Distributionally robust Multi-Model Ensemble Analysis

Trevor Harris

September 4, 2023

Texas A&M University
University of Illinois at Urbana-Champaign

Research interests

- Climate science
  - Long range climate forecasting and model integration with machine learning
  - Climate model validation and assessment
  - Detection and attribution of climate change
  - Model calibration and parameter estimation
- Public health
  - Vector borne disease modeling with graph neural networks
  - Causal analysis, Granger causality, and interrupted time series with deep neural networks
  - Effects of extreme weather on vector borne disease
- Deep learning
  - Uncertainty quantification with Bayesian and conformal methods
  - Robust predictions and out of distribution generalization
  - Semi-supervised learning and small data problems

Multi-model Ensembles

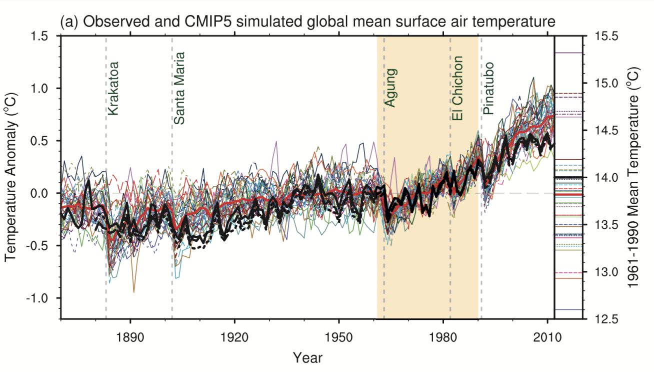


Figure 1: Gobal mean predictions for each CMIP5 model (colored lines), the model mean (red) and observations (black). Different models yield different predictions.

- Multi-model ensemble analysis – how to combine models to best resemble the actual climate?

Multi-model Ensemble Analysis

- Climate models produce spatio-temporal output (discretized to a grid). Combine directly?

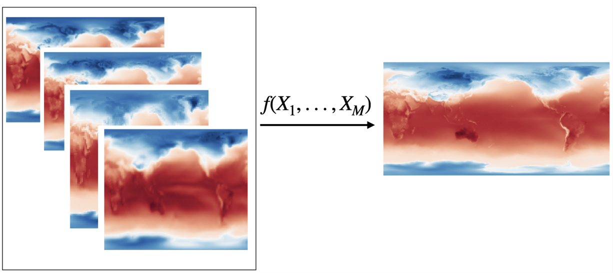


Figure 2: Goal: combine multiple climate fields into a single estimate

- More informative but much more difficult than averaging global means
  - Resize to common grid - introduces bias and lose information
  - Consider correlations between models and observations?
  - Spatially varying weights? Tons of parameters?

Previous Work

- There are many methods for constructing $f : X \mapsto Y$
- Model integration – combining multiple climate projections into a unified projection
  - **Ensemble averaging** – democratic and weighted (Giorgi and Mearns, 2002, 2003; Flato et al., 2014; Abramowitz et al., 2019)
  - **Bayesian methods** (Rougier et al., 2013; Sansom et al., 2017; Bowman et al., 2018)
  - **Regression** (Räisänen et al., 2010; Bracegirdle and Stephenson, 2012) and **Machine Learning** methods (Ghafarianzadeh and Monteleoni, 2013)
- Gaussian process regression (Harris et al., 2023)
  - Climate models are used to predict observational data
  - The predictions constitute an “integration” or “analysis” of the climate models

Distribution Shift

- Most methods are not robust to **distribution shift**.
- Distribution shift occurs when

$$P_{tr}(X, Y) \neq P_{te}(X, Y)$$

  i.e the joint distribution of the predictors X and targets Y is different in the train and test sets.
- If a model is not **robust** or **invariant** to distribution shift, then its loss will generally be higher on test.

$$\mathbb{E}_{(X,Y)\sim P_{tr}}[\ell(f,(X,Y)] \neq \mathbb{E}_{(X,Y)\sim P_{te}}[\ell(f,(X,Y)]$$

- Separate concept from overfitting

Impacts to prediction

- This can have a significant impact on the predictive skill.
- Most methods show increasing error rates over time
- Some models are more robust than others

