Multi-Fidelity Learning With Bayesian Neural Networks

Brown University

Authors:
Mason Lee

Abstract

This study presents the application and exploration of multi-fidelity learning and Bayesian optimization in the context of environmental data prediction and analysis. We successfully applied multi-fidelity learning to approximate the Forester function, demonstrating the advantages of leveraging both low- and high-fidelity data. Furthermore, we implemented active learning within a Bayesian optimization framework, showcasing the potential to reduce uncertainty with minimal sampling. This work includes the implementation of two models, a multi-fidelity neural network (MF-NN) and a multi-fidelity Gaussian Process (MF-GP), for sea surface temperature prediction. Our results suggest that the MF-NN more accurately represents the sea surface temperature, though further research is needed to confirm these findings. Additionally, we investigated the effects of withholding data during training, revealing the significant role of low-fidelity data in reducing the loss function, even in a data-sparse environment. These findings, while demonstrated using simplified examples, offer valuable insights into the potential of multi-fidelity learning and Bayesian optimization for real-world applications.

1 Introduction

Climate science, in its quest to better understand our rapidly changing planet, has benefited enormously from the availability of data from a plethora of remote sensing satellites. These satellites provide observations at varying spatial and temporal scales, offering an invaluable resource for monitoring and predicting environmental parameters such as sea surface temperature (SST). However, the multi-scale and multi-fidelity nature of these observations presents a significant challenge for their effective integration and utilization.

Deep Neural Networks (DNNs) with their impressive expressive power have emerged as an effective tool for learning complex relationships within high-dimensional data. They offer a promising approach to integrate multi-fidelity data and generate precise and accurate predictions. However, traditional DNNs do not inherently provide a measure of uncertainty associated with their predictions.

As we venture into an era where the consequences of climate change are becoming increasingly apparent, the need for accurate and reliable predictions is more critical than ever. Policy makers, in particular, require not only accurate forecasts but also a quantification of the uncertainties associated with these predictions to make informed decisions, especially in the context of natural disasters.

Bayesian Neural Networks (BNNs) have the potential to fill this gap. By treating the weights and biases in the network as random variables, BNNs not only provide point estimates but also probabilistic predictions, offering a measure of uncertainty. This uncertainty quantification can guide decision-making processes and help identify areas where further data collection is needed, informing the optimal placement of sensors.

In this implementation, we harness the power of BNNs in a multi-fidelity setting, demonstrating their potential for integrating multi-fidelity satellite data for SST prediction. Leveraging high-fidelity data from the Massachusetts Water Resource Authority and low-fidelity data from the MODerate-resolution Imaging Spectroradiometer, we train a multi-fidelity BNN and compare it to multi-fidelity Gaussian Process Regression. This is a re-implementation of the architecture and methods outlined in [1], [2].

2 Methodology

2.1 Predicting Sea Surface Temperature

We now utilize the previously described model to approximate the sea surface temperature (SST) in the Massachusetts and Cape Cod Bays following the methods described by [1]. Two sources of data are incorporated: (1) Low-fidelity: The MODerate-resolution Imaging Spectroradiometer (MODIS) Terra on board a NASA satellite providing a spatial resolution of 4×4 km, and (2) High-fidelity (in situ): The Massachusetts Water Resource Authority (MWRA).

In particular, we employ temperature stations which are situated 1m below the sea surface as an approximation for seawater surface temperature. The low- and high-fidelity data are for September 8th, 2015. All the MWRA measurements from different stations are assumed to be taken simultaneously. The MWRA station locations are indicated 1 where 14 measurements are available. Twelve of these are used as the training data (large circles), while the remaining two (xA : (70.86, 42.42), and xB : (70.23, 41.91)) are used to validate the predictions.

In our multi-fidelity modeling approach, we employ a multi-fidelity Bayesian Neural Network (BNN) with two hidden layers, each consisting of 20 neurons, for the low-fidelity data, and a single hidden layer with 30 neurons in the BNN for the high-fidelity data. The surrogate model $\tilde{u}(x_u, \tilde{u}_L; \theta)$ is trained using these configurations.

3 Multi-Fidelity Modeling

The multi-fidelity problem can be posed in a manner that integrates high and low-fidelity information into a cohesive model.

