@rachelafurner

An iterative data-driven model of an ocean General Circulation Model (GCM)

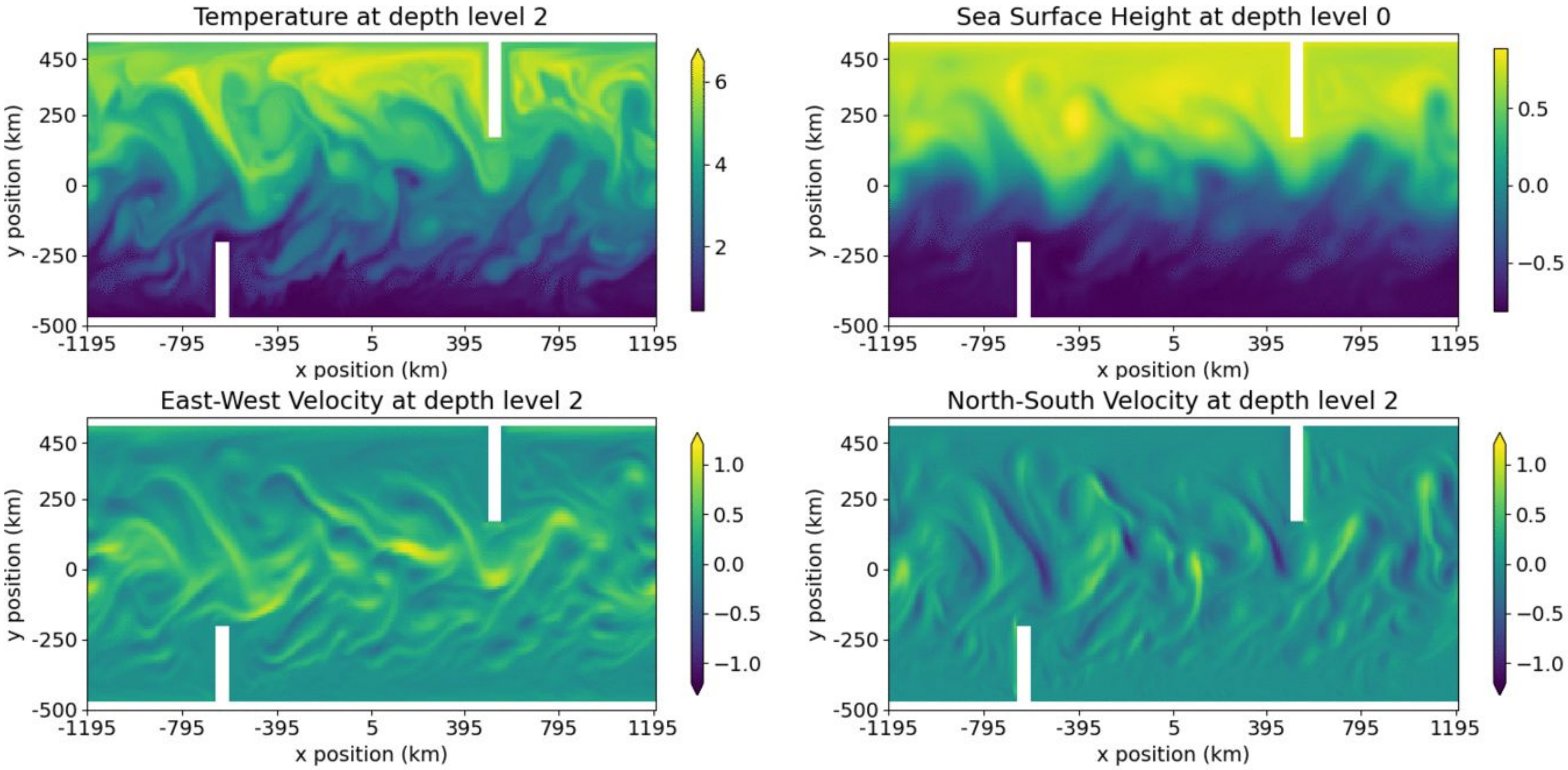
Rachel Furner

Supervisors: Emily Shuckburgh, Peter Haynes, Brooks Paige, Dave Munday, Dan Jones

British Antarctic Survey
NATURAL ENVIRONMENT RESEARCH COUNCIL

UNIVERSITY OF CAMBRIDGE

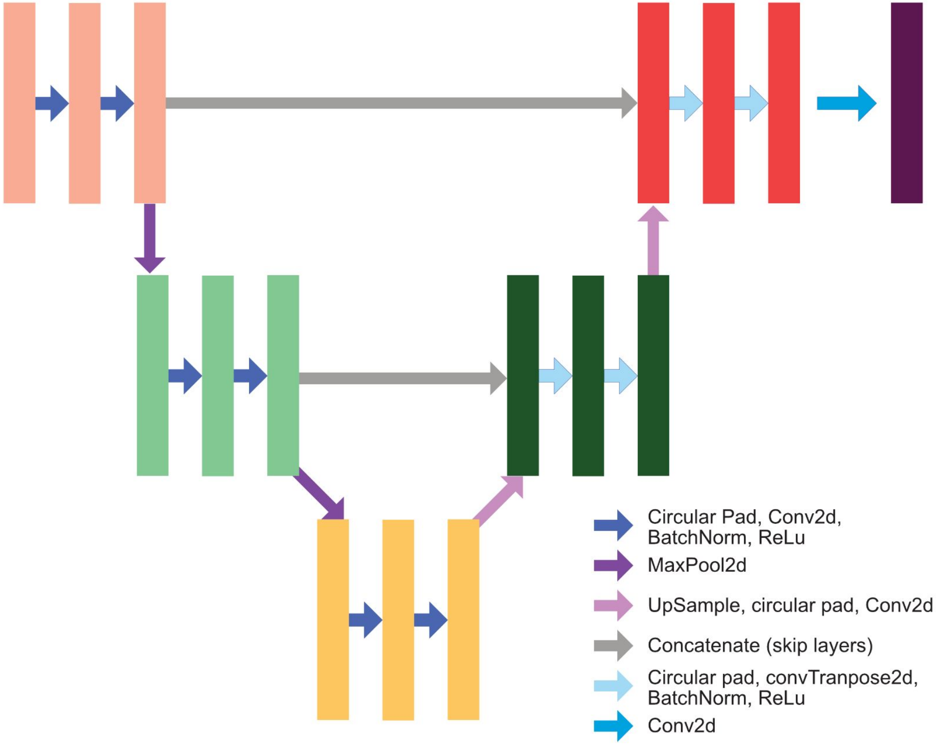
Idealised MITgcm channel model configuration



