Nonlinear Dynamics manuscript No.
(will be inserted by the editor)

What are the most informative data points for predicting extreme events?

Likelihood-Weighted Data Selection to Improve the Prediction of Extreme Events in Complex Dynamical Systems

Bianca Champenois · Themistoklis Sapsis

Received: date / Accepted: date

Abstract Large datasets generated from complex dynamical systems, such as climate models or turbulence simulations, have made machine learning a popular tool for prediction and analysis. However, these datasets are often vast and imbalanced, with extreme events poorly represented. As a result, standard machine learning approaches struggle to accurately capture rare, high-impact behavior. This raises a fundamental question: which data points should we use to train models that effectively learn the dynamics of extremes? To address this, we propose a model-agnostic active data selection approach that sequentially identifies an optimal subset of the most informative data points for model training. Unlike traditional active learning, which assumes the ability to query new data, our method is designed for problems where the dataset is fixed but large — focusing on selection rather than acquisition. Points are scored using a likelihood-weighted uncertainty sampling criterion that prioritizes samples expected to reduce model uncertainty and improve predictions in the tails of the distribution for systems with non-Gaussian statistics. We first validate the method on a benchmark problem, quantifying extreme wave statistics in a turbulent system. We then apply the method to a real-world application: 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. We also introduce analysis techniques to further interpret the optimally selected points. Looking ahead, the approach can serve as a compression algorithm that preserves information associated with extreme events in vast datasets.

Keywords active learning · active data selection · extreme events · complex dynamical systems · climate modeling · machine learning

B. Champenois
Massachusetts Institute of Technology
E-mail: bchamp@mit.edu

T. Sapsis
Massachusetts Institute of Technology

Mathematics Subject Classification (2020) MSC 86A08 · MSC 86-08 · MSC 62G32 · MSC 60G70 · MSC 62L05

1 Introduction

Many important scientific and engineering problems involve complex dynamical systems, such as turbulent flows or the climate. In many such settings, we now have access to terabytes or even petabytes of simulation or observational data. This abundance of data, coupled with the intractable complexity of the underlying physics, has made machine learning (ML) an increasingly popular tool for modeling and analysis of dynamical systems. However, extreme events, high-impact events that lie in the tails of the probability distribution (e.g. rogue waves or extreme weather), are typically underrepresented in datasets not explicitly designed to capture extremes [55]. Standard ML models trained on the available datasets tend to prioritize the regions of the domain where most points exist. As a result, they often fail to capture extremes or converge slowly, resulting in poor generalization where accurate predictions matter most [41]. From this perspective, not all points in a given dataset carry the same *value* of information, so using the full dataset or a random subset of the dataset without proper selection can be inefficient or ineffective for training. Therefore, identifying a subset of training data that is most informative for extreme event prediction can reduce computational cost while improving the model's ability to capture the system's full statistics.

We present an adaptation of the active learning framework for efficient training data selection in non-Gaussian systems. Unlike traditional active learning settings with controlled data acquisition, our method addresses problems where data are vast but predefined, such as large simulation or observational datasets. Our proposed method, introduced in Section 2, assigns value to points in a dataset according to a selection criterion based on the likelihood-weighted uncertainty [54, 6, 56]. Section 2.3 outlines how we quantify the uncertainty needed to evaluate this selection criterion. Section 2.4 describes how we extend the method to systems for which the inputs are high-dimensional functionals. Through this extension, the method demonstrates effectiveness even in the presence of extremely high-dimensional inputs. This capability is particularly important for scientific applications where inputs are inherently high-dimensional, such as turbulent flows or the climate. In Section 3, we apply the framework to predict extreme events in the Majda–McLaughlin–Tabak (MMT) model, a one-dimensional model for dispersive wave turbulence. In Section 4, we apply it to a correction operator for coarse-resolution climate model outputs. In both cases, we introduce methods to interpret the selected points and gain insight into the algorithm.

Overall, this review paper examines likelihood-weighted sampling as a tool for improving the prediction of extreme events in data-driven models of complex dynamical systems. We highlight how this framework shifts the focus from data acquisition to data selection — an important distinction in settings where large simulation or observational datasets already exist. This perspective enables efficient data compression by prioritizing informative samples and provides a systematic methodology for extracting insights from vast datasets. The data selection framework (i) identifies the most informative points in large datasets for predicting a target observable, (ii) reduces training cost by using only high-value data,

(iii) improves model generalization for extreme events, and (iv) allows for the interpretation of the selected points. A key advantage of the approach is that it is model-agnostic and adaptable to any ML architecture. The paper also reviews techniques for extending the framework to high-dimensional inputs typical of dynamical systems. We demonstrate the method's utility through an application to climate modeling, where it is used to improve predictions of extreme weather events — an area of growing interest.

2 Likelihood-Weighted Data Selection

Active learning is a form of supervised machine learning where new data points are sequentially chosen according to a criterion called the acquisition function [34, 12, 20, 60]. Ren et al. [51] provides a survey of active learning in the context of ML classification models. Active learning 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 [22, 26]. Here, we adapt the active learning framework for the case where the new points must be selected from a pre-existing, pre-computed dataset rather than from a continuous domain [45**?** , 2, 21]. This distinction is sometimes referred to as active search, greedy approximations, optimal sampling, or active sampling, but we will refer to it as data selection.

2.1 Data Selection Algorithm

The goal of the data selection algorithm is to find the subset of points $\mathbf{X}$ that are the most "valuable" with respect to predicting a target observable y. We initialize the data selection algorithm (illustrated in Figure 1) with a small training set consisting of randomly chosen candidate points. During each iteration, the model is trained again, and the selection criterion is evaluated on the remaining candidate points. This criterion is computed using the mean and epistemic uncertainty of the target observable predicted by a probabilistic model. Points that maximize the criterion are considered optimal and are added to the training set in batches. More details on batching are provided in Pickering et al. [42]. The loop is repeated until the model error converges or falls below a predefined threshold. The output of the algorithm is a model trained on optimally selected data. At each step, the selected points can be further analyzed to understand what types of points are most informative for the learning task.

2.2 Selection Criterion: Likelihood-Weighted Uncertainty

The key element of the data selection algorithm is the selection criterion that identifies the most valuable points for model training. The choice of the selection criterion can depend on the nature of the system (e.g. non-Gaussian, high-dimensional), the goal of the modeling problem (e.g. optimization, extreme event identification), and other constraints (e.g. computational costs). In general, the selection criterion should strike a balance between exploration and exploitation.

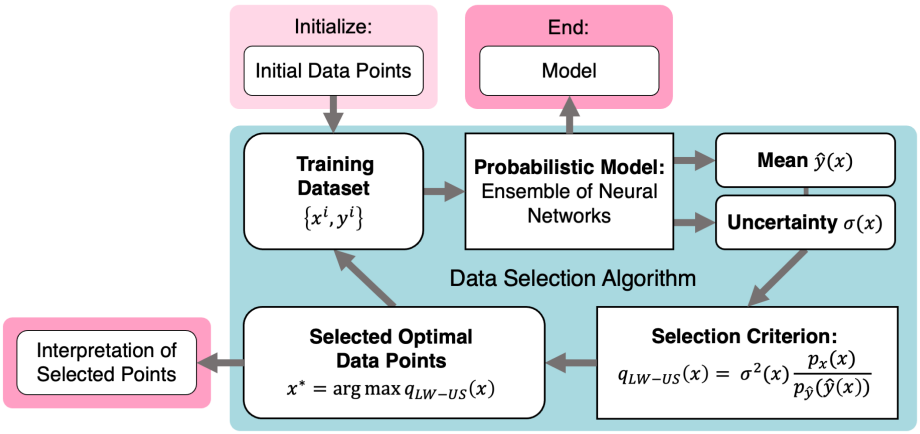

Fig. 1 Data Selection Algorithm. Points are sequentially selected according to the selection criterion and added to the training set to improve model prediction. The output of the algorithm is a model that has been trained on an optimal subset of the data with respect to predicting the traget observable's statistics.

A common choice for the selection criterion is uncertainty sampling (US), where points are ranked by their epistemic variance.

$$q_{\mathrm{US}}(\mathbf{X}) = \sigma^2(\mathbf{X}). \tag{1}$$

A modified version of uncertainty sampling, called input-weighted US, prioritizes points by weighting the epistemic variance with the probability of the input points.

