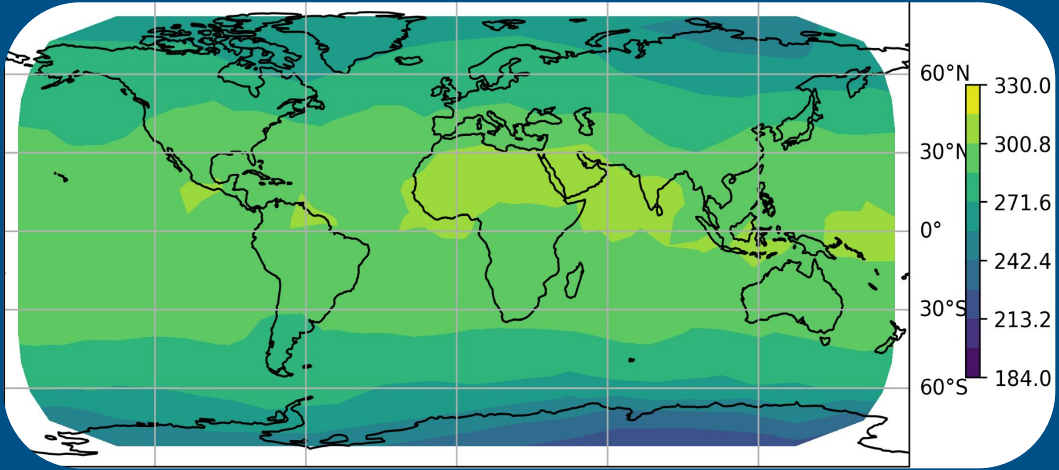
Adopting Physical Laws in ClimSim

Clark Kaminsky
Álvaro Vega Hidalgo



UNIVERSITY OF MICHIGAN

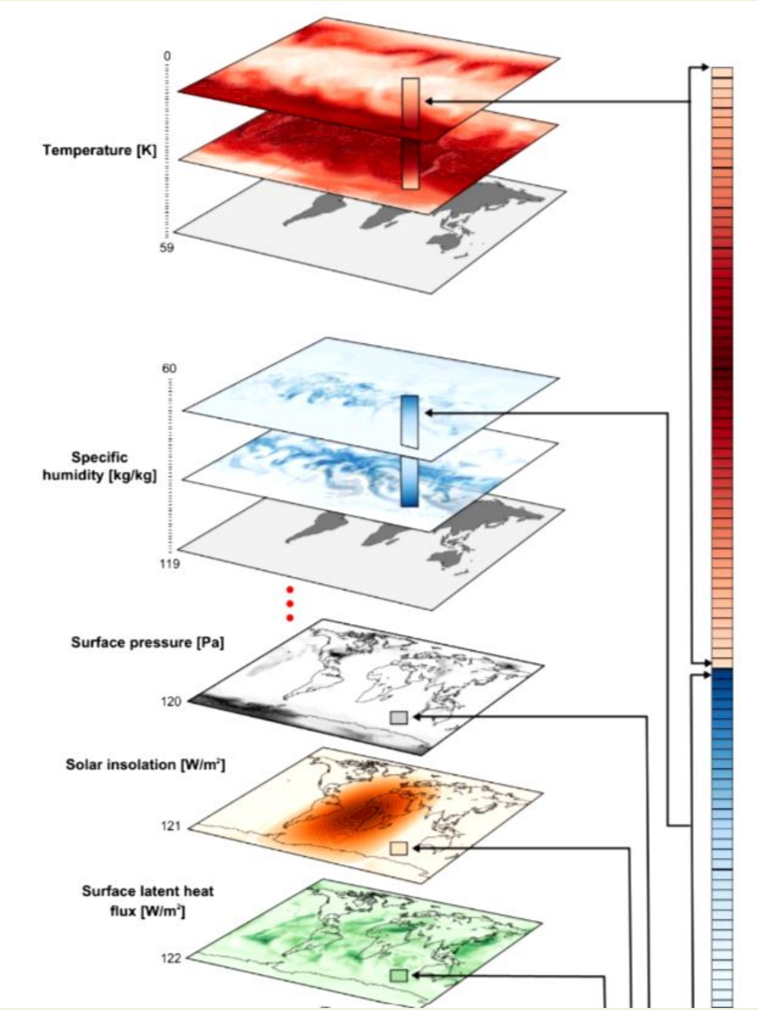
ClimSim Dataset:

High-resolution E3SM (Energy Exascale Earth System Model) climate simulations

20-minute intervals over 10 simulated years

- High resolution: 1.5° x 1.5° horizontal grid (41.2 TB)
- Low-resolution: 11.5° x 11.5° (744 GB)

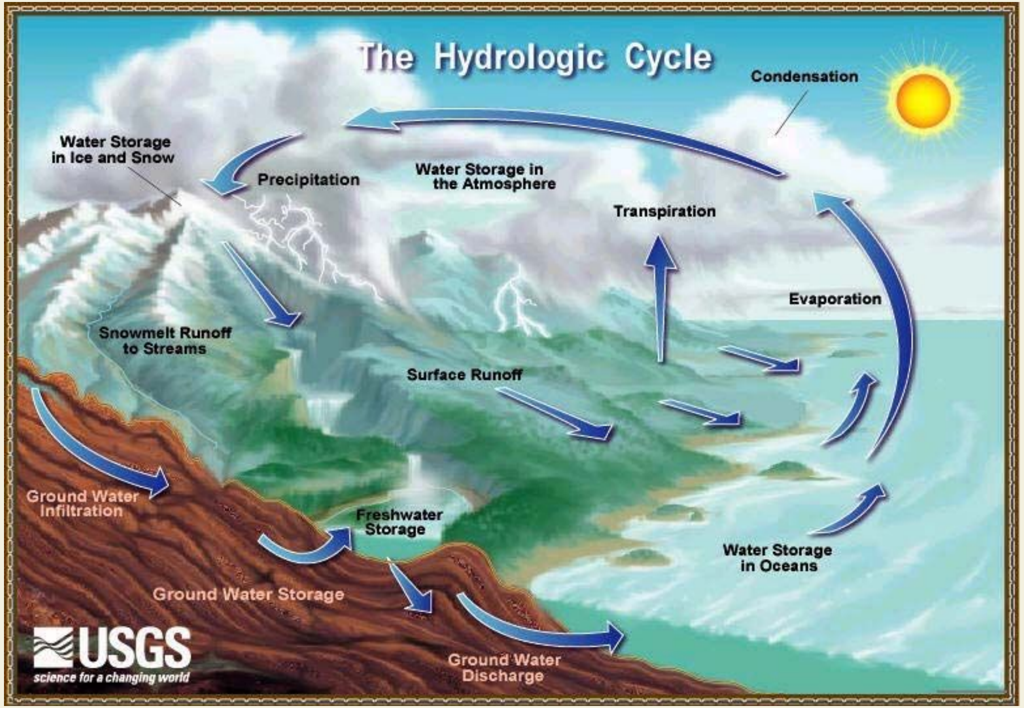
Input	Size	Target	Size
Temperature [K]	60	Heating tendency, dT/dt [K/s]	60
Specific humidity [kg/kg]	60	Moistening tendency, dq/dt [kg/kg/s]	60
Surface pressure [Pa]	1	Net surface shortwave flux, NETSW [W/m^2]	1
Insolation [W/m^2]	1	Downward surface longwave flux, FLWDS [W/m^2]	1
Surface latent heat flux [W/m^2]	1	Snow rate, PRECSC [m/s]	1
Surface sensible heat flux [W/m^2]	1	Rain rate, PRECC [m/s]	1
		Visible direct solar flux, SOLS [W/m^2]	1
		Near-IR direct solar flux, SOLL [W/m^2]	1
		Visible diffused solar flux, SOLSD [W/m^2]	1
		Near-IR diffused solar flux, SOLLD [W/m^2]	1