Let $f_i : \mathbb{R}^d \to \mathbb{R}$ denote the i-th fidelity model, with $i = 0, 1, ..., M$, where M is the highest fidelity level. The lower fidelity models are less accurate but computationally cheaper approximations to the highest fidelity model f_M [3].

$$y_i(x) = f_i(x) + \epsilon_i(x), \quad i = 0, 1, ..., M \tag{1}$$

where ϵ_i is a zero mean Gaussian noise process with variance σ_i^2.

3.1 Multi-Fidelity Gaussian Process

In the Emukit framework, a multi-fidelity Gaussian process (MF-GP) is trained by learning a set of independent Gaussian processes, one for each fidelity level. These are then co-kriged to create a combined model that leverages information from all fidelity levels.

The MF-GP model treats the outputs of the different fidelity models as correlated Gaussian processes. The model is given by:

$$f_i(x) = \rho_i f_{i-1}(x) + \delta_i(x), \quad i = 1, ..., M \tag{2}$$

where ρ_i is the correlation parameter between the i-th and the $(i-1)$-th fidelity levels, and $\delta_i(x)$ is a zero mean Gaussian process.

3.2 Multi-Fidelity Neural Network

DeepXDE utilizes a different approach, integrating the multi-fidelity information into a neural network framework. The neural network is designed such that it includes two components - a low-fidelity component and a high-fidelity component.

Let NN_i denote the neural network associated with the i-th fidelity level. The architecture of the multi-fidelity neural network is given by [2]:

$$NN_M(x) = NN_{M-1}(x) + g_M(x, \theta_M), \quad i = 1, ..., M \tag{3}$$

where g_M is the high-fidelity component, and θ_M are the parameters of the high-fidelity component. The low-fidelity component, NN_{M-1}, provides an initial approximation to the high-fidelity function. The high-fidelity component, g_M, learns to correct the discrepancies between the low-fidelity approximation and the high-fidelity function.

3.3 Forrester Function as a Multi-Fidelity Toy Model

The Forrester function is a commonly used test function in global optimization and surrogate modeling, particularly for problems involving Gaussian process regression and Bayesian optimization. It is defined as follows:

$$f(x) = (6x-2)^2 \sin(12x-4), \quad x \in [0,1] \tag{4}$$

This function has a smooth, non-linear behavior which makes it a suitable candidate for a simple, one-dimensional test case for multi-fidelity modeling.

In order to use the Forrester function as a multi-fidelity model, we create two versions of the function, a low-fidelity version and a high-fidelity version. The low-fidelity function, f_L, is typically a coarser or noisier approximation of the true function, while the high-fidelity function, f_H, is a more accurate or precise representation of the true function.

We can define the low-fidelity function as:

$$f_L(x) = f(x) + \delta, \quad \delta \sim \mathcal{N}(0, \sigma_L^2) \tag{5}$$

where δ is a Gaussian noise term with mean 0 and standard deviation σ_L.

The high-fidelity function can be defined as:

$$f_H(x) = f(x) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_H^2) \tag{6}$$

where ϵ is a Gaussian noise term with mean 0 and standard deviation σ_H, with $\sigma_H < \sigma_L$.

In this setup, the low-fidelity function provides a cheap but less accurate version of the true function, while the high-fidelity function provides a more accurate but also more expensive version of the true function. This serves as a useful toy model for testing and validating multi-fidelity modeling methods.

3.4 Active Learning in Bayesian Optimization Framework

Bayesian optimization (BO) is a model-based optimization methodology intended for optimizing costly-to-evaluate, noisy, and often black-box functions [**?**]. It is particularly well-suited for hyperparameter tuning in machine learning models, optimizing parameters of a robot controller, and similar problems where each function evaluation might require significant computational resources.

Consider a function $f : \mathcal{X} \to \mathbb{R}$ defined on a constrained input space $\mathcal{X} \subseteq \mathbb{R}^d$. The goal of Bayesian optimization is to find the global minimum $x^* \in \mathcal{X}$ of the function f, i.e.,

$$x^* = \arg\min_{x \in \mathcal{X}} f(x) \tag{7}$$

In many practical scenarios, the objective function $f(x)$ is noisy, i.e., $y = f(x) + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2)$, and no gradient information is available.