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

However, neither criterion considers the expected output and, therefore, fails to account for the importance of extreme events. For problems with extreme events, we instead use a likelihood-weighted uncertainty sampling (LW-US) criterion to select optimal training points and quantify the value of points in the dataset. The LW-US selection criterion, like US and input-weighted US, targets input points that are likely and reduce uncertainty, but it also prioritizes points expected to produce extreme outputs. Sampling criteria that take into consideration the output were first introduced in Mohamad and Sapsis [38] and further improved in Sapsis [54] for systems 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 largest uncertainty.

$$D_{\mathrm{Log}^1}(y||y_0; h) = \int_{S_y} \left|\log p_{y_+}(y) - \log p_{y_0}(y)\right| dy. \tag{3}$$

For a bounded domain S_y and a candidate sample point h, this selection criterion asymptotically converges to the desired output statistics, even in regions with low probability of occurrence [56]. However, Equation 3 is computationally expensive

and its lack of smooth gradients makes it unsuitable for gradient-based optimization needed in active learning settings. Instead, we use an upper bound (derived in Sapsis [54]) that has a lower cost of computation and is analytically differentiable

$$q_{\text{LW-US}}(\mathbf{X}) = \int_{S_x} \sigma^2(\mathbf{X}) \frac{p_x(\mathbf{X})}{p_y(y(\mathbf{X}))}. \tag{4}$$

In this modified version, the epistemic variance $\sigma^2(\mathbf{X})$ quantifies uncertainty reduction, the input probability $p_x(\mathbf{X})$ emphasizes points likely to occur, and the inverse output probability $1/p_y(y(\mathbf{X}))$ emphasizes points leading to extreme outcomes. Together, these terms prioritize candidate points that reduce uncertainty, are probable inputs, and most importantly, lead to extreme events. When used for model training, these points result in models that are better able to represent the tails of the distribution in non-Gaussian settings. The criterion can also be thought of as a *scoring* function because it gives priority to data points with the highest *value* with respect to improving the statistics of the target observable.

We measure success in terms of minimizing the error in the tails of the pdf. To do so, we consider the log-pdf error (LPE) that measures the integrated difference between the log of the true pdf obtained from the true y and the log of the conditional pdf of the model's posterior mean $\hat{y}$.

$$\text{LPE} = \int |\log p_y(y) - \log p_{\hat{y}}(\hat{y})| \mathrm{d}y. \tag{5}$$

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 $p_y(y)$.

We benchmark our method against a Monte Carlo (MC) selection criterion that randomly selects points from the candidate set. MC serves as a meaningful benchmark because randomly splitting data into training, validation, and test sets is a common practice in machine learning.

2.3 Probabilistic Model: Ensemble of Neural Networks

The selection criterion requires an estimate for the epistemic uncertainty of the model. In previous works, the uncertainty is quantified using traditional Bayesian supervised learning methods such as Bayesian regression or Gaussian process regression [54, 7, 6, 67]. However, these methods are limited: Bayesian regression can fail when modeling nonlinear systems, and Gaussian process regression suffers from performance issues on high-dimensional or large datasets. Newer neural network architectures that can quantify uncertainty are able to overcome these problems [40, 25, 23, 70, 43, 71, 32]. 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 Angelopoulos and Bates [1].

In the E-NN, multiple models with identical architectures and hyperparameters are trained on the same dataset, each initialized with different random weights. The resulting prediction $\hat{y}$ is the mean of the n neural network predictions $\hat{y}_i$

$$\hat{y}(\mathbf{X}) = \frac{1}{n} \sum_{i=1}^{n} \hat{y}_i(\mathbf{X}) \tag{6}$$

where $\hat{y}_i$ is the prediction of the i^{th} neural network in the ensemble. Similarly, the model uncertainty can be quantified via the variance of the predictions

$$\sigma^2(\mathbf{X}) = \frac{1}{n-1}\sum_{i=1}^{n}\left(\hat{y}_i(\mathbf{X}) - \hat{y}(\mathbf{X})\right)^2. \tag{7}$$

Guth et al. [23] and Pickering and Sapsis [41] showed that even small ensembles of neural networks, as low as $N = 2$, can still perform well. In a D-NN, only one model is trained, but the model includes dropout layers to introduce stochasticity [63]. During the prediction step, multiple predictions are made with different randomly dropped nodes, resulting in higher variance predictions [18]. 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. One advantage of using E-NN or D-NN is that they make use of existing neural networks already developed for the application of interest, eliminating the need to construct a new surrogate model.

2.4 Application to Functional Inputs

Many scientific problems, including the ones we consider, involve functionals — mappings from a (possibly infinite-dimensional) space to a real number. In these settings, the regression task can be viewed as a composition of mappings:

$$\mathbf{X} \mapsto \mathbf{U} \mapsto y,$$

where $\mathbf{X}$ is the high-dimensional input field, $\mathbf{U}$ is a field produced by a physical or ML-based model, and y is a scalar observable extracted from $\mathbf{U}$ (Figure 2). For example, in the MMT application in Section 3, Y is the maximum wave amplitude reached over a given time horizon, determined by evolving the initial wave field $\mathbf{X}$. In the climate modeling application in Section 4, the input $\mathbf{X}$ corresponds to the output of a low-resolution climate model, including fields such as temperature, wind, and humidity. This input is passed through a debiasing or correction operator to produce a higher-fidelity field $\mathbf{U}$. From this corrected output, a scalar observable Y is extracted — for example, the temperature at a specific location or a principal component coefficient representing a dominant mode of variability.

Applying the likelihood-weighted selection criterion in this context requires estimating the input distribution $p_x(\mathbf{X})$. However, this is generally intractable in high-dimensional settings. To overcome this challenge, we project the input onto a lower-dimensional subspace using standard techniques such as Principal Component Analysis (PCA), then estimate densities in that reduced space using either a Gaussian approximation or kernel density estimation (KDE) [8, 64].

Although many reduced-order modeling (ROM) approaches exist, we use PCA due to its simplicity, efficiency, and widespread use in physics-based applications. Importantly, the likelihood-weighted criterion remains general and can incorporate other ROM techniques that preserve the key statistical properties of the input space.

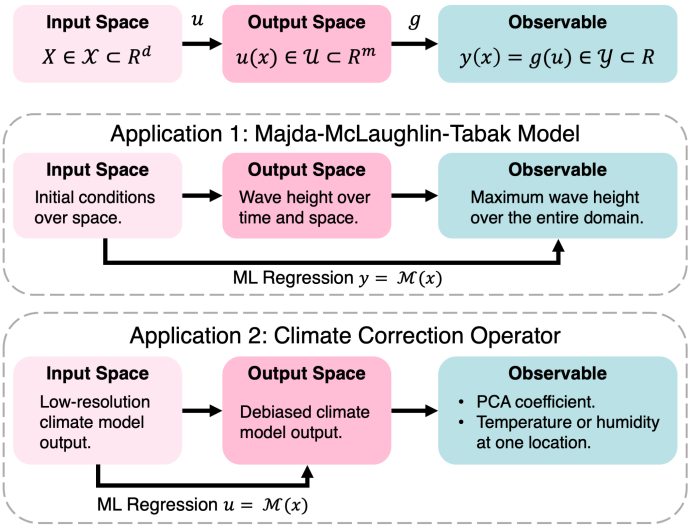

Fig. 2 Schematic of the mapping from input space to observable. Inputs $X \in \mathcal{X}$ are mapped to outputs $U \in \mathcal{U}$ through a forward model, and a scalar observable $y \in \mathbb{R}$ is extracted from U via an observable-specific operator $g(U)$. This framework is illustrated for two applications: in the MMT case (Section 3), the inputs are the initial conditions of the system, and the observable is the maximum wave height over a specified time horizon; in the climate model correction case (Section 4), the input is a coarse-resolution climate model, the output is its debiased counterpart, and the observable is a scalar quantity of interest such as a PCA coefficient or temperature/humidity at a specific location.

2.4.1 Weighted Principal Component Analysis for High-Dimensional Systems

Our input data consists of high- or infinite-dimensional fields, which makes direct estimation of input densities intractable. To overcome this, we first reduce the dimensionality of the input space using weighted principal component analysis (PCA) or, when the covariance function is known *a priori*, a Karhunen–Loève decomposition.

Formally, let $\mathbf{x}(\xi, t)$ denote a vector field defined over a spatial domain indexed by ξ, with temporal mean $\overline{\mathbf{x}}(\xi)$. We define the centered field

$$\mathbf{z}(t, \xi) = \mathbf{x}(t, \xi) - \overline{\mathbf{x}}(\xi),$$

and represent it in terms of orthonormal spatial modes $\{\psi_j(\xi)\}_{j=1}^N$, weighted to respect the domain geometry:

$$\mathbf{z}(t, \xi) = \sum_{j=1}^{N} \alpha_j(t)\psi_j(\xi),$$

where $\alpha_j(t)$ are time-dependent PCA coefficients, and N is the number of retained modes.

To incorporate the spatial geometry, we define a weighted inner product:

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

with a spatial weighting function $w(\xi)$. For example, the MMT application uses uniform weights $w(\xi) = 1$ whereas the climate application accounts for spherical geometry by choosing

$$w(\theta, \phi) = \sqrt{\sin\left(\frac{90^\circ - \theta}{180^\circ}\pi\right)},$$

where θ and ϕ denote latitude and longitude, respectively.