Conservation of Water Mass:

Maintain the net water mass balance between Earth's surface and the atmosphere:

- Water is simply transported across the globe through different phases, not created or destroyed



$$L_{mass} = \frac{1}{N}\sum_{i=1}^{N}\left|\Delta q_{water_i} - (E_i - P_i)\right|$$

change in specific humidity

evaporation rate

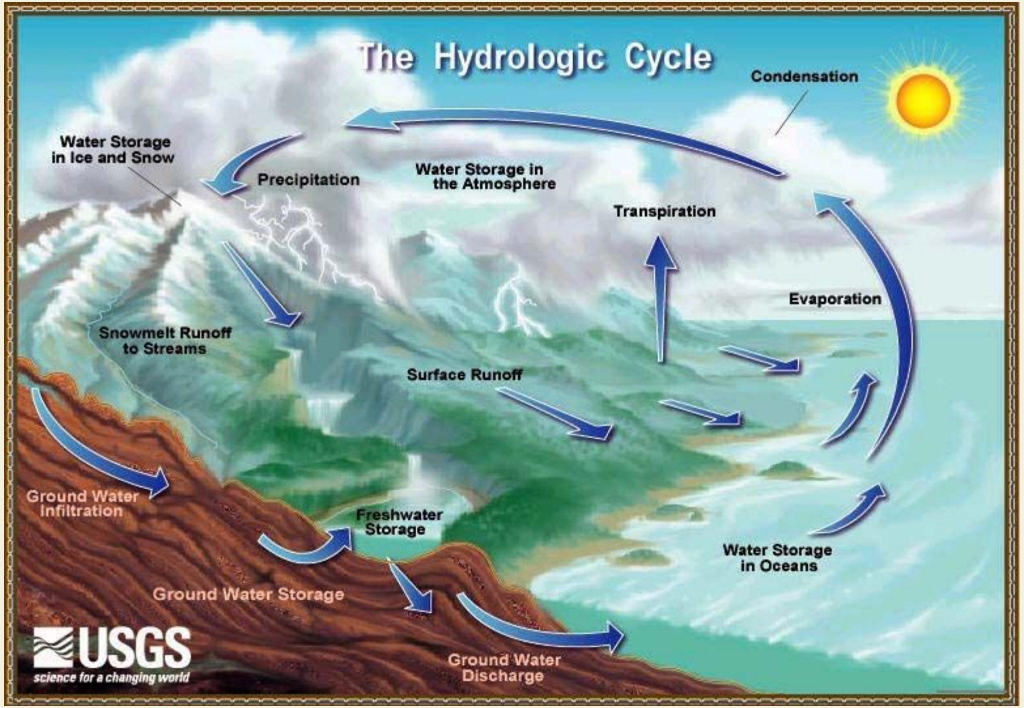
precipitation rate

Conservation of Water Mass:

Maintain the net water mass balance between Earth's surface and the atmosphere:

- Water is simply transported across the globe through different phases, not created or destroyed

If precipitation is higher than evaporation, specific humidity should decrease equivalently



$$\mathrm{L_{mass}} = \frac{1}{N}\sum_{i=1}^{N}\left|\Delta q_{\mathrm{water}_i} - (E_i - P_i)\right|$$

change in specific humidity

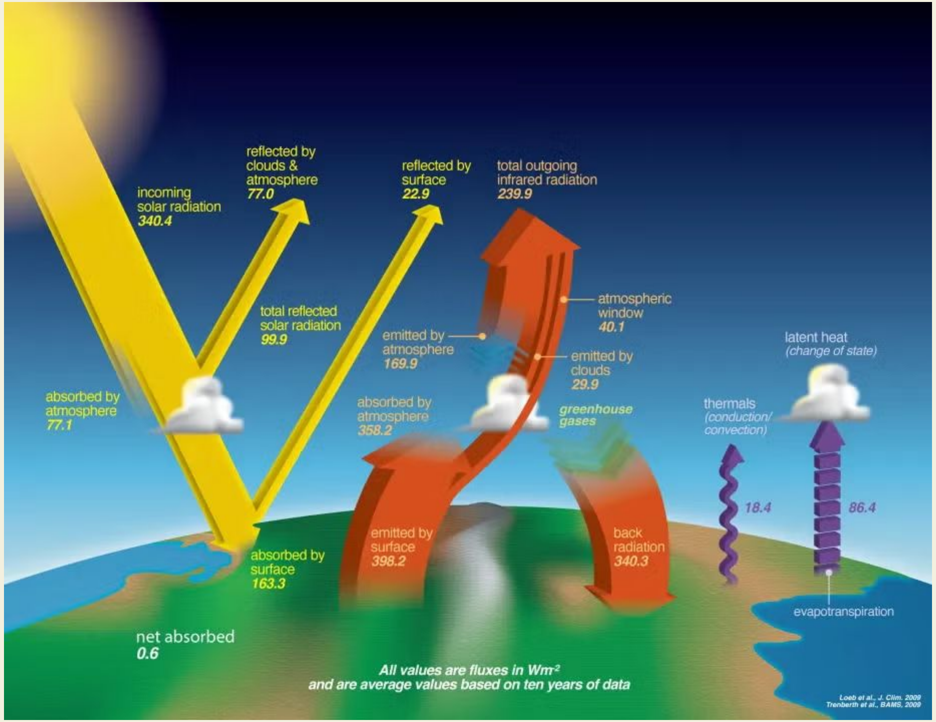
evaporation rate

precipitation rate

Conservation of Energy:

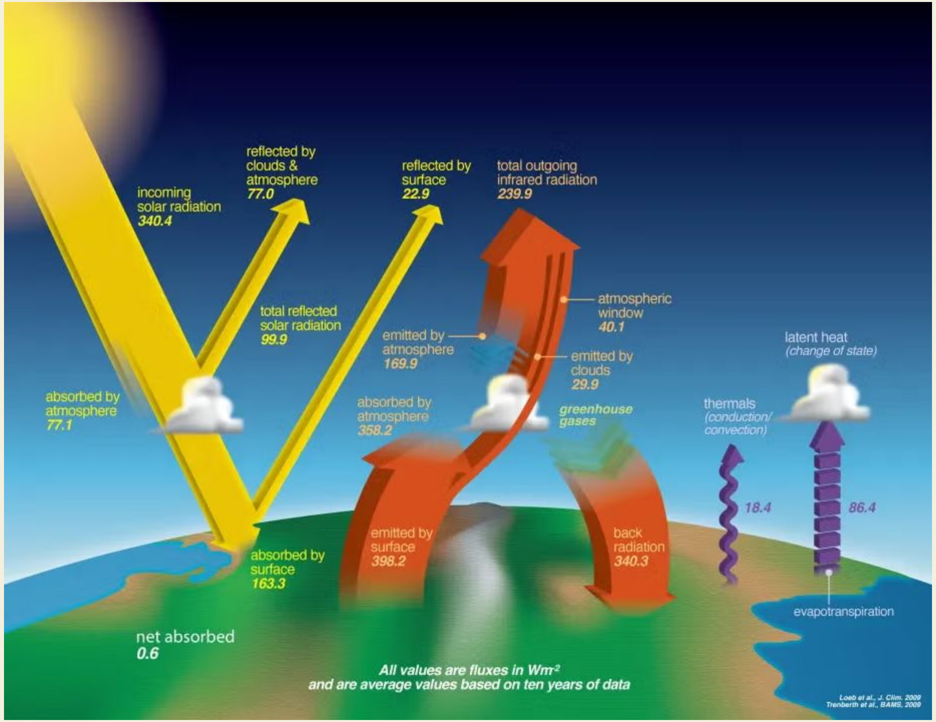
Maintain the net energy balance at the Earth's surface:

Incoming shortwave and longwave radiation = surface fluxes



$$\mathrm{L}_{\mathrm{energy}} = \frac{1}{N}\sum_{i=1}^{N}\left|(\mathrm{NETSW}_i + \mathrm{NETLW}_i) - (\mathrm{SH}_i + \mathrm{LH}_i)\right|$$

Conservation of Energy:



Maintain the net energy balance at the Earth's surface:

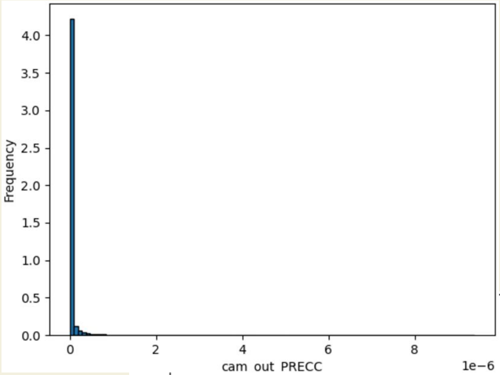
Incoming shortwave and longwave radiation = surface fluxes

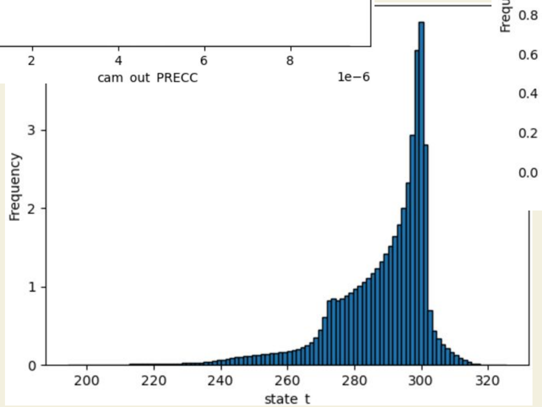
- Latent heat flux – energy required for the phase change of water
- Sensible heat flux – energy moved from difference between the surface and air temperature

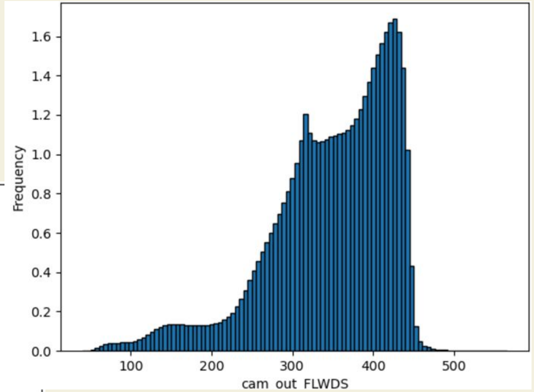
$$L_{\text{energy}} = \frac{1}{N}\sum_{i=1}^{N}\left|(\text{NETSW}_i + \text{NETLW}_i) - (\text{SH}_i + \text{LH}_i)\right|$$

Non-negativity Constraints:

We want to constrain variables to the positive domain when we know they can never be negative:







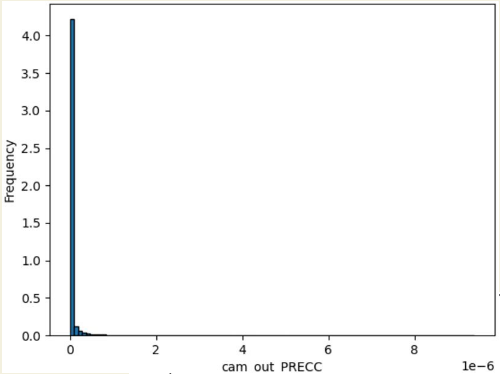
$$\mathrm{L}_{\mathrm{nonneg}} = \sum_{v} \frac{1}{N} \sum_{i=1}^{N} \max\left(0, -v_i\right)$$

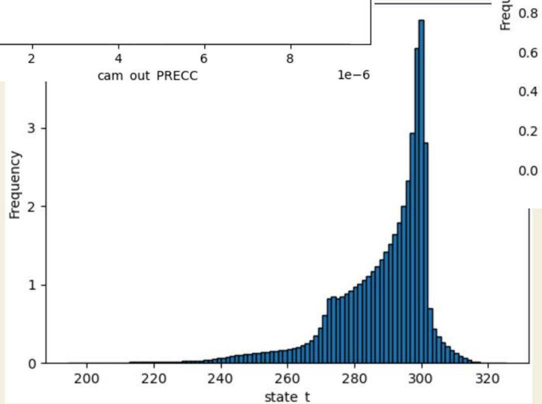
$$v \geq 0 \quad \forall v \in \{\mathrm{TEMP}, \mathrm{PRECC}, \mathrm{PRECSC}, \ldots\}$$

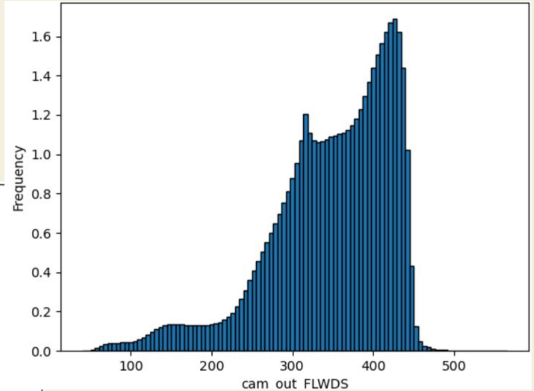
Non-negativity Constraints:

We want to constrain variables to the positive domain when we know they can never be negative:

Ex: snow rate, rain rate, specific humidity, or temperature (°K) are all guaranteed to be greater than 0 – we want our model's loss to be higher if it predicts these terms to be negative.





