Likelihood-Weighted Active Selection of Training Data for Improved Prediction of the Statistics of Extreme Weather Events*

Bianca Champenois† and Themistoklis P. Sapsis†

Abstract. As a result of climate change, extreme weather events have increased in severity and frequency, making the rapid modeling of potential climate scenarios all the more essential for future resource management and planning. However, the broad range of dynamically relevant spatiotemporal scales in the atmosphere makes direct numerical simulations computationally expensive and simplified reduced-order approaches less accurate. Scientific machine learning (ML) methods are a promising alternative, but given the inherent limited representation of extreme events, comprehensive or well-specified training data sets are necessary for model generalizability. To avoid time-consuming learning brought about by large data sets, we use a model-agnostic active learning approach to sequentially select an optimal subset of the most valuable data points for model training. Points are iteratively scored via a likelihood-weighted uncertainty sampling acquisition function which prioritizes points that reduce model uncertainty and improve prediction in the tails of the distribution, i.e. most relevant to the dynamics of extreme events. We first validate the method on a well-studied problem, quantifying the maximum wave magnitude statistics in a synthetic turbulent system. Then, we apply the method to a real-world problem, learning a debiasing operator for coarse-resolution climate simulations. In both cases, the likelihood-weighted active data selection algorithm most accurately reproduces the extreme event statistics using a fraction of the original data points. Looking forward, the approach is useful for improved environmental sampling schemes, and can be used as a compression algorithm that preserves information associated with extreme events in vast data sets.

Key words. active learning, extreme events, climate modeling, dimensionality reduction, weather prediction, machine learning

MSC codes. 86A08, 86-08, 62G32, 60G70, 62L05

1. Introduction.

Climate change is increasing the frequency and severity of extreme weather events such as week-long heatwaves and major rainfall episodes [21, 15]. These events are leading to significant damage to critical infrastructure and numerous premature deaths, and most extremes are occurring in low latitude tropical regions with high population density [46]. To better prepare for and mitigate these catastrophic events, some of which are breaking records by three or more standard deviations, there is a need for high-resolution models that explore the outcomes of different possible greenhouse gas emission scenarios [44, 4, 14, 53]. Historically, researchers have used numerical solvers based on physical equations to emulate climate systems [54, 30, 31, 57, 13, 51, 51, 17]. However, the dynamics can be highly turbulent and involve spatial resolutions ranging from millimeters on the Kolmogorov dissipation scale to tens of thousands of kilometers on the global scale. Numerical solvers require significant computational resources, extensive parameter tuning, and complicated closure terms.

Machine Learning (ML) models provide a useful alternative for traditional computationally

*Submitted to the editors September 3, 2024.

Funding: The work was funded through the AFOSR Award FA9550-23-1-0517. BC was supported through the National Science Foundation Graduate Research Fellowship (Grant No. 2141064).

†Department of Mechanical Engineering and Center for Ocean Engineering, Massachusetts Institute of Technology, Cambridge, MA (bchamp@mit.edu, sapsis@mit.edu)

expensive and complex numerical solvers. Progress in the speed and capability of computers, combined with new ML architectures and algorithms, has improved our ability to create models of highly nonlinear and high-dimensional systems. As a result, significant research has been done on leveraging ML for climate and weather modeling [42, 40, 43, 24, 50, 33, 5, 9, 3, 10, 41, 25]. However, ML remains challenging and expensive when applied to applications for which the available data sets are large and high-dimensional. In many problems, not all points carry the same *value* of information, so it can be inefficient or ineffective to use the entire data set. For some physical systems, certain dynamical mechanisms might be represented by an imbalanced number of samples. Namely, in a system such as the climate, extreme weather events, events in the tails of the probability density function (PDF), are an important example of such dynamics [48]. To achieve adequate representation of these events and understand their relationship to the system, it is often necessary to collect large amounts of data consisting of nearly repetitive, and thus unnecessary, points. These data sets become even larger for problems in high dimensions. Standard neural network (NN) models — typically trained with mean squared error (MSE) — give emphasis to regions of the domain where most points exist, so predictions are worse for phenomena in the tails. This discrepancy is often manifested through slow convergence and bad generalizability properties of ML models with respect to observables that highlight the statistics of extremes [36]. Therefore, identifying a subset of data points most relevant to the dynamics of extreme weather events can reduce model training time while more accurately representing the distribution of the original data.

To overcome the challenges associated with training an ML model given a large data set, we present an adaptation of the active learning framework for effective training data selection. Our active selection framework, introduced in Section 2, is well-suited for systems with extreme events because it quantifies the value of data using a likelihood-weighted uncertainty sampling acquisition (scoring) function [47, 6, 49]. One requirement of the acquisition function is knowledge of epistemic uncertainty, so we provide an overview of probabilistic ML architecture with uncertainty quantification (UQ) capabilities in Section 2.3. In Section 2.4, we explain how to apply the framework to systems with high-dimensional functional inputs. In Sections 3 and 4, we demonstrate the proposed methodology for two applications: i) prediction of extreme events in the Majda-McLaughlin-Tabak (MMT) model, a one-dimensional model for dispersive wave turbulence and ii) a correction operator for coarse-resolution climate model outputs. In both examples, we introduce methods to interpret the optimal points and gain insights into the active selection algorithm.

Overall, we show how our method is able to i) identify the points in a large data set that carry the most valuable information for predicting a specific quantity of interest, ii) reduce the cost of training ML models by using only the most valuable data, iii) improve generalizability properties of the resulting ML models with emphasis on their capacity to capture extreme events, iv) interpret the optimally selected data. One important advantage of the method is that it is model agnostic, so it can be used on any forthcoming ML-based climate model.

2. Data Selection with Active Learning.

Active Learning (AL) is a form of supervised ML in which new points are sequentially chosen to be added to the training set according to a criterion called the acquisition function [28, 11, 18] . Ren et al. provides a survey of AL in the context of ML classification models [45]. AL is part of the same family of algorithms

as Bayesian experimental design (BED) and Bayesian optimization (BO), algorithms that sequentially select the next-best point. However, we adapt the AL algorithm for the case where the new points must be selected from a preexisting, precomputed data set rather than from a continuous domain [39, 52, 2]. This distinction is sometimes referred to as active search, greedy approximations, optimal sampling, or active sampling, but we will refer to it as active data selection or active selection (AS).

2.1. Active Data Selection Algorithm. The AS algorithm (illustrated in Figure 1) is initialized with a small training set consisting of points randomly selected from the set of all candidate points. During each iteration, the model is trained and the acquisition function is evaluated at all remaining candidate training points. To compute the acquisition function, we make use of the predictions for the mean and epistemic uncertainty made by the probabilistic model. Candidate points resulting in the maximum value of the acquisition function are considered optimal, and they are added, as a batch, to the training set. Further details on batching are explained in [35]. The loop is repeated until the model error converges, or until the error reaches a desired threshold. The output of the algorithm is a ML model that has been trained with optimally selected data. At each iteration, the selected input points can be further analyzed to provide insights into what types of data are most useful for modeling.

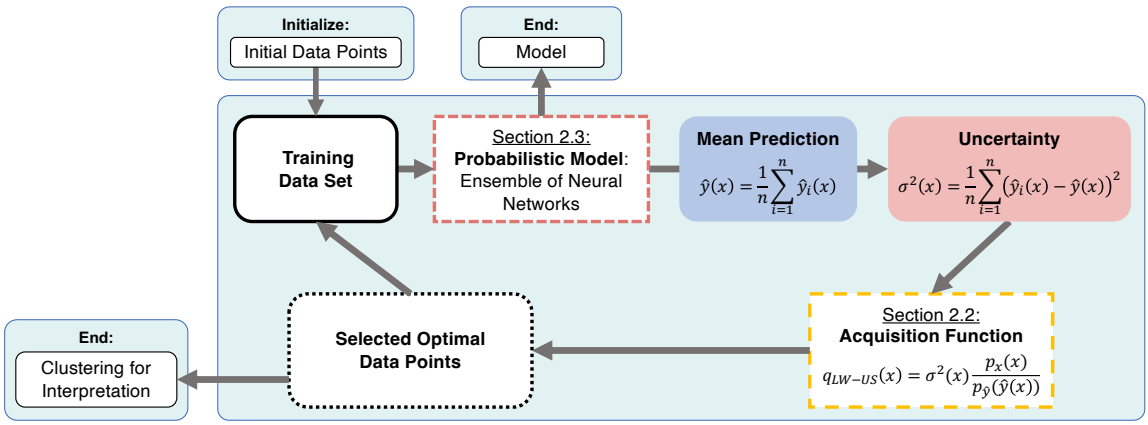

Figure 1. ***Active Data Selection Algorithm.*** *Points are sequentially selected according to the acquisition function and added to the training set to improve model prediction. The output of the algorithm is a model that has been trained on an optimal subset of the data with respect to predicting the statistics of extreme events.*

2.2. Acquisition Function: Likelihood-Weighted Uncertainty Sampling. The key element of the active data selection algorithm is the acquisition function which selects the most valuable points for model training. The choice of the acquisition function can depend on the nature of the system (e.g. nonlinear, high-dimensional, etc.), the goal of the modeling problem (e.g. optimization, extreme event identification, etc.), and many other constraints (e.g. computational costs, etc.). In general, the acquisition function should strike a balance between exploration and exploitation. In the most basic case of uncertainty sampling (US),

the acquisition function is the epistemic variance.

$$q_{\text{US}}(\mathbf{X}) = \sigma^2(\mathbf{X}) \tag{2.1}$$

A modified version of uncertainty sampling (input-weighted) prioritizes points that have a higher chance of occurring by multiplying the epistemic variance by the probability of the input points.

$$q_{\text{US}}(\mathbf{X}) = \sigma^2(\mathbf{X}) p_x(\mathbf{X}) \tag{2.2}$$

However, input-weighted criteria do not take into account the expected output and therefore do not account for the importance of extreme events. Here we choose to use a likelihood-weighted uncertainty sampling (LW-US) criterion to sequentially select optimal training points and quantify the value of points in the data set. The key idea behind the LW-US acquisition function is to pick input points that are likely to occur and reduce uncertainty, but also to take into account points that are likely to lead to extreme outputs. Acquisition functions that take into consideration the output were first introduced in [32] and further improved in [47] for applications to problems with high dimensional input spaces. In the original formulation, the function considers the integrated absolute difference between the log of the distribution of the prediction y_0 and the log of the distribution of a perturbed prediction y_+ made from a model perturbed in the direction of most uncertainty.

$$D_{\text{Log}^1}(y||y_0; h) = \int_{S_y} \left|\log p_{y_+}(y) - \log p_{y_0}(y)\right| dy \tag{2.3}$$

For a bounded domain S_y and a candidate sample point h, this acquisition function asymptotically converges to the desired output statistics, even in regions with low probability of occurrence [49]. However, the function is expensive to compute, and its lack of smooth gradients makes it unsuitable for gradient-based optimization. Instead, we use an upper bound (derived in [47]), which has a lower cost of computation and is analytically differentiable

$$q_{\text{LW-US}}(\mathbf{X}) = \int_{S_x} \sigma^2(\mathbf{X}) \frac{p_x(\mathbf{X})}{p_y(y(\mathbf{X}))} \tag{2.4}$$

In this modified version of the LW-US acquisition function, the epistemic variance $\sigma^2(\mathbf{X})$ is multiplied by the probability of the input points $p_x(\mathbf{X})$ and divided by the probability of the output points $p_y(y(\mathbf{X}))$ to prioritize candidate points that have the potential to reduce the model uncertainty, have a high chance of occurring, and most importantly, result in extreme events. Overall, these points are better able to represent of the tails of the distribution. The criterion can also serve as a "scoring" function because it gives priority to data points with the highest "value" with respect to improving the statistics of a specific observable (e.g. minimizing the error in the probability density function).