The covariance operator is estimated by time-averaging:

$$\mathbf{R}(\xi_1, \xi_2) = \frac{1}{T}\int_0^T \mathbf{z}(t, \xi_1)\mathbf{z}(t, \xi_2)\, dt \approx \frac{1}{n_t}\mathbf{Z}\mathbf{Z}^T,$$

where $\mathbf{Z} \in \mathbb{R}^{N \times n_t}$ is the snapshot matrix. The PCA modes ψ_j can be found by solving the eigenvalue problem

$$\langle \mathbf{R}(\cdot, \xi), \psi_j(\cdot)\rangle_w = \lambda_j \psi_j(\xi),$$

with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N \geq 0$.

The PCA coefficients are computed as projections:

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

The primary purpose of this dimensionality reduction is to enable the use of the data selection algorithm even for systems with very high-dimensional inputs, by projecting the input field onto a lower-dimensional basis that facilitates density estimation. Additionally, the same methodology allows us to define a scalar observable y by selecting appropriate modes or functions ψ. For example, if the quantity of interest is the first PCA coefficient, we set $j = 1$; alternatively, ψ may represent spatial averages, maximum values, or values at specific points. In such cases, we use $\langle x, \cdot\rangle$ to denote the corresponding extraction operator.

In our applications, this manifests as follows: In the climate modeling application (Section 4), weighted PCA is used first to reduce the input field dimensionality from the low-resolution climate model output. Optionally, it is also used to define the scalar observable y when this corresponds to a dominant mode of variability. In the MMT application (Section 3), the input initial conditions have a known covariance function and are obtained from a Karhunen–Loève expansion (equivalent to PCA under these conditions). In both cases, this approach projects the high-dimensional spaces onto a reduced basis, enabling a tractable evaluation of the input probability density and a flexible definition of the observable. In general, we note that the ML models themselves remain high-dimensional.

2.4.2 Evaluation of the Selection Criterion for Functional Inputs

We now bring together all components of the selection criterion for functional inputs. As a reminder, the likelihood-weighted selection criterion depends on both the input $\mathbf{X}$ and the corresponding predicted observable $\hat{y}$, and it incorporates three key terms: (i) the predictive uncertainty $\sigma^2(\mathbf{X})$, (ii) the pdf of the input space $p_x(\mathbf{X})$, and (iii) the pdf of the observable $p_{\hat{y}}(\hat{y}(\mathbf{X}))$:

$$q_{\text{LW-US}}(\mathbf{X}) = \sigma^2(\mathbf{X})\frac{p_x(\mathbf{X})}{p_{\hat{y}}(\hat{y}(\mathbf{X}))}. \tag{8}$$

We consider a labeled dataset $\mathcal{D}^{\text{TR}}$ from which training samples $\mathcal{X}$ — denoting infinite-dimensional input fields — are selected, and a separate set of inputs $\mathcal{D}^{\text{INF}}$ for which the model will be deployed and evaluated during inference. Although the observable's true value in the inference setting is unknown during training, its definition guides the selection process. We therefore optimize the selection of training points with respect to the observable's distribution expected during inference. This affects the output weighting term, which is evaluated over candidate training points but incorporates knowledge of the output distribution anticipated at inference time:

$$q_{\text{LW-US}}(\mathcal{X}) = \sigma^2(\mathcal{X}) \frac{p_{\mathbf{x}}(\mathbf{x})}{p_{\text{inf}}(\hat{y}_{\text{tr}}(\mathcal{X}))}. \tag{9}$$

We define $\mathcal{M}^j_{\mathcal{S}}$ as the members of a probabilistic ensemble trained on $\mathcal{S} = [\mathcal{X}_1, \ldots, \mathcal{X}_p]$, a subset of p selected training inputs from the full labeled dataset $\mathcal{D}^{\text{TR}}$. We evaluate the ensemble on both $\mathcal{X} \subset \mathcal{D}^{\text{TR}}$ and the $\mathcal{D}^{\text{INF}}$ to obtain $\hat{y}_{\text{tr}}$ and $\hat{y}_{\text{inf}}$, respectively. From these, we get

1. *Uncertainty*
   The predictive uncertainty at the candidate training points $\mathcal{X}$ is estimated as the variance of the predicted observable across the ensemble of models $\mathcal{M}^j_{\mathcal{S}}$:

$$\sigma^2(\mathcal{X}) = \frac{1}{n-1} \sum_{j=1}^{n} \left(\hat{y}^j_{\text{tr}}(\mathcal{X}) - \hat{y}_{\text{tr}}(\mathcal{X}) \right)^2 \tag{10}$$

2. *Input Probability*
   The input density of $\mathcal{X}$ is undefined when $\mathcal{X}$ is infinite dimensional, so we approximate the input density $p_{\mathbf{x}}(\mathbf{x})$ in two steps: first, by reducing the high-dimensional inputs $\mathcal{X}$ to their first k principal components, $\mathbf{x}$; and second, by estimating the density of $\mathbf{x}$ using either a Gaussian distribution, a KDE with a Gaussian kernel, or another appropriate approximation:

$$\mathcal{X} \approx \mathbf{x} = \left\langle \mathcal{X}, \{\psi_i\}_{i=1}^{k} \right\rangle_w \tag{11}$$

   The value of k is selected based on the decay of eigenvalues. The purpose of the input distribution is to prioritize likely scenarios, so an approximation is sufficient. Furthermore, although the input distribution is approximated in a reduced space, the full high-dimensional structure of the inputs can be preserved within the machine learning model itself.
3. *Output Probability* The density of the observable, p_{inf}, is estimated using KDE with a Gaussian kernel applied to the model predictions for the inference set $\hat{y}_{\text{inf}}$. KDE is fast and easy to compute for one-dimensional variables, and can be further accelerated using FFT-based methods. For one-dimensional variables, This density is then evaluated at the predicted observable for the training points $\hat{y}_{\text{tr}}$:

$$p_{\text{inf}}(\hat{y}_{\text{tr}}(\mathcal{X})) \tag{12}$$

   We note that this density is approximated using predictions for all of the samples in $\mathcal{D}^{\text{INF}}$, so the resulting estimate is well-resolved and considered representative of the predictive distribution of the observable at inference time.

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 is useful for studying turbulence and rogue waves [35]. More details on the overall system can be found in Cai et al. [9], Zakharov et al. [68], Pushkarev and Zakharov [44], Cousins and Sapsis [13], Zakharov et al. [69]. The system is described by the governing equation

$$iu_t = |\partial_x|^{\alpha} u + \lambda |\partial_x|^{-\beta/4} \left(\left| |\partial_x|^{-\beta/4} u \right|^2 |\partial_x|^{-\beta/4} u \right) + iDu \tag{13}$$

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 that eliminates high wave numbers. For $\alpha = 1/2$ and $\beta = 0$, the equation can be rewritten in the wave number space with forcing $f(k)$

$$\hat{u}(k)_t = -i|k|^{1/2}\hat{u}(k) - i\lambda |\hat{u}(k)|^2 \hat{u}(k) + \widehat{Du}(k) + f(k) \tag{14}$$

where the selective Laplacian is defined as

$$\widehat{Du}(k) = \begin{cases} -(|k| - k^*)^2 \hat{u}(k) & \text{if } |k| > k^* \\ 0 & \text{if } |k| \leq k^* \end{cases}. \tag{15}$$

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 function

$$k(x, x') = \sigma_u^2 \exp\left(i2 \sin^2(\pi(x - x')) \right) \exp\left(-\frac{2 \sin^2(\pi(x - x'))}{l_u^2} \right) \tag{16}$$

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

$$u(x, t = 0) \approx \sum_{j=1}^{m} \alpha_j \sqrt{\lambda_j} \phi_j(x), \ \forall \ x \in [0, 1) \tag{17}$$

to transform the original high-dimensional data into a set of orthogonal components. In this step, the Karhunen-Loève expansion is the continuous analogue of PCA described in Section 2.4.1. 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 Pickering et al. [42] and Guth et al. [23], 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 $\vec{\alpha}$

$$y(\vec{\alpha}) = ||\text{Re}(u(x, T = 50; \vec{\alpha}))||. \tag{18}$$

3.2 Datasets

To better mimic the characteristics of datasets that are found in the real world, we make use of two datasets: 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 datasets. 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 and LW-US sampling, we select candidate training points from $\mathcal{D}_{LHS}$ because this dataset 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 Machine Learning Architecture and Active Learning Hyperparameters

The ML model learns a mapping from the initial conditions to the maximum wave height over a given time horizon. 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 of size 5 (sufficient as shown in Guth et al. [23] and Pickering and Sapsis [41]). 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 value shown to be effective at estimating uncertainty in Guth et al. [23]. 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 dataset 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 selection criterion), 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 dataset) for randomly chosen points (MC), input-weighted uncertainty sampling (US), and likelihood-weighted uncertainty sampling (LW-US) are shown in Figure 3. Because we compute the mean squared error (MSE) with the Latin hypercube sampling dataset $\mathcal{D}_{LHS}$, we weight the error by the input distribution as follows

$$\text{MSE} = \sum_{i=1}^{N} (y_i - \hat{y}_i(\vec{\alpha}_i))^2 p_X(\vec{\alpha}_i). \tag{19}$$

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 3.5. The E-NN outperforms the D-NN, but the D-NN training is faster, making it a useful architecture for more computationally expensive problems.