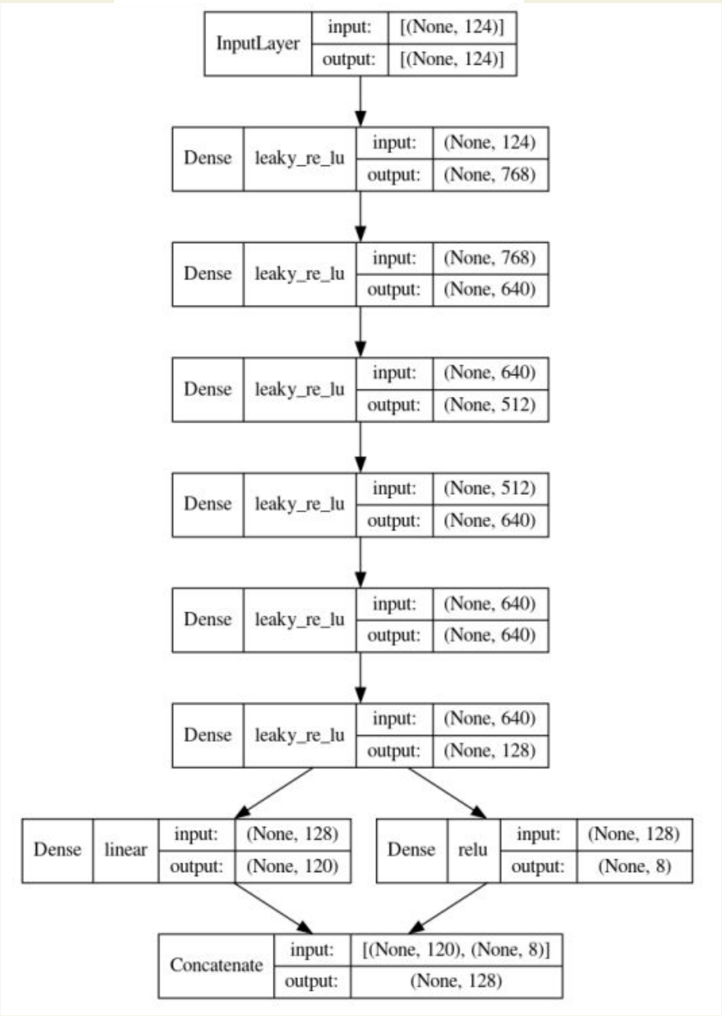


$$\mathrm{L}_{\mathrm{nonneg}} = \sum_{v} \frac{1}{N} \sum_{i=1}^{N} \max\left(0, -v_i\right)$$

$$v \geq 0 \quad \forall v \in \{\mathrm{TEMP}, \mathrm{PRECC}, \mathrm{PRECSC}, \ldots\}$$

Multi-layer Perceptron:

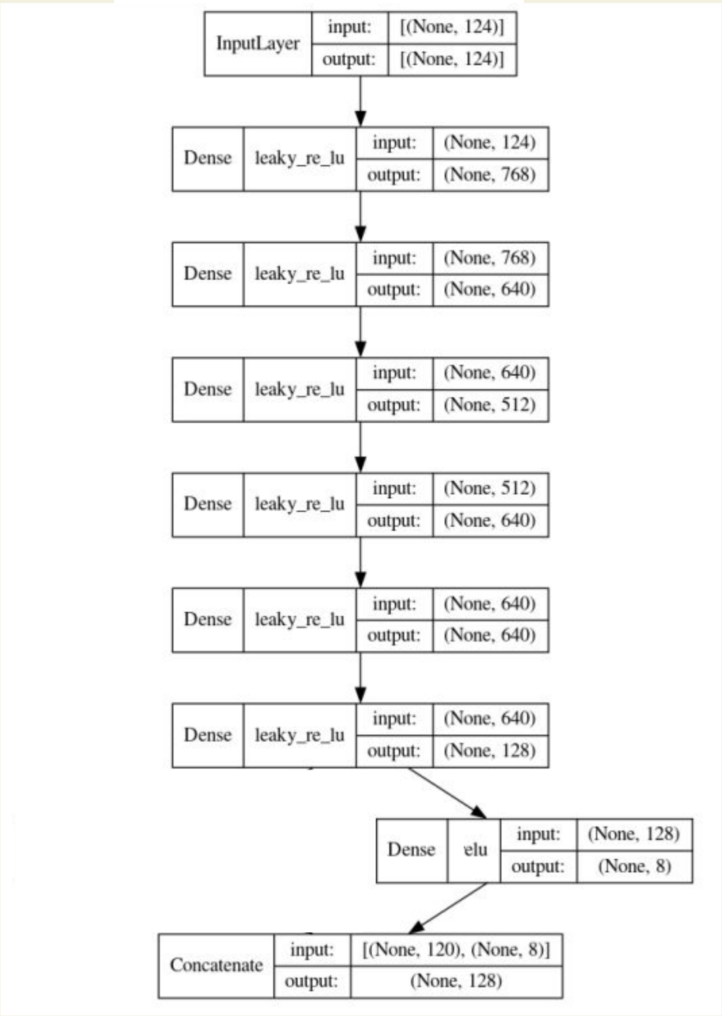
Used the best hyperparameter configuration from the ClimSim analysis – 5 layers with nodes [768, 640, 512, 640, 640], LeakyReLU activation, and RAdam optimizer.



Multi-layer Perceptron:

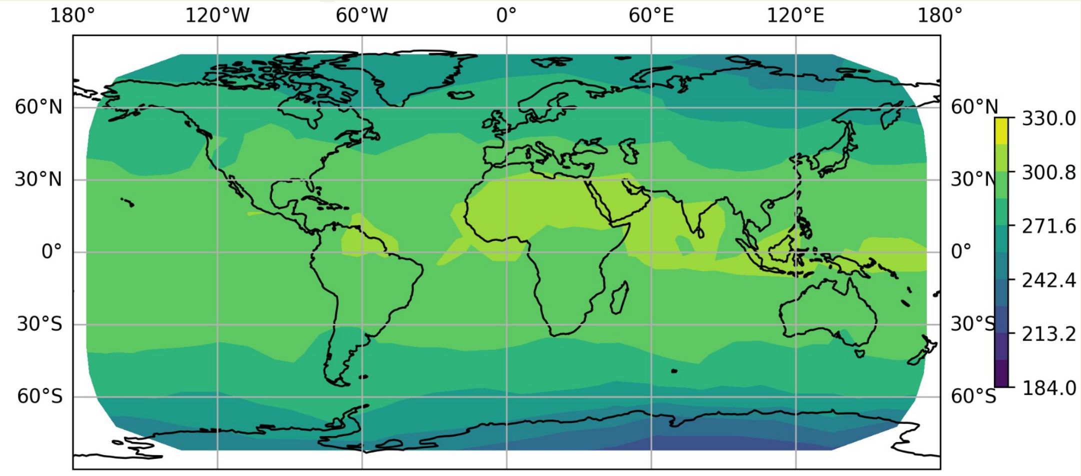
Our changes:

- Excluded some multilevel variables to reduce computational time and memory requirements

