Atmospheric Physics-Guided Machine Learning

Presenter: Tom Beucler (UNIL)

M Pritchard (UCI), S Rasp (Clim. AI),
P Gentine (Columbia), I Ebert (CSU),
N Brenowitz (UW), J Yuval (MIT)…
Towards Data-Driven and Physically-Consistent Models of Atmospheric Convection

How to best combine ML & physical knowledge?
Atmospheric Convection = Atmospheric motion driven by air density differences
Motivation 1: Largest uncertainties in climate projections from clouds

Source: Zelinka et al. (2020), Meehl et al. (In Review), Gentine, Eyring & Beucler (2020)
Motivation 1: Largest uncertainties in climate projections from clouds

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

Motivation 1: Largest uncertainties in climate projections from clouds

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

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

ML of Subgrid-Scale Thermodynamics

Setup: Super-Parameterized climate model with prescribed surface temp. Year 1 for training (42M samples), Year 2 for validation/test.

Neural Network: 20 times faster

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

Temperature

Specific Humidity

Surface Enthalpy Fluxes

Radiative Fluxes

Subgrid Heating

Subgrid Moistening
Truth
Super-param. simulation

Prediction
NN (offline)

600hPa Convective Moistening (W/m²)
600hPa Convective Heating (W/m²)

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

Can we eliminate physics entirely?

Verification: 2018-01-05 00:00 Z
Forecast: 2017-12-10 00:00 Z + 642 h

Image Source: Weyn et al. (2020), See also: Rasp et al. (2020)
Can we eliminate physics entirely?

Maybe for meteorology
Not for climate

Problem 1: ML algorithms violate conservation laws
Problem 2: ML parametrization hard to interpret/trust
Problem 3: ML algorithms fail to generalize
Problem 1: Neural Nets typically violate conservation laws.
Problem 2: ML parametrizations are hard to interpret/trust

Time to Crash: 1.2 day

(a) Near-surface Convective Moistening
(b) Near-surface Convective Heating

See: Brenowitz, Beucler et al. (2020)
Problem 3: ML algorithms fail to generalize

Daily-mean Tropical prediction in reference climate

See: Beucler et al. (2019)
Problem 3: ML algorithms fail to generalize

Daily-mean Tropical prediction in reference climate

See: Beucler et al. (2019)
Problem 3: ML algorithms fail to generalize

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

See: Beucler et al. (2019)
Problem 1: ML algorithms violate conservation laws
Problem 2: ML parametrization hard to interpret/trust
Problem 3: ML algorithms fail to generalize

How can we design interpretable, physically-consistent & data-driven models of convection?

How to best combine ML & physical knowledge?
Physics-Guided ML: Add physical structure to restrict ML output to physically-plausible solutions

Reviews: Willard et al. (2020), Reichstein et al. (2019), Karpatne et al. (2017), Beucler et al. (2021)
Physics-Guided ML: Add physical structure to restrict ML output to physically-plausible solutions

See: Schneider et al. (2017), Reichstein et al. (2019), Camps-Vall et al. (2018), Image Source: CliMA, Caltech
Physics-Guided ML: Add physical structure to restrict ML output to physically-plausible solutions

Bias Correction of Physical Model

Learn Parameters of Physical Model

Physics-Guided ML: Add physical structure to restrict ML output to physically-plausible solutions

Physics-Constrained Loss or Architecture  Bias Correction of Physical Model  Learn Parameters of Physical Model

**Problem 1:** Neural Nets typically violate conservation laws

- Physics-Constrained Loss or Architecture
- Bias Correction of Physical Model
- Learn Parameters of Physical Model

Physics-Constrained Loss Function

Idea: Introduce a penalty for violating conservation (\(\sim\) Lagrange multiplier):

\[
\text{Loss} = \alpha (\text{Squared Residual from conservation laws}) + (1 - \alpha) (\text{Mean squared error})
\]

See: Beucler et al. (2021)
Physics-Constrained Architecture

Idea: Introduce a penalty for violating conservation (≈ Lagrange multiplier):

\[
\text{Loss} = \alpha (\text{Squared Residual from conservation laws}) + (1 - \alpha) (\text{Mean squared error})
\]

Constraint layers to enforce conservation laws to within machine precision!