We measure success in terms of minimizing the error in the tails of the PDF, and we benchmark our method against a Monte Carlo (MC) acquisition function which selects points at random from the available candidate training points. MC is a meaningful benchmark because it is standard practice in many ML applications to randomly select a subset of data

for training and validation. A useful loss function to evaluate the quality of our model is the log-PDF error (LPE) which measures the integrated difference between the log of the true PDF obtained from the true y and the log of the estimated PDF obtained from the prediction $\hat{y}$.

$$\text{LPE} = \int |\log p_y(y) - \log p_{\hat{y}}(\hat{y})| \mathrm{d}y \tag{2.5}$$

This loss function is similar to the Kullback–Leibler divergence, but it more heavily penalizes errors in the tails of the distribution because the metric is not weighted by the output distribution $p_y(y)$.

2.3. Probabilistic Model: Ensemble of Neural Networks. The acquisition function requires an estimate for the epistemic uncertainty of the model. In previous works, traditional Bayesian supervised learning methods such as Bayesian regression or Gaussian process regression have been used to quantify uncertainty for optimal sampling [47, 7, 6, 58]. However, these methods are limited: Bayesian regression can fail when modeling nonlinear systems while Gaussian process regression suffers from performance issues on high dimensional or large data sets. NN-based ML architectures that can quantify uncertainty are able to overcome these problems [34, 22, 19, 62, 37, 63, 26]. We will focus on ensembles of neural networks (E-NN) and dropout neural networks (D-NN), but other methods to create a heuristic measure for uncertainty are summarized in [1]. In the E-NN, multiple models with the same architecture and hyperparameters are trained with the same training data sets but with different random weight initialization. The resulting prediction $\hat{y}$ is the mean of the n NN predictions $\hat{y}_i$

$$\hat{y}(\mathbf{X}) = \frac{1}{n} \sum_{i=1}^{n} \hat{y}_i(\mathbf{X}) \tag{2.6}$$

where $\hat{y}_i$ is the prediction of the i^{th} NN of the ensemble. The model uncertainty can be quantified via the variance of the predictions

$$\sigma^2(\mathbf{X}) = \frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i(\mathbf{X}) - \hat{y}(\mathbf{X}))^2 \tag{2.7}$$

resulting in a probabilistic prediction. In a D-NN, only one model is trained, but the model includes dropout layers [55]. During the prediction step, multiple predictions are made with different randomly dropped nodes [16]. Again, the resulting prediction is the mean of all the predictions, and the variance of the predictions can be used to create a probabilistic prediction. The dropout layers require additional training time, but only one model is trained, so the overall computation time is lower for the D-NN.

2.4. Application to Functional Inputs: Dimensionality Reduction. While the framework is versatile, we explain how to apply it to an output which depends on a functional — a mapping from a (possibly infinite-dimensional) space to a real number. In the MMT application in Section 3, the functional maps the high-dimensional initial conditions to the maximum wave amplitude reached over the given time horizon. In the climate modeling application in

Section 4, the functional maps the high-dimensional field consisting of temperature, humidity, and wind speed fields over the globe to a target quantity of interest, such as, for example, temperature at one spatial location. The challenge of using likelihood-weighted AS for a functional with a complicated input is that there may not be a straightforward or cost-effective way to compute $p_x(\mathbf{X})$.

2.4.1. Weighted Principal Component Analysis. In the applications we consider, the functional inputs are infinite or high-dimensional, so we first reduce the dimensionality by performing weighted principal component analysis (PCA). The spatial weights $w(\xi)$ depend on the problem of interest. For MMT, the weight is trivial $w(\xi) = 1$, so we perform standard principal component analysis to represent the inputs – the initial conditions. For the debiasing operator, we take into account the spherical geometry of the Earth: in the spherical coordinate system, the spatial coordinate ξ represents the polar coordinate $\theta \in (-90^\circ, 90^\circ)$ and azimuthal coordinate $\phi \in (0^\circ,\ 360^\circ)$. At each spatial point ξ, we define the weight $w(\xi) = w(\theta, \phi) = \sqrt{\sin\left(\frac{90^\circ - \theta}{180^\circ}\pi\right)}$.

In the general case, we start with the vector space $\mathbf{x}(\xi, t)$ with temporal mean $\overline{\mathbf{x}}(\xi)$ where ξ is the spatial coordinate. We aim to represent the vector space with an optimal set of N spatial modes (basis functions of dimension N) $\nu_j(\xi)$ with N corresponding time-dependent expansion coefficients $\alpha_j(t)$ (dimension n_t).

$$\mathbf{z}(t, \xi) \triangleq \mathbf{x}(t, \xi) - \overline{\mathbf{x}}(\xi) = \Sigma_{j=1}^{N} \alpha_j(t)\nu_j(\xi) \tag{2.8}$$

We define the weighted inner product between two fields, $\mathbf{x}_1(\xi)$ and $\mathbf{x}_2(\xi)$

$$\langle \mathbf{x}_1, \mathbf{x}_2 \rangle_w \triangleq \int_\xi w^2(\xi)\mathbf{x}_1(\xi)\mathbf{x}_2(\xi)d\xi \tag{2.9}$$

As an example, we show the resulting discretization for the spherical coordinate

$$\langle \mathbf{x}_1, \mathbf{x}_2 \rangle_w \simeq \sum_{\xi_{ij}} w^2(\xi_{ij})\mathbf{x}_1(\xi_{ij})\mathbf{x}_2(\xi_{ij})\delta\theta\delta\phi, \text{ where, } \xi_{ij} = (\theta_i, \phi_j). \tag{2.10}$$

We then define the spatial covariance by averaging over time

$$\mathbf{R}(\xi_1, \xi_2) \triangleq \frac{1}{T}\int_t (\mathbf{x}(t, \xi_1) - \bar{\mathbf{x}})(\mathbf{x}(t, \xi_2) - \bar{\mathbf{x}})dt \simeq \frac{1}{n_t}\mathbf{Z}\mathbf{Z}^T \in \mathbb{R}^{N\times N}, \tag{2.11}$$

where $\mathbf{Z}$ is the concatenated matrix in discrete space time:

$$\mathbf{Z} = [\mathbf{z}(t_1)\ \mathbf{z}(t_2)\ ...\ \mathbf{z}(t_{n_t})] \in \mathbb{R}^{N\times n_t} \tag{2.12}$$

Next, we set up the eigenvalue problem

$$\langle \mathbf{R}(\cdot, \xi), \psi_j(\cdot) \rangle_w = \lambda_j\psi_j(\xi), \psi_j \in \mathbb{R}^N, \lambda_1 \geq \lambda_2 \geq ... \geq \lambda_N \geq 0 \tag{2.13}$$

Finally, we obtain the quantity of interest (observable), the time-dependent PCA coefficient

$$y = \alpha_j(t) = \langle \mathbf{z}(t, \cdot), \psi_j(\cdot) \rangle_w \tag{2.14}$$

If the target quantity is the first PCA coefficient, we use ψ_1. However, we can adjust the mode ψ to focus on other quantities, such as, for example, a specific spatial location. The analysis works for any target quantity of interest that can be described as a functional $\langle \mathbf{z}(t, \cdot),\ \cdot \rangle$.

2.4.2. Evaluation of the Acquisition Function for Functional Inputs. We now explain the process of computing the acquisition function in the case of applying the method to a system with a functional input. We highlight that in many problems, the test set (denoted TS) is possibly different from the training set (denoted TR), which means that the target quantity of interest can be generated with input points that are different from the input points in the training set. The likelihood-weighted acquisition function depends on both an input $\mathbf{X}$ and the predicted $\hat{y}$, and it is made up of i) the uncertainty $\sigma^2(\mathbf{X})$, ii) the weight from the inputs $p_x(\mathbf{X})$, and iii) the weight from the outputs $p_{\hat{y}}(\hat{y}(\mathbf{X}))$.

$$q_{\text{LW-US}}(\mathbf{X}) = \sigma^2(\mathbf{X}) \frac{p_x(\mathbf{X})}{p_{\hat{y}}(\hat{y}(\mathbf{X}))} \tag{2.15}$$

We first define some relevant quantities for our problem set up. We start with a training set $\mathcal{D}^{\text{TR}}$ consisting of candidate samples $\mathcal{X}$ and a test set $\mathcal{D}^{\text{TS}}$ consisting of the points at which we we wish to evaluate the model. The set $\mathcal{U} = [\mathcal{X}_1 \ ... \ \mathcal{X}_p]$ consists of the p points that were selected from all candidate points to train the model. The overall goal is to find which samples $\mathcal{X}$ from $\mathcal{D}^{\text{TR}}$ we should add to $\mathcal{U}$ to improve the prediction of the target output y. At each iteration of the algorithm, we train a model $\mathcal{M}_{\mathcal{U}}$ trained with the selected samples $\mathcal{U}$, a subset of $\mathcal{D}^{\text{TR}}$. We use $\mathcal{M}_{\mathcal{U}}$ to make two sets of predictions: i) predictions for the candidate training points $\hat{\mathcal{Y}}_{\text{tr}} = \mathcal{M}_{\mathcal{U}}(\mathcal{X})$ and ii) predictions for the test points $\hat{\mathcal{Y}}_{\text{ts}} = \mathcal{M}_{\mathcal{U}}(\mathcal{D}^{\text{TS}})$. Then, we use PCA to evaluate the quantities of interest $\hat{y}_{\text{tr}}$ and $\hat{y}_{\text{ts}}$ which are the mean of the predictions of all n members in the NN ensemble

$$\hat{y}_{\text{tr}} = \frac{1}{n}\sum_{j=1}^{n} \left\langle \hat{\mathcal{Y}}_{\text{tr}}^j, \psi \right\rangle_w, \text{ and } \hat{y}_{\text{ts}} = \frac{1}{n}\sum_{j=1}^{n} \left\langle \hat{\mathcal{Y}}_{\text{ts}}^j, \psi \right\rangle_w \tag{2.16}$$

We obtain an estimate for the uncertainty of the predictions made with the candidate training points $\sigma^2(\hat{y}_{\text{tr}}(\mathcal{X}))$ from the model $\mathcal{M}_{\mathcal{U}}$. The uncertainty is the variance of the predictions made by each member $\mathcal{M}_{\mathcal{U}}^j$ of the ensemble.

$$\sigma^2(\mathcal{X}) = \frac{1}{n}\sum_{j=1}^{n} \left(\hat{y}_{\text{tr}}^j(\mathcal{X}) - \hat{y}_{\text{tr}}(\mathcal{X})\right)^2 = \frac{1}{n}\sum_{j=1}^{n} \left(\left\langle \mathcal{M}_{\mathcal{U}}^j(\mathcal{X}), \psi \right\rangle_w - \overline{\langle \mathcal{M}_{\mathcal{U}}(\mathcal{X}), \psi \rangle_w}\right)^2 \tag{2.17}$$

The distribution of the input points $p_{\mathcal{X}}$ is approximated by the distribution of the first k PCA coefficients of the input data set $\mathcal{X}$ where k is selected according to the decay of the eigenvalues. We refer to the truncated version of $\mathcal{X}$ as $\mathbf{x}$, and we estimate its kernel density estimate (KDE) $p_{\mathbf{x}}$ using the python function FFTKDE from the package KDEpy.

$$p_{\mathcal{X}}(\mathcal{X}) \approx p_{\mathbf{x}}(\mathbf{x}) \text{ where } \mathbf{x} = \left\langle \mathcal{X}, \{\psi\}_{i=1}^{k} \right\rangle_w \tag{2.18}$$

The single time distribution $p_{\hat{y}_{\text{ts}}}$ is estimated with the predictions $\hat{y}_{\text{ts}}$. Like $p_{\mathcal{X}}$, $p_{\hat{y}_{\text{ts}}}$ is also computed with KDEpy. We then compute the weight for the output, by evaluating $p_{\hat{y}_{\text{ts}}}$ at the predictions made from the candidate training points $\hat{y}_{\text{tr}}(\mathcal{X})$.

$$p_{\hat{y}_{\text{ts}}}(\hat{y}_{tr}(\mathcal{X})) = p_{\hat{y}_{\text{ts}}}(\langle \mathcal{M}_{\mathcal{U}}(\mathcal{X}), \psi \rangle_w) \tag{2.19}$$

In the end, the acquisition function for the candidate training points $\mathcal{X}$ is computed using all of the quantities above

$$q_{\text{LW-US}}(\mathcal{X}) = \sigma^2(\mathcal{X}) \frac{p_{\mathcal{X}}(\mathcal{X})}{p_{\hat{y}_{\text{ts}}}(\hat{y}_{tr}(\mathcal{X}))} \tag{2.20}$$

As a result, we can evaluate the acquisition function at a low cost for high-dimensional functional inputs $\mathcal{X}$.