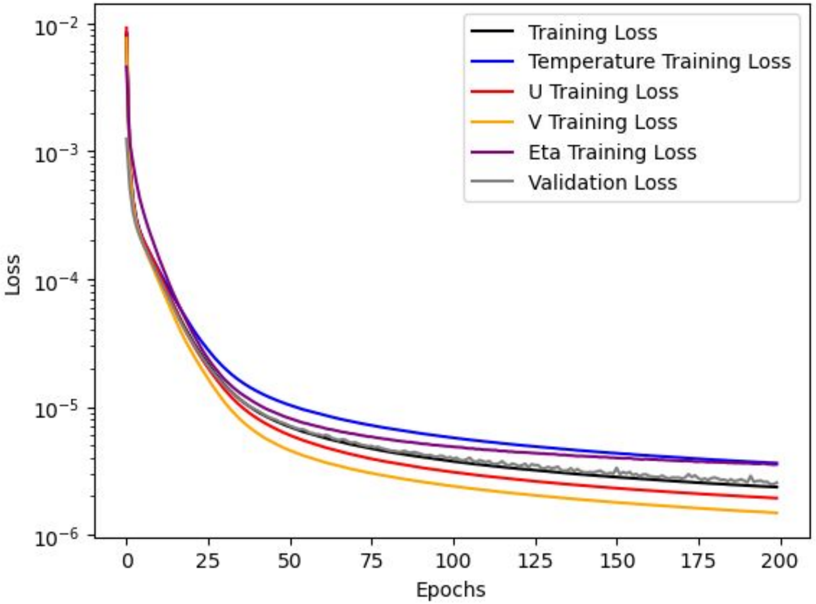
- Periodic boundaries
- Simple forcing
- 3d: 240 x 104 x 38 ‘pixels’
- 10 minute time step, 12 hourly outputting

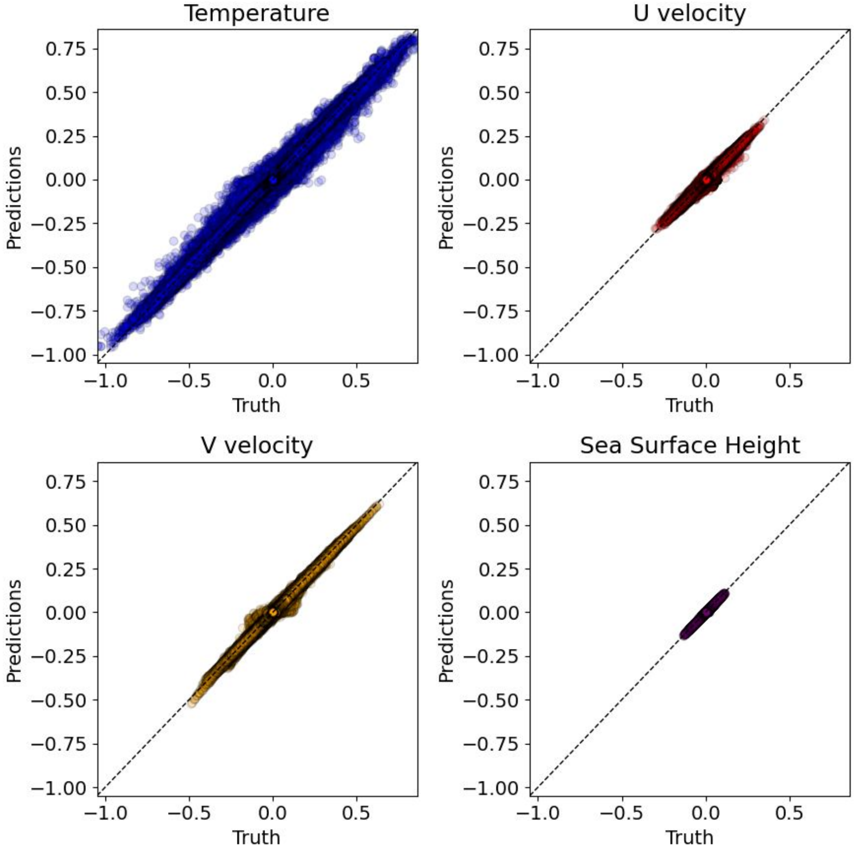
Approach with a 2d U-Net style network

- Combination of 2d convolutional, max pooling, upsampling, & convtranspose layers, and skip connections
- 230 channels (each variable at each depth levels plus land-sea masks)
- **We apply circular padding at East & West boundaries. North & South boundaries aren't padded.**
- **Train to minimise RMS error over all variables at all locations**
- Use Adam optimiser, learning rate of 3.e-6
- 5400 training samples, 1080 validation samples

