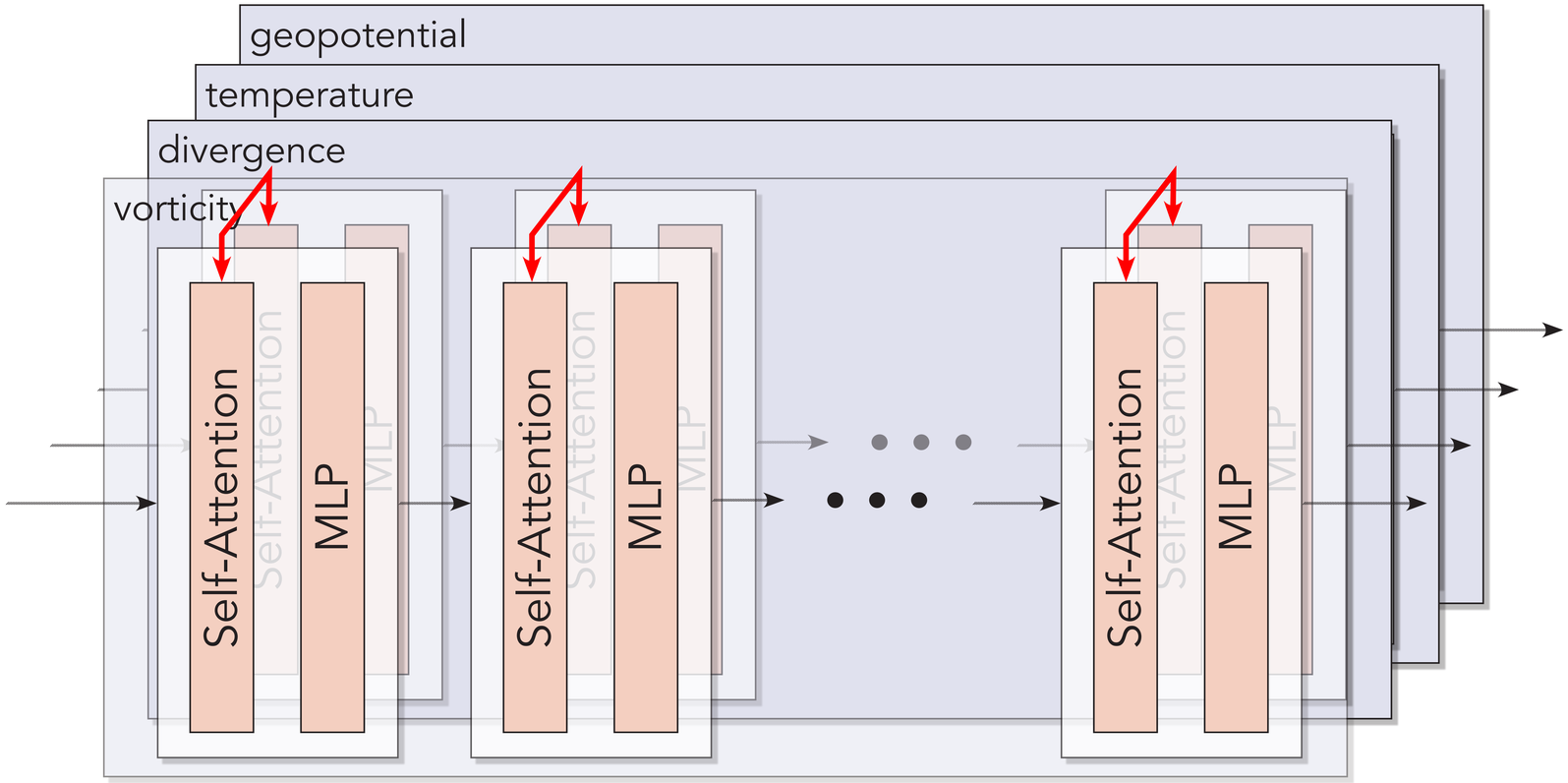
$$\sigma\left(Q\,K^T\right)V$$



Multiformer

Self
attention

$$\sigma\left(Q\,K^T\right)V$$



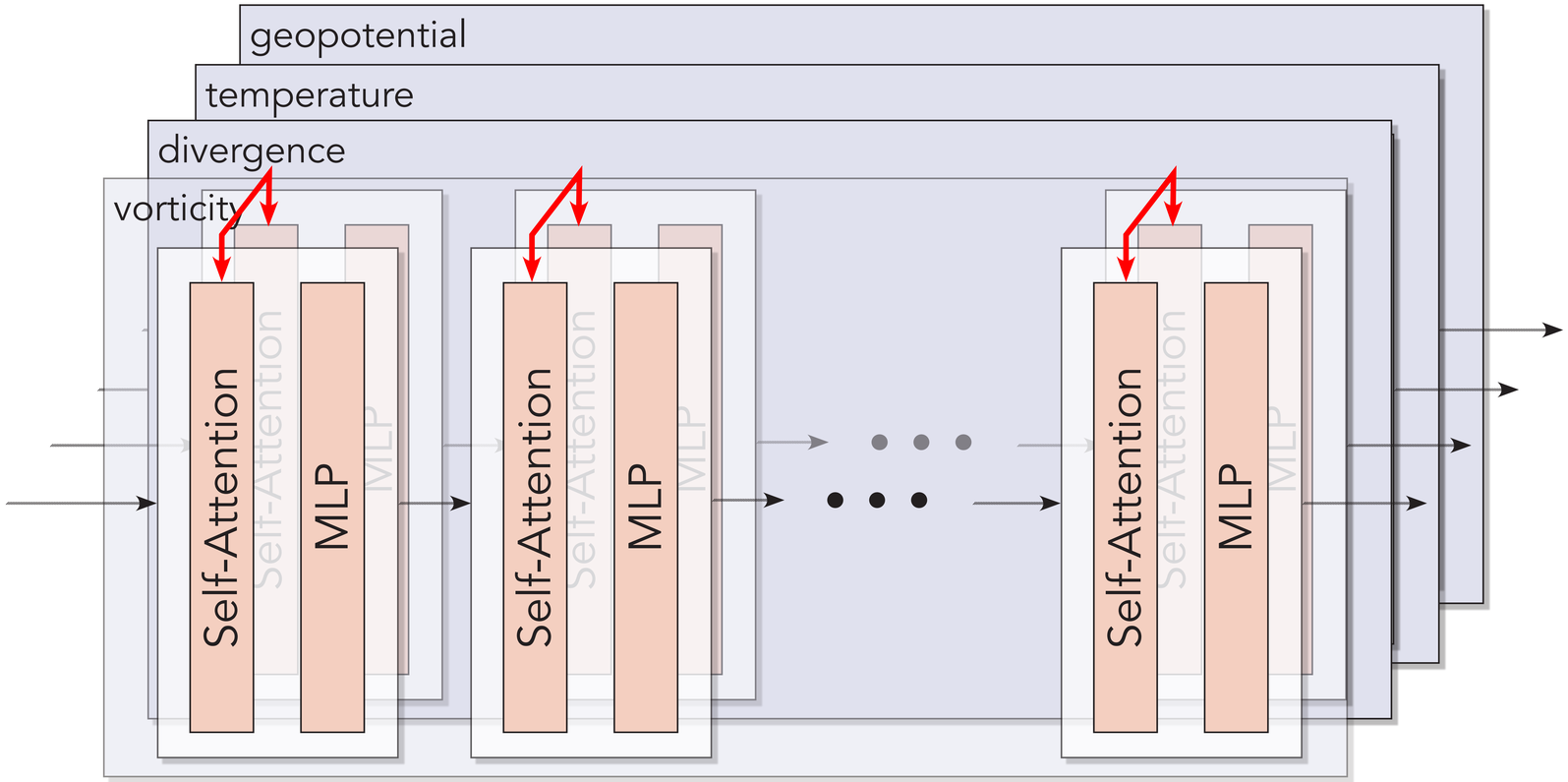
Multiformer

Self attention

$$\sigma\left(Q\,K^T\right)V$$

Cross attention

$$\sigma\left(Q_\zeta\,K_\mu^T\right)V_\mu$$



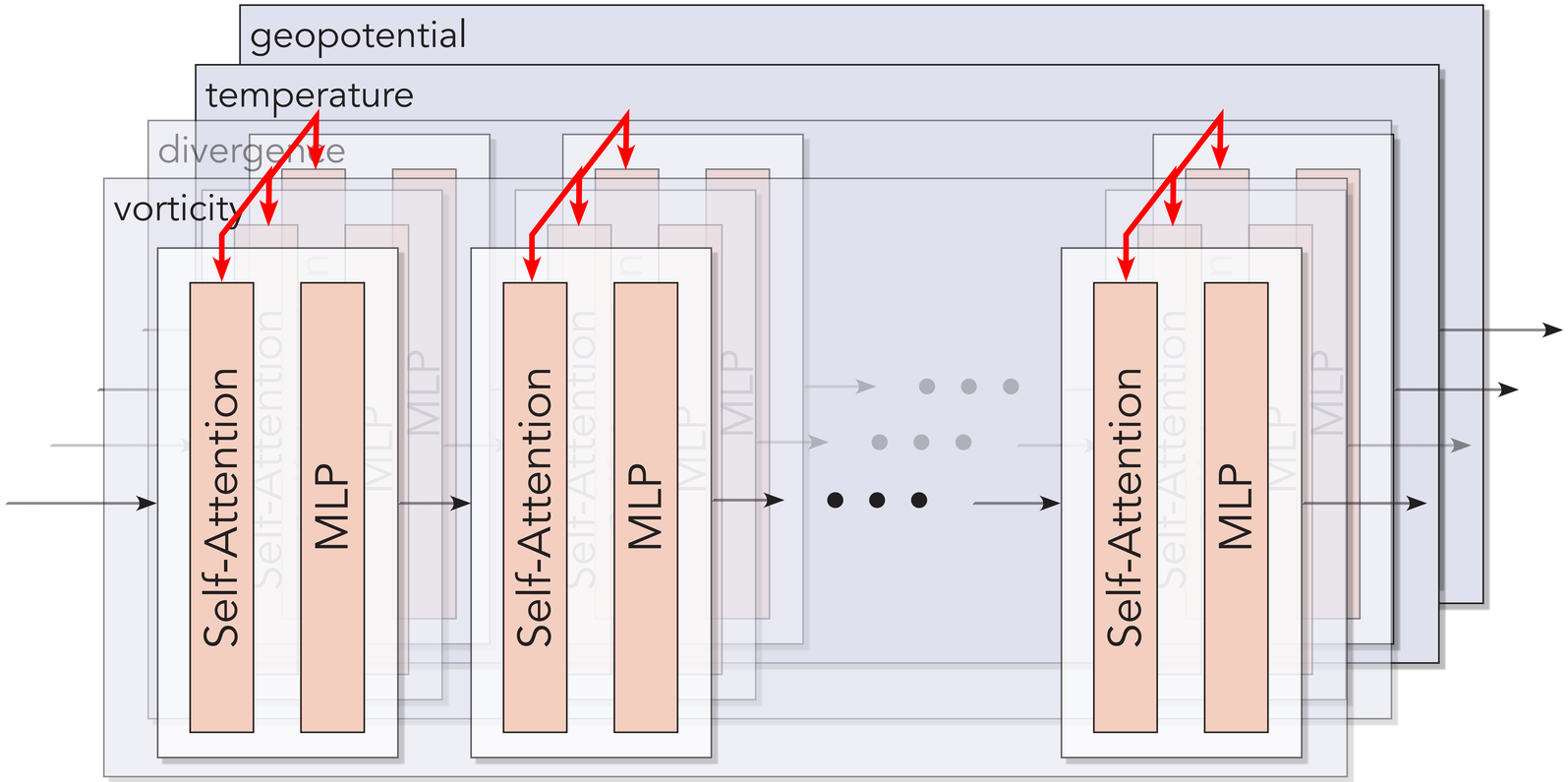
Multiformer

Self attention

$$\sigma\left(Q\,K^T\right)V$$

Cross attention

$$\sigma\left(Q_\zeta\,K_\mu^T\right)V_\mu$$



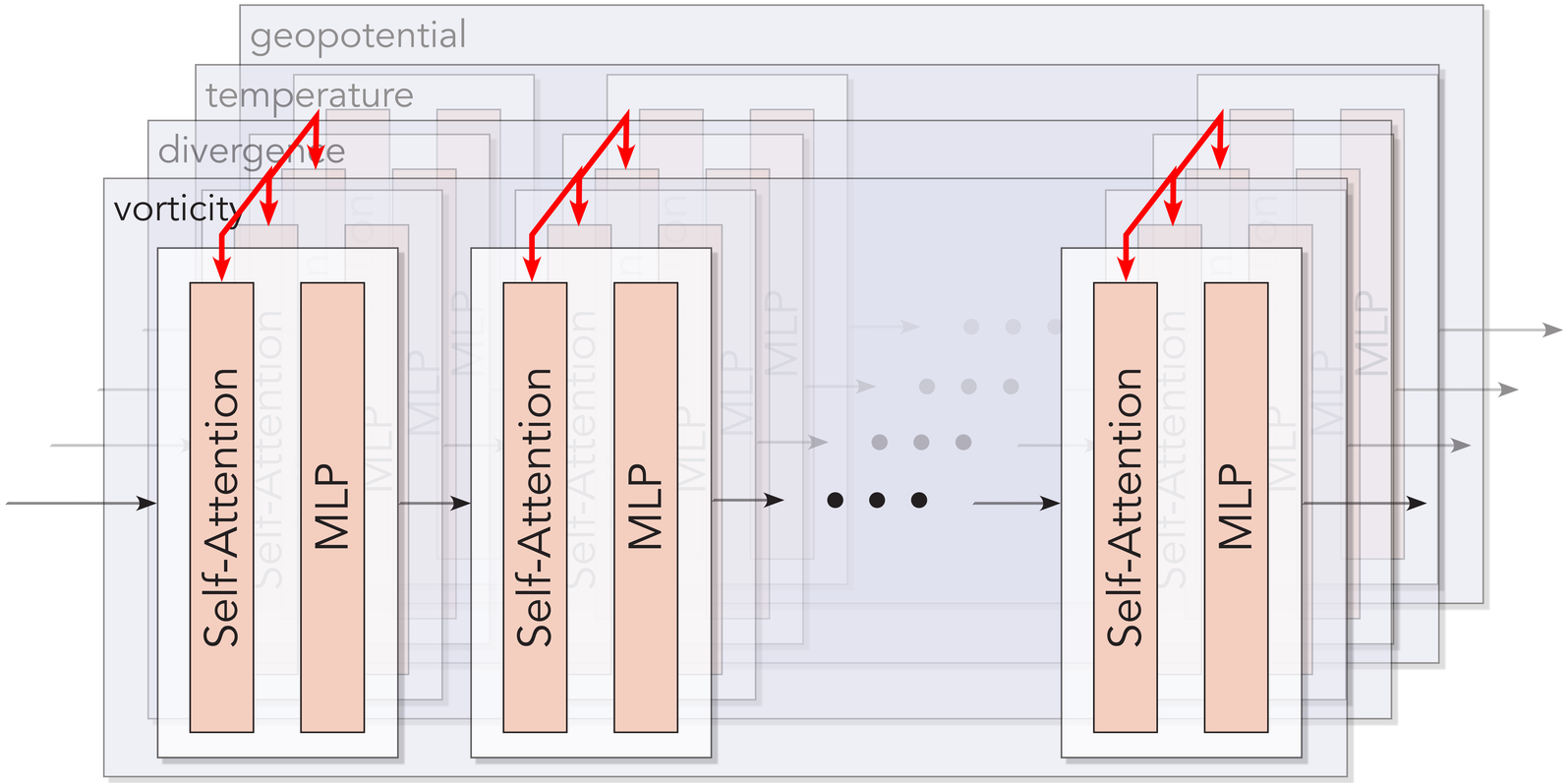
Multiformer

Self attention

$$\sigma\left(Q\,K^T\right)V$$

Cross attention