3. Application to the Majda-McLaughlin-Tabak (MMT) Model.

3.1. MMT System. We first apply the described method to the MMT model, a one-dimensional dispersive nonlinear wave model that, given certain parameters, is useful for studying turbulence and rogue waves [29]. More details on the overall system can be found in [8, 59, 38, 12, 60]. The system is described by the governing equation

$$iu_t = |\partial_x|^{\alpha} u + \lambda |\partial_x|^{-\beta/4} \left(\left| |\partial_x|^{-\beta/4} u \right|^2 |\partial_x|^{-\beta/4} u \right) + iDu \tag{3.1}$$

where the output u is a complex scalar representing the wave amplitude, α and β are parameters of the system, and D is a selective Laplacian which eliminates high wave numbers. For $\alpha = 1/2$ and $\beta = 0$, the equation can be rewritten in the wave number space with forcing $f(k)$

$$\hat{u}(k)_t = -i|k|^{1/2}\hat{u}(k) - i\lambda |\hat{u}(k)|^2 \hat{u}(k) + \widehat{Du}(k) + f(k) \tag{3.2}$$

where the selective Laplacian is defined as

$$\widehat{Du}(k) = \begin{cases} -(|k| - k^*)^2 \hat{u}(k) & \text{if } |k| > k^* \\ 0 & \text{if } |k| \leq k^* \end{cases} \tag{3.3}$$

This operator $\widehat{Du}(k)$ prevents wave numbers above a threshold k^*: for small wave amplitudes, the output PDF appears to be Gaussian, but for large wave amplitudes, the output PDF is very heavy-tailed. The stochastic complex initial conditions $u(x, t = 0)$, which are Gaussian, are obtained from the covariance

$$k(x, x') = \sigma_u^2 \exp\left(i2\sin^2(\pi(x - x'))\right) \exp\left(-\frac{2\sin^2(\pi(x - x'))}{l_u^2}\right) \tag{3.4}$$

with $\sigma_u = 1$ and $l_u = 0.35$, and they are reduced to $2m$ dimensions, m real and m imaginary, using the Karhunen-Loeve (KL) expansion,

$$u(x, t = 0) \approx \sum_{j=1}^{m} \alpha_j \sqrt{\lambda_j} \phi_j(x), \ \forall\ x \in [0, 1) \tag{3.5}$$

which transforms the original high-dimensional data into a set of orthogonal components. The KL expansion is a dimensionality reduction method that maximizes the amount of retained information by only using the most important features of the data. The grid is periodic over

$[0, 1)$ and discretized into 512 points, m is set to 4, the timestep is $dt = 0.001$, the parameters of the equation are $\lambda = -0.5$ and $k^* = 20$, and there is no forcing, $f(k) = 0$. As in [35] and [19], we seek to train a standard fully-connected NN (FC-NN) to predict the maximum future wave amplitude over a given time horizon, an extreme event, as a function of the $2m$ stochastic initial conditions $\vec{\alpha}$

$$y(\vec{\alpha}) = ||\mathrm{Re}(u(x, T = 50; \vec{\alpha}))|| \tag{3.6}$$

3.2. MMT Data Sets. To better mimic the characteristics of data sets that are found in the real world, we make use of two data sets: points obtained from inputs that follow a Gaussian distribution, $\mathcal{D}_{p_X}$, and points obtained with Latin hypercube sampling, $\mathcal{D}_{LHS}$. For Monte Carlo sampling, we select candidate training points from $\mathcal{D}_{p_X}$ because this distribution more closely resembles naturally-occurring data sets. As a result, we compare our proposed method to a more rigorous benchmark (the Monte Carlo sampling performs worse when applied to points from $\mathcal{D}_{LHS}$). For US/LW-US sampling, we select candidate training points from $\mathcal{D}_{LHS}$ because this data set more completely represents all the achievable values, including the tails of the distribution. We evaluate the error metrics on the test set $\mathcal{D}_{LHS}$ to measure the ability of the models to capture the tails of the distribution.

3.3. MMT Machine Learning Architecture and Active Learning Hyperparameters. We test both the E-NN and the D-NN described in Section 2.3. For the E-NN, we use an ensemble of size 2, and for the D-NN, we use an ensemble fo size 5. Even though the size of the D-NN ensemble is higher, the overall process takes less time because only one model is trained. From the results of a simple hyperparameter grid search, we set the number of layers to eight, the number of neurons to 250, the activation to ReLU, the number of epochs to 3000, and the batch size to the floor of half the number of points in the training set. For the D-NN, we set the dropout rate to 50%, a standard choice in many ML papers. The batch size is the only hyperparameter that changes at each iteration, and we choose to update the batch size at each iteration to keep the training error within a reasonable range given a growing data set size and a constant number of epochs. We initialize the algorithm with a training set of 10 randomly chosen points. At each iteration, we add a batch of 10 points to the training set (points that correspond to the maximum value of the acquisition function), and we re-initiate the model to avoid getting stuck in any bad local minima found during early iterations.

3.4. MMT Results. The results obtained from carrying out the algorithm for 150 iterations (up to 1500 points — 1.5% of the full data set) for randomly chosen points (MC), input-weighted uncertainty sampling (US), and likelihood-weighted uncertainty sampling (LW-US) are shown in Figure 2. Because we compute the mean squared error (MSE) with the Latin hypercube sampling data set $\mathcal{D}_{LHS}$, we weight the error by the input distribution as follows

$$\mathrm{MSE} = \sum_{i=1}^{N} (y_i - \hat{y}_i(\vec{\alpha}_i))^2 p_X(\vec{\alpha}_i) \tag{3.7}$$

The LW-US outperforms MC and US with respect to minimizing the error in the tail of the PDF, and this is seen again in Figure 3. The E-NN outperforms the D-NN, but the D-NN training is faster, making it a useful architecture for more computationally expensive problems.

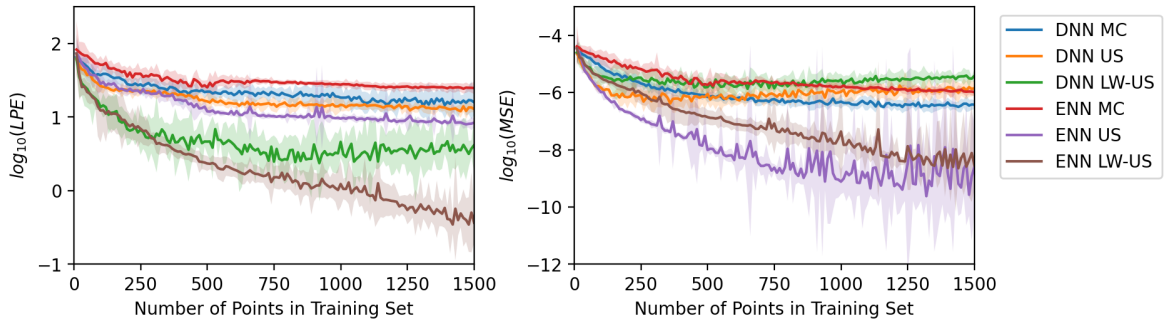

Figure 2. ***Error Convergence Curves of the MMT Predictions.*** *The log of the LPE error (left) and log of the MSE (right) are plotted as a function of the number of points in the training set for both the E-NN and D-NN implementations of the MC, US, and LW-US acquisition functions. LW-US (both E-NN and D-NN) significantly outperforms the other acquisition functions with respect to LPE. E-NN US initially achieves a better MSE, but E-NN LW-US eventually achieves a similar error.*

3.5. Interpreting the Selected Points: Multidimensional Scaling. To gain insights into the behavior of the LW-US active search algorithm, we visualize the eight-dimensional selected input points with multidimensional scaling (MDS). MDS projects high-dimensional points to a two-dimensional subspace with the requirement that a chosen distance metric be preserved between points — points that are more spread apart in the original space must be spread apart in the lower-dimensional space, and vice versa. As is typically done, we use the Euclidian distance as the distance metric. The two-dimensional projection shown in Figure 3 reveals that points chosen by the LW-US acquisition function are farther apart than points chosen by other acquisition functions. The results of MDS suggest that drawing points that are more "spread out" with respect to each other can be helpful for predicting extreme events.

4. Application to Debiasing Operator for Coarse-Resolution Climate Model Outputs. We now show how likelihood-weighted active selection can be used to speed up the training of ML climate models while improving the prediction of extreme weather events. We focus on the model in [3] which learns a debiasing operator that maps trajectories from a free-running coarse-resolution climate simulation to trajectories from a high-resolution fully-resolved climate reanalysis data set [3, 61, 9]. The advantage of this model is that the operator can be used to correct less computationally expensive low-resolution climate simulations. While we focus on this one model, the algorithm is model agnostic and can be used for any ML-based climate model for which the set of possible training points is very large. In addition to reducing computational time and costs, the likelihood-weighted criterion is able to determine which points are most relevant to the dynamics of target extreme weather events.

4.1. Climate Data Sets. The coarse-resolution simulations are obtained from version 2 of the Energy Exascale System Model (E3SM) Atmosphere Model (EAMv2) [13, 17, 56]. The data set consists of temperature (T), specific humidity (Q), zonal velocity (U), and meridional velocity (V) at a 1° (approximately 110km) resolution, and we only consider the vertical layer closest to the surface of the Earth. The high-resolution target data set is the European Centre

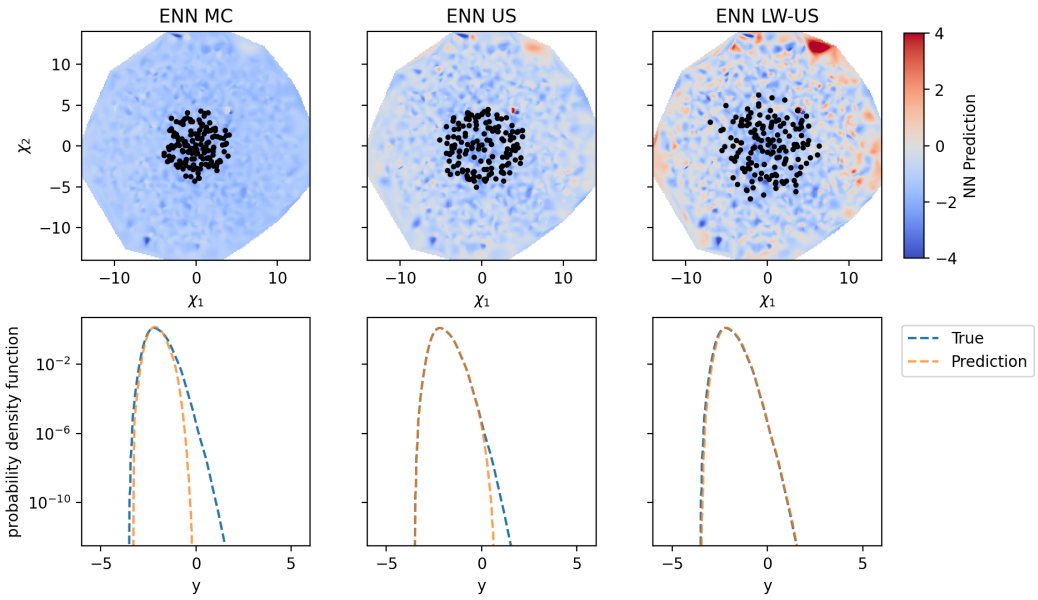

Figure 3. ***Visualization of Selected MMT Input Points.*** *In the top row, the 8D space is projected to a 2D space with multi-dimensional scaling. Each plot shows the spread of the optimally selected points in black over the prediction made from the neural network trained after 150 iterations with training data obtained from MC, US, and LW-US (left to right). The rightmost plot suggests that points chosen by LW-US are more spread out. In the bottom row, the predicted PDF is compared to the true PDF after 150 iterations for MC, US, and LW-US, and LW-US best matches the tail of the distribution.*

for Medium-Range Weather Forecasts (ECMWF) Reanalysis version 5 (ERA5) [20]. ERA5 has a resolution of 0.25° (approximately 31km), but it is projected onto the E3SM grid for the purpose of this model. For all data sets, we use 10 years of data from 2007 to 2017, sampled 8 times per day.