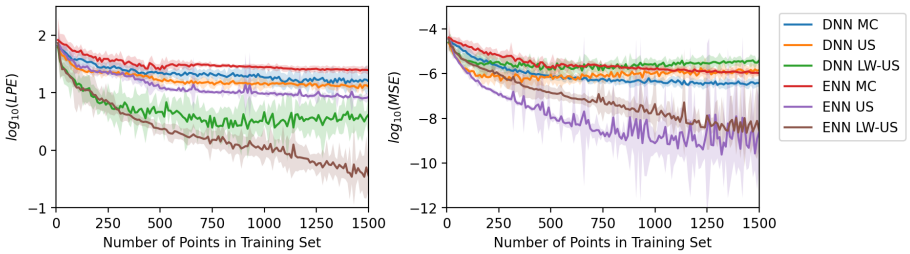

Fig. 3 Error Convergence Curves of the MMT Predictions. The log of the LPE error (left) and log of the MSE (right) are plotted as a function of the number of points in the training set for both the E-NN and D-NN implementations of the MC, US, and LW-US selection criteria. The shading represents ±1 standard deviation of the five experiments. LW-US (both E-NN and D-NN) significantly outperforms the other selection criteria 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 using multidimensional scaling (MDS) [14, 15, 30, 31, 53, 59]. 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. Here, we use the standard Euclidian distance as the distance metric. The two-dimensional projection shown in Figure 4 reveals that points chosen by the LW-US selection criterion are farther apart than points chosen by other selection criteria. The results of MDS suggest that drawing points that are more "spread out" can be valuable for predicting extreme events.

4 Application to Debiasing Operator for Coarse-Resolution Climate Model Outputs

We now apply the likelihood-weighted active selection framework to ML-based climate models to accelerate training and improve the prediction of extreme weather events. Accurate modeling of extreme weather events requires the generation of high-resolution climate model outputs for a range of potential scenarios [50, 52, 4, 17, 61]. Historically, such simulations have been produced from numerical solvers based on physical equations and parameterizations [62, 36, 37, 66, 16, 57, 19]. However, these solvers must resolve turbulent dynamics across scales from millimeters

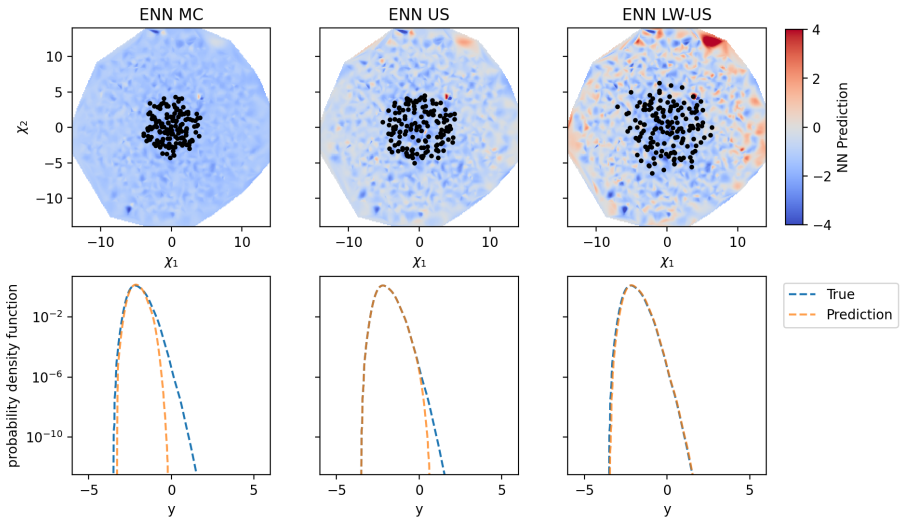

Fig. 4 Visualization of Selected MMT Input Points. In the top row, the 8D space is projected to a 2D space with multi-dimensional scaling. Each plot shows the spread of the optimally selected points in black over the prediction made from the neural network trained after 150 iterations with training data 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.

to thousands of kilometers, making them computationally expensive and reliant on extensive parameter tuning and complicated closure terms.

Recent advances in ML architectures, algorithms, and hardware have enabled alternative data-driven climate models [46, 48, 47, 28, 58, 39, 5, 10, 3, 11, 49, 29]. To build on this growing field, we apply our likelihood-weighted selection approach to a recently proposed debiasing model [3] that learns a correction operator between low-resolution free-running simulations and high-resolution reanalysis data. This operator improves coarse models without requiring full-resolution computation. Although we focus on this particular model, our approach is model-agnostic and broadly applicable to ML-based climate modeling with large candidate training sets. The likelihood-weighted criterion not only reduces training cost but also prioritizes the most informative points for extreme event dynamics.

4.1 Datasets

The coarse-resolution simulations are obtained from version 2 of the Energy Exascale System Model (E3SM) Atmosphere Model (EAMv2) [16, 19, 65]. The dataset 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 dataset is the European Centre for Medium-Range Weather Forecasts (ECMWF) Reanalysis version 5 (ERA5) [24]. 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 datasets, 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 dataset (denoted ERA5), and the input is the free-running dataset 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 Barthel Sorensen et al. [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 dataset ($\mathcal{D}^{\text{ERA5}}$, $\mathcal{D}^{\text{NUDG}}$, and $\mathcal{D}^{\text{CR}}$) is a field over space ξ and time t. Figure 5 shows the mean of the reference reanalysis dataset and the mean of the model output given the test data CR as input.

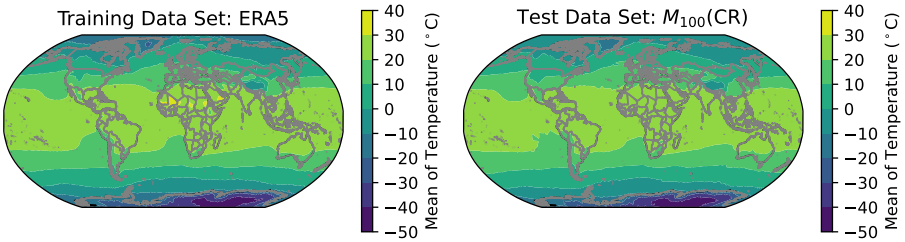

Fig. 5 Mean of ERA5 and Mean of $\mathcal{M}_{100}$(CR) for temperature. During the training phase, the nudged dataset is mapped to the ERA5 dataset. During the testing phase, the coarse-resolution (CR) dataset is provided as input to the trained model.

4.2 Machine Learning Architecture and Active Learning Hyperparameters

The ML model learns a mapping from coarse-resolution climate model outputs to debiased high-fidelity climate predictions. The NN architecture, shown in Figure 6, is an encoder-decoder consisting of two-dimensional 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 (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: $w(\theta) = \sqrt{\sin\left(\frac{90^\circ - \theta}{180^\circ}\pi\right)}$.

The model $\mathcal{M}_{\mathcal{S}}$ is trained to map samples $\mathcal{X}$ from $\mathcal{D}^{\text{NUDG}}$ to the output $\mathcal{D}^{\text{ERA5}}$. During the testing phase, the input of the model is $\mathcal{D}^{\text{CR}}$, and the resulting output is the field $\mathcal{U} = \mathcal{M}_{\mathcal{S}}(\mathcal{D}^{\text{CR}})$. To evaluate the framework, we generate "ground truth" data by training a model with 100% of the samples in $\mathcal{D}^{\text{NUDG}}$

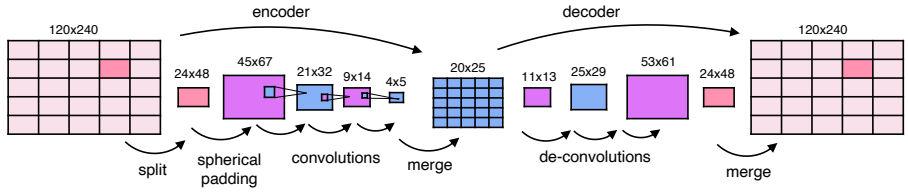

Fig. 6 Climate Debiasing Operator Neural Network Architecture. The NN architecture splits the Earth into sections that are individually passed through convolutional encoder-decoder layers.

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 dataset $\mathcal{D}^{\mathrm{CR}}$: we call this prediction $\mathcal{M}_{100}(\mathcal{D}^{\mathrm{CR}})$. At each iteration, we compute error metrics for $\mathcal{M}_{\mathcal{S}}(\mathcal{D}^{\mathrm{CR}})$ with respect to $\mathcal{M}_{100}(\mathcal{D}^{\mathrm{CR}})$. During the evaluation of the selection criterion, the dimensionality of the inputs is reduced using weighted PCA as in Section 2.4.1 using the weight $w(\xi) = w(\theta, \phi) = \sqrt{\sin\left(\frac{90^\circ - \theta}{180^\circ}\pi\right)}$.