(↓) Mean Squared Error (MSE) - T2M

Model	2030	2040	2050	2060	2070	2080	2090	2100
NN-GPR	1.91 (0.06)	1.97 (0.06)	2.10 (0.07)	2.27 (0.08)	2.37 (0.09)	2.53 (0.11)	2.68 (0.11)	2.84 (0.12)
LM	2.29 (0.11)	2.28 (0.10)	2.38 (0.12)	2.51 (0.13)	2.54 (0.14)	2.57 (0.17)	2.62 (0.17)	2.71 (0.19)
WEA	3.29 (0.22)	3.27 (0.20)	3.40 (0.23)	3.54 (0.25)	3.54 (0.25)	3.60 (0.27)	3.62 (0.28)	3.67 (0.28)
EA	5.98 (0.53)	5.87 (0.50)	5.96 (0.49)	6.04 (0.45)	6.00 (0.45)	6.03 (0.43)	5.97 (0.43)	5.99 (0.42)
GPSE	1.91 (0.06)	2.01 (0.06)	2.26 (0.08)	2.57 (0.09)	2.85 (0.12)	3.23 (0.13)	3.60 (0.15)	3.96 (0.17)
GPEX	1.89 (0.06)	1.97 (0.06)	2.19 (0.07)	2.44 (0.08)	2.65 (0.10)	2.90 (0.11)	3.16 (0.11)	3.40 (0.13)
CNN	2.78 (0.15)	2.75 (0.14)	2.79 (0.17)	2.95 (0.18)	2.94 (0.18)	2.97 (0.22)	3.01 (0.23)	3.08 (0.24)
DELT	3.07 (0.22)	3.05 (0.21)	3.17 (0.23)	3.31 (0.24)	3.30 (0.23)	3.36 (0.25)	3.40 (0.25)	3.46 (0.26)

Figure 3: Decadal MSEs for 8 different model integration methods. Results are averages (std. dev) over 16 different climate model runs.

Impacts to UQ

– Also significantly impacts the uncertainty quantification of these methods

– Most methods show increasing error rates over time

– Some models are more robust than others

(↓) Continuous Ranked Probability Score (CRPS) - T2M

Model	2030	2040	2050	2060	2070	2080	2090	2100
NN-GPR	0.73 (0.01)	0.74 (0.01)	0.76 (0.01)	0.79 (0.01)	0.81 (0.01)	0.83 (0.02)	0.86 (0.02)	0.88 (0.02)
LM	0.68 (0.02)	0.69 (0.02)	0.69 (0.02)	0.72 (0.02)	0.73 (0.02)	0.74 (0.02)	0.74 (0.02)	0.76 (0.02)
WEA	1.15 (0.05)	1.15 (0.05)	1.16 (0.05)	1.18 (0.05)	1.17 (0.04)	1.18 (0.04)	1.18 (0.04)	1.18 (0.04)
EA	1.15 (0.05)	1.15 (0.05)	1.16 (0.05)	1.18 (0.05)	1.17 (0.04)	1.18 (0.04)	1.18 (0.04)	1.18 (0.04)
GPSE	0.73 (0.01)	0.74 (0.01)	0.77 (0.01)	0.81 (0.01)	0.84 (0.01)	0.88 (0.02)	0.92 (0.02)	0.94 (0.02)
GPEX	0.73 (0.01)	0.75 (0.01)	0.78 (0.01)	0.82 (0.01)	0.86 (0.02)	0.92 (0.02)	0.97 (0.02)	1.02 (0.02)
DELT	3.87 (0.04)	3.93 (0.04)	4.00 (0.04)	4.06 (0.04)	4.12 (0.04)	4.16 (0.04)	4.21 (0.04)	4.24 (0.04)

Figure 4: Decadal CRPS for 8 different model integration methods. Results are averages (std. dev) over 16 different climate model runs.

Okay and?

- We expect prediction error and predictive distributions to deteriorate the further (more dissimilar) the test set is from the training set.
  - I.e. the bigger the "gap" between $P_{tr}(X, Y)$ and $P_{te}(X, Y)$, the worse a model will perform
  - There is no way to make a good model that is completely immune to this distribution shift problem

- But we can try to minimize how fast it becomes a problem.