During the training phase, the output is the fine-scale reanalysis data set (denoted ERA5), and the input is the free-running data set from the coarse-scale climate solver that has been *nudged* (denoted NUDG) to match the output. We will not go into the details of the nudging procedure, but it is comprehensively described in [3]. During the testing phase, we use the trained model to predict high-resolution field given the *un-nudged* free-running coarse-resolution climate simulation (denoted CR for coarse-resolution). Each data set ($\mathcal{D}^{\text{ERA5}}$, $\mathcal{D}^{\text{NUDG}}$, and $\mathcal{D}^{\text{CR}}$) is a field over space ξ and time t. Figure 4 shows the mean of the reference reanalysis data set and the mean of the model output given the test data CR as input.

4.2. Machine Learning Architecture and Active Learning Hyperparameters. The NN architecture, in Figure 5, is an encoder-decoder consisting of 2D convolutional layers. The globe is divided into 25 sections (5 × 5 grid), the sections are padded to satisfy spherical periodicity (the Earth is a globe), and the encoding convolutions are applied to each section independently. The encoder is made up of one layer to split the globe, one layer to spherically pad the sections, three convolutional layers applied to each section to capture anistropic local features, one layer to merge the section. Next, the decoder applies "deconvolutional" layers

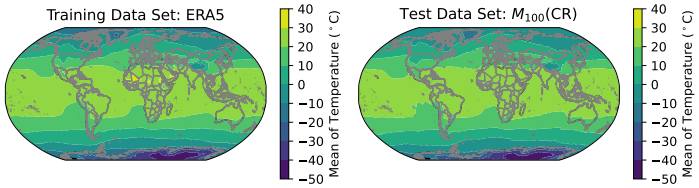

Figure 4. ***Mean of ERA5 and Mean of*** $\mathcal{M}_{100}(\mathrm{CR})$ ***for temperature.*** *During the training phase, the nudged data set is mapped to the ERA5 data set. During the testing phase, the coarse-resolution (CR) data set is provided as input to the trained model.*

(or transpose convolutional layers) to map the latent space back to the desired dimension. Finally, the 25 sections are combined to recreate the full field. The batch size is set to 8, and the number of epochs is set to 150, as was done in [3]. The loss function is the MSE for which spatial points are weighted by latitude θ: $w(\theta) = \sqrt{\sin\left(\frac{90^\circ - \theta}{180^\circ}\pi\right)}$.

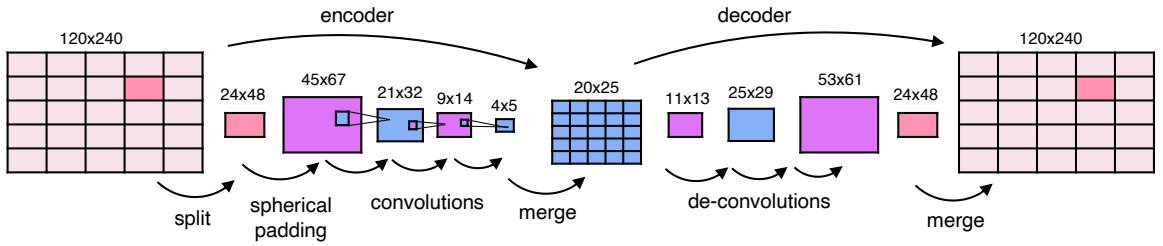

Figure 5. ***Climate Debiasing Operator Neural Network Architecture.*** *The NN architecture splits the Earth into sections which are individually passed through convolutional encoder-decoder layers.*

The model $\mathcal{M}_{\mathcal{U}}$ is trained to map samples $\mathcal{U}$ from $\mathcal{D}^{\mathrm{NUDG}}$ to the output $\mathcal{D}^{\mathrm{ERA5}}$. During the testing phase, the input of the model is $\mathcal{D}^{\mathrm{CR}}$, and the resulting functional output is the field $\mathcal{Y} = \mathcal{M}_{\mathcal{U}}(\mathcal{D}^{\mathrm{CR}})$. To evaluate the framework, we generate "ground truth" data by training a model with 100% of the samples in $\mathcal{D}^{\mathrm{NUDG}}$ and $\mathcal{D}^{\mathrm{ERA5}}$: we call this model $\mathcal{M}_{100}$. Then, we use the model $\mathcal{M}_{100}$ to make a prediction from the un-nudged coarse resolution data set $\mathcal{D}^{\mathrm{CR}}$: we call this prediction $\mathcal{M}_{100}(\mathcal{D}^{\mathrm{CR}})$. At each iteration, we compute error metrics for $\mathcal{M}_{\mathcal{U}}(\mathcal{D}^{\mathrm{CR}})$ with respect to $\mathcal{M}_{100}(\mathcal{D}^{\mathrm{CR}})$.

The probabilistic model is an E-NN of size two, a choice that was shown to be preferable in [35]. The prediction is the mean of the outputs of each member of the ensemble, and the uncertainty is their variance. We initialize the algorithm with a training set of ten randomly chosen points, and at each iteration, we add ten points to the training set (points that correspond to the maximum value of the acquisition function). For the MC case, we add twenty random points at each iteration because future iterations do not depend on previous iterations. There are 29,200 (10 years $\times$ 365 days $\times$ 8 measurements per day) possible samples that can be chosen by the acquisition function, but we only evaluate the method up to 750 points in the training set (2.6% of all data). For each test case, we perform five or six experiments so

that we can take the average MSE and LPE over the different experiments.

4.3. Results. For the climate application, we test the method on three cases: i) the first PCA coefficient for temperature over the entire globe (Figure 6), ii) temperature in Paris (Figure 9), and iii) specific humidity in Miami (Figure 12). These last two locations were chosen randomly from a list of cities that have experienced extreme heat waves (in the case of temperature) or extreme floods (in the case of specific humidity) in the last few decades.

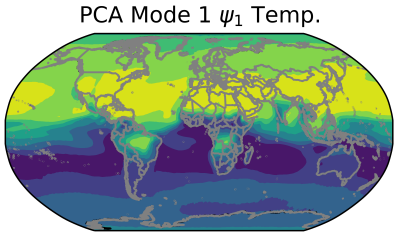

Figure 6. *First weighted PCA mode of the global temperature field.*

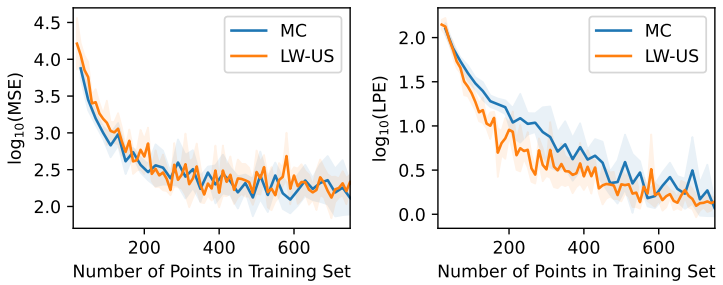

Figure 7. *The log of the MSE and LPE are shown for MC and LW-US with respect to predicting the first PCA coefficient for temperature given a global model.*

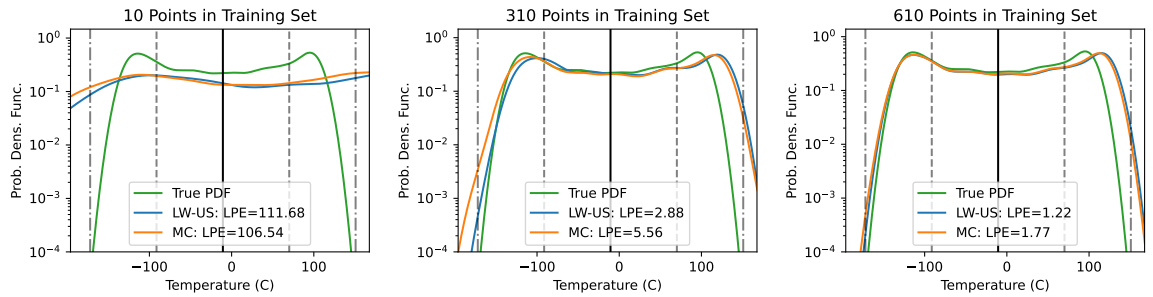

Figure 8. *The true PDF of the first PCA coefficient for temperature is compared to the PDF obtained from predictions made with MC and LW-US. The black vertical line denotes the mean of the true distribution, and the dashed lines denote the 1σ and 2σ. LW-US is able to better match the left tail of the true PDF.*

Looking at the LPE as a function of number of points in the training set in Figures 7, 10, and 13, we see that LW-US outperforms MC in all cases. In some cases, the improvement is more significant (e.g. temperature in Paris Figure 10). The improvement obtained from using LW-US can also be seen in the plots of the PDF (Figures 8, 11, and 14)— LW-US does a better job at matching the tails of the distribution. We also observe that the improvement obtained from using LW-US occurs at different number of iterations for different test cases. In the case of MSE, the error is similar for cases involving temperature (Figures 7 and 10), but worse for cases involving humidity (Figure 13). When using LW-US to improve the prediction of temperature, there are no losses in the MSE and great improvement in the LPE. We note that as the number of data points increase, the two methods give very similar results, as expected.