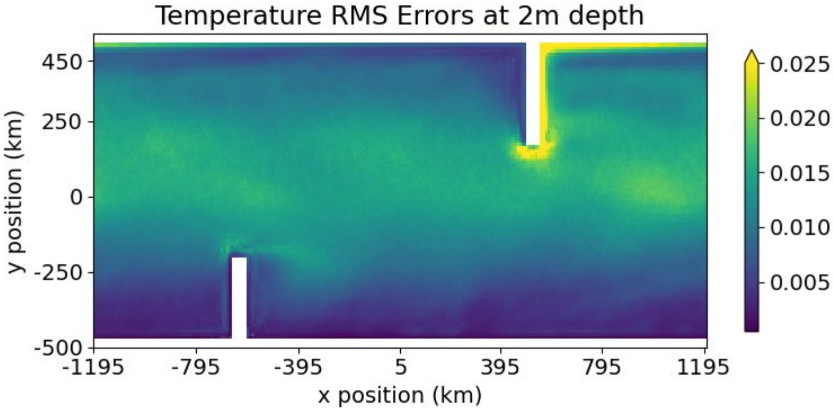


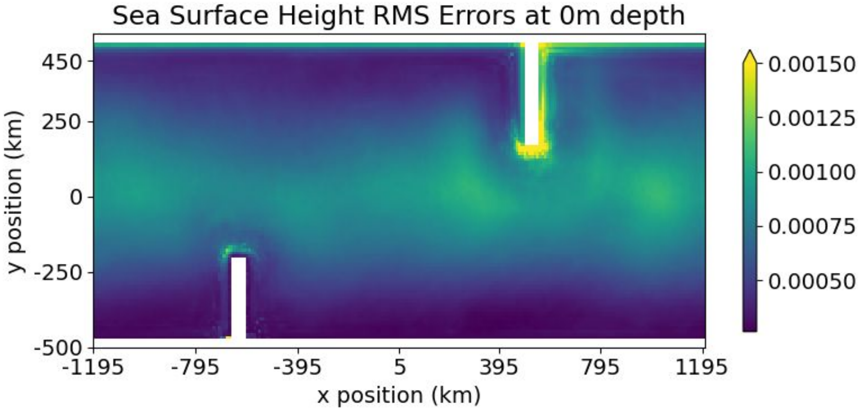
Network Performance

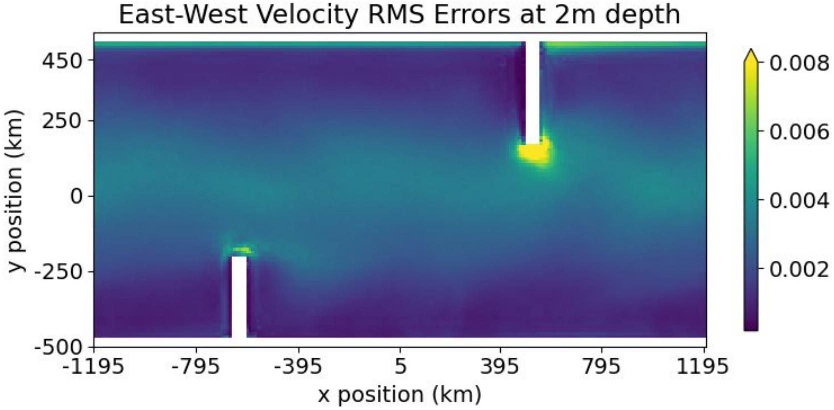


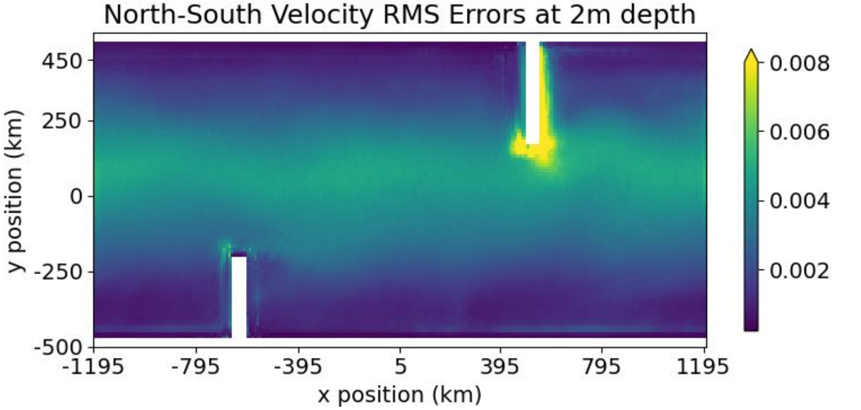


RMS Errors

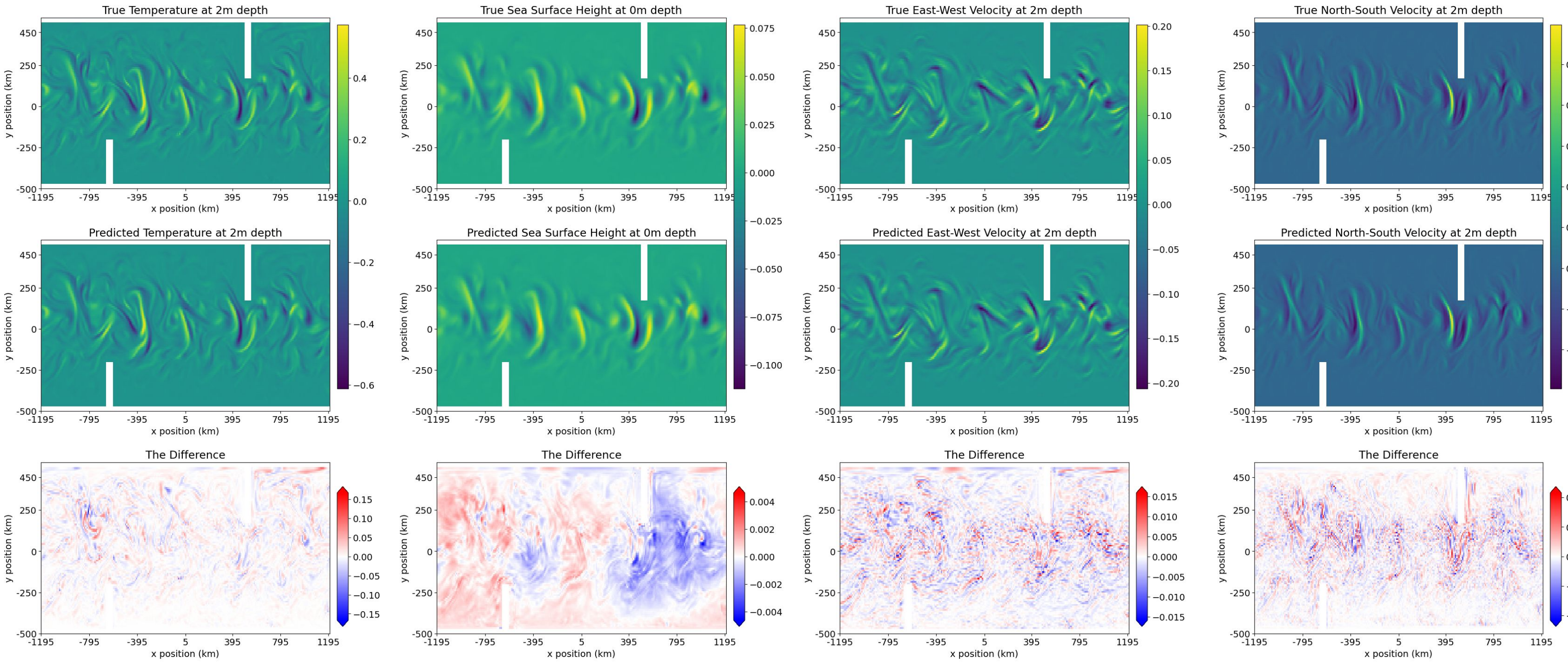

Temperature RMS Errors at 2m depth
y position (km)
450
250
0
-250
-500
x position (km)
-1195
-795
-395
5
395
795
1195
0.025
0.020
0.015
0.010
0.005


Sea Surface Height RMS Errors at 0m depth
y position (km)
450
250
0
-250
-500
x position (km)
-1195
-795
-395
5
395
795
1195
0.00150
0.00125
0.00100
0.00075
0.00050