- **Goal**: A model who's error rates increase **very slowly** over time
  - Increased forecasting skill improves long term model integration
  - Increased UQ skill narrows long term model projection uncertainty

Proposal

Three stage model: downsampling, prediction and upsampling

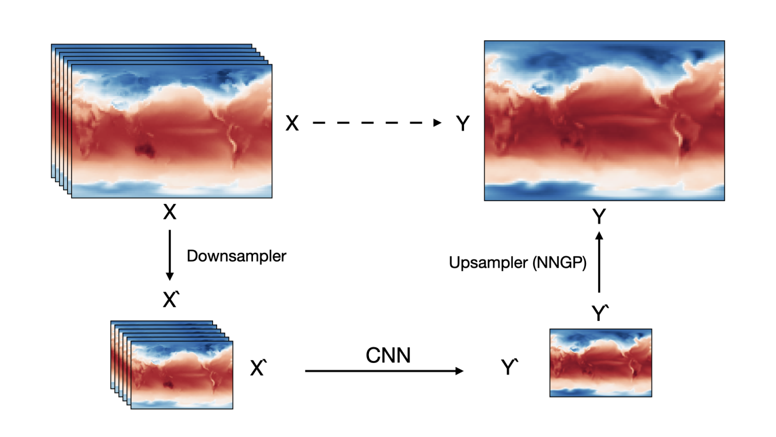


Figure 5: Model schematic showing how an ensemble of climate models is downsampled, used to predict a downsampled target, then finally re-upsampled to the target resolution.

Proposal

Putting it all together. Our overall goal is to learn a map $f : X \mapsto Y$.
We break this down into three stages as $f(X) = g \circ h \circ l(x)$

1. Downsample $l : X \mapsto X'$ (bicubic)
2. Forecast $h : X' \mapsto Y'$ (CNN)
3. Upsample $g : Y' \mapsto Y$ (nngp)

– Downsampler is not trained (image resizing).

– Component 2 (forecasting) and 3 (upsampling) are trained separately.

– We call our model “dCNN” for downscaled CNN

Why decompose?

- The GP model we use in our previous work simultaneously predicted a target field given an ensemble of climate models.
    - Automatically upscaled the inputs to match the dimension of the output.
    - Fairly sensitive to distribution shift (but better than other GPs!)
- Empirical testing shows that the CNN is relatively **robust** to distributional shifts that are less than (or equal) to what our data exhibits. I.e. a CNN (apparently) mitigates the distributional shift issue. (we're not sure why)
- However, the CNN struggles to upscale (blurry), which was an area that our GP model excelled at.

Deep Kernel Learning

- Feeding the inputs through a neural network then through a GP is known as **Deep Kernel Learning**
- Deep Kernel Learning is a powerful technique for learning complex kernel

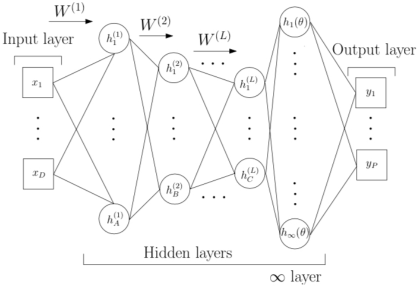
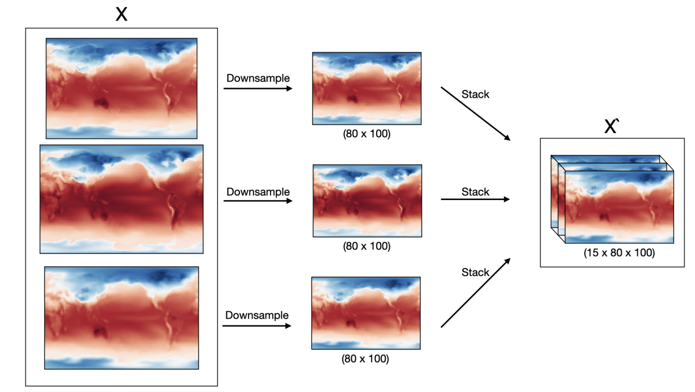


Figure 1: Deep Kernel Learning: A Gaussian process with a deep kernel maps D dimensional inputs $\mathbf{x}$ through L parametric hidden layers followed by a hidden layer with an infinite number of basis functions, with base kernel hyperparameters $\boldsymbol{\theta}$. Overall, a Gaussian process with a deep kernel produces a probabilistic mapping with an infinite number of adaptive basis functions parametrized by $\boldsymbol{\gamma} = \{\mathbf{w}, \boldsymbol{\theta}\}$. All parameters $\boldsymbol{\gamma}$ are learned through the marginal likelihood of the Gaussian process.