See: Beucler et al. (2021)
Problem 1: Neural Nets typically violate conservation laws

We can enforce conservation laws in NNs
Conservation of mass, energy, and radiation

See: Beucler et al. (2021)
Problem 2: For climate modeling, we need trustworthy/interpretable parametrizations.

*awkward silence*

Why did you predict 42 for this data point?

Source: Interpretable Machine Learning, C. Molnar (2019)
Problem 2: ML parametrizations are hard to interpret/trust

Idea: Tailor 2 NN interpretability methods to parameterization convection

Partial Dependence Plots confirm that at fixed l.t. stability, mid-tropospheric moisture fuels convection

\[ QM = 20.0 \text{kg/m}^2 \]

See: Brenowitz, Beucler et al. (2020)
Partial Dependence Plots confirm that at fixed l.t. stability, mid-tropospheric moisture fuels convection

QM = 30.5 kg/m²
Partial Dependence Plots confirm that at fixed l.t. stability, mid-tropospheric moisture fuels convection

\[ QM = 34.7 \text{ kg/m}^2 \]

See: Brenowitz, Beucler et al. (2020)
Jacobian calculated via automatic differentiation helps interpret and stabilize parameterization.

Spurious unstable propagating modes couple to linearized gravity wave dynamics.

See: Kuang (2018, 2007), Herman and Kuang (2013), Beucler et al. (2018), Brenowitz, Beucler et al. (2020)
Problem 2: ML parametrizations are hard to interpret/trust

We can tailor interpretability methods
Partial Dependence Plots + Gradients
Also applies to Attribution Maps

See: Brenowitz, Beucler et al. (2020)
Problem 3: ML algorithms fail to generalize

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

Convective moistening (W m⁻²)
Idea: Break the model even more!
Generalization Experiment: Uniform +8K warming

Training and Validation on cold aquaplanet simulation

Test on warm aquaplanet simulation

Images: Rashevskyi Viacheslav, Sebastien Decoret
Generalization Experiment: Uniform +8K warming

Surface Temperature (K, fixed)

Latitudinal distribution of surface temperature showing a peak and trough, with a dashed line indicating a baseline temperature.
Generalization Experiment: Uniform +8K warming

Surface Temperature (K, fixed)

- Cold
- Warm (+8K)
Generalization Experiment: Uniform +8K warming

**Surface Temperature (K, fixed)**

- Cold
- Warm (+8K)
Generalization Experiment: Uniform +8K warming

Trained on cold climate

Tested out-of-sample

- Cold
- Warm (+8K)
Problem 3: NNs fail to generalize to unseen climates

Daily-mean Tropical prediction in cold climate

![Graph showing daily-mean Tropical prediction in cold climate](image)

- **Y-axis:** Pressure (hPa)
- **X-axis 1:** Moistening (W m⁻²)
- **X-axis 2:** Heating (W m⁻²)

- **Graph:**
  - **Line:** Truth

Problem 3: NNs fail to generalize to unseen climates

Daily-mean Tropical prediction in cold climate

- Brute Force
- Truth

Moistening (W m$^{-2}$)

Heating (W m$^{-2}$)
Daily-mean Tropical prediction in warm climate

![Graph showing moisture and heating variations over pressure range.](image)
Daily-mean Tropical prediction in warm climate
Physically rescale the data to convert extrapolation into interpolation.

**Specific humidity \((p)\)**
- Temperature \((p)\)
- Surface Pressure
- Solar Insolation
- Latent Heat Flux
- Sensible Heat Flux

**Subgrid moistening \((p)\)**
- Subgrid heating \((p)\)
- Radiative fluxes

---

**Brute Force: Not Climate-Invariant**
Physically rescale the data to convert extrapolation into interpolation

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

\[
\begin{bmatrix}
\text{Specific humidity (}\tilde{p}\text{)} \\
\text{Temperature (}\tilde{p}\text{)} \\
\text{Surface Pressure} \\
\text{Solar Insolation} \\
\text{Latent Heat Flux} \\
\text{Sensible Heat Flux}
\end{bmatrix}
\mapsto
\begin{bmatrix}
\text{Subgrid moistening (}\tilde{p}\text{)} \\
\text{Subgrid heating (}\tilde{p}\text{)} \\
\text{Radiative fluxes}
\end{bmatrix}
\]
Physically rescale the data to convert extrapolation into interpolation