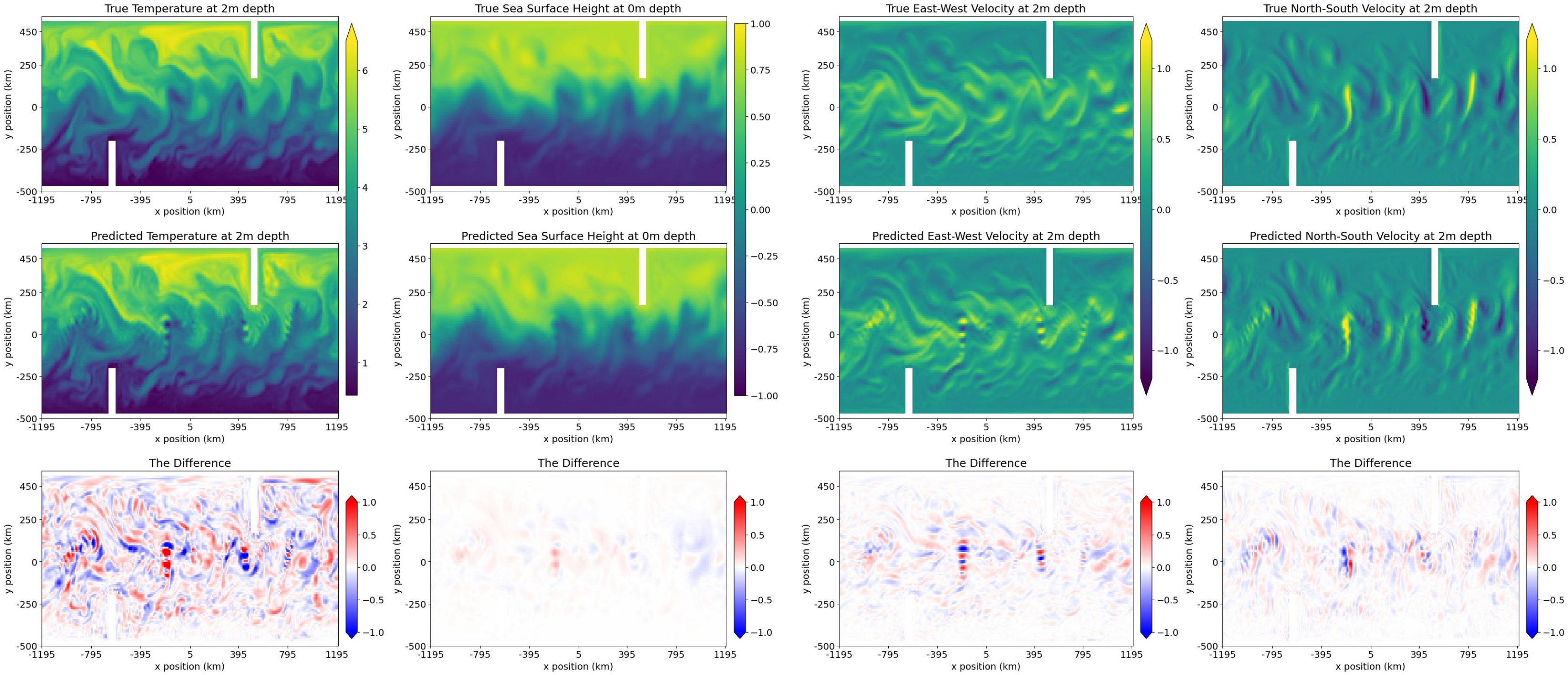

East-West Velocity RMS Errors at 2m depth
y position (km)
450
250
0
-250
-500
x position (km)
-1195
-795
-395
5
395
795
1195
0.008
0.006
0.004
0.002


North-South Velocity RMS Errors at 2m depth
y position (km)
450
250
0
-250
-500
x position (km)
-1195
-795
-395
5
395
795
1195
0.008
0.006
0.004
0.002

Example predictions



Iterating the model - 2 weeks



Adams–Bashforth Second Order Iteration

Previously we used a very simple autoregressive iteration scheme:

$$f(y_t) = \Delta y_t \qquad y_{t+1} = y_t + f(y_t)$$

Adams–Bashforth Second Order Iteration

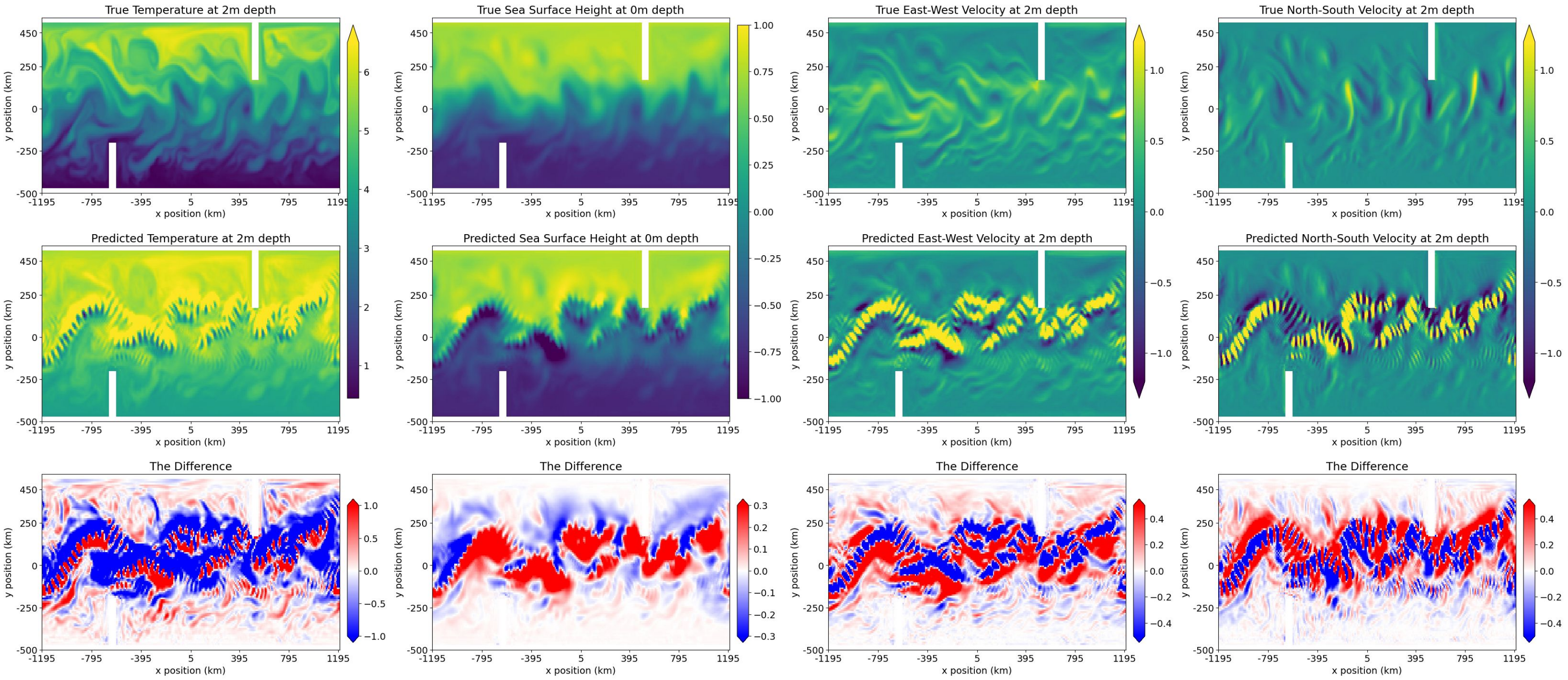
Previously we used a very simple autoregressive iteration scheme:

$$f(y_t) = \Delta y_t \qquad y_{t+1} = y_t + f(y_t)$$

Instead with try with a second order AB method:

$$y_{t+1} = y_t + \frac{3}{2} f(y_t) - \frac{1}{2} f(y_{t-1})$$

Adams–Bashforth Second Order Iteration
Iterating for 2 weeks



We also tried…..

- A Runge-Kutta iteration scheme

We also tried…..

- A Runge-Kutta iteration scheme

- Smoothing - within simulators we apply mixing/smoothing to compensate for sub-grid scale processes and to ensure stability

We also tried…..