Bicubic Downsampling

- For downsampling we use a bicubic interpolator to "resize" each climate field from its native resolution to an 80x100 pixel image. Downscaler is not trained
- Each climate model is observed on its own native resolution, so this is necessary to create a stack of models anyways



CNN forecasting

- For testing purposes we use a small CNN (32 x 5 x 32 x 32 x 32 x 1) with relu activations. Trained with adam on minibatches.
- Qualitative empirical findings
  - Minibatching is essential for generalization (batch size 32)
  - The bottleneck layer (5 channels) is necessary for generalization
  - relus improves generalization over tanh, sigmoid, leakyrelus
  - Further regularization (weight decay and dropout) does not seem to matter much (but might be helpful for getting the average error rate lower)
- CNN converts our stack of 15 models (treated as channels), X', into a single (1 channel) image, $\hat{Y}'$.
- Minimize MSE loss $||Y' - \hat{Y}'||_2$

Gaussian Process Upsampling

- For the GP we use a neural network GP (NNGP) kernel. This was shown in our previous work to be more robust than standard exponential and squared exp kernels.
- This time we learn a GP to map $g : Y \mapsto Y'$

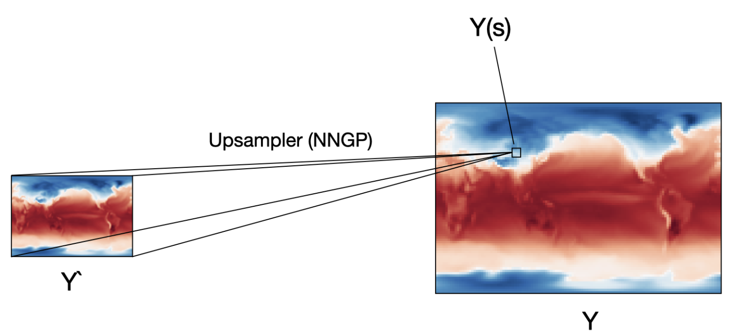


Figure 6: NNGP upscales climate models by using the downscaled model to predict each pixel of the upscaled model separately.

Training

- Dataset consists of monthly aggregate 2-meter surface temperature (T2M) as output from 16 different climate models (one output from each model).
  - We hold one model out as the "target" and use the remaining 15 models as predictors.
  - Repeat for each model as target. 16 "perfect model" experiments in total.

- For each experiment...
  - Train on historical period (1979 - 2015) match reanalysis data availability
  - Test on future simulations (2015-2100) based on SSP245
  - SSP245 – Shared Socioeconomic Pathway 2 with Representative Concentration Pathway (RCP) 4.5 (medium plausible scenario)

Training

For each model experiment, training occurs in two stages.

- Stage 1 - CNN
  - We first use the downsampler to convert all 15 predictor models X, into a tensor X'.
  - We also use the downsampler to conver the held out model Y into a low res field Y'.
  - Train the CNN to minimize the MSE $||Y' - \hat{Y}'||_2$
- Stage 2 - Upscaler
  - We then train the upscaler (NNGP) to predict Y from the low res version Y'
  - this is performed completely independently from the CNN (for now)

Experiments

- Test methods ability to accurately predict future climate under many "perfect model" scenarios
  - Given 16 global climate models. Treat one model as the "truth". Treat other 15 as multi-model ensemble.
  - Cycle through / repeat for all models as the "truth".
- We consider two separate comparisons
  - Evaluate the test MSE of the dCNN vs an NNGP model trained to predict Y' from X' (low res forecasting)
  - Evaluate the test MSE of the dCNN against an NNGP trained to directly predict Y from X (hi res forecasting)