The probabilistic model is a two-member ensemble neural network (E-NN), where the prediction is given by the mean of the ensemble outputs and the uncertainty by 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 selection criterion). 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 selection criterion, 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 to evaluate the statistics of the MSE and LPE over the different experiments.

4.3 Results

We test the method on three cases: (i) the first PCA coefficient for temperature over the entire globe, obtained using weighted PCA (Figure 7), (ii) temperature in Paris (Figure 10), and (iii) specific humidity in Ankara (Figure 13). 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. For each example, we plot the true distribution in green, the predicted distribution obtained from MC sampling in blue, and the predicted distribution obtained from LW sampling in orange. We also plot a Gaussian distribution with the same mean and standard deviation as the true distribution, shown as a dashed gray line. When both tails are heavy or non-Gaussian, we compute the LPE over the full distribution; however, if only one tail is heavy, we compute the LPE solely for the half of the distribution corresponding to the heavy tail.

Looking at the LPE as a function of number of points in the training set in Figures 8, 11, and 14, we see that LW-US outperforms MC in all cases. The

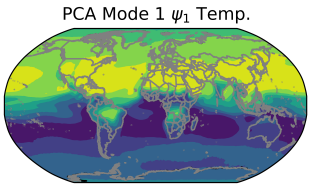

Fig. 7 First weighted PCA mode of the global temperature field.

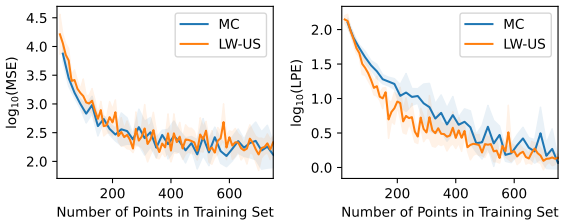

Fig. 8 The log of the MSE and LPE are shown for MC and LW-US with respect to predicting the first PCA coefficient of global temperature. The shading represents ±1 standard deviation across five experiments.

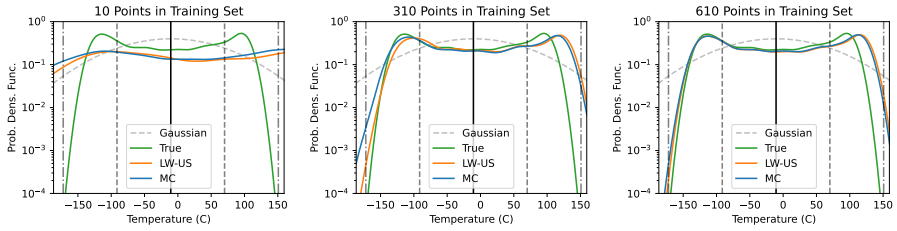

Fig. 9 The true pdf (green) of the first PCA coefficient for temperature is compared to the pdf obtained from predictions made with MC (blue) and LW-US (orange). The Gaussian pdf is also shown in the dashed gray line. The black vertical line denotes the mean of the true distribution, and the dashed lines denote the 1σ and 2σ. LW-US is able to better match the left tail of the true pdf.

improvement obtained from using LW-US can also be seen in the plots of the pdf (Figures 9, 12, and 15) — LW-US does a better job at matching the tails of the distribution, especially in cases where the distribution is non-Gaussian or exhibits heavy tails. We also observe that the improvement obtained from using LW-US occurs at a different number of iterations for the different test cases. Looking at the MSE, the error is similar to MC for cases involving temperature (Figures 8 and 11), but worse for cases involving humidity (Figure 14). However, LPE, not MSE, is the primary metric of interest in our framework, as it better reflects the performance on the tails of the distribution. It is not expected that our method performs well with respect to minimizing the MSE.

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 interested 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.

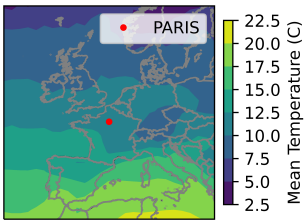

Fig. 10 Mean temperature in the region surrounding Paris, France.

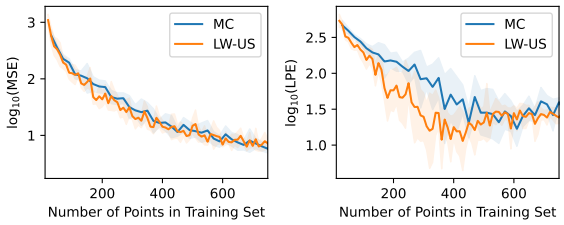

Fig. 11 The log of the MSE and LPE are shown for MC and LW-US with respect to predicting the temperature in Paris given a global model. The shading represents ±1 standard deviation across six experiments.

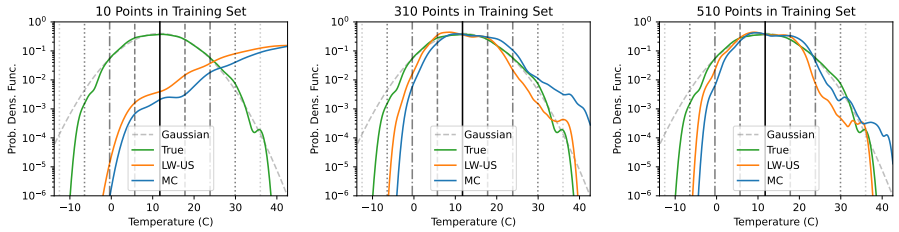

Fig. 12 The true pdf (green) of the temperature in Paris is compared to the pdf obtained from predictions made with MC (blue) and LW-US (orange). The Gaussian pdf is also shown in the dashed gray line. The black vertical line denotes the mean of the true distribution, and the dashed lines denote the 1σ, 2σ, 3σ, and 4σ. LW-US is able to better match the tails of the true pdf with just 310 points.

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 datasets in other applications [33]. 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 dataset 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. 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 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 16 and 17, the six clusters are mapped in order of most occurring in the full dataset (Cluster #1) to least occurring in the full dataset (Cluster #6). For temperature in Paris (Figure 16), we found that points belonging to Cluster #6 are more relevant to the dynamics of extreme weather events (Figure 16). Upon further examination, the shape of Cluster #6 suggests a potential heat dome over Paris and the surrounding region [27]. In the case of specific humidity in Ankara, Cluster #4 contains most of the optimal training points, and the distribution of points between clusters is significantly more skewed. By applying clustering to the

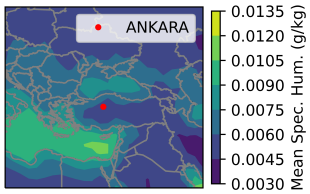

Fig. 13 Mean specific humidity in the region surrounding Ankara, USA.

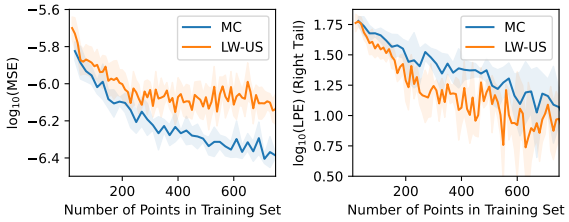

Fig. 14 The log of the MSE and LPE (over the right tail) are shown for MC and LW-US with respect to predicting the specific humidity in Ankara given a global model. Here, we report the LPE for the right tail only, as it exhibits heavy-tailed behavior. The shading represents ± 1 standard deviation across five experiments. LW-US is better for minimizing LPE, but worse for minimizing MSE in the case of specific humidity.

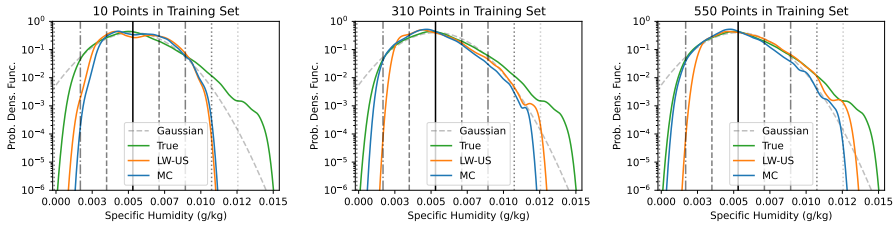

Fig. 15 The true pdf (green) of the specific humidity in Ankara is compared to the pdf obtained from predictions made with MC (blue) and LW-US (orange). The Gaussian pdf is also shown in the dashed gray line as a way to assess how heavy each tail is. The black vertical line denotes the mean of the true distribution, and the dashed lines denote the 1σ, 2σ, 3σ, and 4σ. LW-US is able to better match the heavy tail (right tail) of the true pdf.

points selected by the algorithm, we are able to pick out extreme weather events in an unsupervised way.