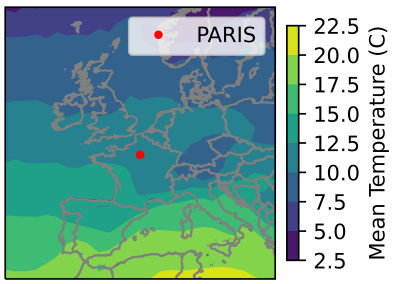

Figure 9. *Mean temperature in the region surrounding Paris, France.*

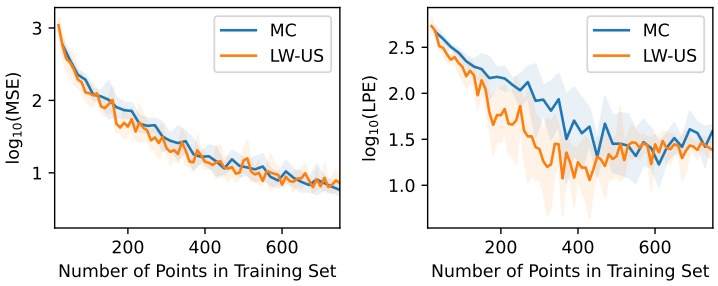

Figure 10. *The log of the MSE and LPE are shown for MC and LW-US with respect to predicting the temperature in Paris given a global model.*

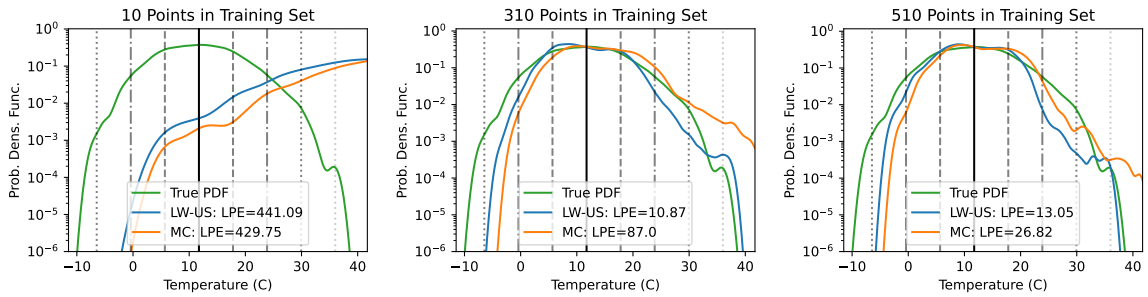

Figure 11. *The true PDF of the temperature in Paris is compared to the PDF obtained from predictions made with MC and LW-US. The black vertical line denotes the mean of the true distribution, and the dashed lines denote the 1σ, 2σ, 3σ, and 4σ. LW-US is able to better match the tails of the true PDF with just 310 points.*

4.4. Interpreting the most informative data points: Clustering. Upon selecting the training points, the subsequent goal is to determine if the points that were chosen by the algorithm have any relevant physical meaning. For example, scientists could be interesting in determining if these points are related to important system dynamics, if they can be attributed to physical phenomena (e.g. turbulence, atmospheric rivers, tropical cyclones, etc.), or if their physical interpretation depends on the target's predicted output. Understanding why the optimal points were selected also reduces some of the "black box" nature of the ML-based algorithm.

We present a clustering framework to mechanistically identify and define the dynamics of these points of interest. Clustering, a form of reduced-order modeling in which observations are clustered around centroids, has been used for climate data sets in other applications [27]. In the case of a dynamic system like the climate, the observations (or samples) are snapshots in time of the system. We select cluster centroids from the entire reference data set of PCA time coefficients $\alpha_j(t) = \left\langle \mathcal{D}^{\mathrm{ERA5}}, \psi_j^{\mathrm{ERA5}} \right\rangle_w$ using the standard k-means algorithm with k-means++ for seed initialization. We set the number of clusters to six for all cases. The resulting cluster centroids are projected back onto the PCA modes to visualize the spatial

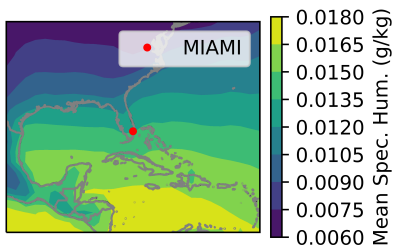

Figure 12. *Mean specific humidity in the region surrounding Miami, USA.*

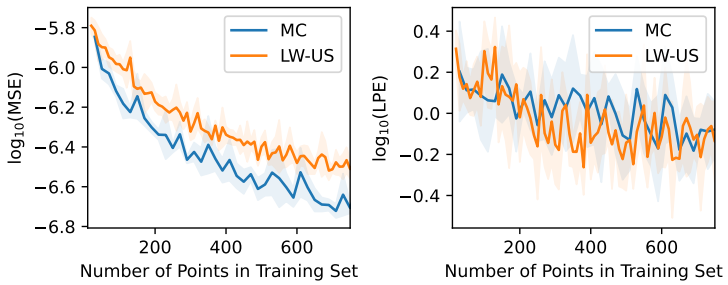

Figure 13. *The log of the MSE and LPE are shown for MC and LW-US with respect to predicting the specific humidity in Miami given a global model. LW-US is better for minimizing LPE, but worse for minimizing MSE in the case of specific humidity.*

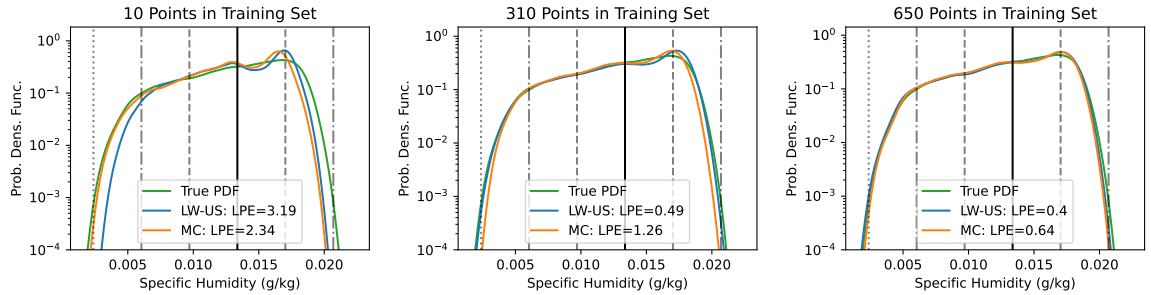

Figure 14. *The true PDF of the specific humidity in Miami is compared to the PDF obtained from predictions made with MC and LW-US. The black vertical line denotes the mean of the true distribution, and the dashed lines denote the 1σ and 2σ. LW-US is able to better match the tails of the true PDF.*

patterns. These cluster centroids are then used to predict the cluster labels of the new subset of the data chosen by the algorithm. This step assigns the optimal points to relevant cluster centers which allows us to determine if the points chosen by the algorithm are associated with noteworthy dynamical phenomena. The ultimate goal is to interpret the physical meaning of the points that were chosen for training.

In Figures 15 and 16, the six clusters are mapped in order of most occurring in the whole data set (Cluster #1) to least occurring in the whole data set (Cluster #6). For temperature in Paris (Figure 15), we found that points belonging to Cluster #6 are more relevant to the dynamics of extreme weather events (Figure 15). Upon further examination, the shape of Cluster #6 suggests a potential heat dome over Paris and the surrounding region [23]. By using clustering, we are able to pick out extreme weather events using the active search as an unsupervised algorithm. For specific humidity in Miami (Figure 16), we observe a similar behavior: around 30% of the optimal points belong to Cluster #1 which resembles standard zonal flow while around 50% of the optimal points belong to Cluster #6 which resembles a blocking pattern. We also observe that only Clusters #1 and #6 are well-represented in the optimal points which might provide an explanation as to why the MSE from LW-US is higher

for humidity in Miami.

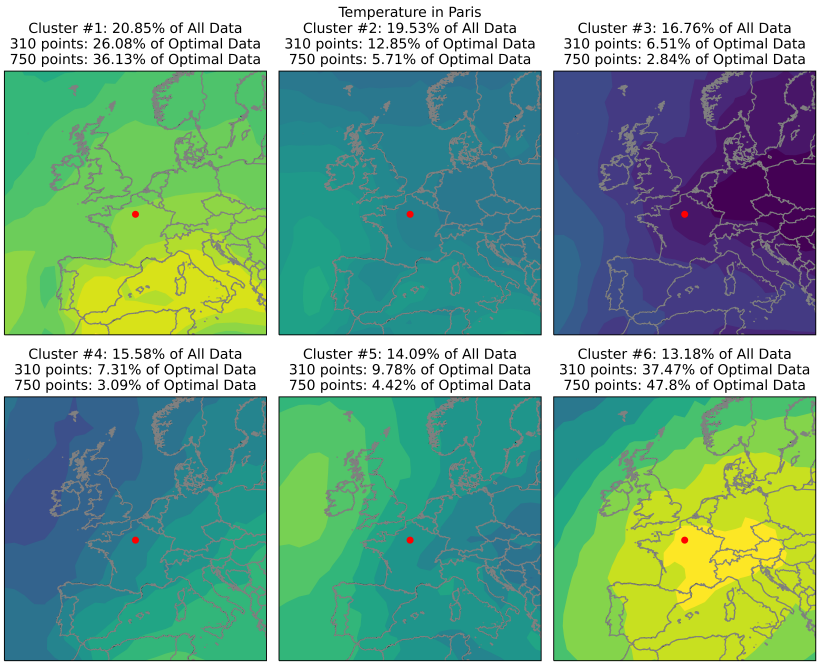

Figure 15. ***Clusters for Temperature in Paris*** *With 310 points in the training set, Cluster #6 only represents 13.18% of all data but 37.47% of the optimal data. Cluster #6 exhibits a blocking pattern over most of France. The next most occurring cluster is Cluster #1 which represents standard zonal flow, typical for normal weather events.*

5. Conclusions. To better prepare for the impacts of climate change on humans, infrastructure, and ecosystems, there is a pressing demand for improved climate models. These models need to be fast so that they can be used for a variety of potential emission scenarios, and they need to be accurate, even with respect to capturing the statistics of lower-probability extreme weather events. Given both i) the rise in ML-based weather and climate models and ii) the vast number of samples in high-resolution comprehensive climate data sets, our ability to develop models can be significantly improved by more intelligently selecting training data. To address this gap in the field, we introduced a likelihood-weighted active data selection framework which sequentially selects optimal training points to improve prediction of extreme event statistics (i.e. tails of the distribution). The framework is model agnostic and suitable for high-dimensional data sets. We demonstrated the success of the framework on both a synthetic problem and a real-world problem. In both cases, the likelihood-weighted active data selection achieved a lower error in the tails of the probability distribution with fewer training points, which reduces model uncertainty and brings down computational costs. In the real-world problem, our method was also able to identify the dynamics relevant to extreme weather events for added interpretability. The developed approach has the potential to be useful for improved environmental sampling schemes, as well as compression algorithms that preserve the information associated with extreme events in extensive data sets.

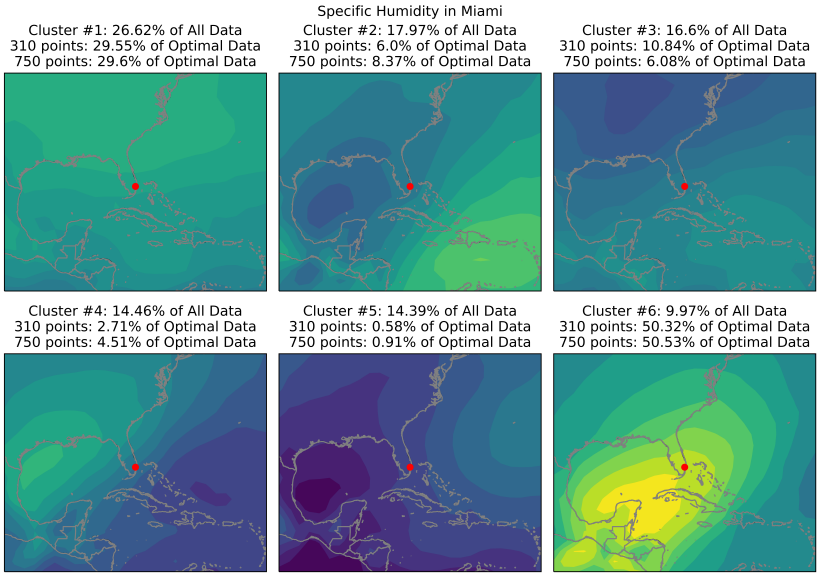

Figure 16. ***Clusters for Specific Humidity in Miami*** *With 310 points in the training set, Cluster #6, which resembles a blocking pattern, only represents 9.97% of all data but 50.32% of the optimal data. The next most occurring cluster is Cluster #1 which resembles standard zonal flow, typical for normal weather events. The spread of the frequency of optimal points does not change much between 310 points and 750 points.*

Appendix A. Supplementary Figures.