- A Runge-Kutta iteration scheme
- Smoothing - within simulators we apply mixing/smoothing to compensate for sub-grid scale processes and to ensure stability
- Multimodel averaging - training multiple NNs from different random seed starts, and averaging output

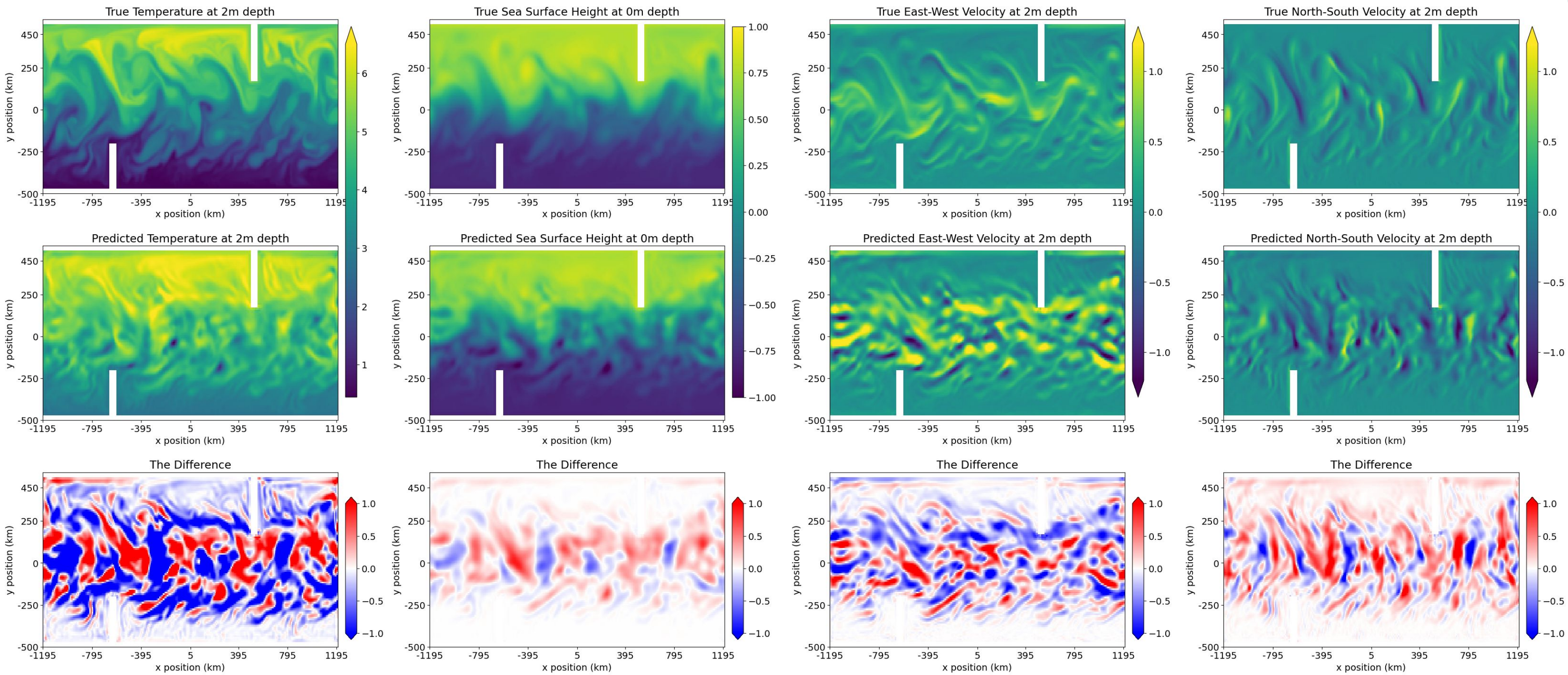
Is separation of prediction & iteration a flawed premise?

- Our network architecture and design, and our loss function needs to address the actual question we are interested in

Is separation of prediction & iteration a flawed premise?

- Our network network architecture and design, and our loss function needs to address the actual question we are interested in
- Conv LSTM layers offer a more ‘natural’ way to predict for 2-d timeseries data

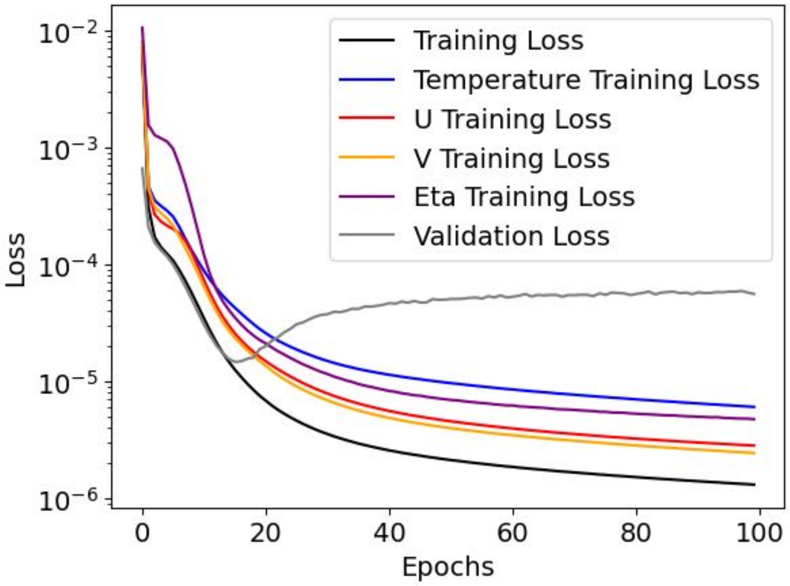
UNet-ConvLSTM model Iterating for 2 weeks