5 Conclusions

To address the challenge of training ML models on vast datasets, we introduced a likelihood-weighted active data selection framework that sequentially selects optimal training points to improve prediction of extreme event statistics (i.e. tails of the distribution). Points are selected according to a likelihood-weighted selection criterion. The uncertainty in the model is quantified according to the variance of an ensemble of neural networks, and the dimensionality of the high-dimensional inputs is reduced using weighted principal component analysis. The framework is model-agnostic and suitable for high-dimensional datasets. We demonstrated the success of the framework on both a synthetic problem (one-dimensional model for dispersive wave turbulence) and a real-world problem (coarse-resolution climate model correction operator). In both cases, likelihood-weighted active data selection achieved a lower error in the tails of the probability distribution with fewer training points. In the real-world problem, clustering showed that the method was selecting points relevant to extreme weather events. Down the line, the developed

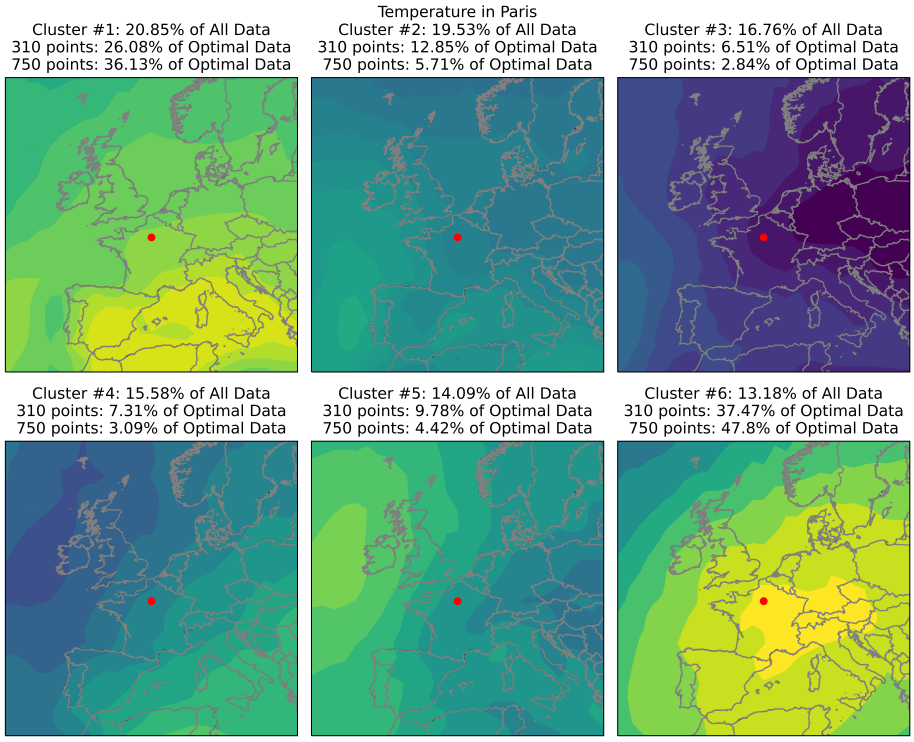

Fig. 16 Clusters for Temperature in Paris With 310 points in the training set, Cluster #6 only represents 13.18% of all data but 37.47% of the optimal data. Cluster #6 exhibits a blocking pattern over most of France. The next most occurring cluster is Cluster #1 that represents standard zonal flow, typical for normal weather events.

approach has the potential to be used as a compression algorithm that preserves the information associated with extreme events in vast datasets.

Acknowledgements The work was funded through the AFOSR Award FA9550-23-1-0517 and the National Science Foundation Graduate Research Fellowship Grant No. 2141064.

We implement multidimensional scaling with the `MATLAB` `mdscale` function. We implement the kernel density estimate with the python function `FFTKDE` from the package `KDEpy`. The neural networks are designed using the `tensorflow` library. We implement `k-means++` with `scikit-learn`.

References

1. Angelopoulos AN, Bates S (2022) A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification. DOI 10.48550/arXiv.2107.07511, URL http://arxiv.org/abs/2107.07511, arXiv:2107.07511 [cs, math, stat]
2. Banerjee A, Dunson D, Tokdar S (2011) Efficient Gaussian Process Regression for Large Data Sets. URL http://arxiv.org/abs/1106.5779

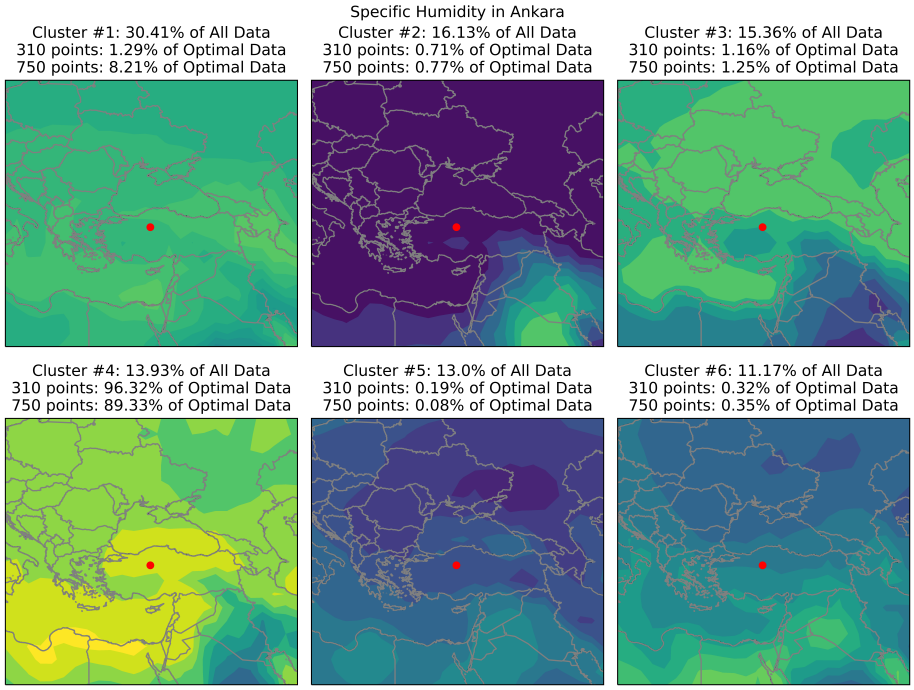

Fig. 17 Clusters for Specific Humidity in Ankara With 310 points in the training set, Cluster #4 only represents 13.93% of all data but 96.32% of the optimal data. The next most occurring cluster is Cluster #1. Between 310 points and 750 points, more points are chosen from Cluster #1.

3. Barthel Sorensen B, Charalampopoulos A, Zhang S, Harrop BE, Leung LR, Sapsis TP (2024) 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(3):e2023MS004122
4. Bauer P, Stevens B, Hazeleger W (2021) A digital twin of Earth for the green transition. Nature Climate Change 11(2):80–83, DOI 10.1038/s41558-021-00986-y, URL https://www.nature.com/articles/s41558-021-00986-y, publisher: Nature Publishing Group
5. Bi K, Xie L, Zhang H, Chen X, Gu X, Tian Q (2023) Accurate medium-range global weather forecasting with 3D neural networks. Nature 619(7970):533–538, DOI 10.1038/s41586-023-06185-3, URL https://doi.org/10.1038/s41586-023-06185-3
6. Blanchard A, Sapsis T (2021) Bayesian optimization with output-weighted optimal sampling. Journal of Computational Physics 425:109901, DOI 10.1016/j.jcp.2020.109901
7. Blanchard A, Sapsis T (2021) Output-Weighted Optimal Sampling for Bayesian Experimental Design and Uncertainty Quantification. SIAM/ASA Journal on Uncertainty Quantification 9(2):564–592, DOI 10.1137/20M1347486, URL https://epubs.siam.org/doi/10.1137/20M1347486