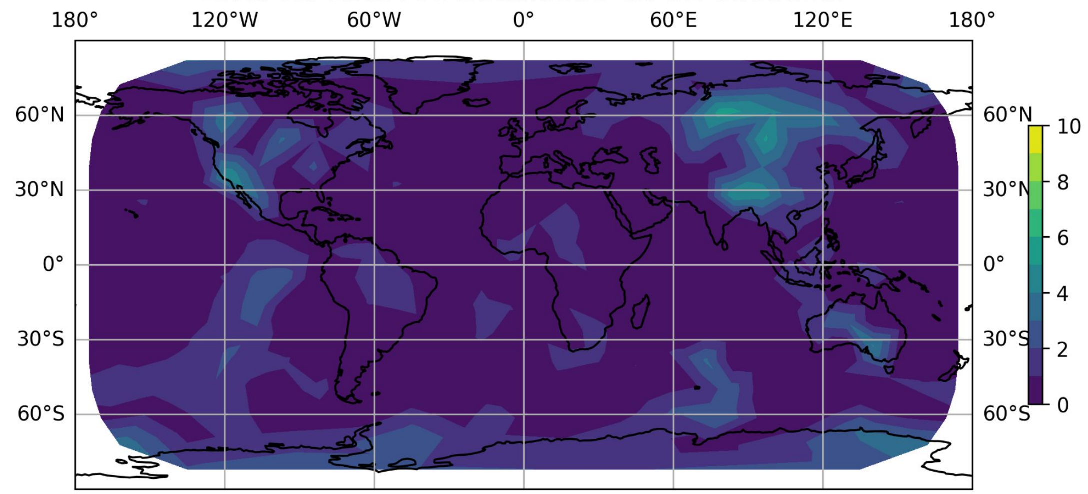


Temperature Prediction Example:

Surface Temperature (K) Prediction for May 1:



Absolute error for predictions:

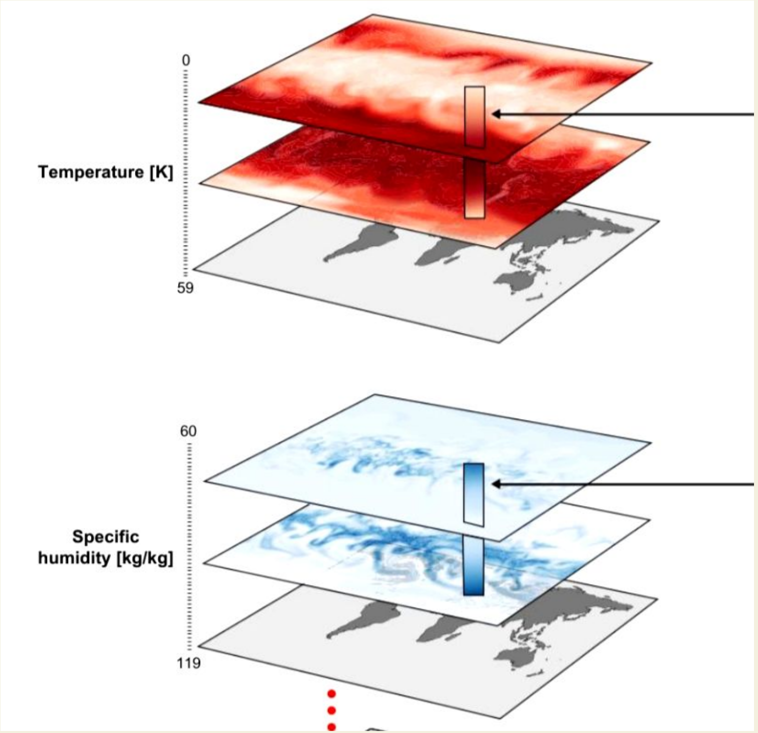


Methods (preprocessing):

Task: predicting output variables (temperature, rain rate, specific humidity) at each of 384 global locations 20 minutes in the future

Preprocessing:

- Normalized the variables for training stability
- Subsampled multilevel variables to 1 altitude level (surface) instead of 60, used only 1% of the available low resolution dataset



Methods (loss incorporation):

Task: predicting output variables (temperature, rain rate, specific humidity) at each of 384 global locations 20 minutes in the future

Incorporate physics-informed losses: balance, then multiply each with scaling factors (λ)

Balancing multiple loss functions:

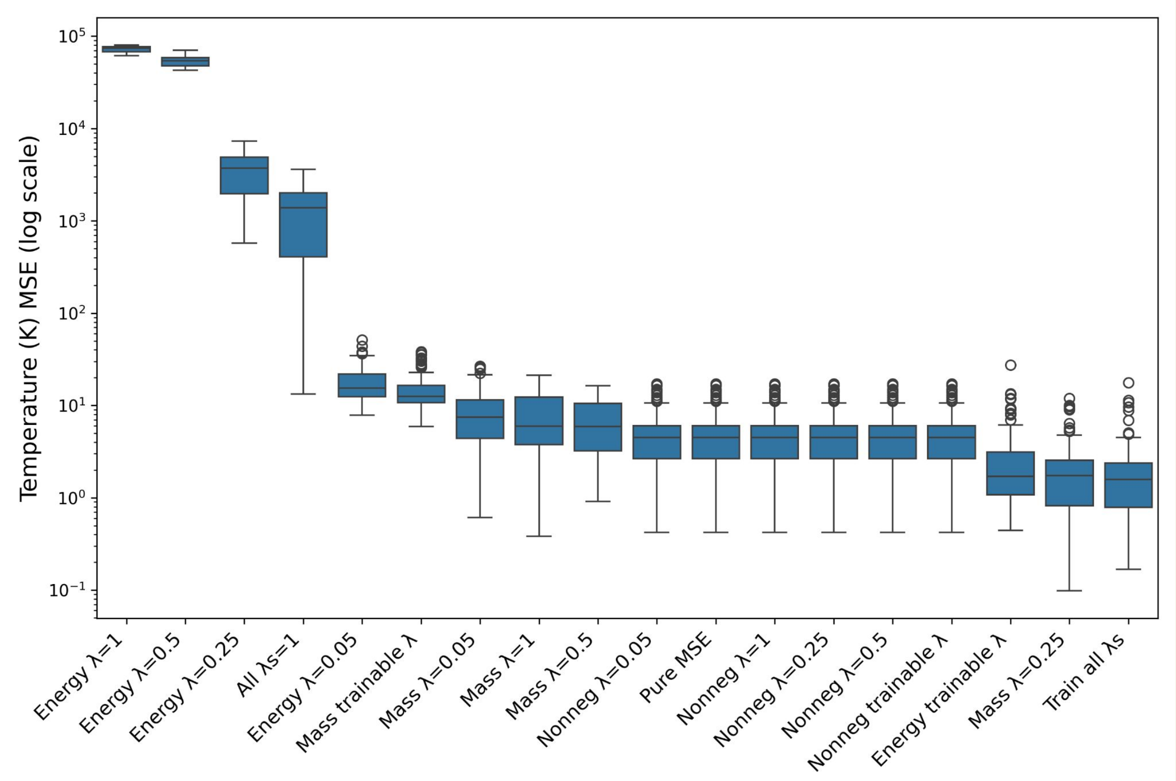
- Estimate the initial loss with the first few batches, scale each loss value with the reciprocal of its initial value

$$\mathcal{L} = \sum_{i=1}^{n} \frac{\mathcal{L}_i}{\mathcal{L}_i^{(\text{initial})}}$$

$$L_{total} = L_{data} + \lambda_1 L_{energy} + \lambda_2 L_{mass} + \lambda_3 L_{nonneg}$$