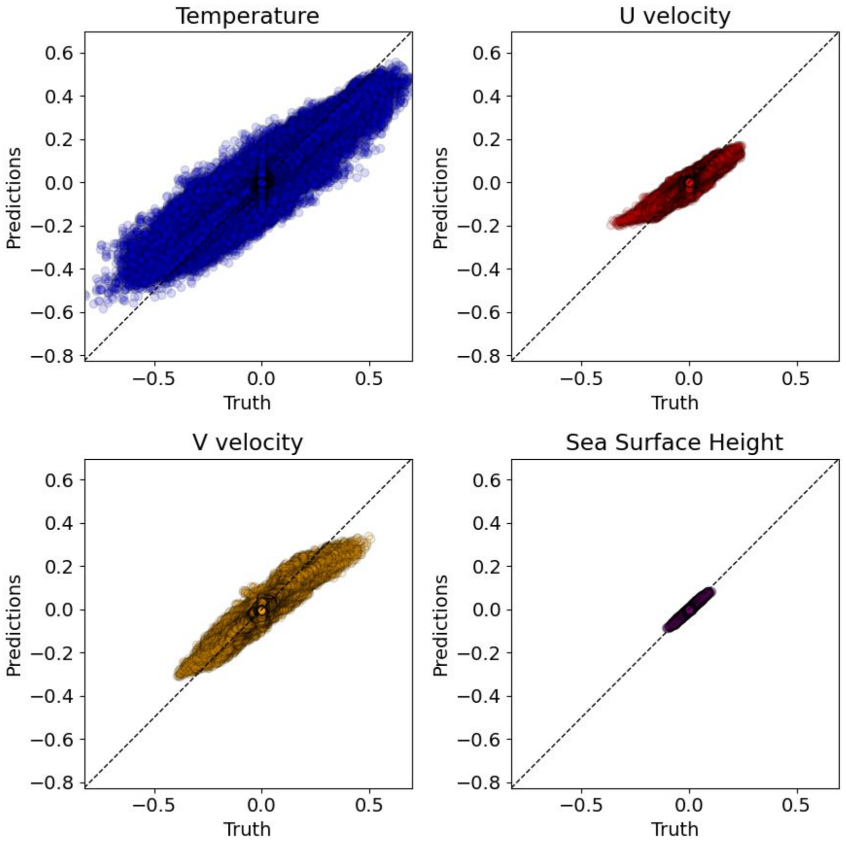


Is separation of prediction & iteration a flawed premise?

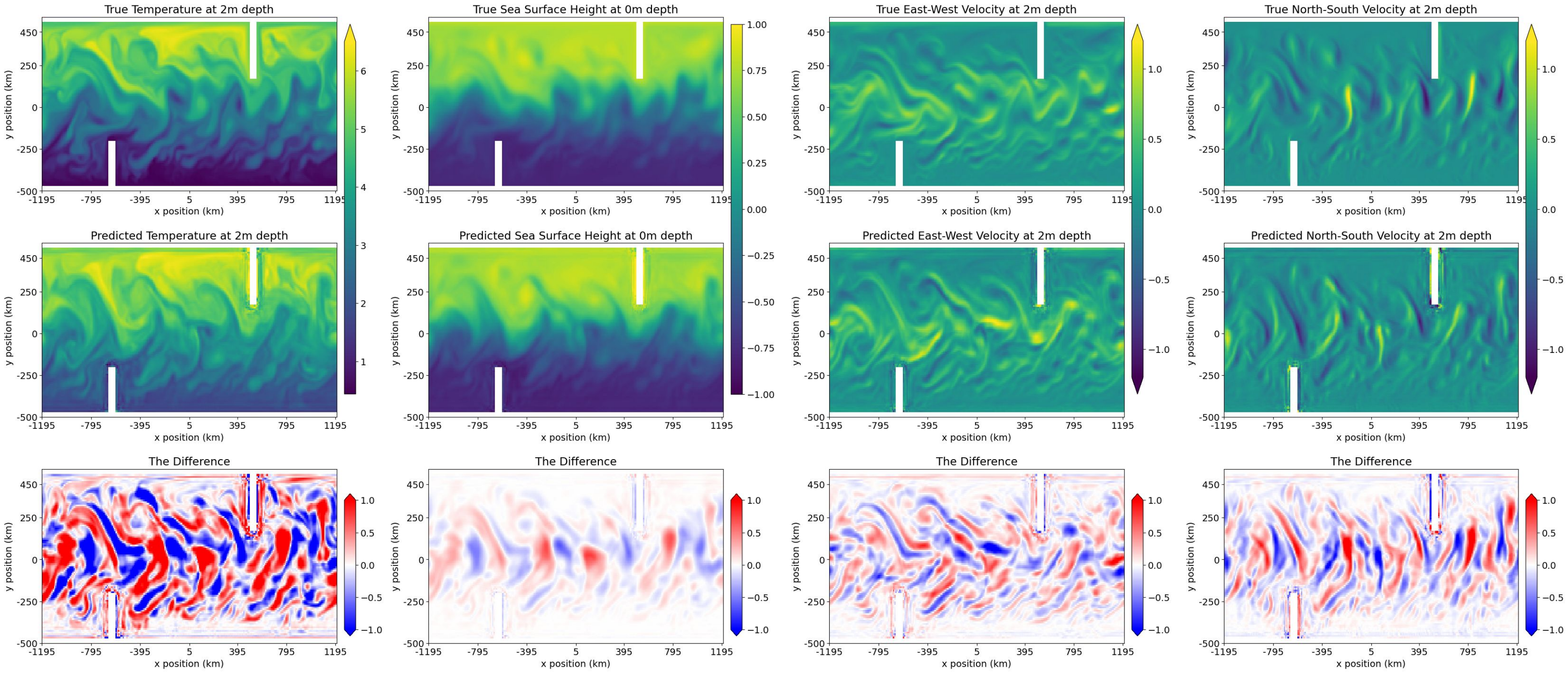
- Our network network architecture and design, and our loss function needs to address the actual question we are interested in
- Conv LSTM layers offer a more ‘natural’ way to predict for 2-d timeseries data
- We are actually interested in the errors when applying the network over time, so creating a loss function which better represents this is key