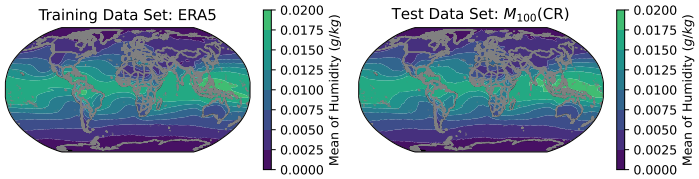

Figure 17. ***Mean of ERA5 and Mean of*** $\mathcal{M}_{100}(\mathrm{CR})$ ***for humidity.*** *During the training phase, the nudged data set is mapped to the ERA5 data set. During the testing phase, the coarse-resolution (CR) data set is provided as input to the trained model.*

446

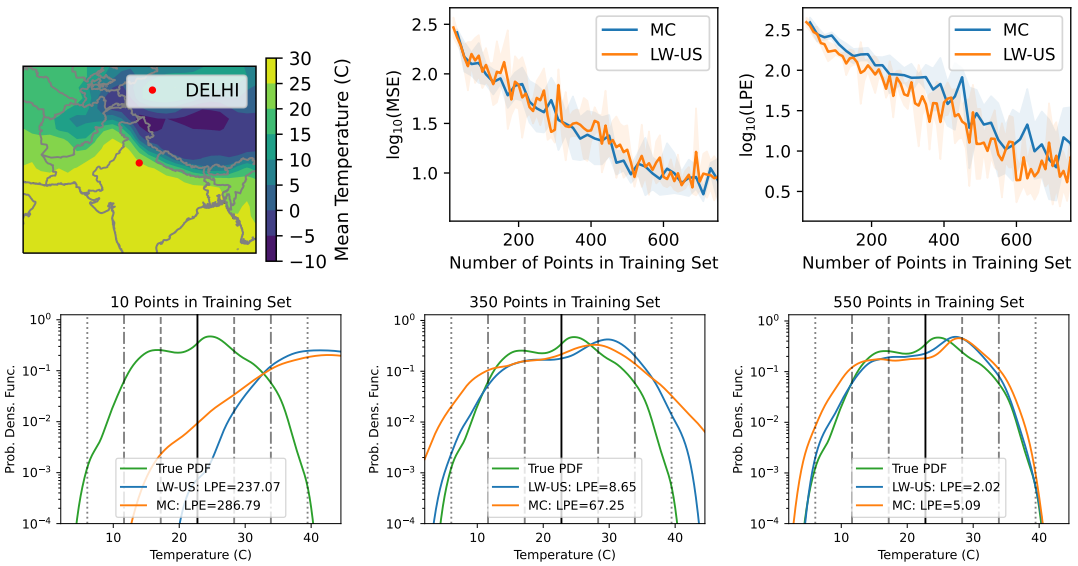

Figure 18. ***Temperature in Delhi.*** *In the top row, the MSE and LPE are shown for MC and LW-US. In the bottom row the true PDF is compared to the PDF obtained from predictions made with MC and LW-US. The black vertical line denotes the mean of the true distribution, and the dashed lines denote the* 1σ*,* 2σ*, and* 3σ*.*

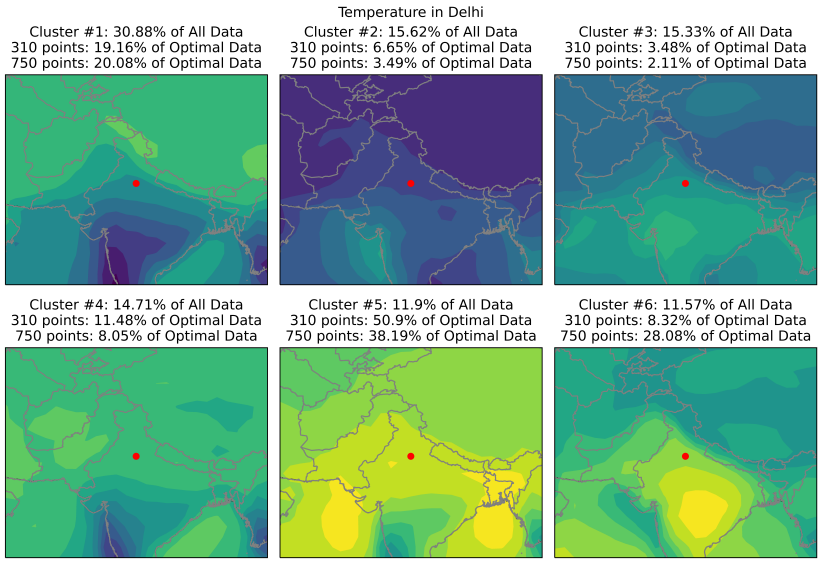

Figure 19. ***Temperature in Delhi****: With 310 points in the training set, Cluster #5 only represents 11.9% of all data but 50.9% of the optimal data. The next most occurring cluster is Cluster #1. With 750 points in the training set, there is a significant increase in the presence of Cluster #6.*

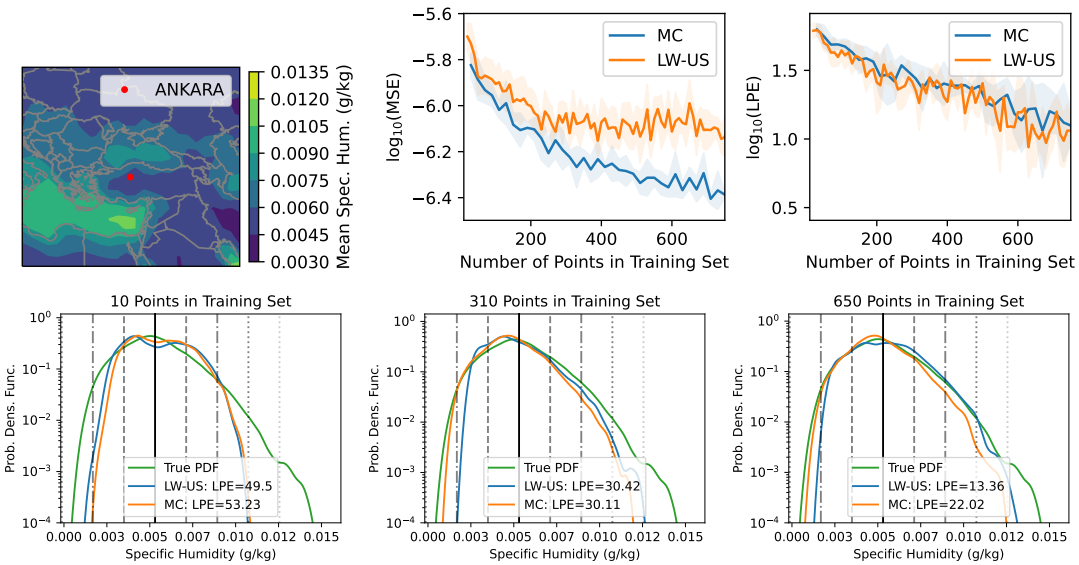

Figure 20. ***Specfic Humidity in Ankara****. In the top left, the MSE and LPE are shown for MC and LW-US. In the bottom row the true PDF is compared to the PDF obtained from predictions made with MC and LW-US. The black vertical line denotes the mean of the true distribution, and the dashed lines denote the* 1σ*,* 2σ*,* 3σ*, and* 4σ*.*

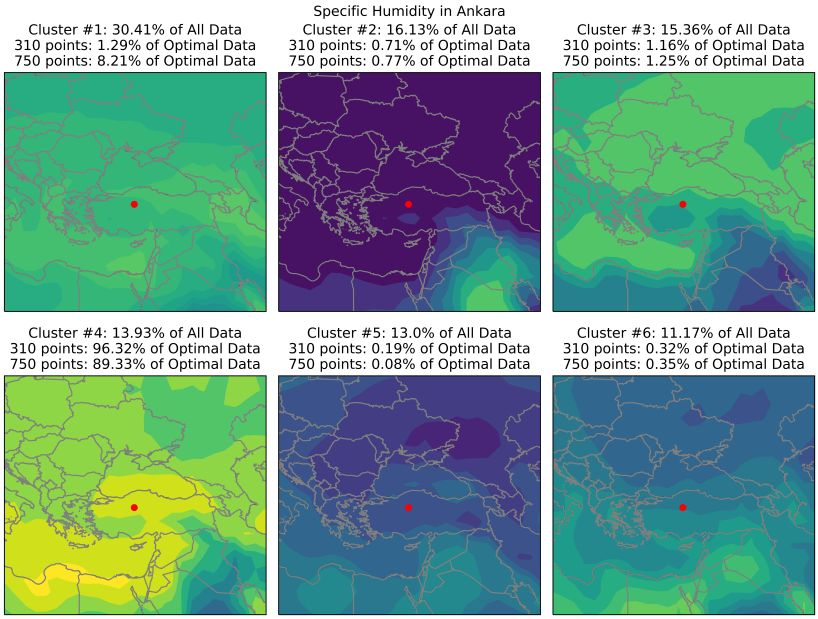

Figure 21. ***Humidity in Ankara****: With 310 points in the training set, Cluster #4 only represents 13.93% of all data but 96.32% of the optimal data. With 750 points in the training set, Cluster #4 remains heavily represented, but a larger portion of the optimally selected points are coming from Cluster #1.*

REFERENCES

[1] A. N. Angelopoulos and S. Bates, *A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification*, Dec. 2022, https://doi.org/10.48550/arXiv.2107.07511, http://arxiv.org/abs/2107.07511 (accessed 2024-07-12). arXiv:2107.07511 [cs, math, stat].

[2] A. Banerjee, D. Dunson, and S. Tokdar, *Efficient Gaussian Process Regression for Large Data Sets*, June 2011, http://arxiv.org/abs/1106.5779 (accessed 2023-04-28). arXiv:1106.5779 [stat].

[3] B. Barthel Sorensen, A. Charalampopoulos, S. Zhang, B. E. Harrop, L. R. Leung, and T. P. Sapsis, *A Non-Intrusive Machine Learning Framework for Debiasing Long-Time Coarse Resolution Climate Simulations and Quantifying Rare Events Statistics*, Journal of Advances in Modeling Earth Systems, 16 (2024), p. e2023MS004122, https://doi.org/10.1029/2023MS004122, https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2023MS004122 (accessed 2024-07-12).

[4] P. Bauer, B. Stevens, and W. Hazeleger, *A digital twin of Earth for the green transition*, Nature Climate Change, 11 (2021), pp. 80–83, https://doi.org/10.1038/s41558-021-00986-y, https://www.nature.com/articles/s41558-021-00986-y (accessed 2024-07-12). Publisher: Nature Publishing Group.

[5] K. Bi, L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian, *Accurate medium-range global weather forecasting with 3D neural networks*, Nature, 619 (2023), pp. 533–538, https://doi.org/10.1038/s41586-023-06185-3, https://doi.org/10.1038/s41586-023-06185-3.

[6] A. Blanchard and T. Sapsis, *Bayesian optimization with output-weighted optimal sampling*, Journal of Computational Physics, 425 (2021), p. 109901, https://doi.org/10.1016/j.jcp.2020.109901, https://www.sciencedirect.com/science/article/pii/S0021999120306756 (accessed 2024-07-12).

[7] A. Blanchard and T. Sapsis, *Output-Weighted Optimal Sampling for Bayesian Experimental Design and Uncertainty Quantification*, SIAM/ASA Journal on Uncertainty Quantification, 9 (2021), pp. 564–592, https://doi.org/10.1137/20M1347486, https://epubs.siam.org/doi/10.1137/20M1347486 (accessed 2024-07-12).