Results - Downsampled

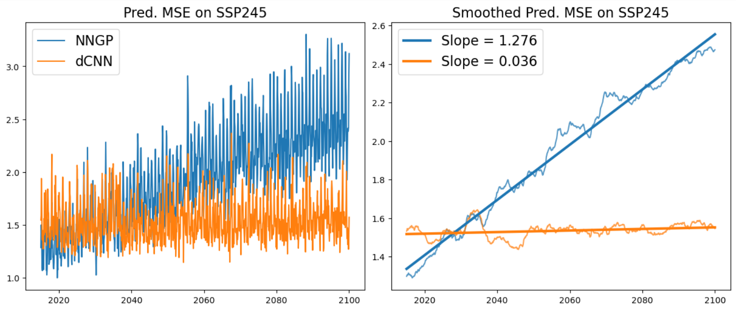


Figure 7: dCNN vs NNGP prediction MSE targeting a single climate model

– NNGP has a lower starting error, but is relative high at the end

– dCNN has almost an entirely flat error rate over the test set. CNN is evidently robust to the distribution shift present in the data.

– Architecture improvements might bring CNN error rate down (ongoing work)

Results - Downsampled

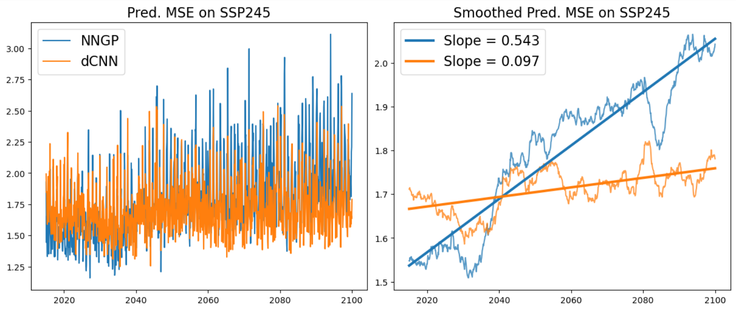


Figure 8: dCNN vs NNGP prediction MSE targeting a different climate model

– Overall performance can vary depending on the target

– Still shows improvements in the error slope over NNGP

– Architecture improvements might bring CNN error rate down (ongoing work)

Results - Downsampled

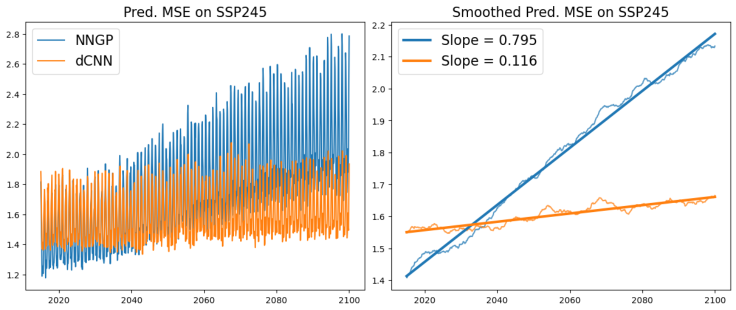


Figure 9: Average dCNN vs NNGP prediction MSE across all model runs. Average error rates are comparable but the slope of the dCNN is much lower.

– Average error rates over all time tend to be comparable

– Lower dCNN error rates are possible with architecture improvements in the CNN. (Not true for NNGP)

Results - Upsampled

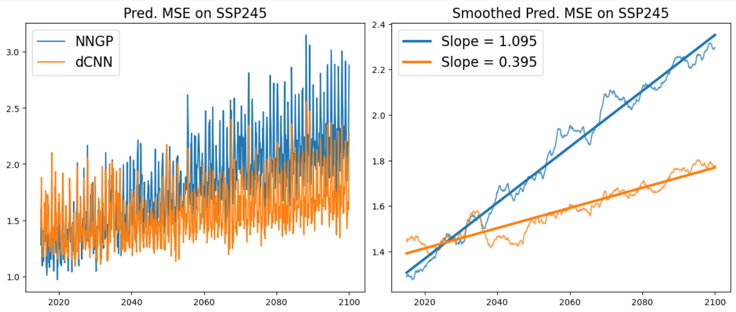


Figure 10: dCNN vs NNGP prediction MSE targeting a single climate model

– Upscale the dCNN predictions v.s. a direct NNGP approach, Error rates are much lower, but show an upward trend now.