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

How to choose the physical rescaling?
Specific humidity ($z$) $\rightarrow$ Relative humidity ($\tilde{z}$)

Generalization improves dramatically!
Specific humidity ($z$) $\rightarrow$ Relative humidity ($z$)

![Graph showing specific humidity and relative humidity](image)

- **X-axis:** Mean Sq Err Moistening [$10^3$ W$^2$ m$^{-4}$]
- **Y-axis:** Pressure [hPa]
- **Legend:**
  - Cold
  - Warm

![Graph showing PDF of specific humidity](image)

- **X-axis:** Relative humidity
- **Y-axis:** Specific humidity (kg kg$^{-1}$)
- **Legend:**
  - Cold
  - Warm
Observations suggest a strong relationship between buoyancy & moist convection across scales

\[
\text{Buoyancy} (z) \overset{\text{def}}{=} g \times \frac{\text{Temp}_\text{parcel} - \text{Temp}(z)}{\text{Temp}(z)}
\]

Latent Heat Flux $\rightarrow \frac{L_H}{L_v(Near-surf\ q\ disequilibrium)}$

![Graph showing pressure vs. mean square error in moistening and normalized LHF](image)
Climate-Invariant NNs generalization error close to NN trained in warm climate
Problem 3: Physically Rescaling Inputs allows NNs to generalize from cold to warm climate.

See: Beucler et al. (Under review)
Physically-Rescaled Neural Networks Generalize Better Across Climates in Earth-like configurations

Without Rescaling

With Physical Rescaling

Near-Surface Subgrid Heating
Physically-Rescaled Neural Networks Generalize Better Across Climates in *Earth-like* configurations

**Without Rescaling**

**With Physical Rescaling**

**Mid-Tropospheric** Subgrid Heating
Outlook 1: Extracting Physics from Data

Incorporate physical knowledge into ML

Use ML to extract physical knowledge from data

See: Barnes & Ebert-Uphoff (2020)
Climate-invariant NNs more local than Brute-Force NNs

- Subgrid Moistening
- Subgrid Heating

<table>
<thead>
<tr>
<th>Buoyancy</th>
<th>Relative Humidity</th>
<th>Temperature</th>
<th>Specific Humidity</th>
</tr>
</thead>
</table>

SHAP Feature Importance Matrix [W m⁻²]
Extracting convective regimes from cloud-resolving data

Cloud Types Determined By VAE

Source: Mooers, Tuyls, Mandt, Pritchard, & Beucler (2020)
Outlook 2: Transferring knowledge across climates/geographies/models/observations

![Graph showing correlation skill over forecast lead (months) with and without transfer learning.]

Adapted from: Ham et al. (2019), See: Rasp & Thuerey (2021)
**Problem:** Observations of convection are sparse

Images: NASA, NOAA

- **Specific humidity (kg/kg)**
- **Temperature (K)**

Global Observing System vs. Global Climate Model
Problem: Observations of convection are sparse

Global Observing System

Moistening tendency (W/m²)

Heating tendency (W/m²)

Images: NASA, NOAA
Problem: Observations of convection are sparse

Field Campaigns

Moistening tendency (W/m²)

Heating tendency (W/m²)

Global Climate Model

Images: EUREC⁴A, NOAA
Climate-Invariant NNs learn transferable mappings
Climate-Invariant NNs learn transferable mappings

Log of Mean-Squared Error tested in

Clim. Inv. NN trained in
Outlook 2: Physics-informed ML may assist the data assimilation of sparse observations

Moistening tendency (W/m²)
Heating tendency (W/m²)

Images: EUREC4A, NOAA
Atmospheric Physics can Help Machine Learning

1) Enforce physical constraints approx. (loss) or exactly (architecture)

2) Tailor ML interpretability methods for emulation of physical processes

3) Help NNs generalize by physically rescaling inputs & outputs

4) Rescaled ML learns more general mappings/facilitates transfer learning

Images: NASA, NOAA
Thank you

www.unil.ch/dawn
tom.beucler@unil.ch
Bonus Slides
Summary
Neural Network = Non-linear regression tool

High-dimensional Inputs $x$ \quad \rightarrow \quad High-dimensional Outputs $y$

$x \mapsto y$

\[ \text{min } \text{Loss function} (y_{\text{Predicted}}, y_{\text{Truth}}) \]