[8] D. Cai, A. J. Majda, D. W. McLaughlin, and E. G. Tabak, *Spectral bifurcations in dispersive wave turbulence*, Proceedings of the National Academy of Sciences, 96 (1999), pp. 14216–14221, https://doi.org/10.1073/pnas.96.25.14216, https://www.pnas.org/doi/abs/10.1073/pnas.96.25.14216 (accessed 2024-07-12). Publisher: Proceedings of the National Academy of Sciences.

[9] A.-T. Charalampopoulos, S. Zhang, B. Harrop, L.-y. R. Leung, and T. Sapsis, *Statistics of extreme events in coarse-scale climate simulations via machine learning correction operators trained on nudged datasets*, Apr. 2023, https://doi.org/10.48550/arXiv.2304.02117, http://arxiv.org/abs/2304.02117 (accessed 2024-07-12). arXiv:2304.02117 [physics].

[10] L. Chen, F. Du, Y. Hu, Z. Wang, and F. Wang, *SwinRDM: Integrate SwinRNN with Diffusion Model towards High-Resolution and High-Quality Weather Forecasting*, Proceedings of the AAAI Conference on Artificial Intelligence, 37 (2023), pp. 322–330, https://doi.org/10.1609/aaai.v37i1.25105, https://ojs.aaai.org/index.php/AAAI/article/view/25105.

[11] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, *Active Learning with Statistical Models*, Feb. 1996, https://doi.org/10.48550/arXiv.cs/9603104, http://arxiv.org/abs/cs/9603104 (accessed 2023-08-31). arXiv:cs/9603104.

[12] W. Cousins and T. P. Sapsis, *Quantification and prediction of extreme events in a one-dimensional nonlinear dispersive wave model*, Physica D: Nonlinear Phenomena, 280-281 (2014), pp. 48–58, https://doi.org/https://doi.org/10.1016/j.physd.2014.04.012, https://www.sciencedirect.com/science/article/pii/S016727891400092X.

[13] J. M. Dennis, J. Edwards, K. J. Evans, O. Guba, P. H. Lauritzen, A. A. Mirin, A. St-Cyr, M. A. Taylor, and P. H. Worley, *CAM-SE: A scalable spectral element dynamical core for the Community Atmosphere Model*, The International Journal of High Performance Computing Applications, 26 (2012), pp. 74–89, https://doi.org/10.1177/1094342011428142, https://doi.org/10.1177/1094342011428142 (accessed 2024-07-12). Publisher: SAGE Publications Ltd STM.

[14] T. Fiedler, A. J. Pitman, K. Mackenzie, N. Wood, C. Jakob, and S. E. Perkins-Kirkpatrick, *Business risk and the emergence of climate analytics*, Nature Climate Change, 11 (2021), pp. 87–94, https://doi.org/10.1038/s41558-020-00984-6, https://www.nature.com/articles/s41558-020-00984-6 (accessed 2024-07-12). Publisher: Nature Publishing Group.

[15] E. M. Fischer, S. Sippel, and R. Knutti, *Increasing probability of record-shattering climate extremes*,

Nature Climate Change, 11 (2021), pp. 689–695, https://doi.org/10.1038/s41558-021-01092-9, https://www.nature.com/articles/s41558-021-01092-9 (accessed 2024-07-12). Publisher: Nature Publishing Group.

[16] Y. Gal and Z. Ghahramani, *Dropout as a bayesian approximation: Representing model uncertainty in deep learning*, in international conference on machine learning, PMLR, 2016, pp. 1050–1059.

[17] J.-C. Golaz, L. P. Van Roekel, X. Zheng, A. F. Roberts, J. D. Wolfe, W. Lin, A. M. Bradley, Q. Tang, M. E. Maltrud, R. M. Forsyth, C. Zhang, T. Zhou, K. Zhang, C. S. Zender, M. Wu, H. Wang, A. K. Turner, B. Singh, J. H. Richter, Y. Qin, M. R. Petersen, A. Mametjanov, P.-L. Ma, V. E. Larson, J. Krishna, N. D. Keen, N. Jeffery, E. C. Hunke, W. M. Hannah, O. Guba, B. M. Griffin, Y. Feng, D. Engwirda, A. V. Di Vittorio, C. Dang, L. M. Conlon, C.-C.-J. Chen, M. A. Brunke, G. Bisht, J. J. Benedict, X. S. Asay-Davis, Y. Zhang, M. Zhang, X. Zeng, S. Xie, P. J. Wolfram, T. Vo, M. Veneziani, T. K. Tesfa, S. Sreepathi, A. G. Salinger, J. E. J. Reeves Eyre, M. J. Prather, S. Mahajan, Q. Li, P. W. Jones, R. L. Jacob, G. W. Huebler, X. Huang, B. R. Hillman, B. E. Harrop, J. G. Foucar, Y. Fang, D. S. Comeau, P. M. Caldwell, T. Bartoletti, K. Balaguru, M. A. Taylor, R. B. McCoy, L. R. Leung, and D. C. Bader, *The DOE E3SM Model Version 2: Overview of the Physical Model and Initial Model Evaluation*, Journal of Advances in Modeling Earth Systems, 14 (2022), p. e2022MS003156, https://doi.org/10.1029/2022MS003156, https://doi.org/10.1029/2022MS003156 (accessed 2024-07-12). Publisher: John Wiley & Sons, Ltd.

[18] R. B. Gramacy and H. K. H. Lee, *Adaptive Design and Analysis of Supercomputer Experiments*, Technometrics, 51 (2009), pp. 130–145, https://doi.org/10.1198/TECH.2009.0015, https://doi.org/10.1198/TECH.2009.0015 (accessed 2024-07-12). Publisher: Taylor & Francis _eprint: https://doi.org/10.1198/TECH.2009.0015.

[19] S. Guth, A. Mojahed, and T. P. Sapsis, *Quality measures for the evaluation of machine learning architectures on the quantification of epistemic and aleatoric uncertainties in complex dynamical systems*, Computer Methods in Applied Mechanics and Engineering, 420 (2024), p. 116760, https://doi.org/10.1016/j.cma.2024.116760, https://www.sciencedirect.com/science/article/pii/S0045782524000161.

[20] H. Hersbach, B. Bell, P. Berrisford, S. Hirahara, A. Horányi, J. Muñoz-Sabater, J. Nicolas, C. Peubey, R. Radu, D. Schepers, A. Simmons, C. Soci, S. Abdalla, X. Abellan, G. Balsamo, P. Bechtold, G. Biavati, J. Bidlot, M. Bonavita, G. De Chiara, P. Dahlgren, D. Dee, M. Diamantakis, R. Dragani, J. Flemming, R. Forbes, M. Fuentes, A. Geer, L. Haimberger, S. Healy, R. J. Hogan, E. Hólm, M. Janisková, S. Keeley, P. Laloyaux, P. Lopez, C. Lupu, G. Radnoti, P. de Rosnay, I. Rozum, F. Vamborg, S. Villaume, and J.-N. Thépaut, *The ERA5 global reanalysis*, Quarterly Journal of the Royal Meteorological Society, 146 (2020), pp. 1999–2049, https://doi.org/https://doi.org/10.1002/qj.3803, https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3803. _eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/qj.3803.

[21] T. Houser, S. Hsiang, R. Kopp, K. Larsen, M. Delgado, A. Jina, M. Mastrandrea, S. Mohan, R. Muir-Wood, D. J. Rasmussen, J. Rising, and P. Wilson, *Economic Risks of Climate Change: An American Prospectus*, Columbia University Press, Aug. 2015, https://doi.org/10.7312/hous17456, https://www.degruyter.com/document/doi/10.7312/hous17456/html (accessed 2024-07-12).

[22] E. Hüllermeier and W. Waegeman, *Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods*, Machine learning, 110 (2021), pp. 457–506. Publisher: Springer.

[23] L.-A. Kautz, O. Martius, S. Pfahl, J. G. Pinto, A. M. Ramos, P. M. Sousa, and T. Woollings, *Atmospheric blocking and weather extremes over the Euro-Atlantic sector – a review*, Weather and Climate Dynamics, 3 (2022), pp. 305–336, https://doi.org/10.5194/wcd-3-305-2022, https://wcd.copernicus.org/articles/3/305/2022/ (accessed 2024-04-24). Publisher: Copernicus GmbH.

[24] R. Keisler, *Forecasting Global Weather with Graph Neural Networks*, 2022, https://arxiv.org/abs/2202.07575.

[25] D. Kochkov, J. Yuval, I. Langmore, P. Norgaard, J. Smith, G. Mooers, M. Klöwer, J. Lottes, S. Rasp, P. Düben, S. Hatfield, P. Battaglia, A. Sanchez-Gonzalez, M. Willson, M. P. Brenner, and S. Hoyer, *Neural general circulation models for weather and climate*, Nature, (2024), https://doi.org/10.1038/s41586-024-07744-y, https://doi.org/10.1038/s41586-024-07744-y.

[26] B. Lakshminarayanan, A. Pritzel, and C. Blundell, *Simple and Scalable Predictive Uncertainty*

Estimation using Deep Ensembles, 2017, https://arxiv.org/abs/1612.01474.

[27] H. Li, D. Fernex, R. Semaan, J. Tan, M. Morzyński, and B. R. Noack, *Cluster-based network model*, Journal of Fluid Mechanics, 906 (2021), p. A21, https://doi.org/10.1017/jfm.2020.785, https://www.cambridge.org/core/journals/journal-of-fluid-mechanics/article/clusterbased-network-model/8252D04A5438ED01E624A7C41CCF81BB (accessed 2024-07-12).

[28] D. J. C. MacKay, *Information-Based Objective Functions for Active Data Selection*, Neural Computation, 4 (1992), pp. 590–604, https://doi.org/10.1162/neco.1992.4.4.590.

[29] A. J. Majda, D. W. McLaughlin, and E. G. Tabak, *A one-dimensional model for dispersive wave turbulence*, Journal of Nonlinear Science, 7 (1997), pp. 9–44, https://doi.org/10.1007/BF02679124, https://doi.org/10.1007/BF02679124.

[30] S. Manabe, J. Smagorinsky, and R. F. Strickler, *SIMULATED CLIMATOLOGY OF A GENERAL CIRCULATION MODEL WITH A HYDROLOGIC CYCLE*, (1965), https://journals.ametsoc.org/view/journals/mwre/93/12/1520-0493_1965_093_0769_scoagc_2_3_co_2.xml (accessed 2024-07-12). Section: Monthly Weather Review.

[31] Y. Mintz, *Very Long-Term Global Integration of the Primitive Equations of Atmospheric Motion: An Experiment in Climate Simulation*, in Causes of Climatic Change: A collection of papers derived from the INQUA—NCAR Symposium on Causes of Climatic Change, August 30–31, 1965, Boulder, Colorado, D. E. Billings, W. S. Broecker, R. A. Bryson, A. Cox, P. E. Damon, W. L. Donn, E. Eriksson, M. Ewing, J. O. Fletcher, W. Hamilton, M. Jerzykiewicz, J. E. Kutzbach, E. N. Lorenz, Y. Mintz, J. M. Mitchell, B. Saltzman, K. Serkowski, W. C. Shen, H. E. Suess, W. F. Tanner, P. K. Weyl, L. V. Worthington, and J. M. Mitchell, eds., American Meteorological Society, Boston, MA, 1968, pp. 20–36, https://doi.org/10.1007/978-1-935704-38-6_3, https://doi.org/10.1007/978-1-935704-38-6_3 (accessed 2024-07-12).

[32] M. A. Mohamad and T. P. Sapsis, *Sequential sampling strategy for extreme event statistics in nonlinear dynamical systems*, Proceedings of the National Academy of Sciences, 115 (2018), pp. 11138–11143, https://doi.org/10.1073/pnas.1813263115, https://www.pnas.org/doi/full/10.1073/pnas.1813263115 (accessed 2024-07-12). Publisher: Proceedings of the National Academy of Sciences.