Results:

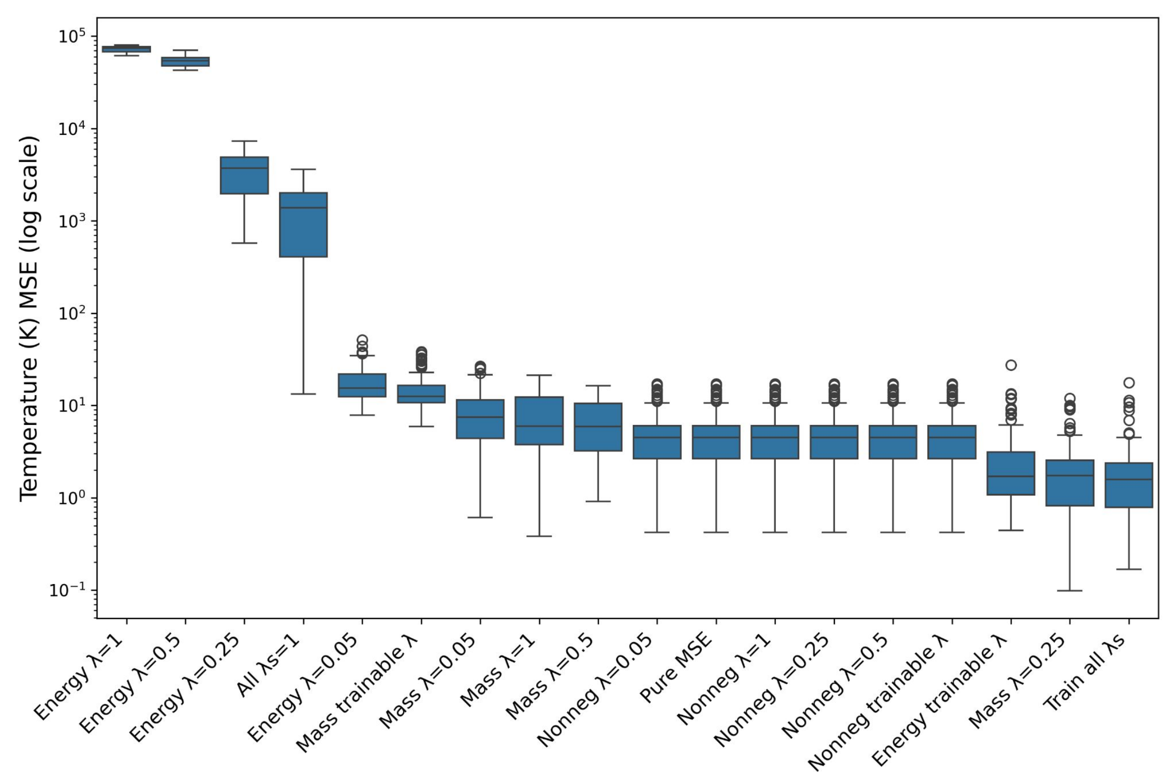
Baseline MLP ('PureMSE') uses only data loss to guide its optimization, and performs relatively well



Results:

Baseline MLP (‘PureMSE’) uses only data loss to guide its optimization, and performs relatively well

Incorporating all of the loss terms yields a significant improvement over the baseline

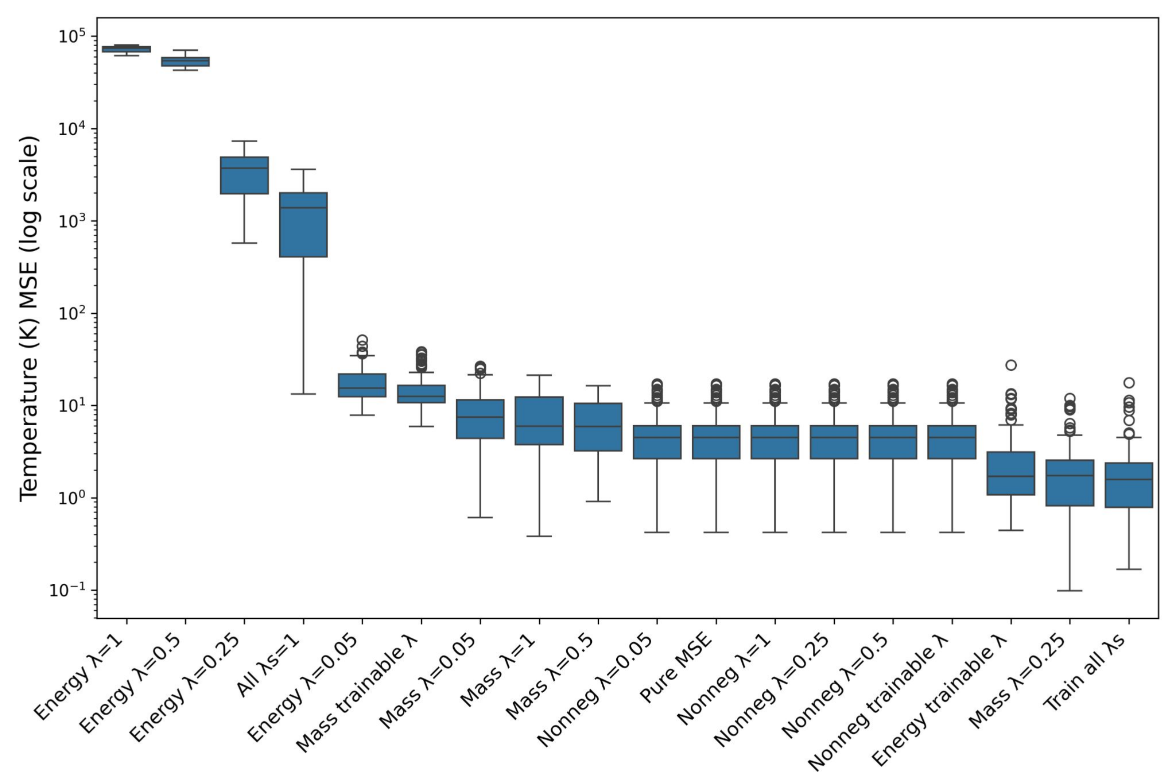


Results:

Baseline MLP ('PureMSE') uses only data loss to guide its optimization, and performs relatively well

Incorporating all of the loss terms yields a significant improvement over the baseline