*Image source: Kathuria (Paperspace)*
Linear Response Function

Stability Diagram (Offline)
Climate-Invariant nets: Rescale inputs/outputs so that (extrapolation) $\rightarrow$ (interpolation)
Climate-Invariant neural networks:

- Learn more general mappings
- Facilitate transfer learning
Soft Constraints (Loss) vs Hard Constraints (Architecture)

**Loss:** Introduce a penalty for violating conservation (\( \sim \) Lagrange multiplier):

\[
\text{Loss} = \alpha \left( \text{Squared Residual from conservation laws} \right) + \left( 1 - \alpha \right) \left( \text{Mean squared error} \right)
\]

**Architecture:** Constraints layers to enforce conservation laws to machine precision

**Diagram:**

Inputs:
\[
\begin{bmatrix}
x_1 \\
\vdots \\
x_m
\end{bmatrix}
\]

Standard NN (Optimizable)

Direct Outputs:
\[
\begin{bmatrix}
y_1 \\
\vdots \\
y_{p-n}
\end{bmatrix}
\]
Loss: Trade-off between **physical constraints** and **performance**

\[
\text{Loss} = \alpha (\text{Squared Residual from conservation laws}) + (1 - \alpha) (\text{Mean squared error})
\]
Mean-Squared Error (skill) for unconstrained network

Loss: Trade-off between physical constraints and performance

Loss = $\alpha (\text{Squared Residual from conservation laws}) + (1 - \alpha) \text{ (Mean squared error)}$
Mean-Squared Error (skill) for multiple linear regression

Loss: Trade-off between physical constraints and performance

\[
\text{Loss} = \alpha \left( \text{Squared Residual from conservation laws} \right) + (1 - \alpha) \left( \text{Mean squared error} \right)
\]
Loss: Trade-off between physical constraints and performance

\[ \text{Loss} = \alpha (\text{Squared Residual from conservation laws}) + (1 - \alpha) (\text{Mean squared error}) \]
Loss: Trade-off between physical constraints and performance

Loss = \alpha(Squared Residual from conservation laws) + (1 - \alpha)(Mean squared error)
Performs worse and worse

Loss: Trade-off between **physical constraints** and **performance**

\[
\text{Loss} = \alpha (\text{Squared Residual from conservation laws}) + (1 - \alpha) (\text{Mean squared error})
\]
Loss: Trade-off between physical constraints and performance

Architecture: Constraints enforced & competitive performance

See: Beucler et al. (2019)
Problem 2: Even when physically constrained, NNs fail to generalize
Algorithms: Custom Data Generators/Layers

Inputs
\[
\begin{bmatrix}
q_v,1 \\
q_v,2 \\
\vdots \\
SHF \\
LHF
\end{bmatrix}
\]

Outputs
\[
\begin{bmatrix}
\dot{q}_v,1 \\
\dot{q}_v,2 \\
\vdots \\
P \\
P_i
\end{bmatrix}
\]

- Only one training/validation/test data despite multiple rescalings
- Test different rescalings quickly using multi-linear/logistic regressions
- Keep the rescalings that yield the best generalization
Start with clear link to climate impact/remote sensing

Link = Transfer Learning
Why Integrate Physics into ML/Stat Algorithms?

• Physical consistency (definitions, conservation laws...)
• Ability to generalize outside of the training set
• Interpretability
• Stability
• Data limitations

Reviews: Willard et al. (2020), Reichstein et al. (2019), Karpatne et al. (2017), Beucler et al. (2021)
Aquaplanet (SPCAM3)  Earth-like (SPCAM5)  Hypohydrostatic (SAM)

(a) Brute Force

(-4K)  (-4K)  (-4K)

(-4K)  (+4K)  (+4K)

(b) Climate-Invariant

(-4K)  (-4K)  (-4K)

(-4K)  (+4K)  (+4K)

500-hPa Subgrid Heating

Coefficient of determination \( R^2 \)
Near-surface heating

Aquaplanet (SPCAM3)  Earth-like (SPCAM5)  Hypohydrostatic (SAM)

(a) Brute Force

(-4K)  (-4K)  (-4K)

(+4K)  (+4K)  (+0K)

(b) Climate-Invariant

(-4K)  (-4K)  (-4K)

(+4K)  (+4K)  (+0K)

Coefficient of determination $R^2$