$$\sigma\left(Q_\zeta\,K_\mu^T\right)V_\mu$$



Multiformer

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

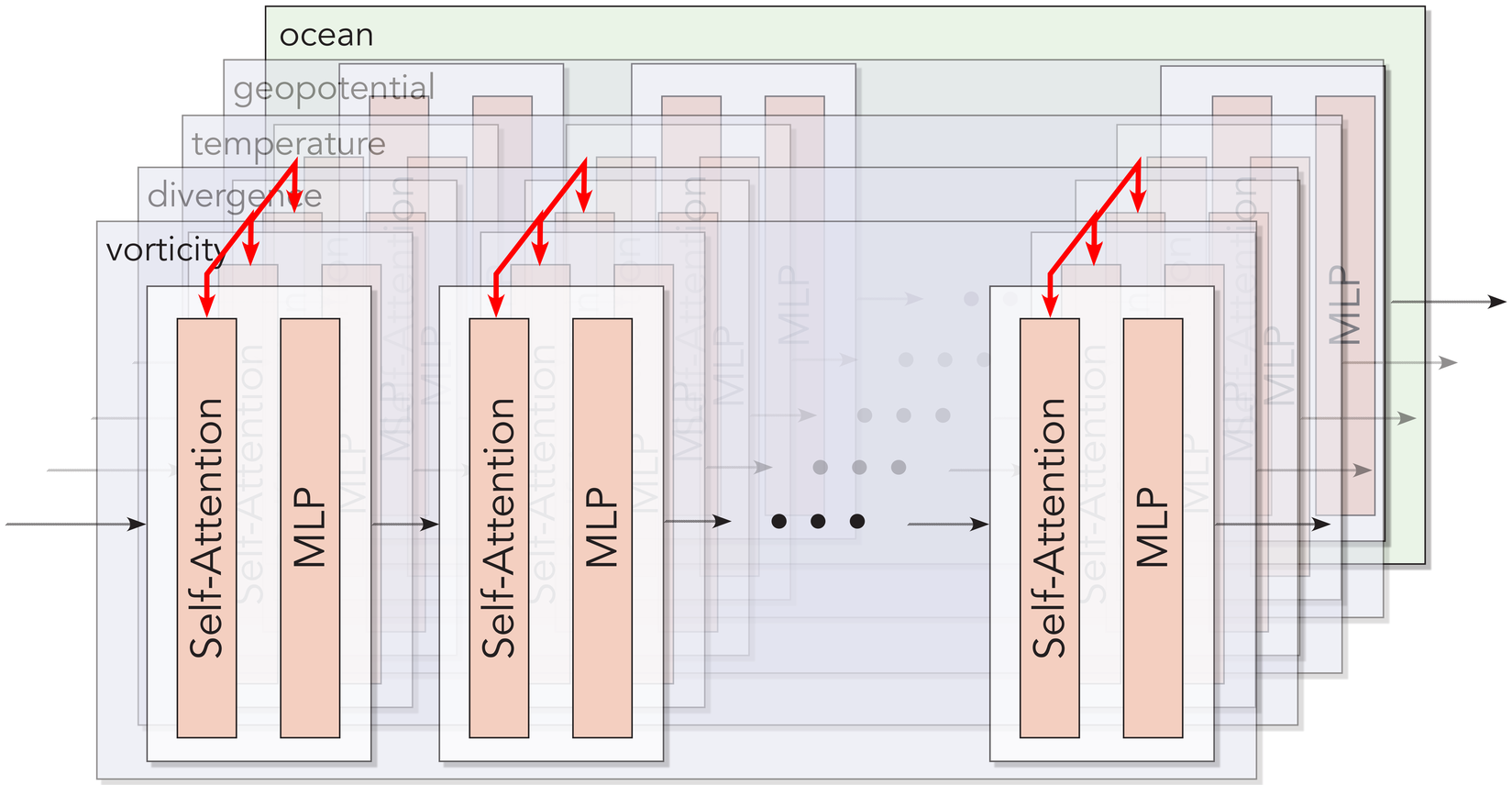
Multiformer

Self attention

$$\sigma\left(Q\,K^T\right)V$$

Cross attention

$$\sigma\left(Q_\zeta\,K_\mu^T\right)V_\mu$$



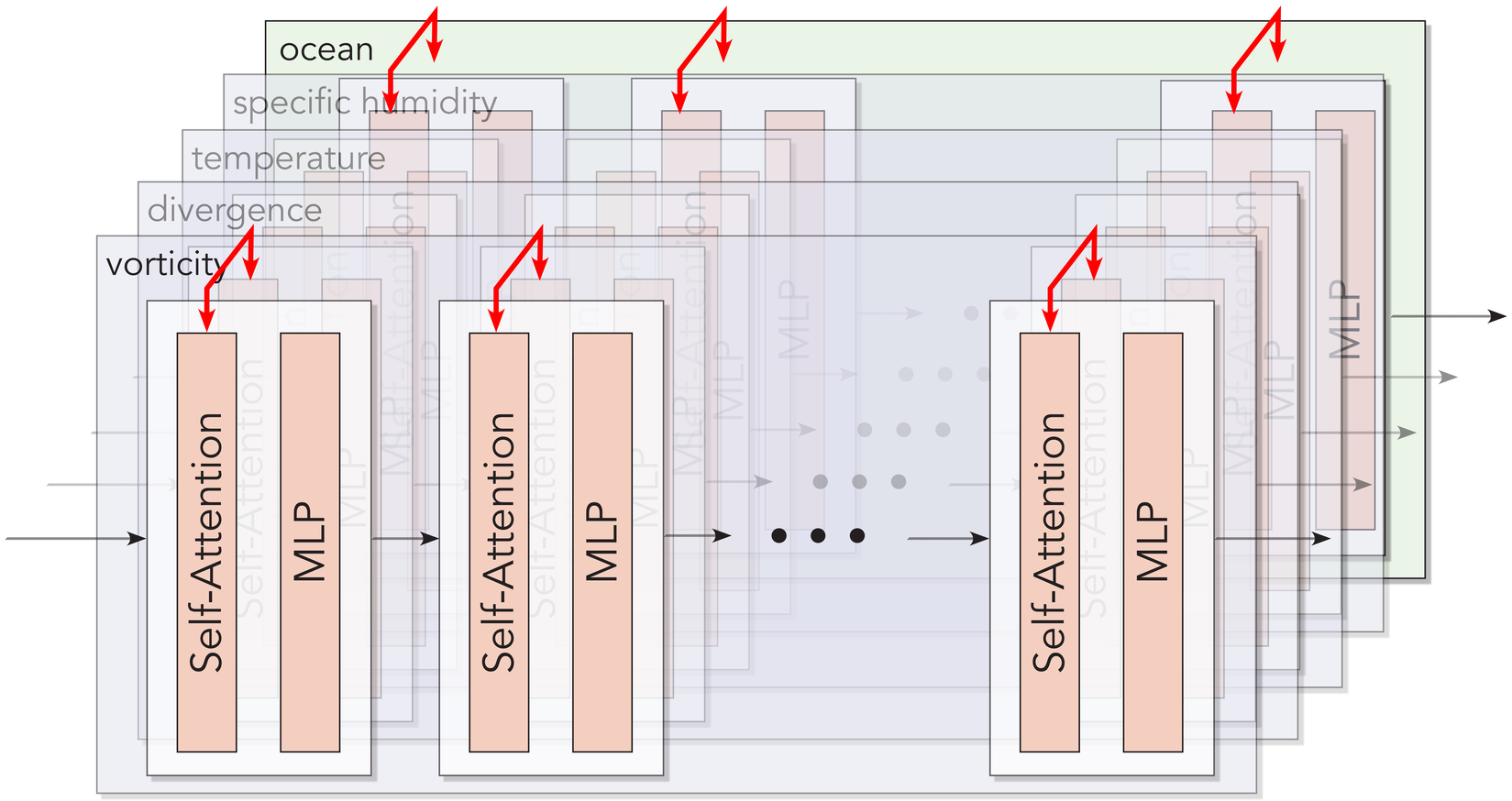
Multiformer

Self attention

$$\sigma\left(Q\,K^T\right)V$$

Cross attention

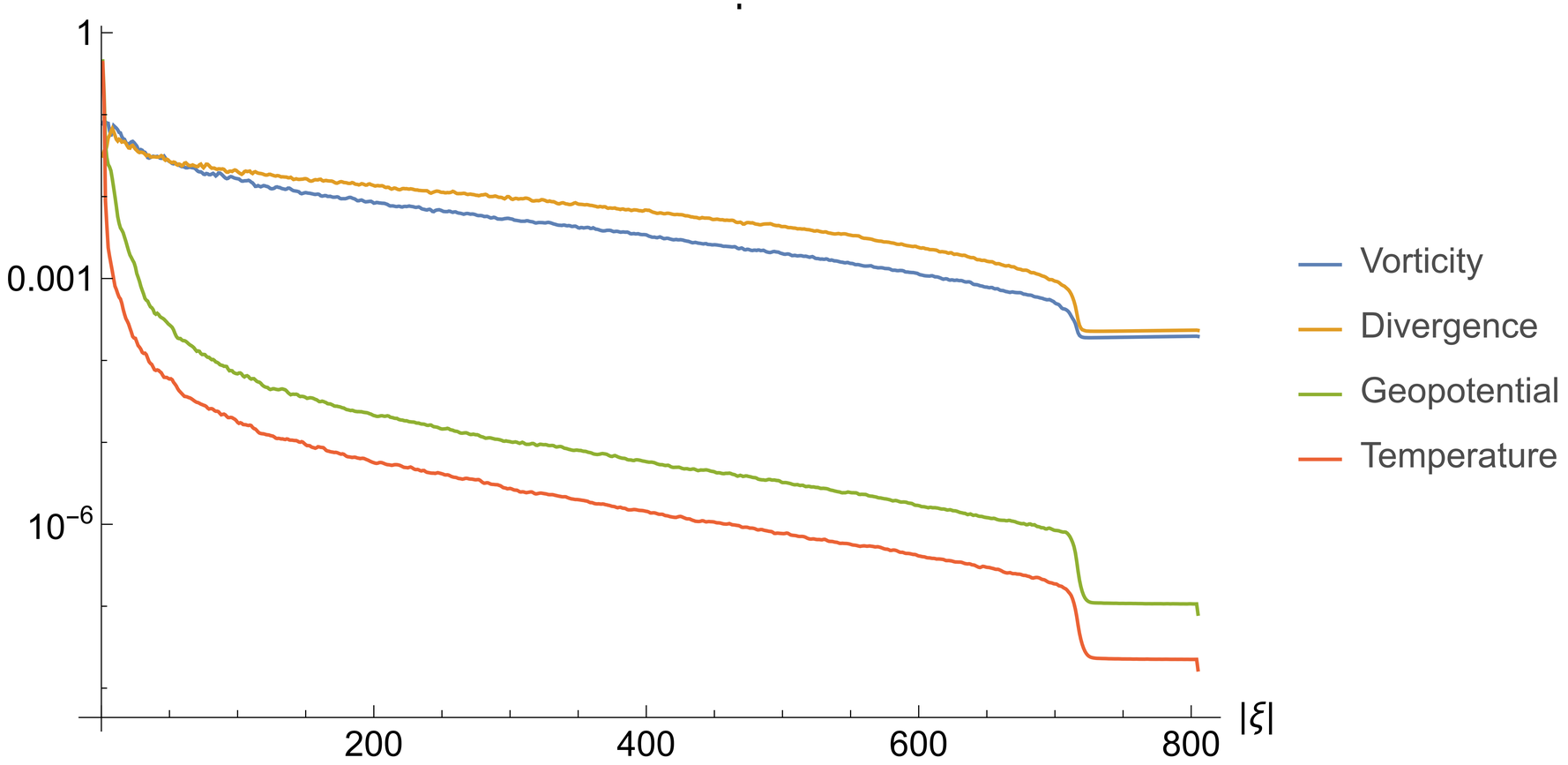
$$\sigma\left(Q_\zeta\,K_\mu^T\right)V_\mu$$



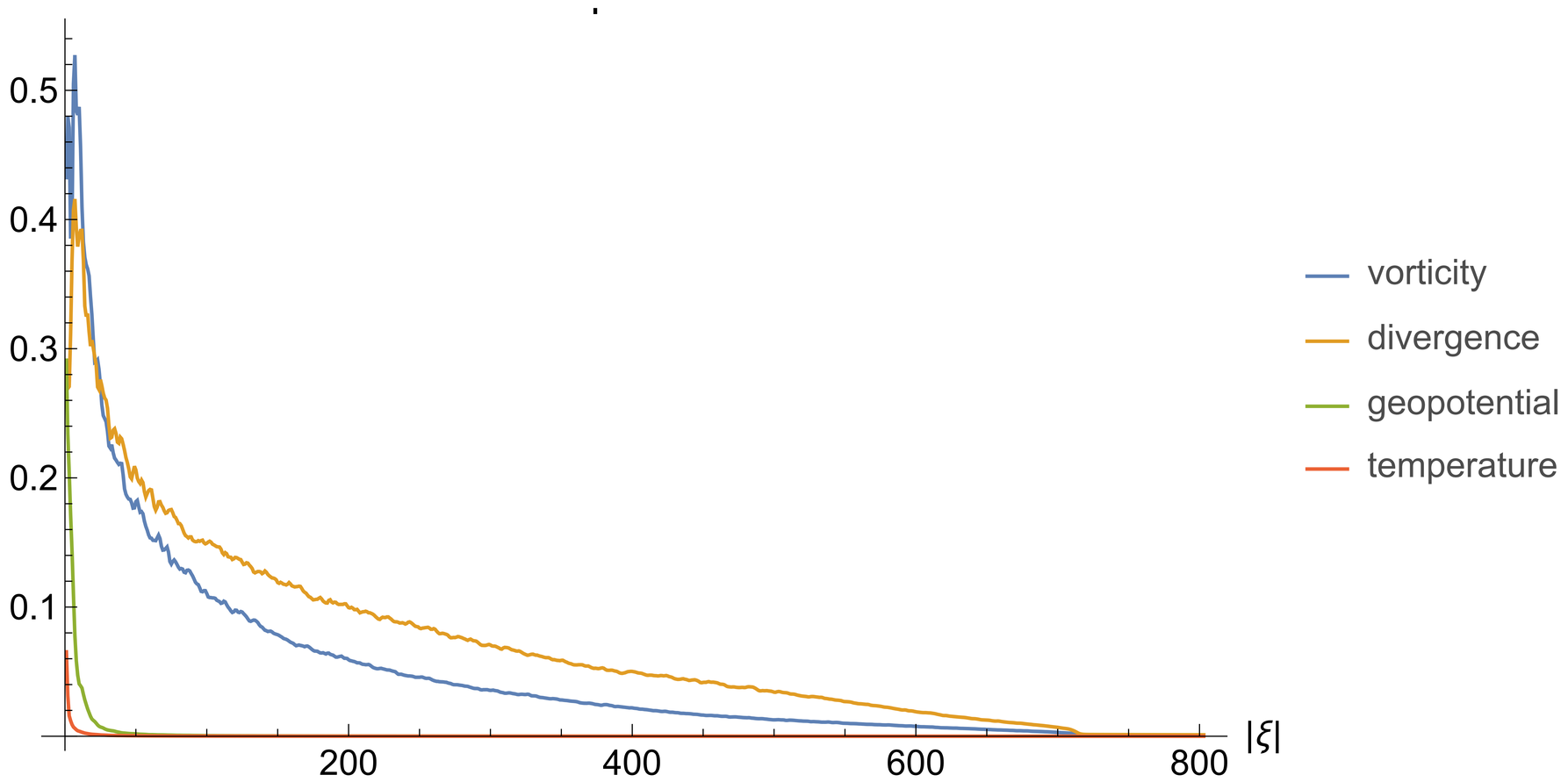
Multiformer

- 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)