Non-negativity constraint is never worse than the baseline



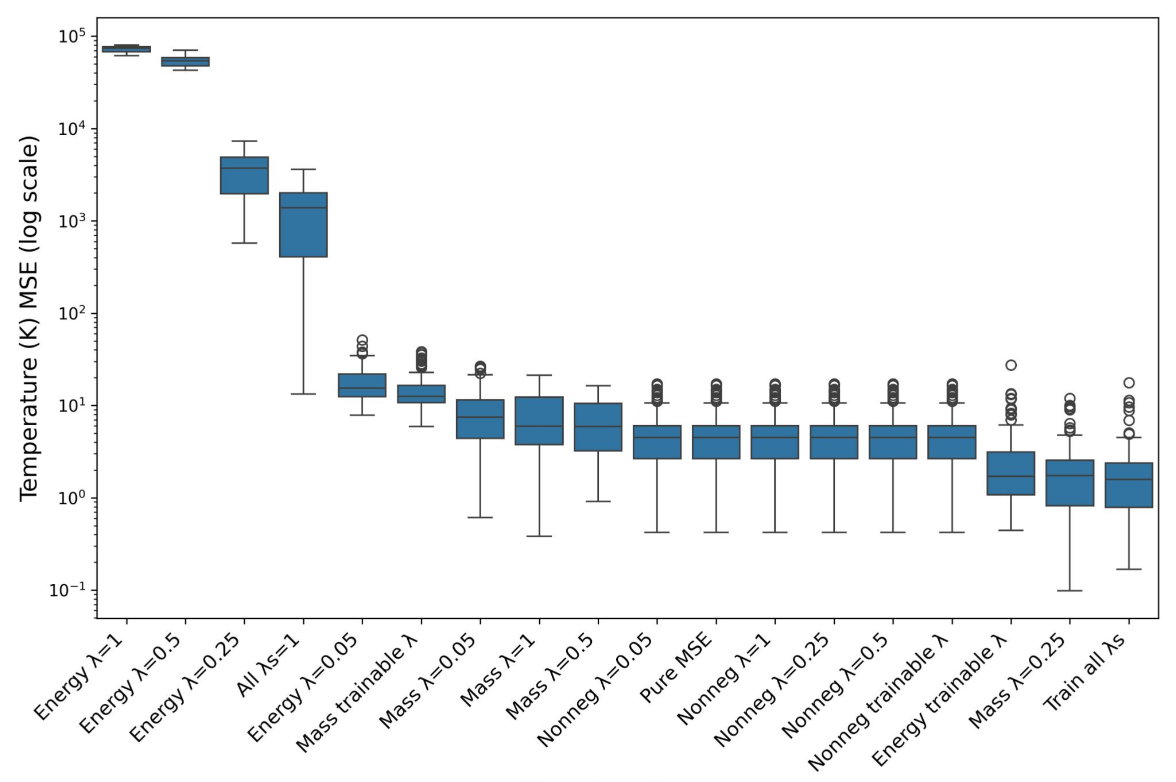
Results:

Baseline MLP ('PureMSE') uses only data loss to guide its optimization, and performs relatively well

Incorporating all of the loss terms yields a significant improvement over the baseline

Non-negativity constraint is never worse than the baseline

Still have to look into why some trainable scaling factors were worse than setting explicitly (see water mass conservation λ = .25)



Results:

Baseline MLP ('PureMSE') uses only data loss to guide its optimization, and performs relatively well

Incorporating all of the loss terms yields a significant improvement over the baseline

Non-negativity constraint is never worse than the baseline

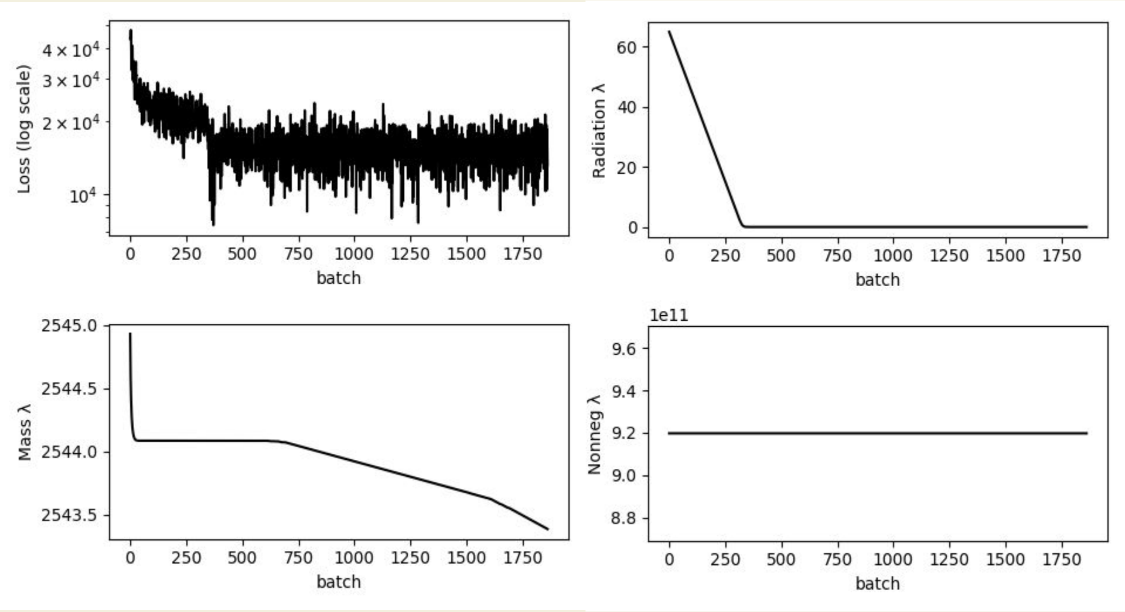
Still have to look into why some trainable scaling factors were worse than setting explicitly (see water mass conservation λ = .25)

Experiment name	Temperature MSE
Energy λ=1	72712.51
Energy λ=0.5	54177.82
Energy λ=0.25	3514.68
All λs=1	1321.56
Energy λ=0.05	17.73
Mass trainable λ	14.40
Mass λ=0.05	8.48
Mass λ=1	7.87
Mass λ=0.5	6.84
Nonneg λ=0.25	5.10
Pure MSE	5.10
Nonneg λ=0.05	5.10
Nonneg λ=1	5.10
Nonneg λ=0.5	5.10
Nonneg trainable λ	5.10
Energy trainable λ	2.70
Mass λ=0.25	2.14
Train all λs	**2.05**