[33] S. K. Mukkavilli, D. S. Civitarese, J. Schmude, J. Jakubik, A. Jones, N. Nguyen, C. Phillips, S. Roy, S. Singh, C. Watson, R. Ganti, H. Hamann, U. Nair, R. Ramachandran, and K. Weldemariam, *AI Foundation Models for Weather and Climate: Applications, Design, and Implementation*, Sept. 2023, https://doi.org/10.48550/arXiv.2309.10808, http://arxiv.org/abs/2309.10808 (accessed 2024-07-12). arXiv:2309.10808 [physics].

[34] K. P. Murphy, *Probabilistic machine learning: an introduction*, MIT press, 2022.

[35] E. Pickering, S. Guth, G. E. Karniadakis, and T. P. Sapsis, *Discovering and forecasting extreme events via active learning in neural operators*, Nature Computational Science, 2 (2022), pp. 823–833, https://doi.org/10.1038/s43588-022-00376-0, https://doi.org/10.1038/s43588-022-00376-0.

[36] E. Pickering and T. P. Sapsis, *Information FOMO: The unhealthy fear of missing out on information. A method for removing misleading data for healthier models*, Sept. 2022, https://doi.org/10.48550/arXiv.2208.13080, http://arxiv.org/abs/2208.13080 (accessed 2023-05-15). arXiv:2208.13080 [physics, stat].

[37] A. F. Psaros, X. Meng, Z. Zou, L. Guo, and G. E. Karniadakis, *Uncertainty quantification in scientific machine learning: Methods, metrics, and comparisons*, Journal of Computational Physics, 477 (2023), p. 111902, https://doi.org/https://doi.org/10.1016/j.jcp.2022.111902, https://www.sciencedirect.com/science/article/pii/S0021999122009652.

[38] A. Pushkarev and V. E. Zakharov, *Quasibreathers in the MMT model*, Physica D: Nonlinear Phenomena, 248 (2013), pp. 55–61, https://doi.org/10.1016/j.physd.2013.01.003, https://www.sciencedirect.com/science/article/pii/S0167278913000079 (accessed 2024-07-12).

[39] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, Nov. 2005, https://doi.org/10.7551/mitpress/3206.001.0001, https://doi.org/10.7551/mitpress/3206.001.0001.

[40] S. Rasp, P. D. Dueben, S. Scher, J. A. Weyn, S. Mouatadid, and N. Thuerey, *WeatherBench: A Benchmark Data Set for Data-Driven Weather Forecasting*, Journal of Advances in Modeling Earth Systems, 12 (2020), p. e2020MS002203, https://doi.org/https://doi.org/10.1029/2020MS002203, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2020MS002203.

[41] S. Rasp, S. Hoyer, A. Merose, I. Langmore, P. Battaglia, T. Russell, A. Sanchez-Gonzalez, V. Yang, R. Carver, S. Agrawal, M. Chantry, Z. Ben Bouallegue, P. Dueben, C. Bromberg, J. Sisk, L. Barrington, A. Bell, and F. Sha, *WeatherBench 2: A Benchmark for the Next Generation of Data-Driven Global Weather Models*, Journal of Advances in Modeling Earth Systems, 16 (2024), p. e2023MS004019, https://doi.org/https://doi.org/10.1029/2023MS004019, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2023MS004019.

[42] S. Rasp and S. Lerch, *Neural Networks for Postprocessing Ensemble Weather Forecasts*, Monthly Weather Review, 146 (2018), pp. 3885 – 3900, https://doi.org/10.1175/MWR-D-18-0187.1, https://journals.ametsoc.org/view/journals/mwre/146/11/mwr-d-18-0187.1.xml.

[43] S. Rasp and N. Thuerey, *Data-Driven Medium-Range Weather Prediction With a Resnet Pretrained on Climate Simulations: A New Model for WeatherBench*, Journal of Advances in Modeling Earth Systems, 13 (2021), p. e2020MS002405, https://doi.org/https://doi.org/10.1029/2020MS002405, https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2020MS002405.

[44] C. Raymond, R. M. Horton, J. Zscheischler, O. Martius, A. AghaKouchak, J. Balch, S. G. Bowen, S. J. Camargo, J. Hess, K. Kornhuber, M. Oppenheimer, A. C. Ruane, T. Wahl, and K. White, *Understanding and managing connected extreme events*, Nature Climate Change, 10 (2020), pp. 611–621, https://doi.org/10.1038/s41558-020-0790-4, https://www.nature.com/articles/s41558-020-0790-4 (accessed 2024-07-12). Publisher: Nature Publishing Group.

[45] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang, *A Survey of Deep Active Learning*, ACM Comput. Surv., 54 (2021), https://doi.org/10.1145/3472291, https://doi.org/10.1145/3472291. Place: New York, NY, USA Publisher: Association for Computing Machinery.

[46] A. Robinson, J. Lehmann, D. Barriopedro, S. Rahmstorf, and D. Coumou, *Increasing heat and rainfall extremes now far outside the historical climate*, npj Climate and Atmospheric Science, 4 (2021), pp. 1–4, https://doi.org/10.1038/s41612-021-00202-w, https://www.nature.com/articles/s41612-021-00202-w (accessed 2024-07-12). Publisher: Nature Publishing Group.

[47] T. P. Sapsis, *Output-weighted optimal sampling for Bayesian regression and rare event statistics using few samples*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 476 (2020), p. 20190834, https://doi.org/10.1098/rspa.2019.0834, https://royalsocietypublishing.org/doi/10.1098/rspa.2019.0834 (accessed 2024-07-12).

[48] T. P. Sapsis, *Statistics of Extreme Events in Fluid Flows and Waves*, Annual Review of Fluid Mechanics, 53 (2021), pp. 85–111, https://doi.org/10.1146/annurev-fluid-030420-032810, https://www.annualreviews.org/content/journals/10.1146/annurev-fluid-030420-032810 (accessed 2024-07-12).

[49] T. P. Sapsis and A. Blanchard, *Optimal criteria and their asymptotic form for data selection in data-driven reduced-order modelling with Gaussian process regression*, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 380 (2022), p. 20210197, https://doi.org/10.1098/rsta.2021.0197, https://royalsocietypublishing.org/doi/10.1098/rsta.2021.0197 (accessed 2024-07-12).

[50] T. Schneider, S. Behera, G. Boccaletti, C. Deser, K. Emanuel, R. Ferrari, L. R. Leung, N. Lin, T. Müller, A. Navarra, O. Ndiaye, A. Stuart, J. Tribbia, and T. Yamagata, *Harnessing AI and computing to advance climate modelling and prediction*, Nature Climate Change, 13 (2023), pp. 887–889, https://doi.org/10.1038/s41558-023-01769-3, https://doi.org/10.1038/s41558-023-01769-3.

[51] T. Schneider, S. Lan, A. Stuart, and J. Teixeira, *Earth System Modeling 2.0: A Blueprint for Models That Learn From Observations and Targeted High-Resolution Simulations*, Geophysical Research Letters, 44 (2017), pp. 12,396–12,417, https://doi.org/10.1002/2017GL076101, https://onlinelibrary.wiley.com/doi/abs/10.1002/2017GL076101 (accessed 2024-07-12). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2017GL076101.

[52] B. Settles, *Active Learning Literature Survey*, technical Report, University of Wisconsin-Madison Department of Computer Sciences, 2009, https://minds.wisconsin.edu/handle/1793/60660 (accessed 2023-06-02).

[53] J. Slingo, P. Bates, P. Bauer, S. Belcher, T. Palmer, G. Stephens, B. Stevens, T. Stocker, and G. Teutsch, *Ambitious partnership needed for reliable climate prediction*, Nature Climate Change, 12 (2022), pp. 499–503, https://doi.org/10.1038/s41558-022-01384-8, https://www.nature.

com/articles/s41558-022-01384-8 (accessed 2024-07-12). Publisher: Nature Publishing Group.

[54] J. Smagorinsky, S. Manabe, and J. L. Holloway, *NUMERICAL RESULTS FROM A NINE-LEVEL GENERAL CIRCULATION MODEL OF THE ATMOSPHERE 1*, (1965), https://journals.ametsoc.org/view/journals/mwre/93/12/1520-0493_1965_093_0727_nrfanl_2_3_co_2.xml (accessed 2024-07-12). Section: Monthly Weather Review.

[55] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*, Journal of Machine Learning Research, 15 (2014), pp. 1929–1958, http://jmlr.org/papers/v15/srivastava14a.html.

[56] M. A. Taylor, A. S. Cyr, and A. Fournier, *A Non-oscillatory Advection Operator for the Compatible Spectral Element Method*, in Computational Science – ICCS 2009, G. Allen, J. Nabrzyski, E. Seidel, G. D. van Albada, J. Dongarra, and P. M. A. Sloot, eds., Berlin, Heidelberg, 2009, Springer Berlin Heidelberg, pp. 273–282.

[57] H. Tomita, H. Miura, S. Iga, T. Nasuno, and M. Satoh, *A global cloud-resolving simulation: Preliminary results from an aqua planet experiment*, Geophysical Research Letters, 32 (2005), https://doi.org/10.1029/2005GL022459, https://onlinelibrary.wiley.com/doi/abs/10.1029/2005GL022459 (accessed 2024-07-12). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2005GL022459.

[58] Y. Yang, A. Blanchard, T. Sapsis, and P. Perdikaris, *Output-weighted sampling for multi-armed bandits with extreme payoffs*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 478 (2022), p. 20210781, https://doi.org/10.1098/rspa.2021.0781, https://royalsocietypublishing.org/doi/10.1098/rspa.2021.0781 (accessed 2024-07-12).

[59] V. Zakharov, F. Dias, and A. Pushkarev, *One-dimensional wave turbulence*, Physics Reports, 398 (2004), pp. 1–65, https://doi.org/10.1016/j.physrep.2004.04.002, https://www.sciencedirect.com/science/article/pii/S0370157304002212 (accessed 2024-07-12).

[60] V. E. Zakharov, P. Guyenne, A. N. Pushkarev, and F. Dias, *Wave turbulence in one-dimensional models*, Physica D: Nonlinear Phenomena, 152-153 (2001), pp. 573–619, https://doi.org/10.1016/S0167-2789(01)00194-4, https://www.sciencedirect.com/science/article/pii/S0167278901001944 (accessed 2024-07-12).

[61] S. Zhang, B. E. Harrop, L. R. Leung, A.-T. Charalampopoulos, B. Barthel, W. W. Xu, and T. Sapsis, *A Machine Learning Bias Correction of Large-scale Environment of Extreme Weather Events in E3SM Atmosphere Model*, https://www.authorea.com/doi/full/10.22541/essoar.170067232.22392274?commit=0d2c5e24d88d73abb13e16b7022dd81d72be0977 (accessed 2024-07-12).

[62] X. Zhou, H. Liu, F. Pourpanah, T. Zeng, and X. Wang, *A Survey on Epistemic (Model) Uncertainty in Supervised Learning: Recent Advances and Applications*, Neurocomputing, 489 (2022), pp. 449–465, https://doi.org/10.1016/j.neucom.2021.10.119, http://arxiv.org/abs/2111.01968 (accessed 2023-05-15).

[63] Z. Zou, X. Meng, A. F. Psaros, and G. E. Karniadakis, *NeuralUQ: A Comprehensive Library for Uncertainty Quantification in Neural Differential Equations and Operators*, SIAM Review, 66 (2024), pp. 161–190, https://doi.org/10.1137/22M1518189, https://doi.org/10.1137/22M1518189 (accessed 2024-07-10). Publisher: Society for Industrial and Applied Mathematics.