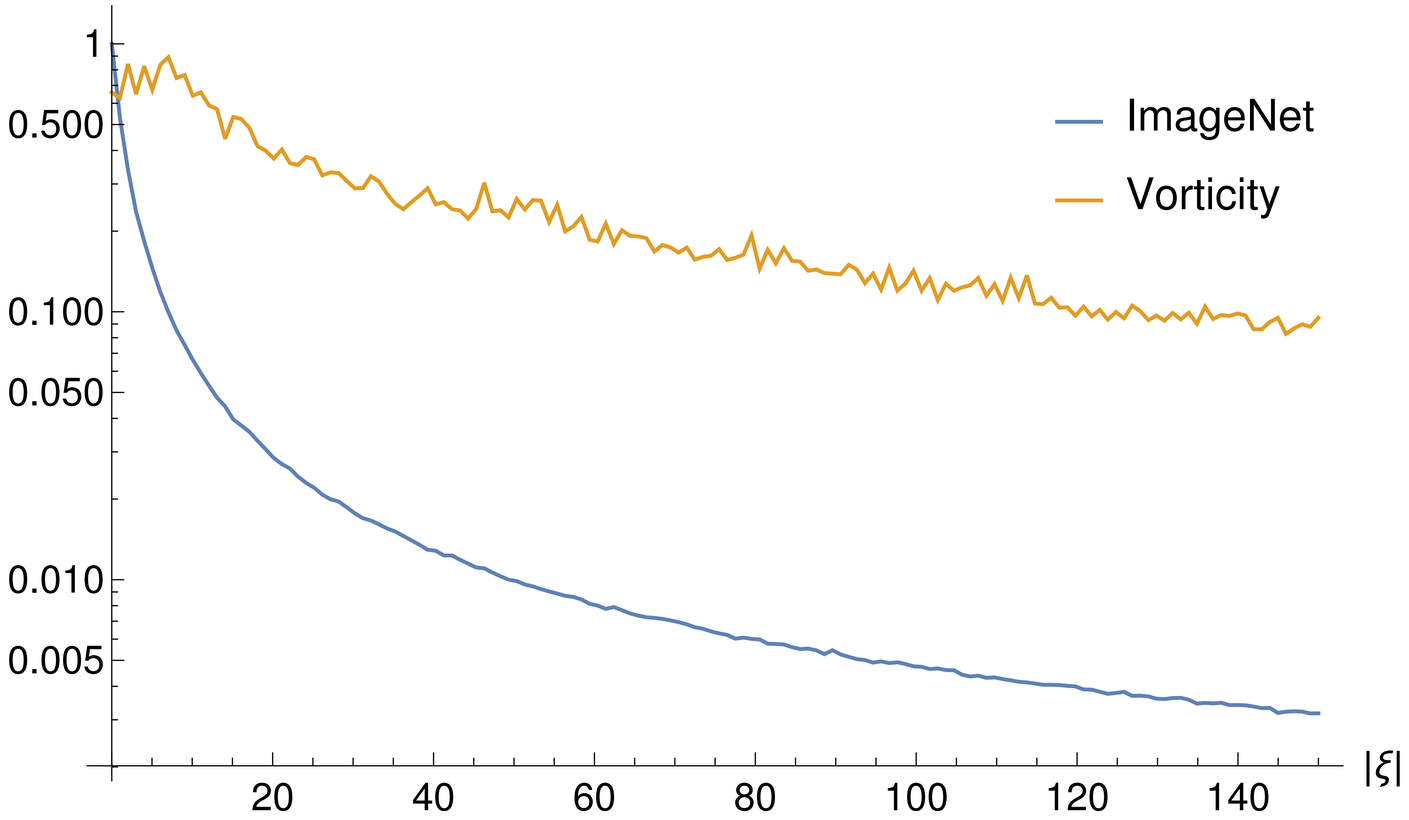
AtmoRep data



AtmoRep data



ERA5 versus ImageNet



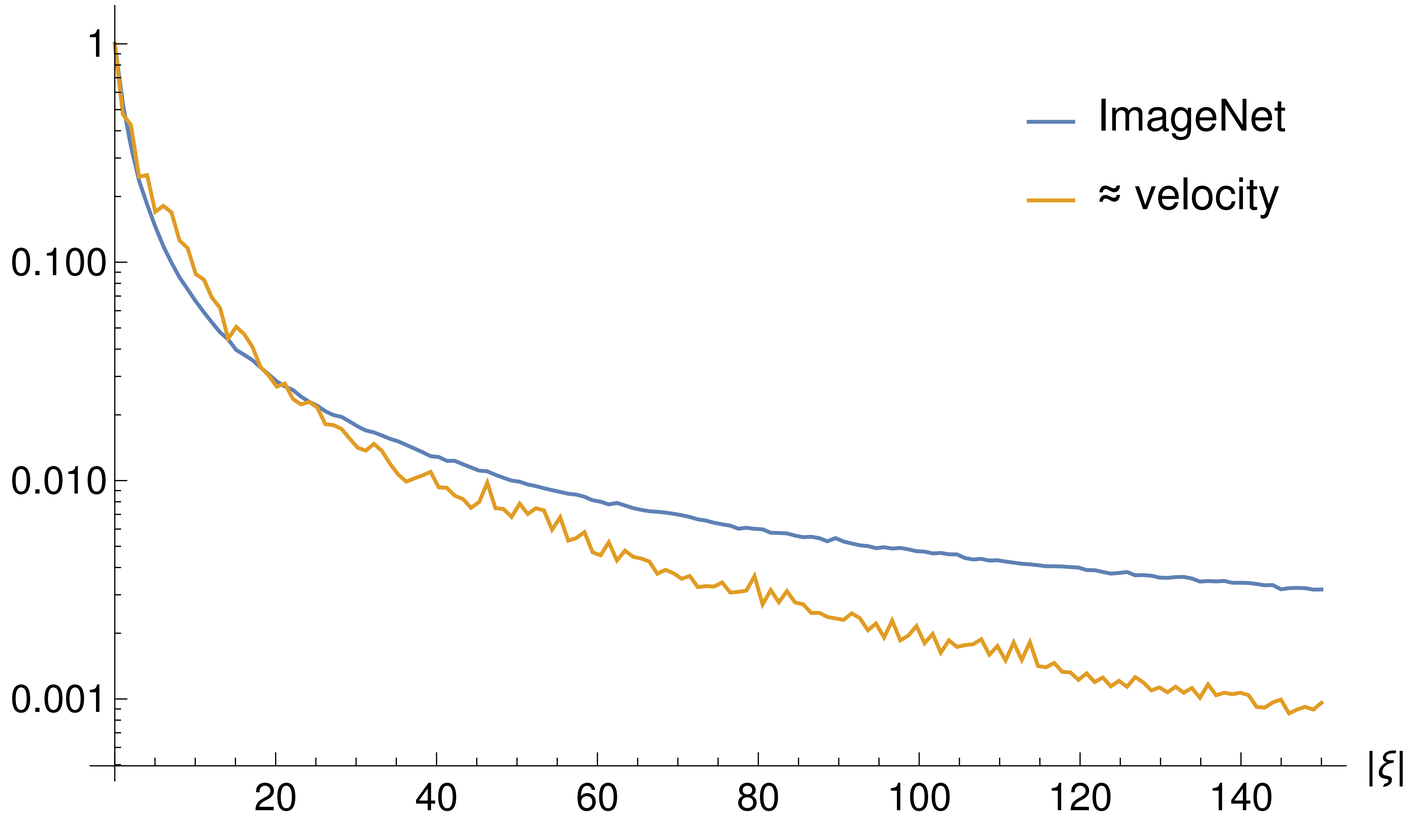
ERA5 versus ImageNet

stream function
velocity potential

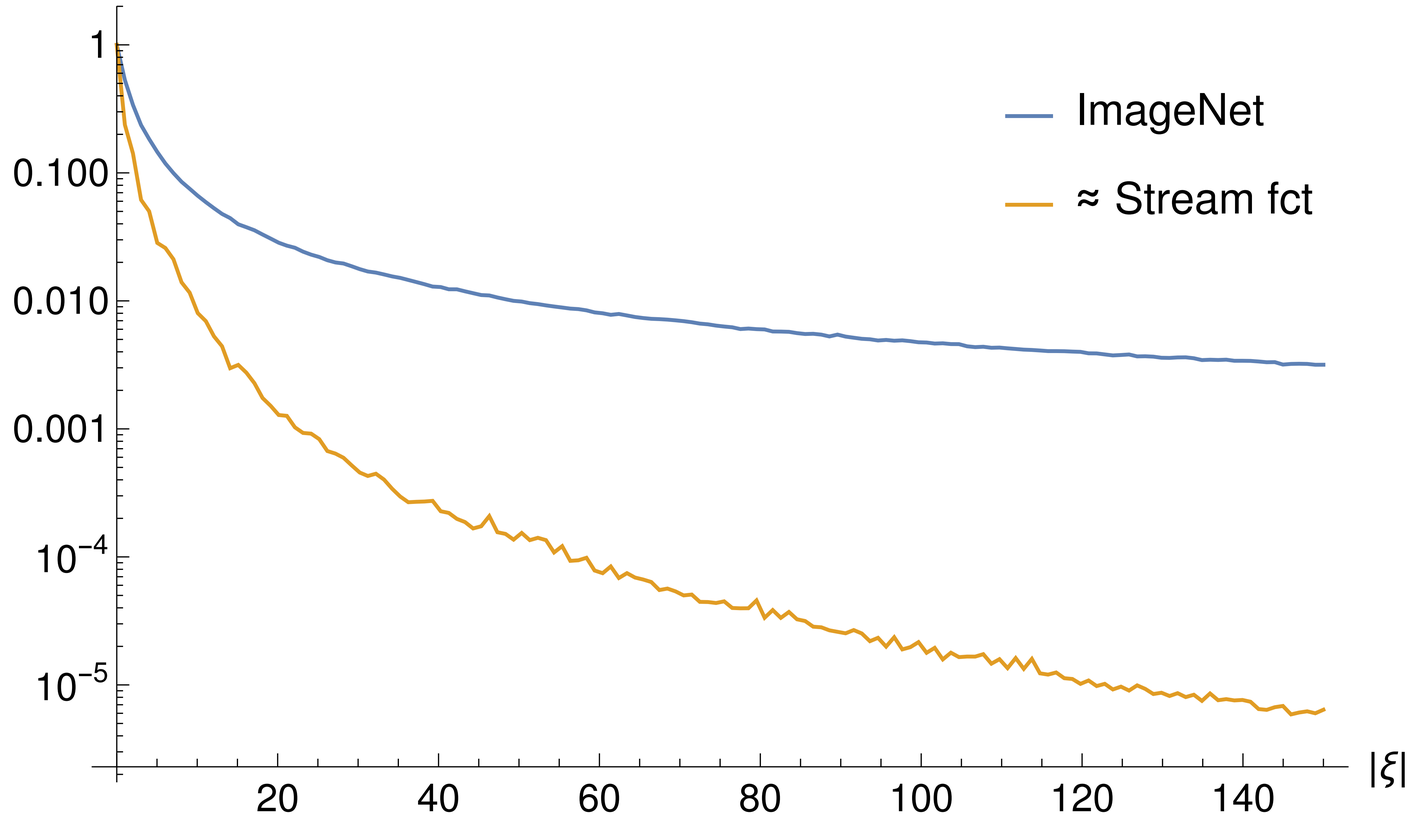
velocity
vector field

vorticity
divergence

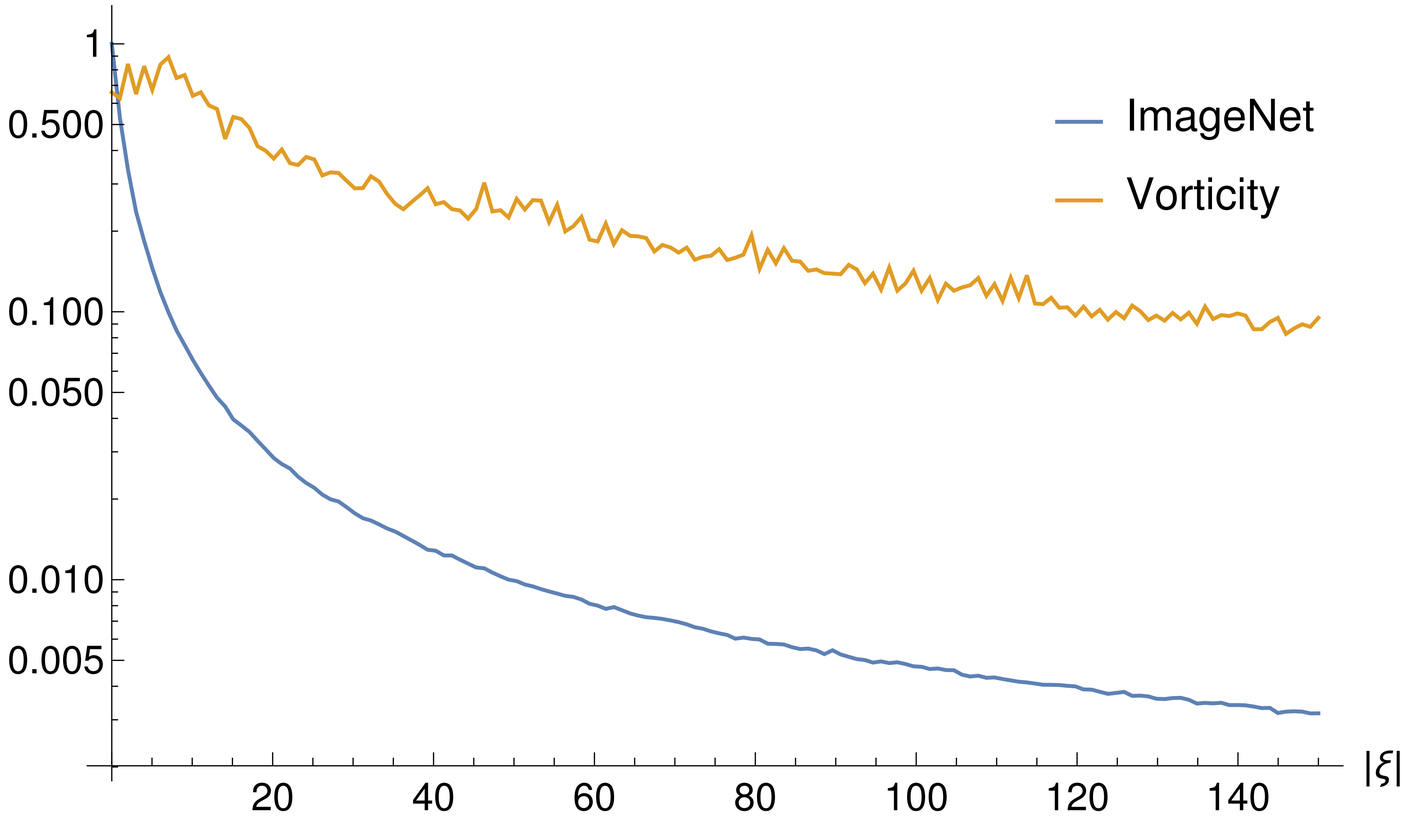
ERA5 versus ImageNet



ERA5 versus ImageNet

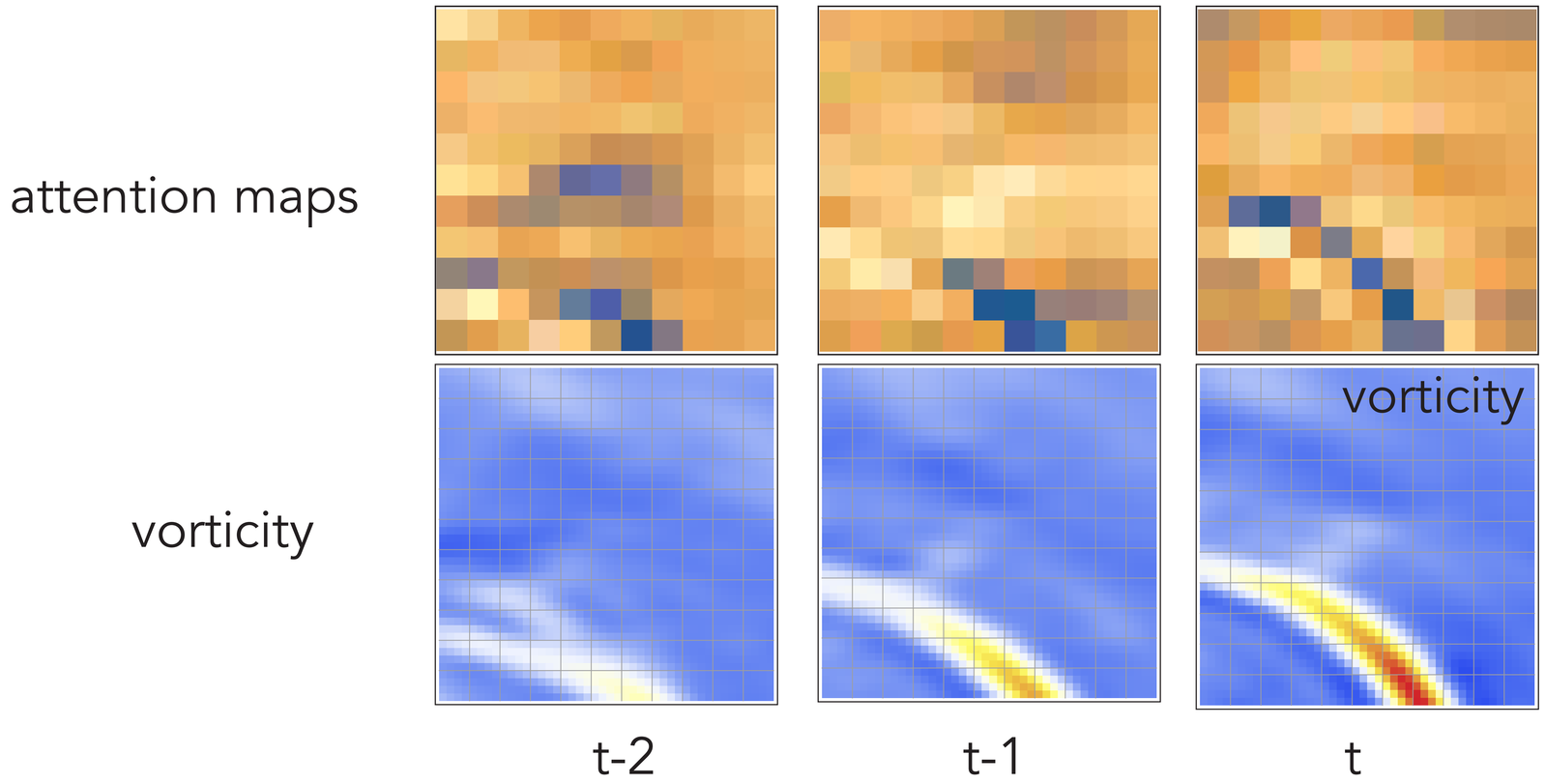


AtmoRep data



Statistical loss

- Attention maps:



Statistical loss

- Statistical loss:

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

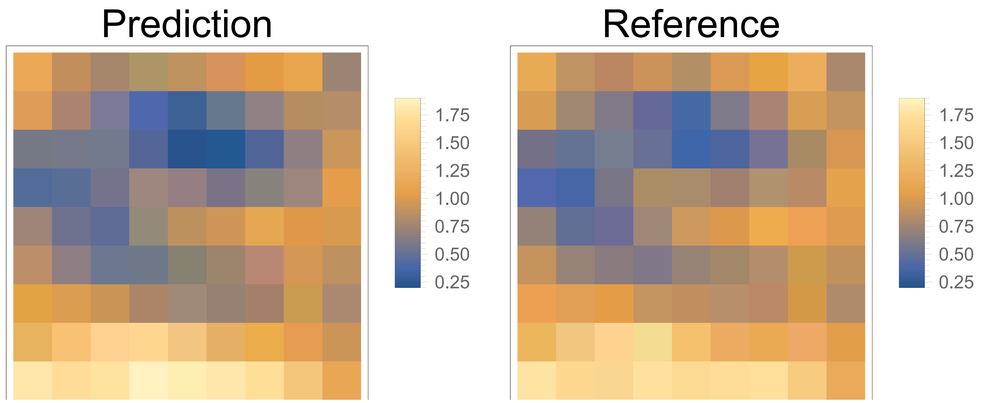
- CRPS:[1]

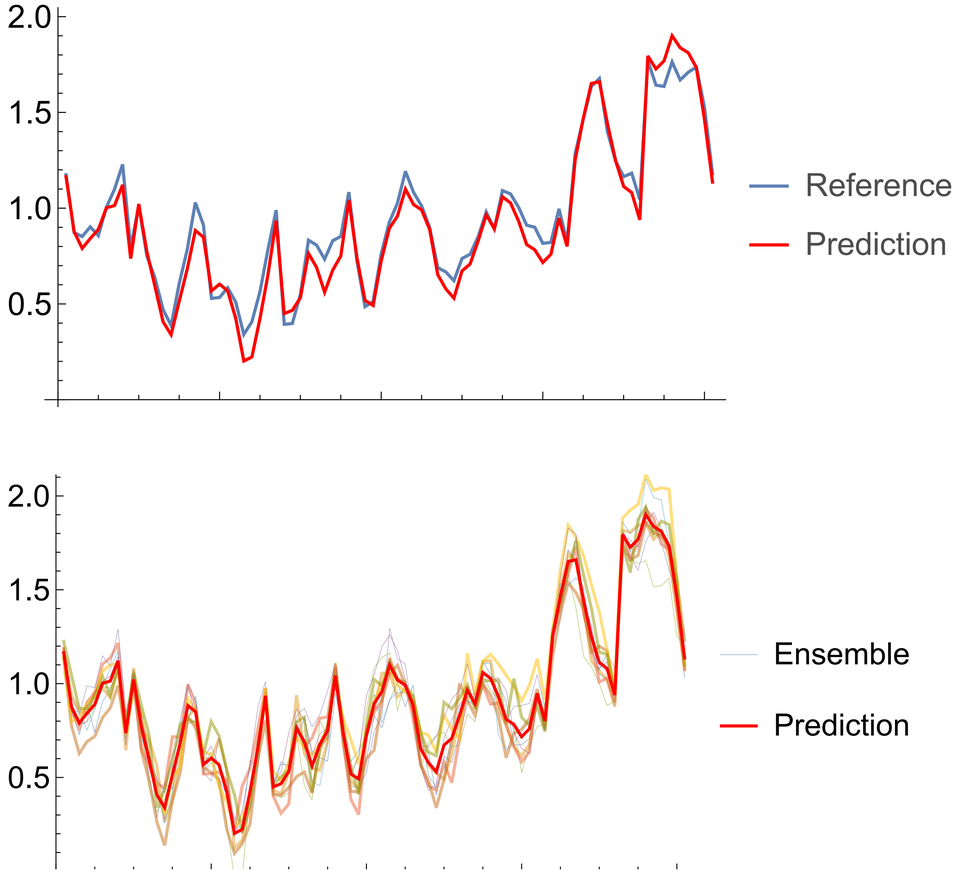
$$\mathcal{L}_{\text{CRPS}} = \int_{\mathrm{R}} \left| H_y(x)\, \mathrm{erf}_{\tilde{\mu},\tilde{\sigma}}(x) \right|^2 \mathrm{d}x$$

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

Statistical loss

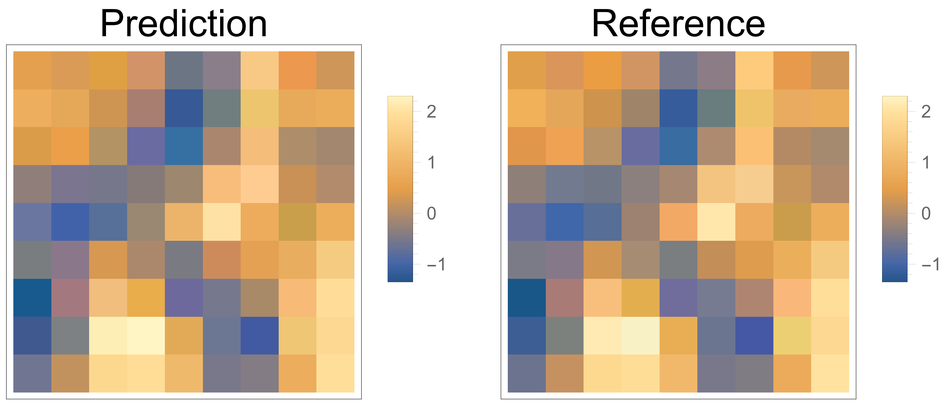
- Predictions:

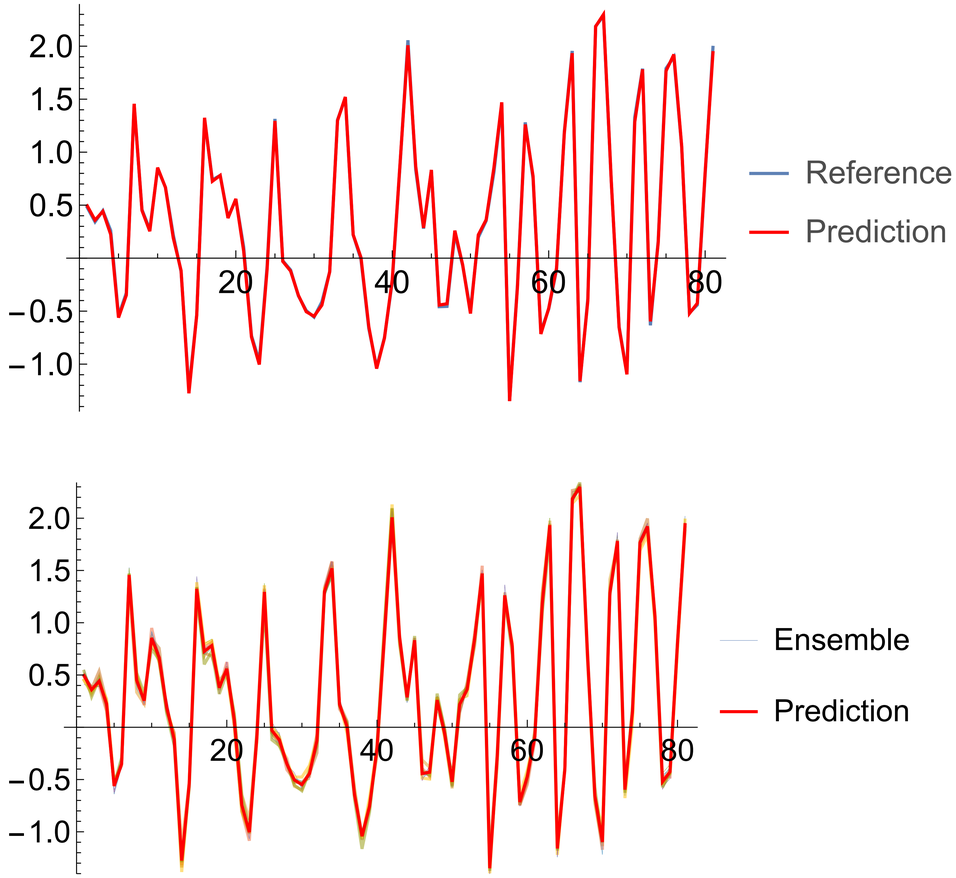




Statistical loss

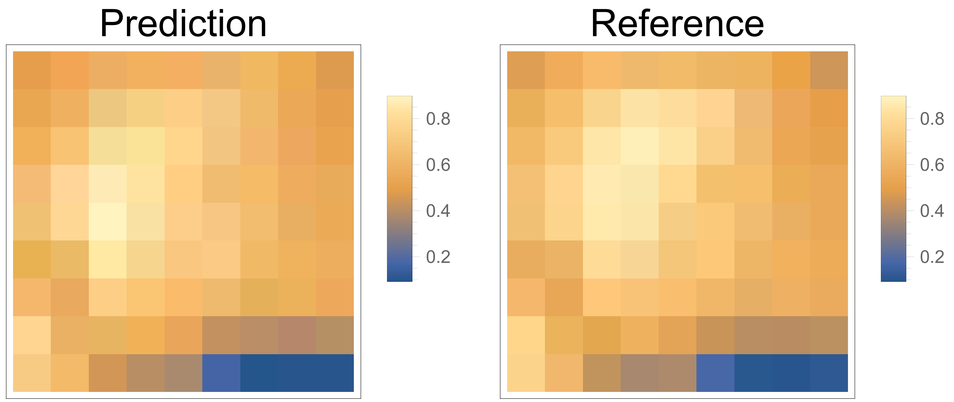
- Predictions:

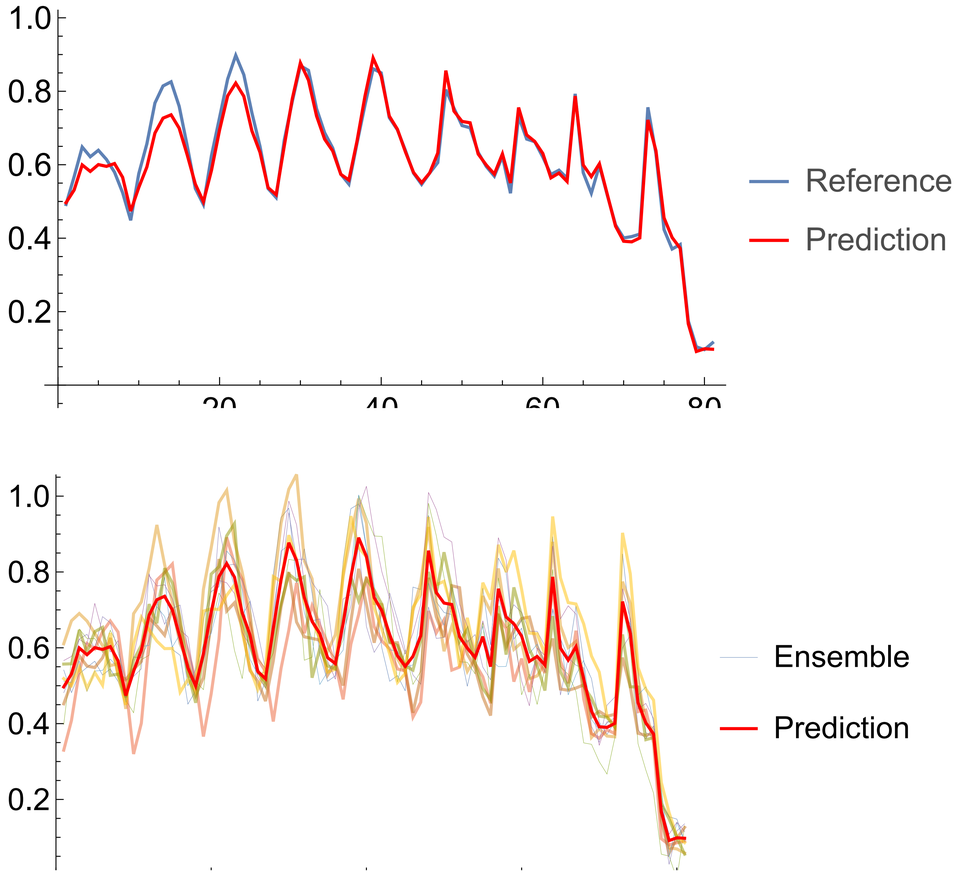




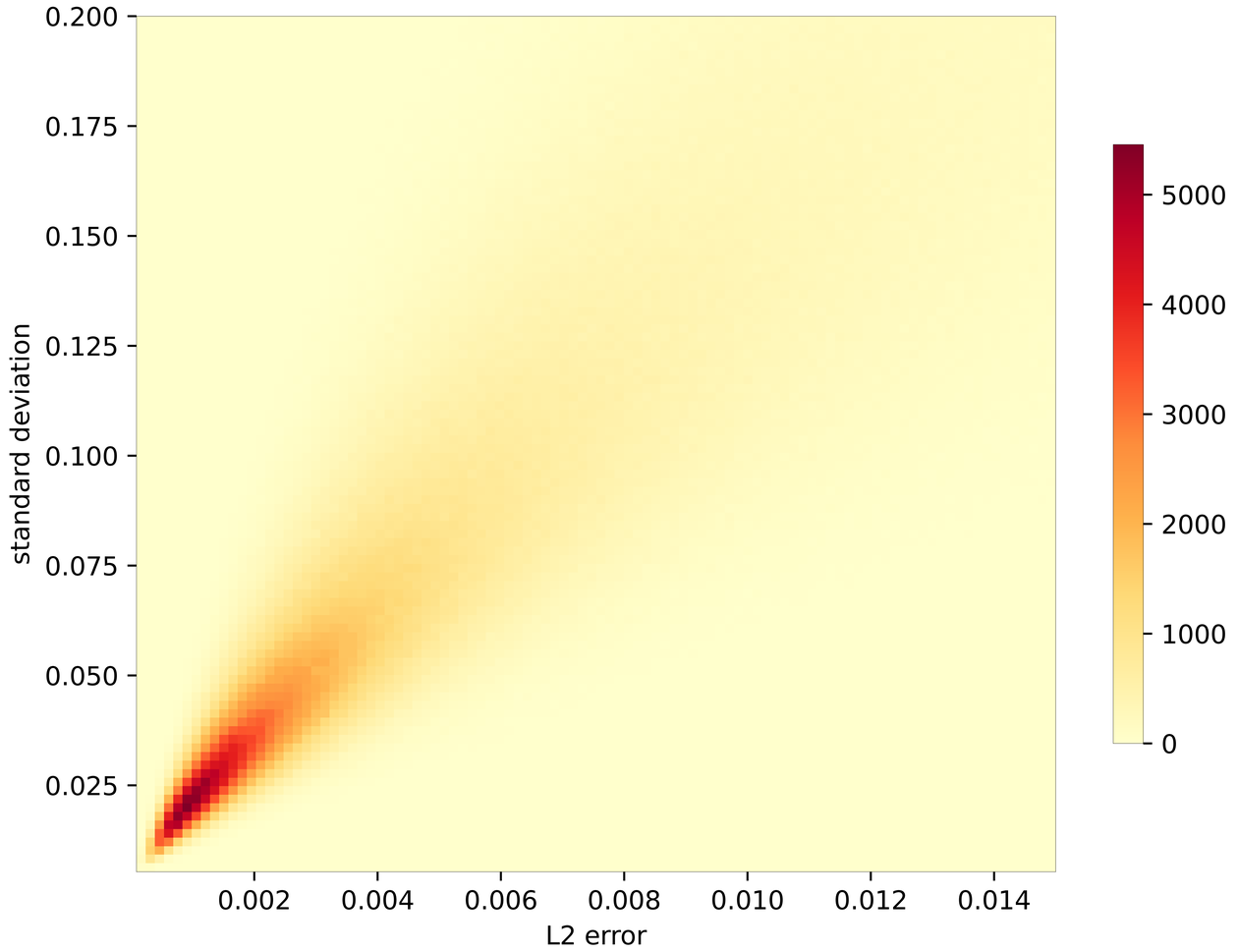
Statistical loss

- Predictions:



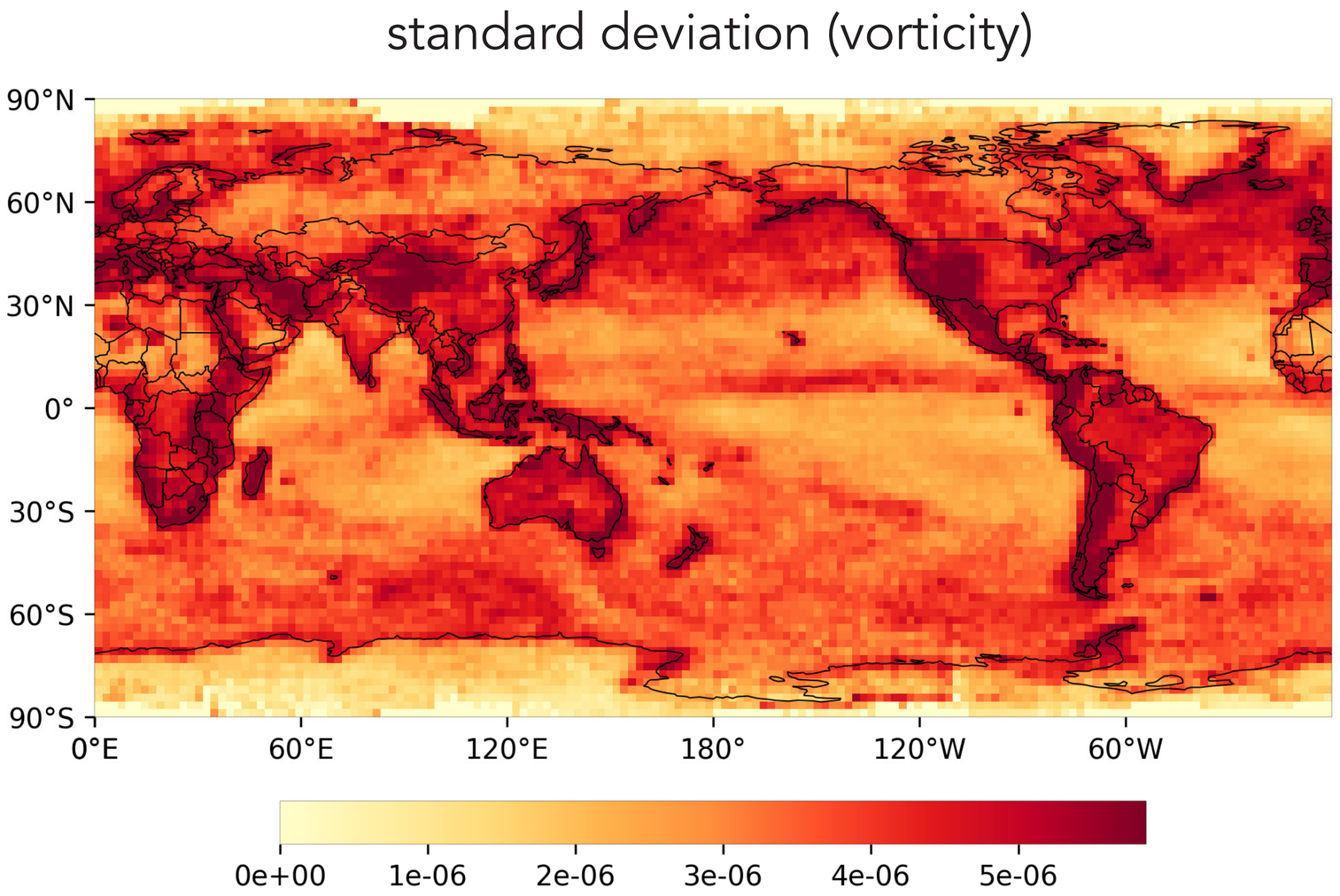


Statistical loss



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

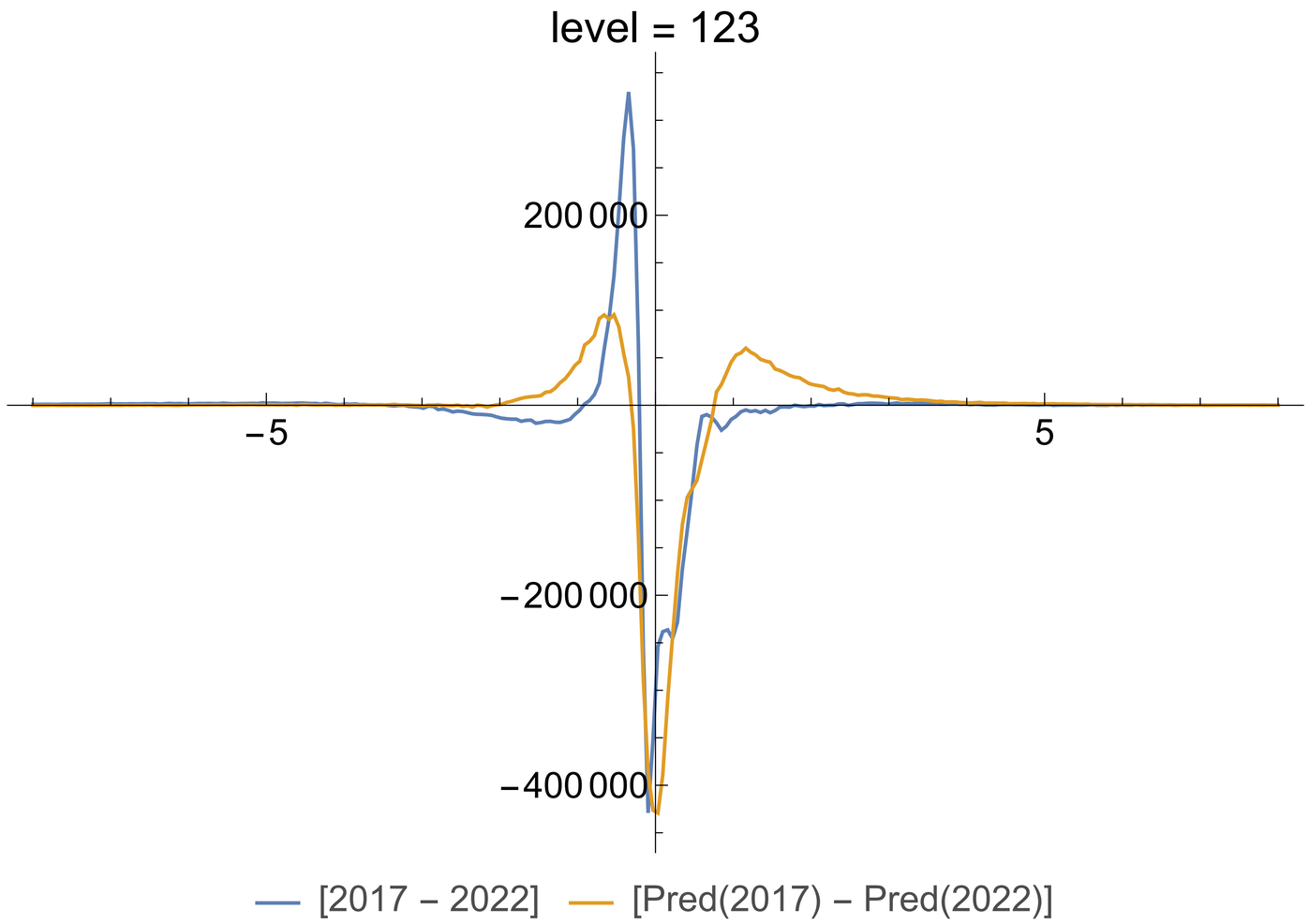
Statistical loss



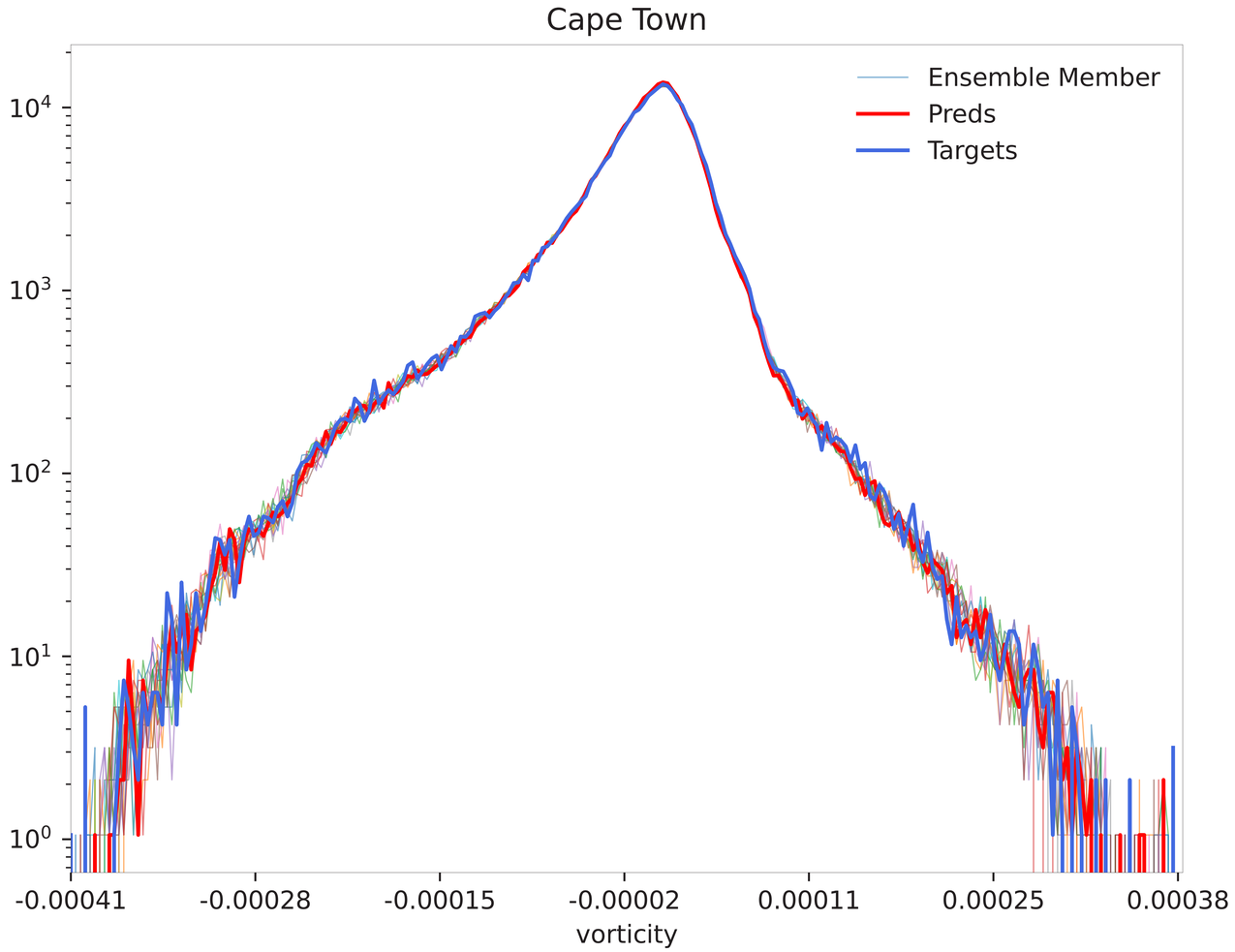
Some observations on training

- 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

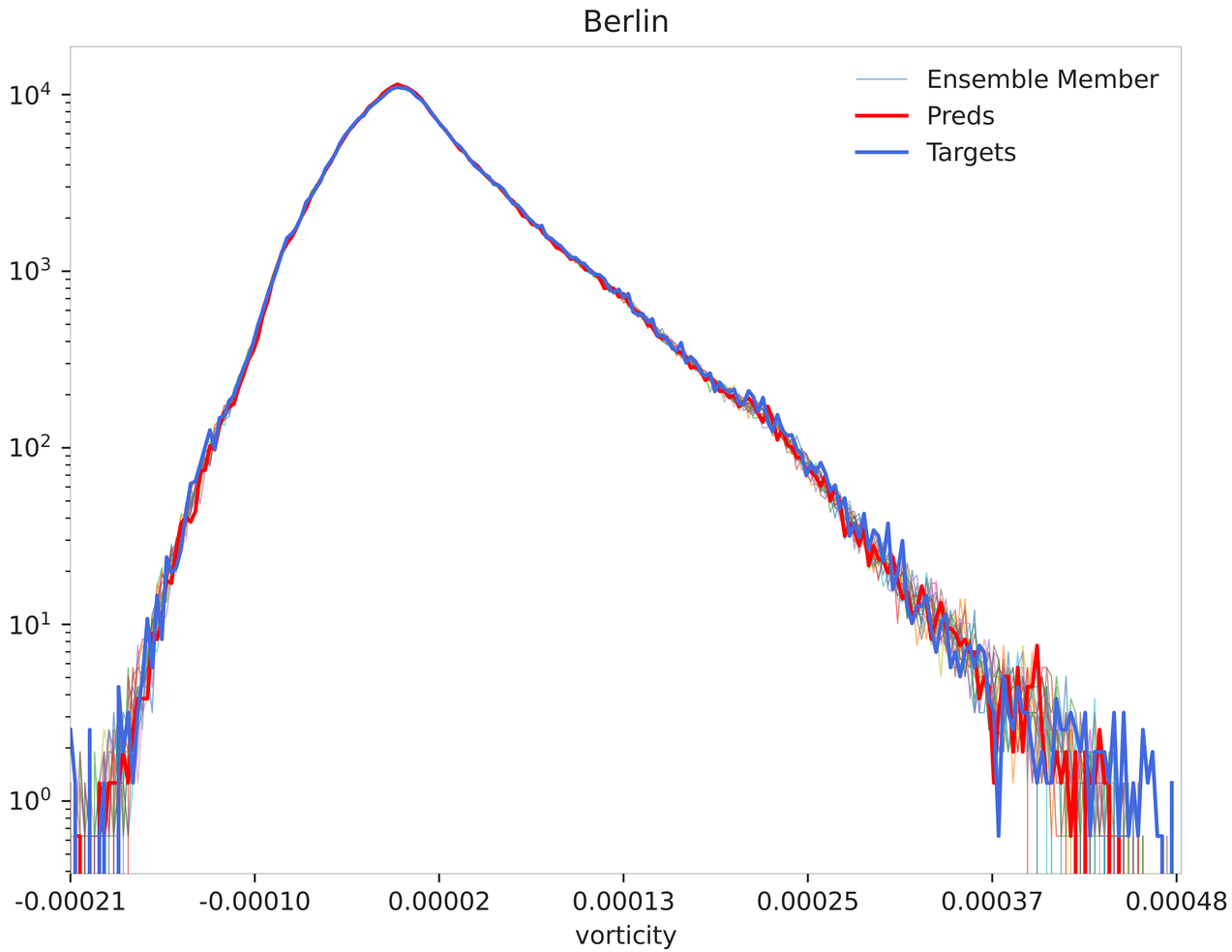
Counterfactuals: extrapolation



Pre-training results



Pre-training results



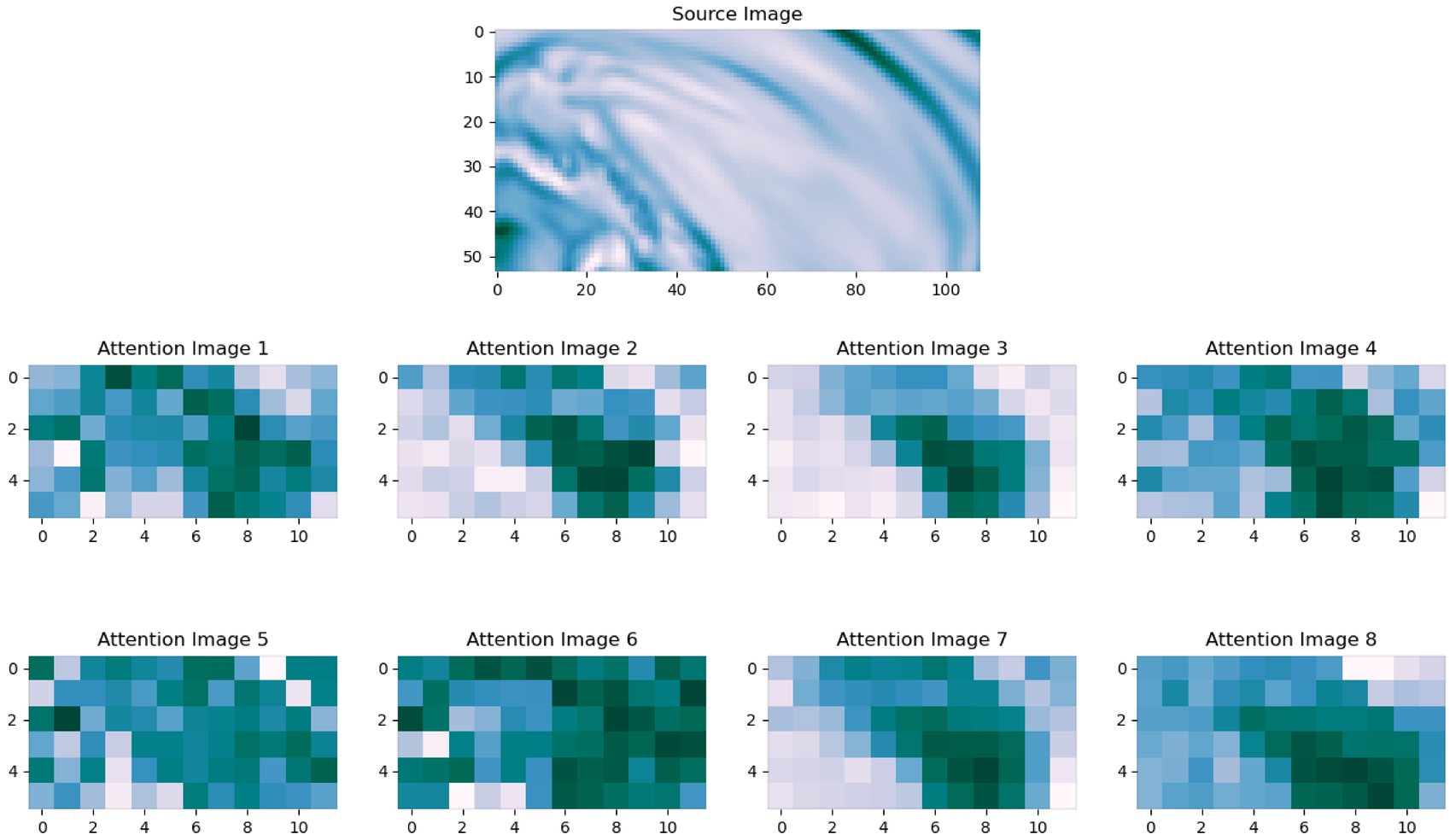
Training on observations

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

Can one continuous update a model with observations?

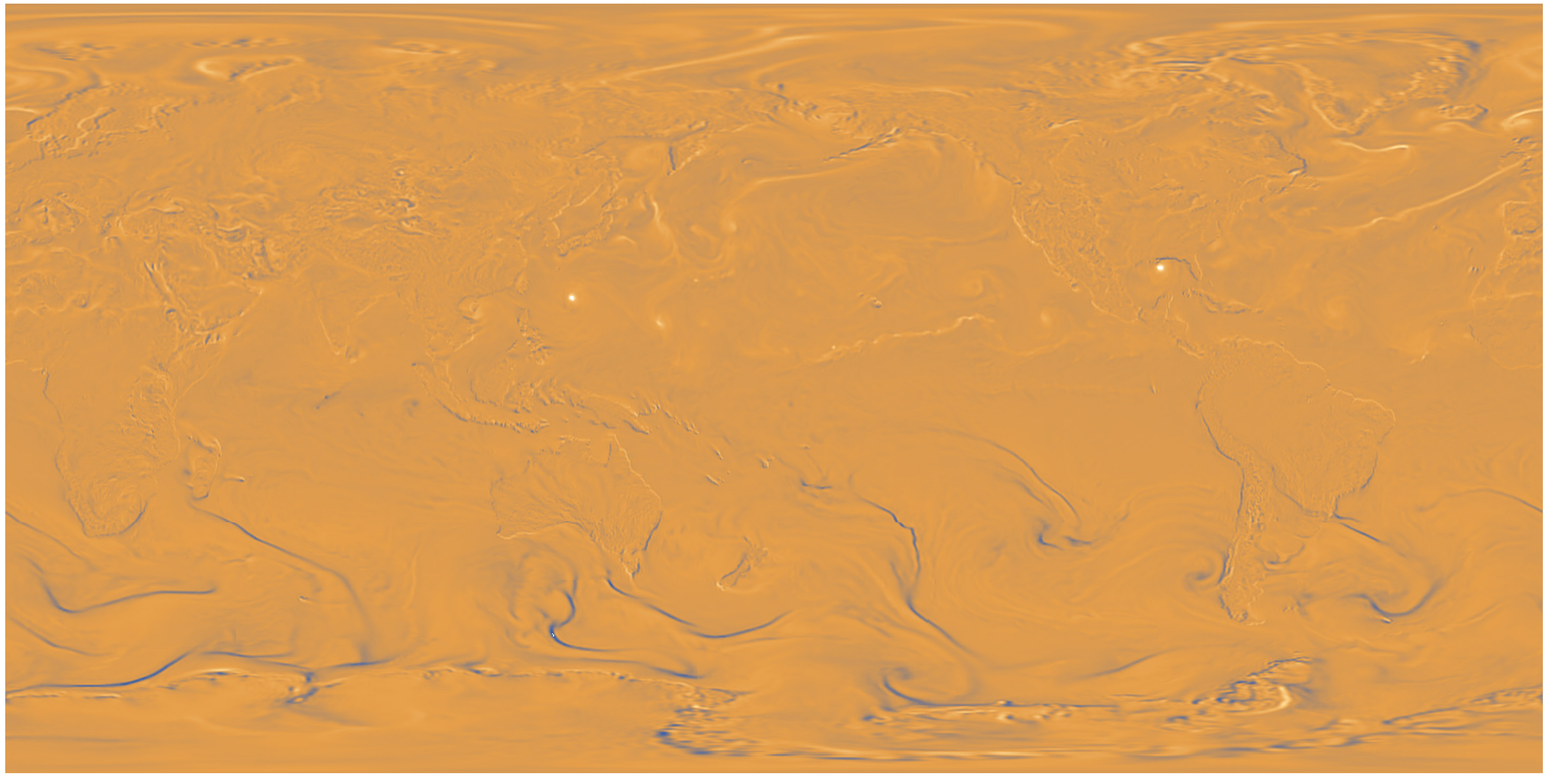
How to handle and propagate uncertainties?

Physics of trained network



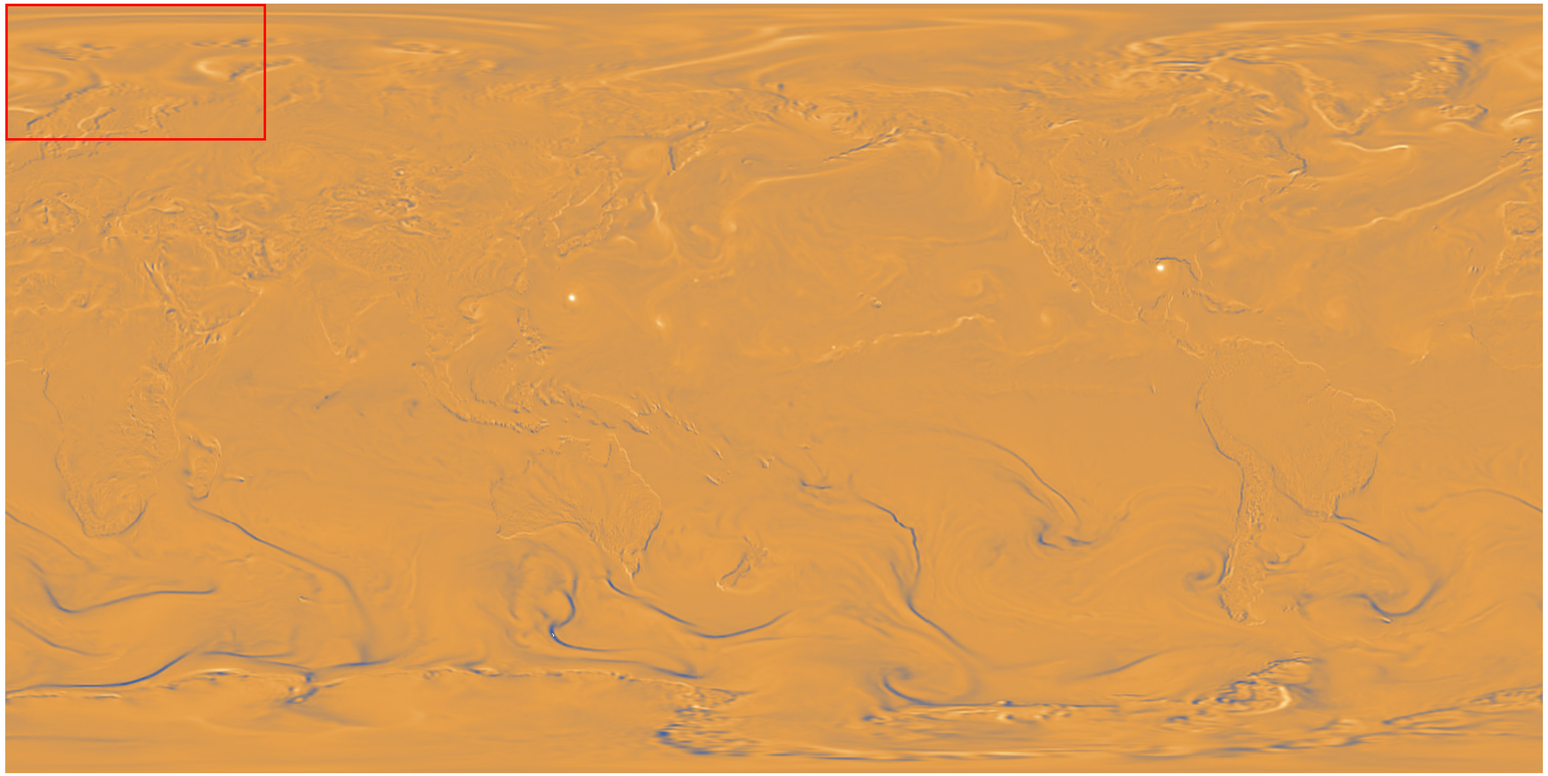
Medium range forecasting

- How to do global forecasts with a local model?