Bayesian optimization leverages two key components:

1. A prior probability measure $\mathcal{P}$ over f, representing our prior beliefs about f. As new data $D = \{(x_i, y_i)\}_{i=1}^N$ is observed, the prior is updated to a posterior $\mathcal{P}(f|D)$ using the Bayes' rule.
2. An acquisition function $\alpha : \mathcal{X} \to \mathbb{R}$, which for each point in the input space quantifies the utility of evaluating this point. The acquisition function balances exploration, i.e., probing regions where the model is uncertain, and exploitation, i.e., exploiting regions where the model is confident to have good performance.

Bayesian optimization algorithm iterates the following steps until a predefined stopping criterion is met:

1. Fit the model $\mathcal{P}$ on the currently available data D.
2. Find the point x_{next} to evaluate next by optimizing the acquisition function, i.e., $x_{\text{next}} = \arg\max_{x \in \mathcal{X}} \alpha(x|D)$.

3. Evaluate the objective function at x_{next}, obtain $y_{\text{next}} = f(x_{\text{next}}) + \epsilon$ and add the new observation $(x_{\text{next}}, y_{\text{next}})$ to the data D.

4 Results

Our study begins with an exploration of the geographical distribution of low- and high-fidelity data across the Boston Bay, depicted in Figure 1. The color bar represents the sea surface temperature (SST), with larger circles indicating high-fidelity data points.

To understand the effects of data quality and quantity on prediction performance, we trained a multi-fidelity neural network (MF-NN) under three different data scenarios. Figure 4 shows the training history of the MF-NN for these scenarios. The training process reveals the influence of withholding different parts of the dataset on the model's learning capability.

As part of the methodological development, we also applied Bayesian optimization techniques. Figure 2 provides insights into different aspects of this process, including the use of Gaussian Process (GP) regression models to estimate the objective function and its uncertainty, as well as the evaluation of various acquisition functions: Expected Improvement (EI), Negative Lower Confidence Bound (NLCB), and Probability of Improvement (PI). The subplots collectively depict the adaptive sampling strategy used for efficient optimization of the objective function.

We further validated the performance of the MF-NN using a synthetic dataset derived from the Forester function [3]. As shown in Figure 3, the MF-NN successfully approximates the function using both low- and high-fidelity data, demonstrating the model's ability to leverage data of varying fidelity.

In a real-world application, we employed the MF-NN to predict SST over the Boston Bay, as shown in Figure 9. However, the model experienced challenges in handling high-fidelity data points randomly introduced into the low-fidelity output. This difficulty emphasizes the need for a well-structured training regime that considers the spatial distribution and fidelity of the training data.

Finally, to better understand the impact of fidelity on training, we created a bar chart (Figure 10) that presents the effect of withholding data on the MF-NN network's performance. It is clear that low-fidelity data plays a vital role in reducing the loss function, underlining its importance in multi-fidelity modeling.

In the first subplot, titled "GP Bayesian Optimization Results," the observations are shown as red dots, the objective function as a black line, and the model prediction as a blue line. The shaded regions around the model prediction represent the uncertainty.

The second subplot, titled "Acquisition Functions Compared," compares different acquisition functions: Expected Improvement (EI) shown in green, Negative Lower Confidence Bound (NLCB) in purple, and Probability of Improvement (PI) in dark orange. The functions are normalized to better visualize the differences.

The third subplot, titled "Initial Uncertainty," presents the initial observations as red dots, the objective function as a black line, and the model prediction as a blue line. Similar to the first subplot, the shaded regions represent the uncertainty.

Despite these promising results, it is important to note that our study primarily used synthetic datasets and toy examples. While these offer a controlled environment to understand the capabilities of multi-fidelity learning and Bayesian optimization, they do not fully represent the complexities of real-world applications. Therefore, further research is needed to extend these methods to more complex, real-world datasets and scenarios. To implement active-learning in scenarios like the Boston Bay, we need to have the ability to collect data. Temporal and seasonal factors will affect the MF model's ability to learn.