8. Béranger B, Duong T, Perkins-Kirkpatrick SE, Sisson SA (2019) Tail density estimation for exploratory data analysis using kernel methods. Journal of Nonparametric Statistics 31(1):144–174, DOI 10.1080/10485252.2018.1537442, URL https://www.tandfonline.com/doi/full/10.1080/10485252.2018.1537442
9. Cai D, Majda AJ, McLaughlin DW, Tabak EG (1999) Spectral bifurcations in dispersive wave turbulence. Proceedings of the National Academy of Sciences 96(25):14216–14221, DOI 10.1073/pnas.96.25.14216, URL https://www.pnas.org/doi/abs/10.1073/pnas.96.25.14216, publisher: Proceedings of the National Academy of Sciences
10. Charalampopoulos AT, Zhang S, Harrop B, Leung LyR, Sapsis T (2023) Statistics of extreme events in coarse-scale climate simulations via machine learning correction operators trained on nudged datasets. DOI 10.48550/arXiv.2304.02117, URL http://arxiv.org/abs/2304.02117, arXiv:2304.02117 [physics]
11. Chen L, Du F, Hu Y, Wang Z, Wang F (2023) SwinRDM: Integrate SwinRNN with Diffusion Model towards High-Resolution and High-Quality Weather Forecasting. Proceedings of the AAAI Conference on Artificial Intelligence 37(1):322–330, DOI 10.1609/aaai.v37i1.25105, URL https://ojs.aaai.org/index.php/AAAI/article/view/25105
12. Cohn DA, Ghahramani Z, Jordan MI (1996) Active Learning with Statistical Models. DOI 10.48550/arXiv.cs/9603104, URL http://arxiv.org/abs/cs/9603104
13. Cousins W, Sapsis TP (2014) Quantification and prediction of extreme events in a one-dimensional nonlinear dispersive wave model. Physica D: Nonlinear Phenomena 280-281:48–58, DOI https://doi.org/10.1016/j.physd.2014.04.012
14. Cox T, Cox M (2000) Multidimensional Scaling, 0th edn. Chapman and Hall/CRC, DOI 10.1201/9780367801700, URL https://www.taylorfrancis.com/books/9781420036121
15. Davison ML (1992) Multidimensional scaling. xiv, 242 p., Krieger Pub. Co., Malabar, Fla., URL https://catalog.hathitrust.org/Record/009131389
16. Dennis JM, Edwards J, Evans KJ, Guba O, Lauritzen PH, Mirin AA, St-Cyr A, Taylor MA, Worley PH (2012) CAM-SE: A scalable spectral element dynamical core for the Community Atmosphere Model. The International Journal of High Performance Computing Applications 26(1):74–89
17. Fiedler T, Pitman AJ, Mackenzie K, Wood N, Jakob C, Perkins-Kirkpatrick SE (2021) Business risk and the emergence of climate analytics. Nature Climate Change 11(2):87–94, DOI 10.1038/s41558-020-00984-6, URL https://www.nature.com/articles/s41558-020-00984-6, publisher: Nature Publishing Group
18. Gal Y, Ghahramani Z (2016) Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: international conference on machine learning, PMLR, pp 1050–1059
19. Golaz JC, Van Roekel LP, Zheng X, Roberts AF, Wolfe JD, Lin W, Bradley AM, Tang Q, Maltrud ME, Forsyth RM, Zhang C, Zhou T, Zhang K, Zender CS, Wu M, Wang H, Turner AK, Singh B, Richter JH, Qin Y, Petersen MR, Mametjanov A, Ma PL, Larson VE, Krishna J, Keen ND, Jeffery N, Hunke EC, Hannah WM, Guba O, Griffin BM, Feng Y, Engwirda D, Di Vittorio AV, Dang C, Conlon LM, Chen CCJ, Brunke MA, Bisht G, Benedict JJ, Asay-Davis XS, Zhang Y, Zhang M, Zeng X, Xie S, Wolfram PJ, Vo T, Veneziani M,

Tesfa TK, Sreepathi S, Salinger AG, Reeves Eyre JEJ, Prather MJ, Mahajan S, Li Q, Jones PW, Jacob RL, Huebler GW, Huang X, Hillman BR, Harrop BE, Foucar JG, Fang Y, Comeau DS, Caldwell PM, Bartoletti T, Balaguru K, Taylor MA, McCoy RB, Leung LR, Bader DC (2022) The DOE E3SM Model Version 2: Overview of the Physical Model and Initial Model Evaluation. Journal of Advances in Modeling Earth Systems 14(12):e2022MS003156, DOI 10.1029/2022MS003156, URL https://doi.org/10.1029/2022MS003156

20. Gramacy RB, Lee HKH (2009) Adaptive Design and Analysis of Supercomputer Experiments. Technometrics 51(2):130–145, DOI 10.1198/TECH.2009.0015, URL https://doi.org/10.1198/TECH.2009.0015, publisher: Taylor & Francis _eprint: https://doi.org/10.1198/TECH.2009.0015
21. Guerra N, Nelsen NH, Yang Y (2025) Learning Where to Learn: Training Distribution Selection for Provable OOD Performance. DOI 10.48550/ARXIV.2505.21626, URL https://arxiv.org/abs/2505.21626, version Number: 1
22. Guth S, Champenois B, Sapsis TP (2022) Application of gaussian process multi-fidelity optimal sampling to ship structural modeling. In: 34th Symp. on Naval Hydrodynamics, Washington, DC, June
23. Guth S, Mojahed A, Sapsis TP (2024) 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 420:116760
24. Hersbach H, Bell B, Berrisford P, Hirahara S, Horányi A, Muñoz-Sabater J, Nicolas J, Peubey C, Radu R, Schepers D, Simmons A, Soci C, Abdalla S, Abellan X, Balsamo G, Bechtold P, Biavati G, Bidlot J, Bonavita M, De Chiara G, Dahlgren P, Dee D, Diamantakis M, Dragani R, Flemming J, Forbes R, Fuentes M, Geer A, Haimberger L, Healy S, Hogan RJ, Hólm E, Janisková M, Keeley S, Laloyaux P, Lopez P, Lupu C, Radnoti G, de Rosnay P, Rozum I, Vamborg F, Villaume S, Thépaut JN (2020) The ERA5 global reanalysis. Quarterly Journal of the Royal Meteorological Society 146(730):1999–2049, DOI https://doi.org/10.1002/qj.3803, URL https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.3803
25. Hüllermeier E, Waegeman W (2021) Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. Machine learning 110(3):457–506
26. Katsidoniotaki E, Guth S, Göteman M, Sapsis TP (2025) Reduced order modeling of wave energy systems via sequential Bayesian experimental design and machine learning. Applied Ocean Research 155:104439, DOI 10.1016/j.apor.2025.104439, URL https://linkinghub.elsevier.com/retrieve/pii/S0141118725000276
27. Kautz LA, Martius O, Pfahl S, Pinto JG, Ramos AM, Sousa PM, Woollings T (2022) Atmospheric blocking and weather extremes over the Euro-Atlantic sector – a review. Weather and Climate Dynamics 3(1):305–336, DOI 10.5194/wcd-3-305-2022, URL https://wcd.copernicus.org/articles/3/305/2022/
28. Keisler R (2022) Forecasting Global Weather with Graph Neural Networks. URL https://arxiv.org/abs/2202.07575
29. Kochkov D, Yuval J, Langmore I, Norgaard P, Smith J, Mooers G, Klöwer M, Lottes J, Rasp S, Düben P, Hatfield S, Battaglia P, Sanchez-Gonzalez

A, Willson M, Brenner MP, Hoyer S (2024) Neural general circulation models for weather and climate. Nature DOI 10.1038/s41586-024-07744-y, URL https://doi.org/10.1038/s41586-024-07744-y
30. Kruskal JB (1964) Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypothesis. Psychometrika 29(1):1–27, DOI 10.1007/BF02289565
31. Kruskal JB (1964) Nonmetric Multidimensional Scaling: A Numerical Method. Psychometrika 29(2):115–129, DOI 10.1007/BF02289694
32. Lakshminarayanan B, Pritzel A, Blundell C (2017) Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. URL https://arxiv.org/abs/1612.01474
33. Li H, Fernex D, Semaan R, Tan J, Morzyński M, Noack BR (2021) Cluster-based network model. Journal of Fluid Mechanics 906:A21, DOI 10.1017/jfm.2020.785
34. MacKay DJC (1992) Information-Based Objective Functions for Active Data Selection. Neural Computation 4(4):590–604, DOI 10.1162/neco.1992.4.4.590
35. Majda AJ, McLaughlin DW, Tabak EG (1997) A one-dimensional model for dispersive wave turbulence. Journal of Nonlinear Science 7(1):9–44, DOI 10.1007/BF02679124, URL https://doi.org/10.1007/BF02679124
36. Manabe S, Smagorinsky J, Strickler RF (1965) SIMULATED CLIMATOLOGY OF A GENERAL CIRCULATION MODEL WITH A HYDROLOGIC CYCLE Section: Monthly Weather Review
37. Mintz Y (1968) Very Long-Term Global Integration of the Primitive Equations of Atmospheric Motion: An Experiment in Climate Simulation. In: Billings DE, Broecker WS, Bryson RA, Cox A, Damon PE, Donn WL, Eriksson E, Ewing M, Fletcher JO, Hamilton W, Jerzykiewicz M, Kutzbach JE, Lorenz EN, Mintz Y, Mitchell JM, Saltzman B, Serkowski K, Shen WC, Suess HE, Tanner WF, Weyl PK, Worthington LV, Mitchell JM (eds) 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, American Meteorological Society, Boston, MA, pp 20–36
38. Mohamad MA, Sapsis TP (2018) Sequential sampling strategy for extreme event statistics in nonlinear dynamical systems. Proceedings of the National Academy of Sciences 115(44):11138–11143, DOI 10.1073/pnas.1813263115, URL https://www.pnas.org/doi/full/10.1073/pnas.1813263115, publisher: Proceedings of the National Academy of Sciences
39. Mukkavilli SK, Civitarese DS, Schmude J, Jakubik J, Jones A, Nguyen N, Phillips C, Roy S, Singh S, Watson C, Ganti R, Hamann H, Nair U, Ramachandran R, Weldemariam K (2023) AI Foundation Models for Weather and Climate: Applications, Design, and Implementation. DOI 10.48550/arXiv.2309.10808, URL http://arxiv.org/abs/2309.10808, arXiv:2309.10808 [physics]
40. Murphy KP (2022) Probabilistic machine learning: an introduction. MIT press
41. Pickering E, Sapsis TP (2024) Information FOMO: The Unhealthy Fear of Missing Out on Information—A Method for Removing Misleading Data for Healthier Models. Entropy 26(10):835, DOI 10.3390/e26100835, URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11507899/
42. Pickering E, Guth S, Karniadakis GE, Sapsis TP (2022) Discovering and forecasting extreme events via active learning in neural operators. Nature