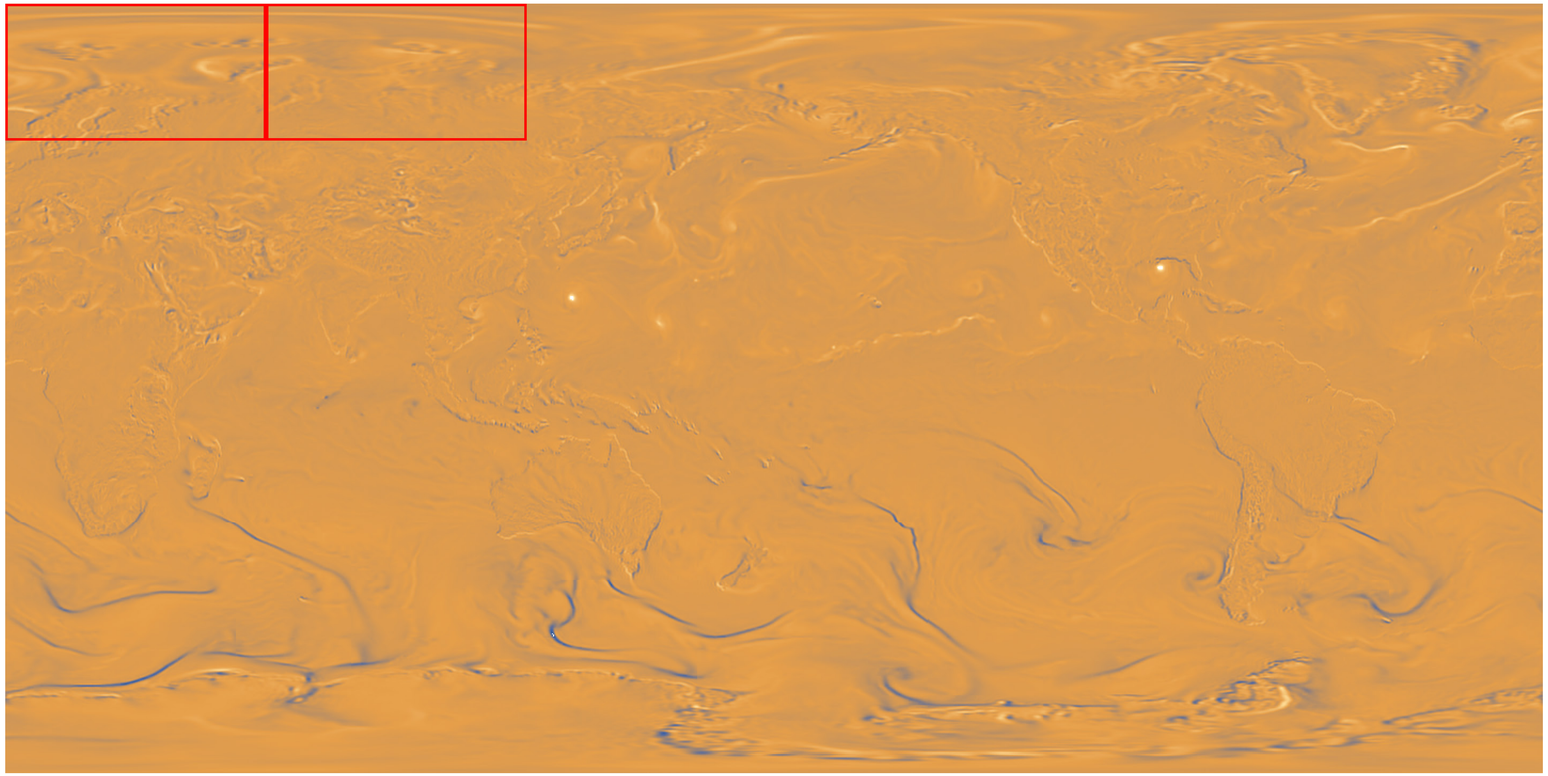
Medium range forecasting

- How to do global forecasts with a local model?



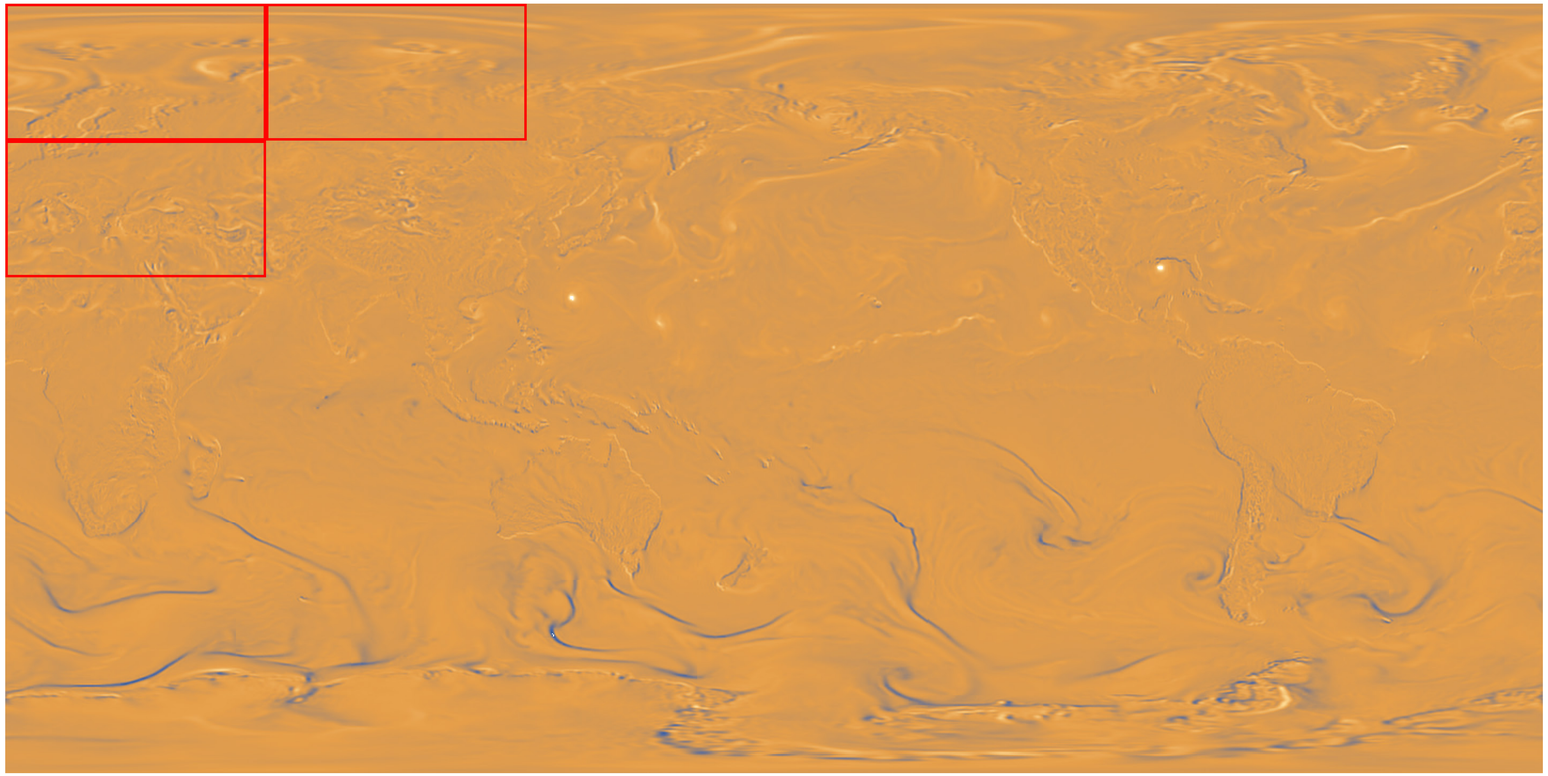
Medium range forecasting

- How to do global forecasts with a local model?



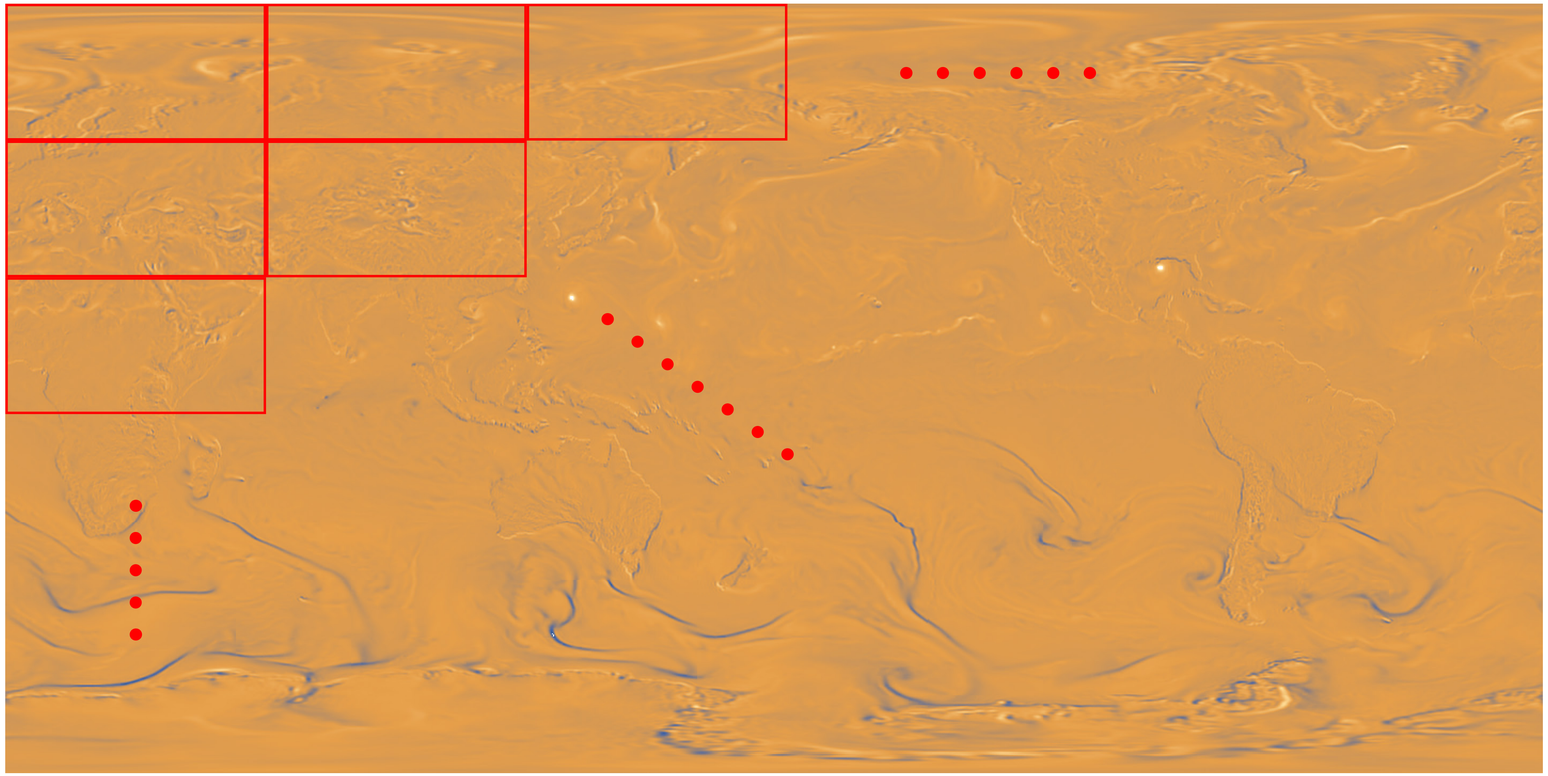
Medium range forecasting

- How to do global forecasts with a local model?



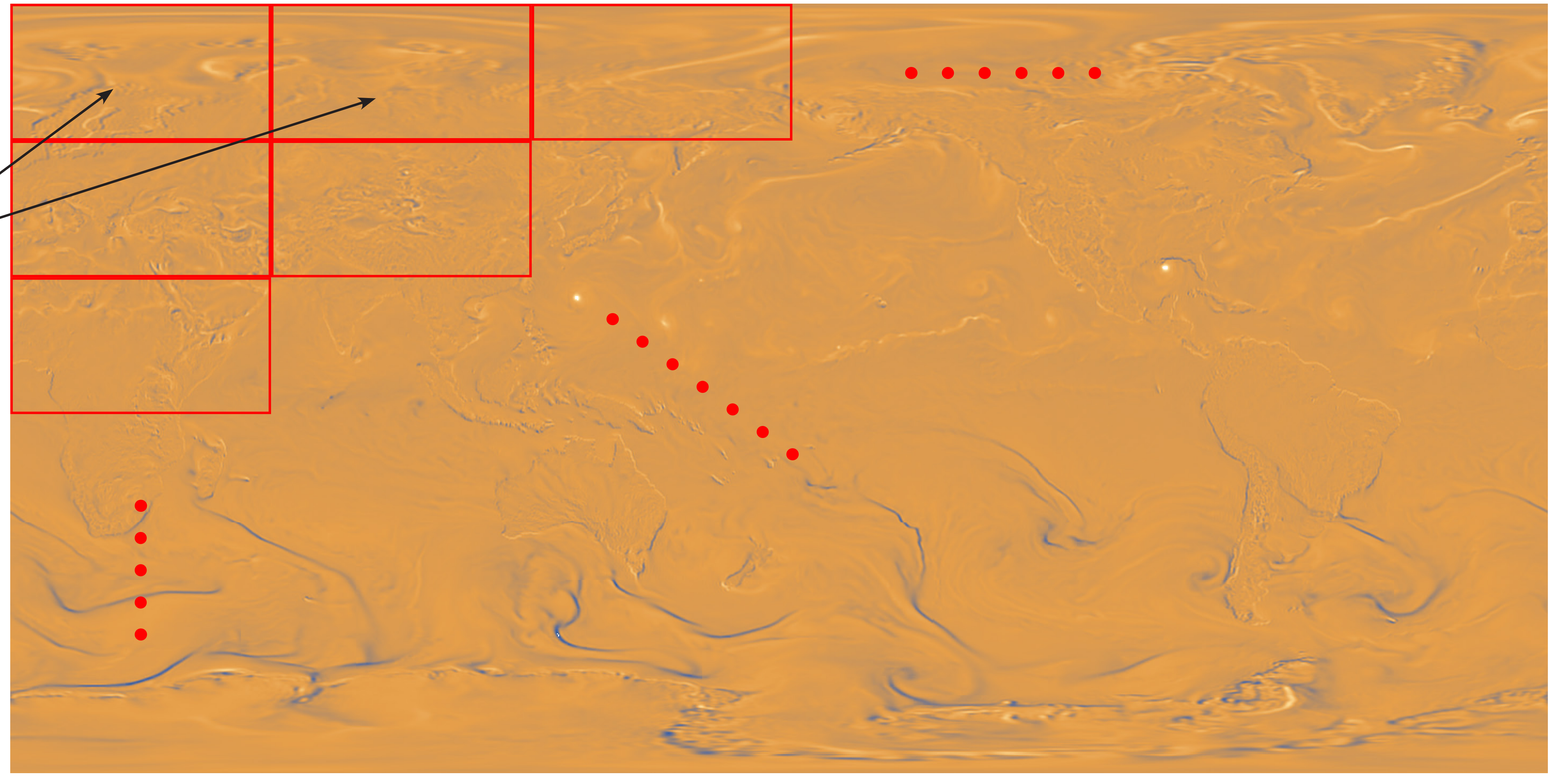
Medium range forecasting

- How to do global forecasts with a local model?