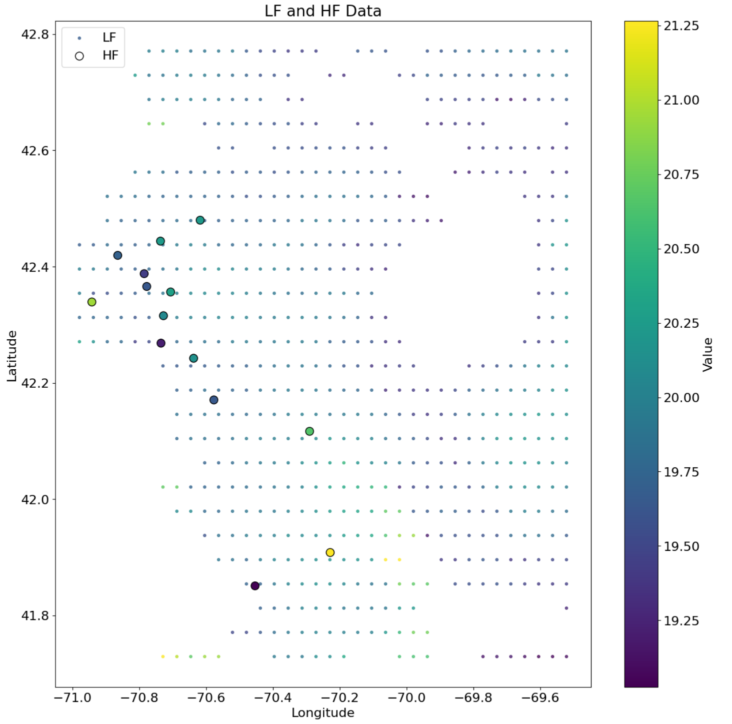

Figure 1: Scatter plot of the low and high fidelity locations across the Boston Bay. The color bar represents the sea surface temprature. THe high fidelity predictions are represented by larger circles.

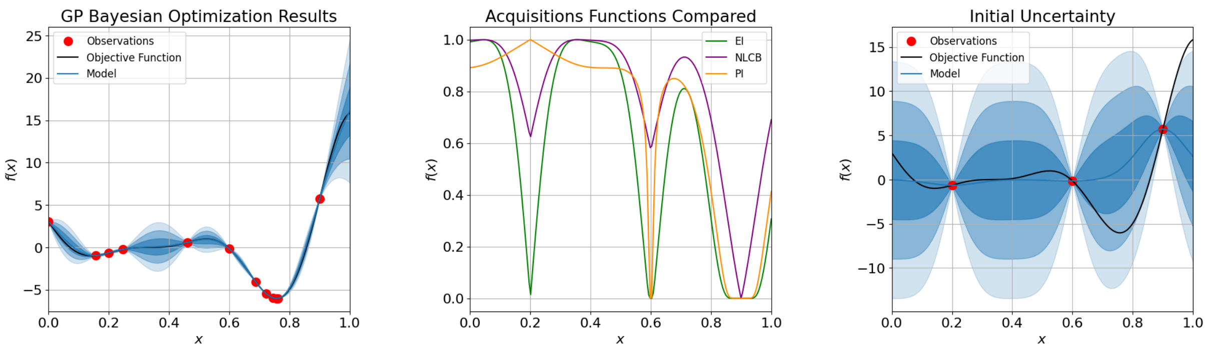

Figure 2: The figure showcases three subplots representing different aspects of Bayesian optimization. In the first subplot, GPBayesianOptimization is employed to model and optimize the objective function based on observations. The second subplot illustrates the evaluation of three acquisition functions: Expected Improvement (EI), Negative Lower Confidence Bound (NLCB), and Probability of Improvement (PI). The third subplot showcases the GPyModelWrapper, which utilizes a Gaussian Process (GP) regression model to estimate the objective function and its uncertainty. Each subplot provides valuable insights into the Bayesian optimization process, aiding in efficient optimization of the objective function.