Computational Science 2(12):823–833, DOI 10.1038/s43588-022-00376-0, URL https://doi.org/10.1038/s43588-022-00376-0
43. Psaros AF, Meng X, Zou Z, Guo L, Karniadakis GE (2023) Uncertainty quantification in scientific machine learning: Methods, metrics, and comparisons. Journal of Computational Physics 477:111902, DOI https://doi.org/10.1016/j.jcp.2022.111902
44. Pushkarev A, Zakharov VE (2013) Quasibreathers in the MMT model. Physica D: Nonlinear Phenomena 248:55–61, DOI 10.1016/j.physd.2013.01.003
45. Rasmussen CE, Williams CKI (2005) Gaussian Processes for Machine Learning. The MIT Press, DOI 10.7551/mitpress/3206.001.0001, URL https://doi.org/10.7551/mitpress/3206.001.0001
46. Rasp S, Lerch S (2018) Neural Networks for Postprocessing Ensemble Weather Forecasts. Monthly Weather Review 146(11):3885 – 3900, DOI 10.1175/MWR-D-18-0187.1
47. Rasp S, Thuerey N (2021) 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(2):e2020MS002405, DOI https://doi.org/10.1029/2020MS002405, URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2020MS002405
48. Rasp S, Dueben PD, Scher S, Weyn JA, Mouatadid S, Thuerey N (2020) WeatherBench: A Benchmark Data Set for Data-Driven Weather Forecasting. Journal of Advances in Modeling Earth Systems 12(11):e2020MS002203
49. Rasp S, Hoyer S, Merose A, Langmore I, Battaglia P, Russell T, Sanchez-Gonzalez A, Yang V, Carver R, Agrawal S, Chantry M, Ben Bouallegue Z, Dueben P, Bromberg C, Sisk J, Barrington L, Bell A, Sha F (2024) WeatherBench 2: A Benchmark for the Next Generation of Data-Driven Global Weather Models. Journal of Advances in Modeling Earth Systems 16(6):e2023MS004019, DOI https://doi.org/10.1029/2023MS004019
50. Raymond C, Horton RM, Zscheischler J, Martius O, AghaKouchak A, Balch J, Bowen SG, Camargo SJ, Hess J, Kornhuber K, Oppenheimer M, Ruane AC, Wahl T, White K (2020) Understanding and managing connected extreme events. Nature Climate Change 10(7):611–621, DOI 10.1038/s41558-020-0790-4, URL https://www.nature.com/articles/s41558-020-0790-4, publisher: Nature Publishing Group
51. Ren P, Xiao Y, Chang X, Huang PY, Li Z, Gupta BB, Chen X, Wang X (2021) A Survey of Deep Active Learning. ACM Comput Surv 54(9), DOI 10.1145/3472291, URL https://doi.org/10.1145/3472291
52. Robinson A, Lehmann J, Barriopedro D, Rahmstorf S, Coumou D (2021) Increasing heat and rainfall extremes now far outside the historical climate. npj Climate and Atmospheric Science 4(1):1–4, DOI 10.1038/s41612-021-00202-w, URL https://www.nature.com/articles/s41612-021-00202-w, publisher: Nature Publishing Group
53. Sammon J (1969) A Nonlinear Mapping for Data Structure Analysis. IEEE Transactions on Computers C-18(5):401–409, DOI 10.1109/T-C.1969.222678
54. Sapsis TP (2020) 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(2234):20190834, URL https://royalsocietypublishing.org/doi/10.1098/rspa.2019.0834

55. Sapsis TP (2021) Statistics of Extreme Events in Fluid Flows and Waves. Annual Review of Fluid Mechanics 53(Volume 53, 2021):85–111
56. Sapsis TP, Blanchard A (2022) 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(2229):20210197, DOI 10.1098/rsta.2021.0197, URL https://royalsocietypublishing.org/doi/10.1098/rsta.2021.0197
57. Schneider T, Lan S, Stuart A, Teixeira J (2017) Earth System Modeling 2.0: A Blueprint for Models That Learn From Observations and Targeted High-Resolution Simulations. Geophysical Research Letters 44(24):12,396–12,417, DOI 10.1002/2017GL076101, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2017GL076101
58. Schneider T, Behera S, Boccaletti G, Deser C, Emanuel K, Ferrari R, Leung LR, Lin N, Müller T, Navarra A, Ndiaye O, Stuart A, Tribbia J, Yamagata T (2023) Harnessing AI and computing to advance climate modelling and prediction. Nature Climate Change 13(9):887–889, DOI 10.1038/s41558-023-01769-3, URL https://doi.org/10.1038/s41558-023-01769-3
59. Seber GAF (1984) Multivariate Observations, 1st edn. Wiley Series in Probability and Statistics, Wiley, DOI 10.1002/9780470316641, URL https://onlinelibrary.wiley.com/doi/book/10.1002/9780470316641
60. Settles B (2009) Active Learning Literature Survey. Technical Report, University of Wisconsin-Madison Department of Computer Sciences, URL https://minds.wisconsin.edu/handle/1793/60660
61. Slingo J, Bates P, Bauer P, Belcher S, Palmer T, Stephens G, Stevens B, Stocker T, Teutsch G (2022) Ambitious partnership needed for reliable climate prediction. Nature Climate Change 12(6):499–503, DOI 10.1038/s41558-022-01384-8, URL https://www.nature.com/articles/s41558-022-01384-8, publisher: Nature Publishing Group
62. Smagorinsky J, Manabe S, Holloway JL (1965) NUMERICAL RESULTS FROM A NINE-LEVEL GENERAL CIRCULATION MODEL OF THE ATMOSPHERE 1 Section: Monthly Weather Review
63. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research 15(56):1929–1958, URL http://jmlr.org/papers/v15/srivastava14a.html
64. Stein ML (2021) A parametric model for distributions with flexible behavior in both tails. Environmetrics 32(2):e2658, DOI 10.1002/env.2658, URL https://onlinelibrary.wiley.com/doi/10.1002/env.2658
65. Taylor MA, Cyr AS, Fournier A (2009) A Non-oscillatory Advection Operator for the Compatible Spectral Element Method. In: Allen G, Nabrzyski J, Seidel E, van Albada GD, Dongarra J, Sloot PMA (eds) Computational Science – ICCS 2009, Springer Berlin Heidelberg, Berlin, Heidelberg, pp 273–282
66. Tomita H, Miura H, Iga S, Nasuno T, Satoh M (2005) A global cloud-resolving simulation: Preliminary results from an aqua planet experiment. Geophysical Research Letters 32(8)
67. Yang Y, Blanchard A, Sapsis T, Perdikaris P (2022) Output-weighted sampling for multi-armed bandits with extreme payoffs. Proceedings of the Royal Society A: Mathematical, Physical and Engineer-

ing Sciences 478(2260):20210781, DOI 10.1098/rspa.2021.0781, URL https://royalsocietypublishing.org/doi/10.1098/rspa.2021.0781
68. Zakharov V, Dias F, Pushkarev A (2004) One-dimensional wave turbulence. Physics Reports 398(1):1–65, DOI 10.1016/j.physrep.2004.04.002
69. Zakharov VE, Guyenne P, Pushkarev AN, Dias F (2001) Wave turbulence in one-dimensional models. Physica D: Nonlinear Phenomena 152-153:573–619, DOI 10.1016/S0167-2789(01)00194-4
70. Zhou X, Liu H, Pourpanah F, Zeng T, Wang X (2022) A Survey on Epistemic (Model) Uncertainty in Supervised Learning: Recent Advances and Applications. Neurocomputing 489:449–465, DOI 10.1016/j.neucom.2021.10.119, URL http://arxiv.org/abs/2111.01968
71. Zou Z, Meng X, Psaros AF, Karniadakis GE (2024) NeuralUQ: A Comprehensive Library for Uncertainty Quantification in Neural Differential Equations and Operators. SIAM Review 66(1):161–190, DOI 10.1137/22M1518189, URL https://doi.org/10.1137/22M1518189, publisher: Society for Industrial and Applied Mathematics