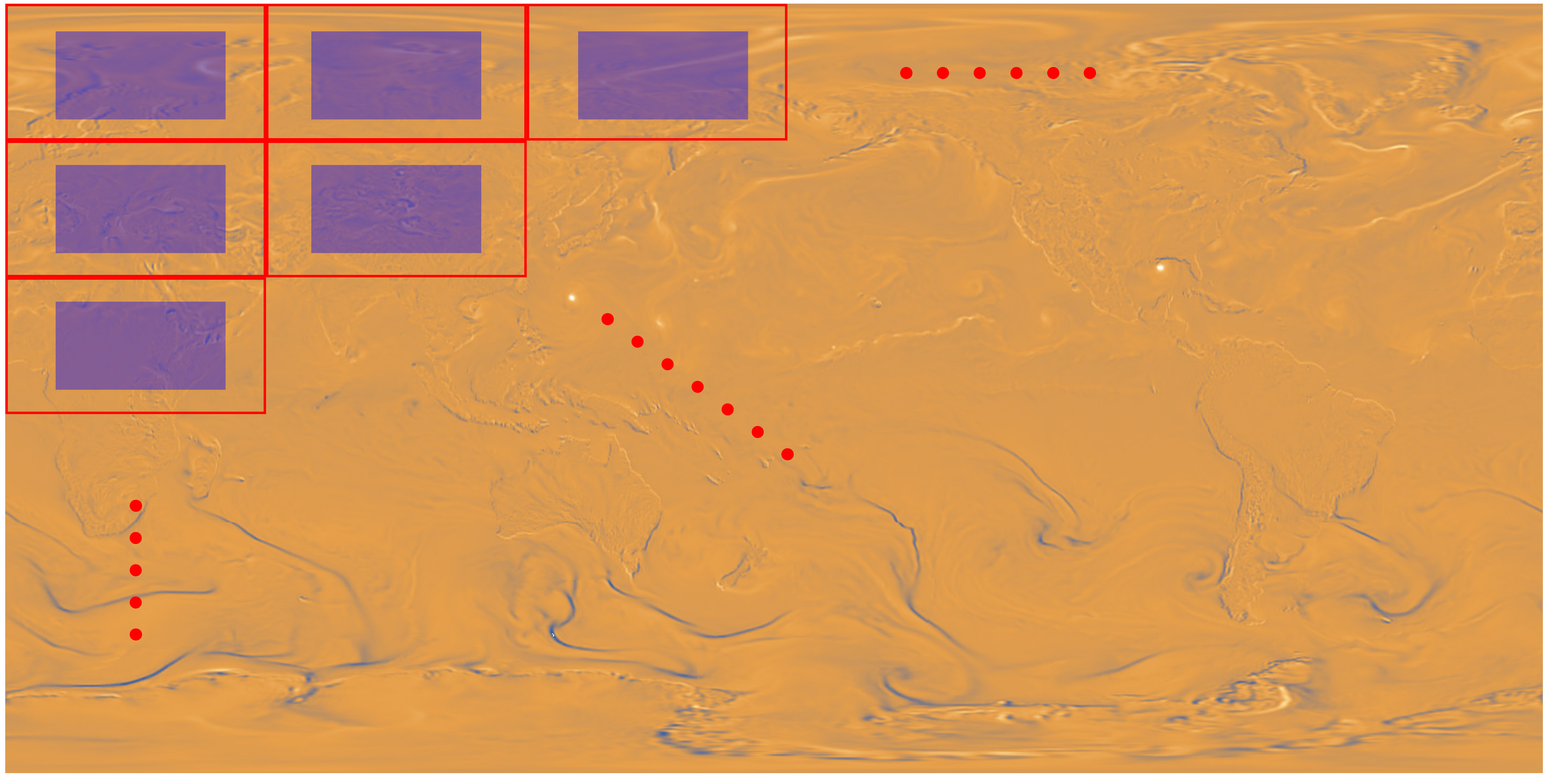
Medium range forecasting

- How to do global forecasts with a local model?



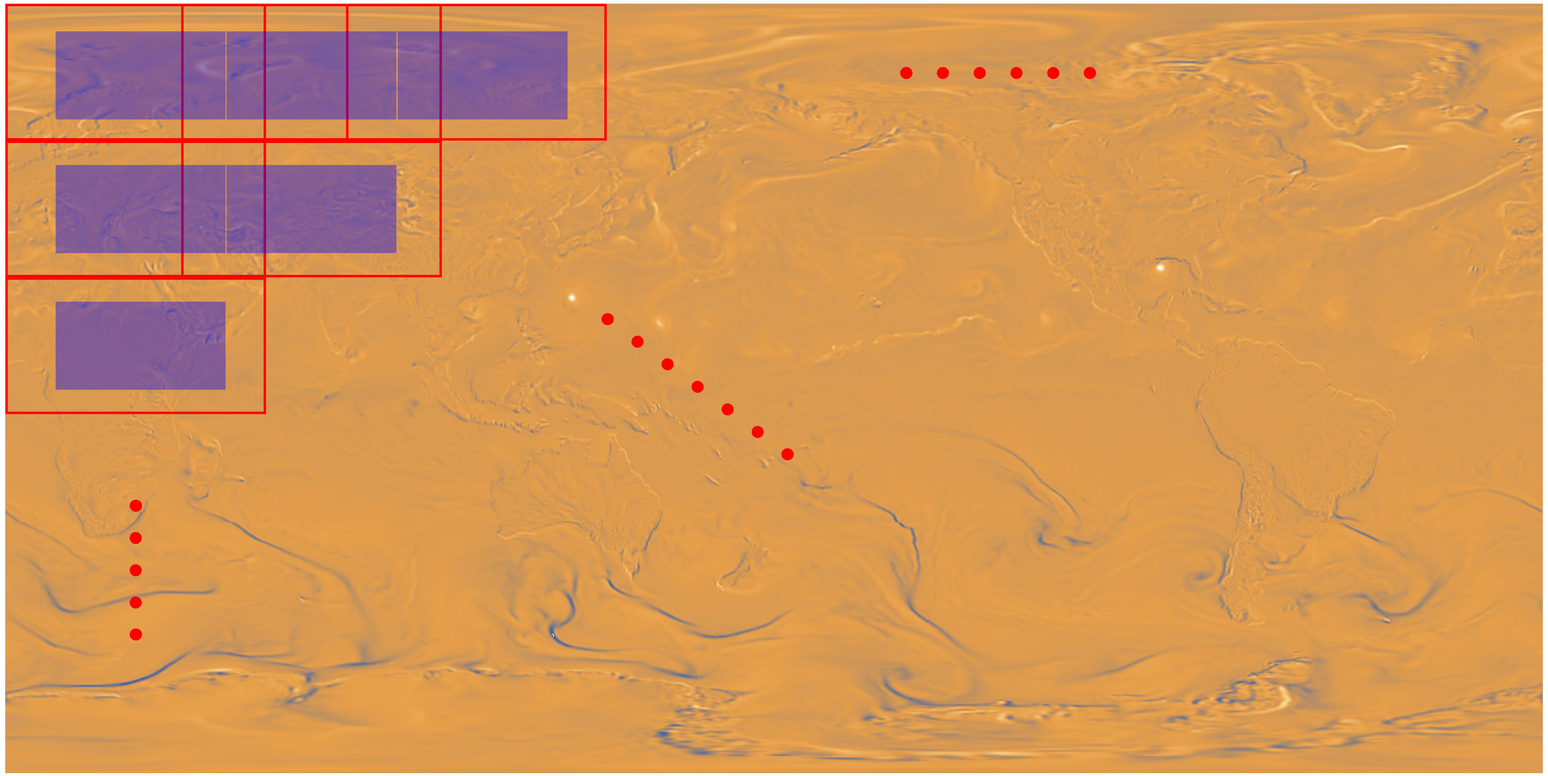
Medium range forecasting

- How to do global forecasts with a local model?



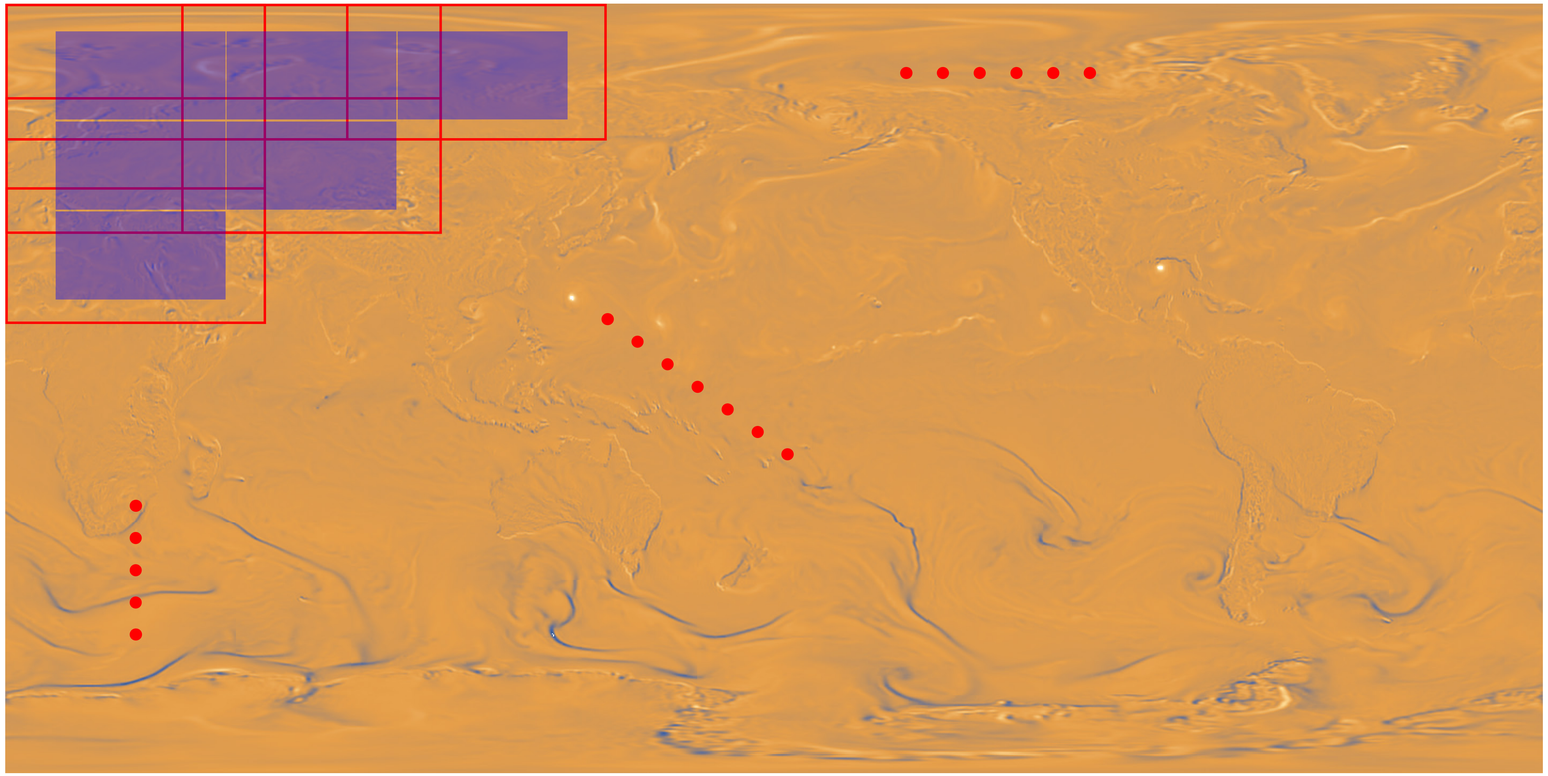
Medium range forecasting

- How to do global forecasts with a local model?



Medium range forecasting

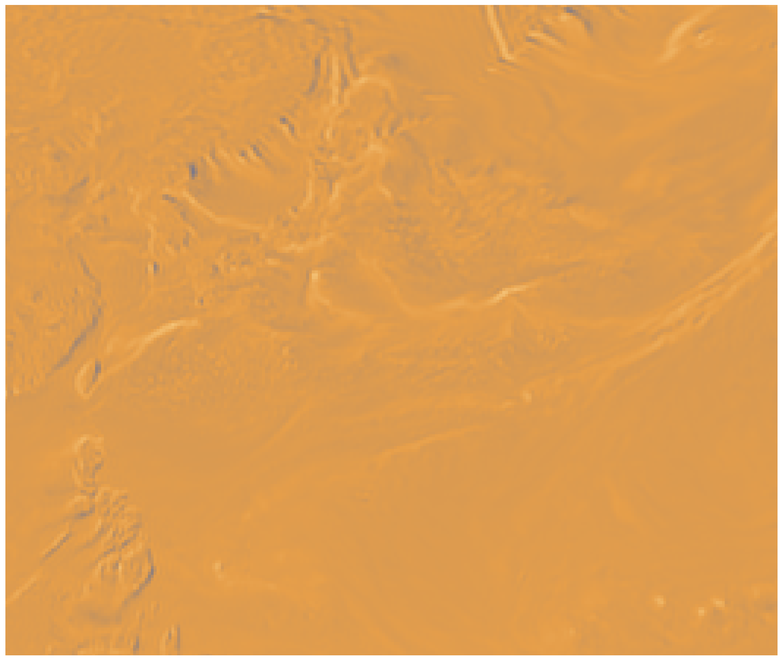
- How to do global forecasts with a local model?



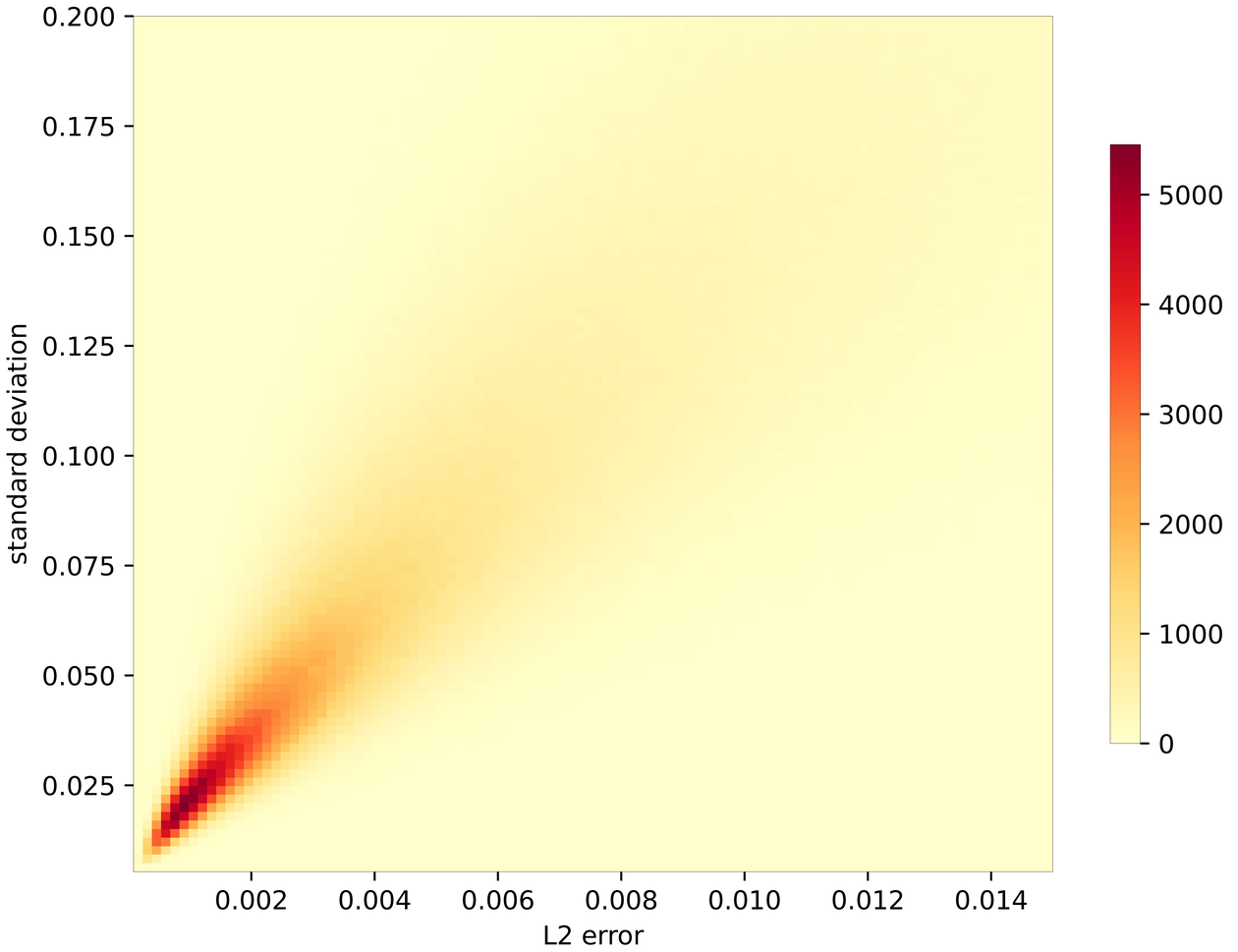
Model correction

network input: ERA5

network input: IFS



Statistical loss



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