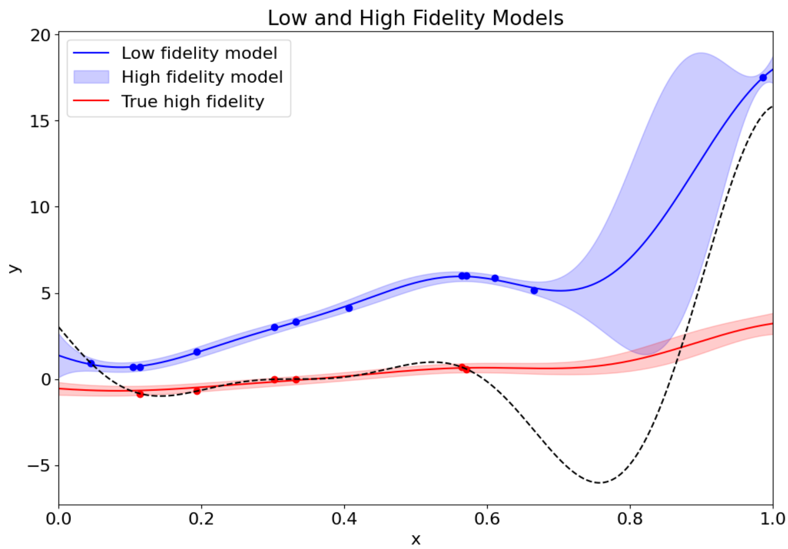

Figure 3: Multi-Fidelity Neural Network Learning the Forester Function. Information from low and high fidelity are used for function approximation

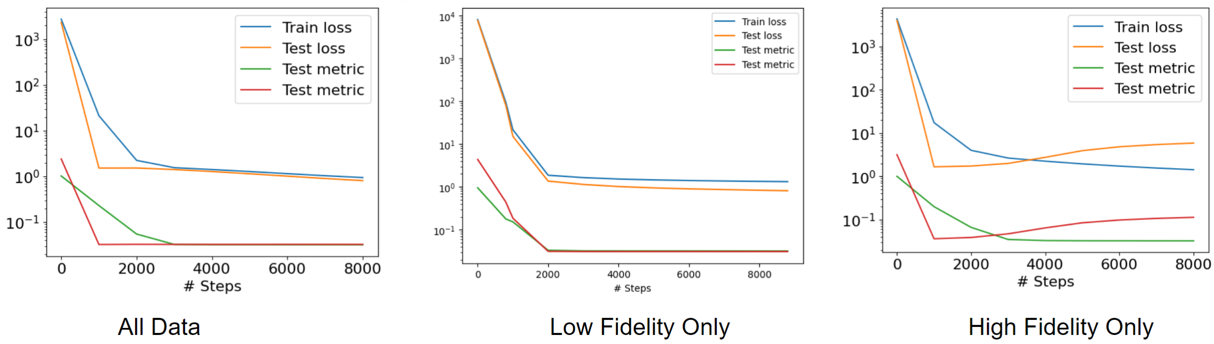

Figure 4: The training history of the deepXDE multi-fidelity neural network over different data scenarios. Each subplot depicts the effect of withholding parts of the dataset