– Conclusion: The NNGP upscaler is responsible for the decreased performance / weakness to distribution shift (look to replace?)

Quantifying uncertainty

- The NNGP approach has an inbuilt mechanism for quantifying uncertainty via the posterior predictive distribution

- Unfortunately in our case, in order to make things scalable, we assumed the variance is shared at every spatial location.
  - I.e. variance is constant over the spatial output domain (bad approximation).
  - Overestimates variance in low variability regions, underestimates in high variability regions.

- Our new approach, involving downsampling, a CNN, and upsampling with GPs seems hopeless for UQ

Functional Conformal Inference

- Conformal inference is a framework for constructing exact prediction intervals in finite samples.
- The only requirement is exchangeability (and, in practice, enough to data to sample split)
- That is, given a level α and a new input X conformal inference constructs a set $C_\alpha(X)$ such that

$$P(Y \in C_\alpha(X)) \geq 1 - \alpha$$

and in many cases

$$P(Y \in C_\alpha(X)) < 1 - \alpha + 1/(1 + n)$$

Proposal

A split conformal approach for black box regression with high dimensional targets

1. Partition our original training dataset $D = \{(X_i, Y_i)\}_{i=1}^n$ into
$$D_{train} = \{(X_i, Y_i)\}_{i=1}^m$$
$$D_{val} = \{(X_i, Y_i)\}_{i=m+1}^n$$
2. Train the dCNN model f on D_{train}
3. Compute the residual fields $R_i = Y_i - \hat{Y}_i$ on D_{val}
4. Find the set of the $(1-\alpha)\%$ set of **most central** residual fields R_i
5. We predict each $Y_j \in D_{test}$ with the set $\{\hat{Y}_j + R_i\}_{i=m+1}^n$

As long as $R_i = Y_i - \hat{Y}_i$ on D_{val} and $R_j = Y_j - \hat{Y}_j$ on D_{test} are exchangeable, the $(1-\alpha)\%$ central region estimated on D_{val} will also have $(1-\alpha)\%$ coverage on D_{test}.

Results

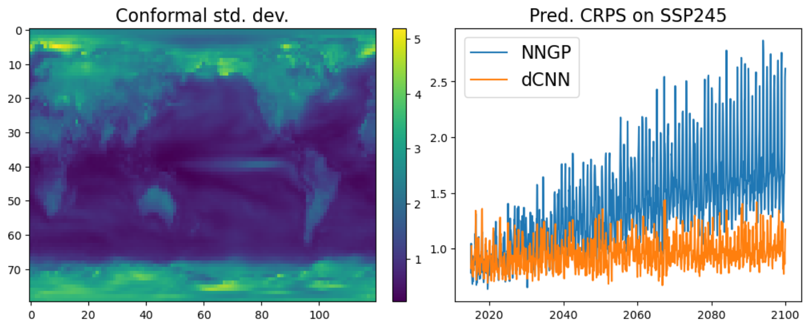


Figure 11: dCNN vs NNGP prediction CRPS targeting a single climate model

– Continuous Ranked Probability Score (CRPS) measures the quality of ensemble forecasts. Lower CRPS represents better UQ.

– As a consequence of mitigating distribution shift, our conformal based prediction sets have much better UQ

Conclusion

- Distribution shift has to be considered when applying models to future climate data
- GP models (like NNGP) have strong performance when there is little distribution shift. Degrade quickly with increasing distribution shift.
  - Modifying the architecture of the NN does little to change things.
- CNN based models are (evidently) more robust to distribution shift than GP models (for this problem), but require more effort to train
- More work is needed to improve the overall error rates of the CNN based approach
  - Bigger CNNs with modern tricks and data augmentation approaches
  - Semi-supervised learning approaches and invariance learning
  - Replace the CNN with an NNGP using a CNN kernel?
- UQ still under development!