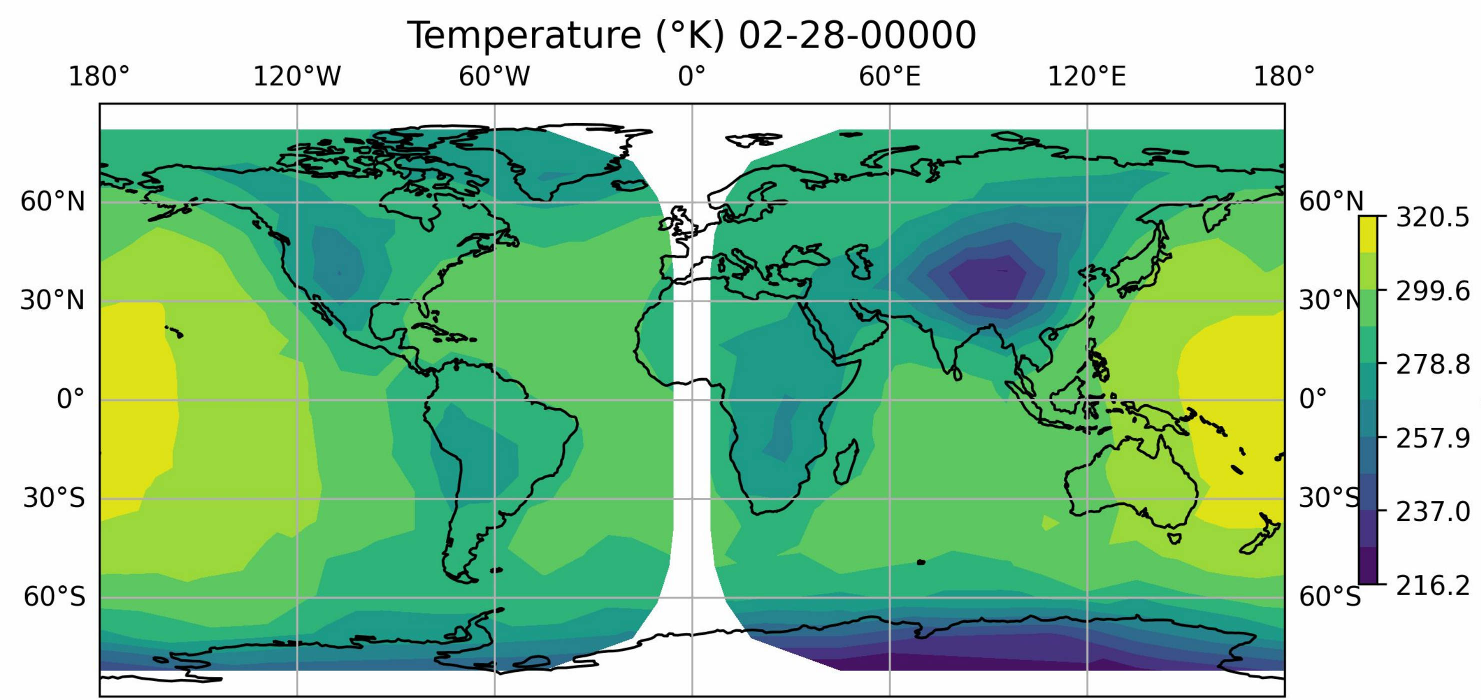
Loss and learnable scaling factor:

Both water conservation and non-negativity constraints learned to use a high scaling factor, greatly outweighing the data loss

Energy (radiation) scaling factor was brought down to nearly 0



Prediction over one full day (Feb 2):



Conclusions

Including trainable scaling factors for radiation, mass, and non-negativity loss landed the best performance, improving predictions over solely data loss

Performing experiments with physical losses in the ClimSim dataset opens a lot of questions:

- When do these losses perform well, and when do they actually hurt our model?

Data loss performs almost too well in our task (predicting output variables only 20 minutes in the future) → longer forecast window might be more interesting, where our physics-informed losses could take over

References

References

[1] Josh Cowls, Andreas Tsamados, Mariarosaria Taddeo, and Luciano Floridi. 2023. The AI gambit: leveraging artificial intelligence to combat climate change—opportunities, challenges, and recommendations. Ai & Society (2023), 1–25.

[2] Sungduk Yu, Walter Hannah, Liran Peng, Jerry Lin, Mohamed Aziz Bhouri, Ritwik Gupta, Björn Lütjens, Justus C. Will, Gunnar Behrens, Julius Busecke, Nora Loose, Charles Stern, Tom Beucler, Bryce Harrop, Benjamin Hillman, Andrea Jenney, Savannah L. Ferretti, Nana Liu, Animashree Anandkumar, Noah Brenowitz, Veronika Eyring, Nicholas Geneva, Pierre Gentine, Stephan Mandt, Jaideep Pathak, Akshay Subramaniam, Carl Vondrick, Rose Yu, Laure Zanna, Tian Zheng, Ryan Abernathey, Fiaz Ahmed, David Bader, Pierre Baldi, Elizabeth Barnes, Christopher Bretherton, Peter Caldwell, Wayne Chuang, Yilun Han, YU HUANG, Fernando Iglesias-Suarez, Sanket Jantre, Karthik Kashinath, Marat Khairoutdinov, Thorsten Kurth, Nicholas Lutsko, Po-Lun Ma, Griffin Mooers, J. David Neelin, David Randall, Sara Shamekh, Mark Taylor, Nathan Urban, Janni Yuval, Guang Zhang, and Mike Pritchard. 2023. ClimSim: A large multi-scale dataset for hybrid physics-ML climate emulation. In Advances in Neural Information Processing Systems, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (Eds.), Vol. 36. Curran Associates, Inc., 22070–22084.
https://proceedings.neurips.cc/paper_files/paper/2023/file/45fbcc01349292f5e059a0b8b02c8c3f-Paper-Datasets_and_Benchmarks.pd

Appendix

ClimSim baseline results:

	(Variables)	MLPv1	MLPv2	MLPv1-ne30	MLPv2-ne30
MAE	dT/dt	2.688	2.305	2.799	2.886
	dq/dt	4.503	4.030	4.231	4.068
	dq_l/dt	N/A	0.689	N/A	0.697
	dq_i/dt	N/A	0.384	N/A	0.330
	du/dt	N/A	1.34E-04	N/A	2.68E-04
	dv/dt	N/A	1.09E-04	N/A	2.66E-04
	NETSW	13.47	8.339	15.47	11.04
	FLWDS	5.118	4.134	5.318	4.891
	PRECSC	2.645	1.539	3.115	3.009
	PRECC	33.89	23.74	42.49	29.62
	SOLS	7.942	5.774	8.484	6.866
	SOLL	10.30	8.190	10.582	8.993
	SOLSD	4.587	3.230	5.056	4.360
	SOLLD	4.834	3.977	4.963	4.553
R2	dT/dt	0.590	0.663	0.626	0.536
	dq/dt	-	-	-	-
	dq_l/dt	N/A	-	N/A	-
	dq_i/dt	N/A	-	N/A	-
	du/dt	N/A	-	N/A	-
	dv/dt	N/A	-	N/A	-
	NETSW	0.982	0.993	0.977	0.988
	FLWDS	0.927	0.945	0.914	0.924
	PRECSC	-	-	-0.117	-0.117
	PRECC	-1.494	0.833	-0.115	-0.115
	SOLS	0.962	0.978	0.963	0.976
	SOLL	0.948	0.964	0.953	0.965
	SOLSD	0.955	0.976	0.950	0.965
	SOLLD	0.866	0.905	0.874	0.899
RMSE	dT/dt	4.437	3.756	5.199	4.958
	dq/dt	7.337	6.521	7.550	7.135
	dq_l/dt		1.192		1.489
	dq_i/dt		0.812		0.940
	du/dt		2.80E-04		6.45E-04
	dv/dt		2.25E-04		6.72E-04
	NETSW	26.95	17.24	30.48	21.18
	FLWDS	6.803	5.532	7.136	6.540
	PRECSC	4.656	2.955	7.791	7.509
	PRECC	73.16	53.47	119.8	83.22
	SOLS	17.39	12.84	18.51	14.74
	SOLL	21.96	17.89	22.71	19.27
	SOLSD	9.474	6.837	10.42	8.724
	SOLLD	10.14	8.486	10.62	9.526

Table 4: Similar to Table 2 in the main text but for comparing MAR, R2, and RMSE of different MLP models: MLP v1 (subset emulation) and the MLP v2 (full vector emulation) built with the low-resolution (ne4) and the high-resolution datasets (ne30). dq_l/dt, dq_i/dt, du/dt, and dv/dt correspond to the tendencies of state_q0002, state_q0003, state_u, and state_v, respectively, in Table SI1.