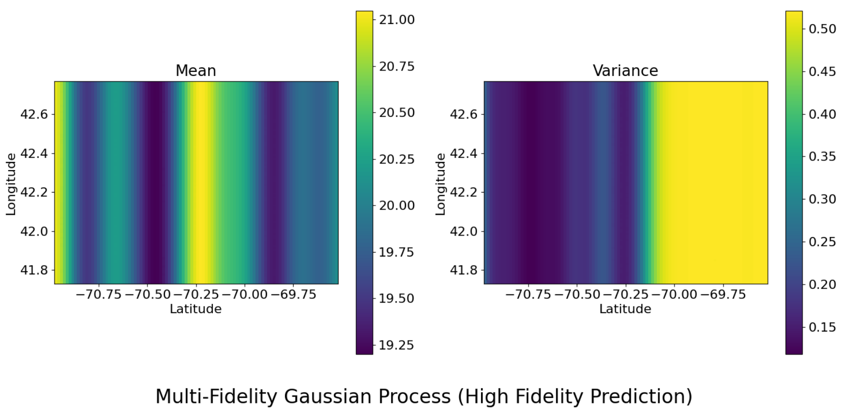

Figure 5: High-Fidelity Prediction Using the MF-GP

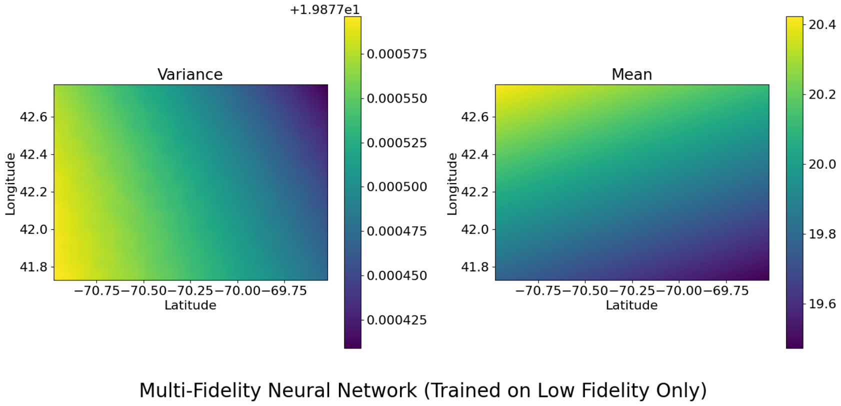

Figure 6: Low-Fidelity Prediction Using the MF-NN

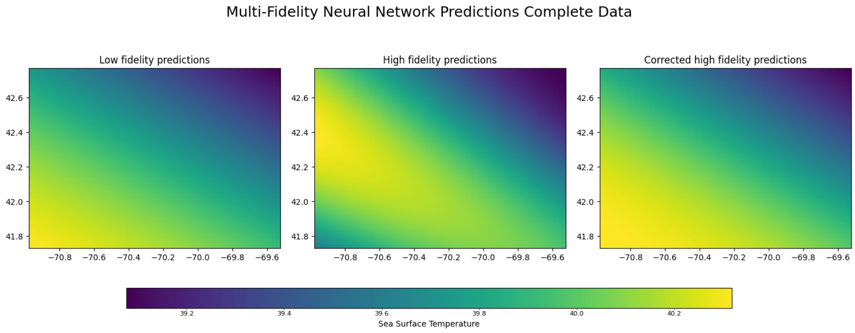

Figure 7: Correct Sea-Surface Prediction Using the MF-NN

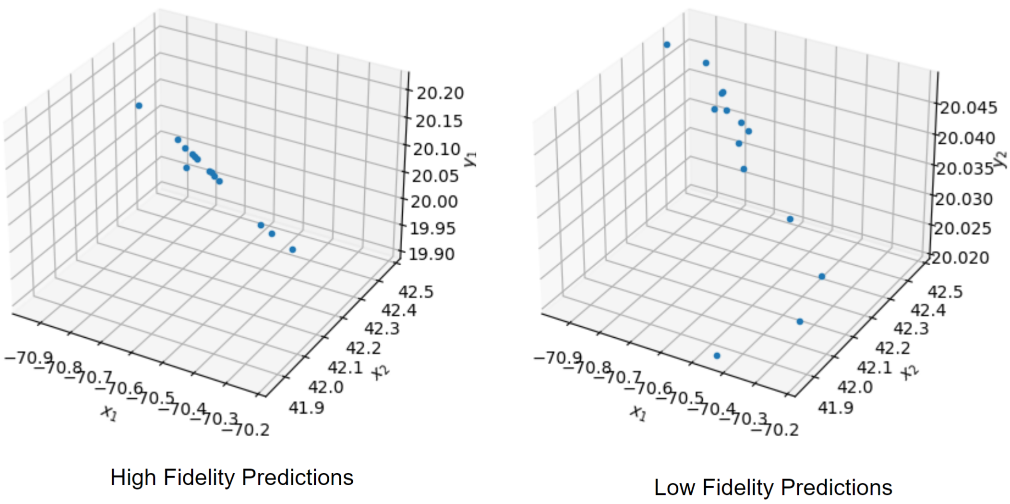

Figure 8: Correlations Between Neural Network Predictions Using the MF-NN

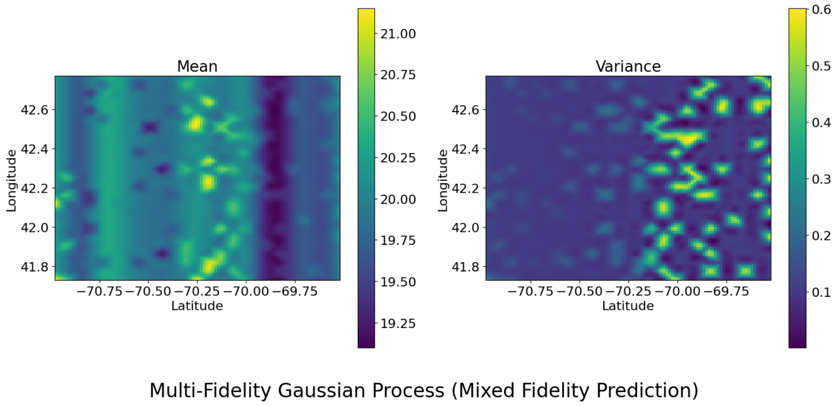

Figure 9: Multi-Fidelity Gaussian Process Predicting the mixed fidelity sea surface temperatures over the Boston Bay. When we add random high fidelity points into the output, the model struggles to handle these predictions

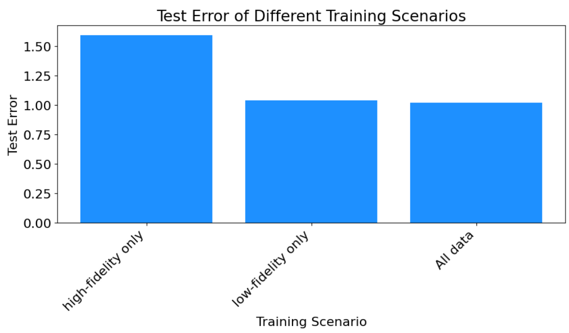

Figure 10: Bar chart depicting the effect of with-holding data on the MF-NN network. Low-fidelity data has the most prominent affect on reducing the loss

5 Discussion

Our successful implementation of multi-fidelity learning to approximate the Forester function (Figure 3) demonstrated the power of combining low- and high-fidelity data. The combined fidelity approach resulted in a more accurate approximation than either fidelity could achieve alone, confirming the importance of multi-fidelity strategies in complex function approximation tasks.

The active learning approach employed in our Bayesian optimization framework (Figure 2) introduced a cost-acquisition function. This function facilitated the selection of points to maximally reduce uncertainty with minimal sampling. Such an approach could have far-reaching implications in various fields. For instance, in reinforcement learning, it could improve the efficiency of learning algorithms by prioritizing exploration in the most uncertain areas of the state-action space. Moreover, in environmental decision-making, this method could enable more efficient sampling and data collection strategies, thereby maximizing the information yield per unit of sampling effort.

The sea surface temperature prediction task further highlighted the differences between our multi-fidelity neural network (MF-NN) and multi-fidelity Gaussian Process (MF-GP) models (Figure 9). Our findings indicate that the MF-NN more accurately represents the sea surface temperature. However, this performance difference may be contingent on the nature and quality of the data, emphasizing the need for further research to understand the conditions under which each model may excel.