Loss function = RMS over 3 iterative forecast steps



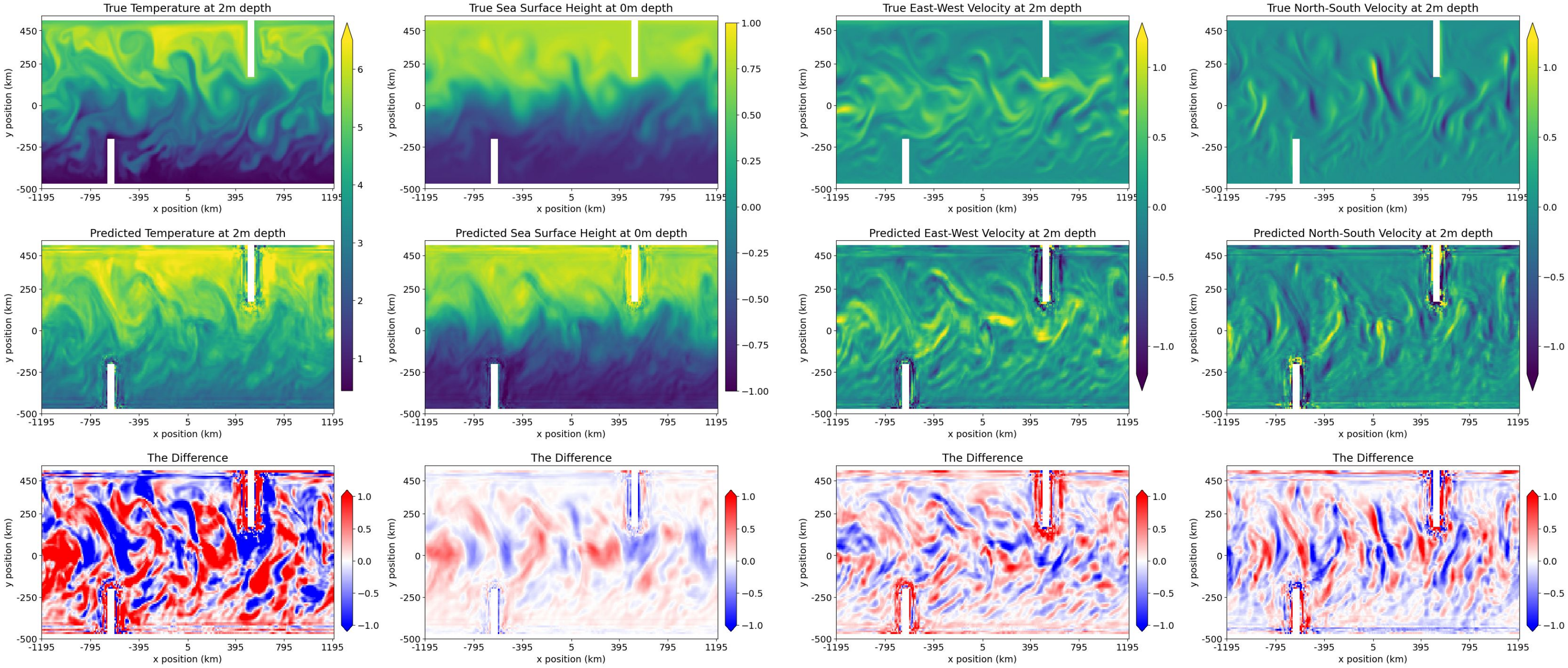


Loss function = RMS over 3 iterative forecast steps
Iterating for 2 weeks



Loss function = RMS over 3 iterative forecast steps

Iterating for **4 weeks**



Is separation of prediction & iteration a flawed premise?

- Our network network architecture and design, and our loss function needs to address the actual question we are interested in
- Conv LSTM layers offer a more ‘natural’ way to predict for 2-d timeseries data
- We are actually interested in the errors when applying the network over time, so creating a loss function which better represents this is key
- ….but not trivial, at future time steps we are interested in feature resolution over location, RMS loss functions lead to overfitting, and/or oversmoothing issues.

Conclusions

- Creating a data-driven analogue model of a physics based ocean simulator works well in an idealised use cases, when assessing predictive skill over a single time step

Conclusions

- Creating a data-driven analogue model of a physics based ocean simulator works well in an idealised use cases, when assessing predictive skill over a single time step
- However, these models are most useful when used in an iterative way, and performance here is shows problems

Conclusions

- Creating a data-driven analogue model of a physics based ocean simulator works well in an idealised use cases, when assessing predictive skill over a single time step
- However, these models are most useful when used in an iterative way, and performance here is shows problems
- Separating the problem of predicting and iterating is a dangerous approach when using data-driven methods

Conclusions

- Creating a data-driven analogue model of a physics based ocean simulator works well in an idealised use cases, when assessing predictive skill over a single time step
- However, these models are most useful when used in an iterative way, and performance here is shows problems
- Separating the problem of predicting and iterating is a dangerous approach when using data-driven methods
- We need to more carefully consider the optimisation question we are asking of ML techniques:
  - What metrics are best used to train (and validate) our models
  - RMS error is not sufficient, and for longer forecast horizons, is rarely useful and can be detrimental