Our investigation into the effects of withholding data during training (Figure 4 and Figure 10) revealed the significant role of low-fidelity data in reducing the loss function. This finding suggests that, even in a data-sparse environment, low-fidelity data may contribute more to the model's performance than high-fidelity data. However, this conclusion should be interpreted with caution, and further research is required to verify these findings.

While the above results were demonstrated using simplified toy examples, they serve as a solid foundation for future work. They showcase the potential of multi-fidelity learning, active learning, and Bayesian optimization in handling real-world problems. There are several obvious extensions to improve these methods, such as incorporating more advanced acquisition functions in the Bayesian optimization process, developing more efficient training regimes for the MF-NN model, and devising strategies to handle high-fidelity data in data-sparse environments. The journey to fully exploit these methodologies in practical applications has only just begun.

6 Future Work

Several promising avenues remain unexplored in our current study, which will be the primary focus of our future work.

6.1 Multi-Fidelity Active Learning

Our current work did not explore the potential of multi-fidelity active learning due to time constraints. Multi-fidelity approaches can be highly beneficial in scenarios where high-fidelity data is expensive or time-consuming to acquire, enabling the model to learn effectively from lower fidelity data. This concept could be integrated within our machine learning framework, providing a more efficient and cost-effective solution.

6.2 Physics-Informed Neural Networks for Uncertainty Quantification

The usage of Physics-Informed Neural Networks (PINNs) could offer considerable advancements in uncertainty quantification. PINNs incorporate known physical laws into the structure of the neural network, thereby improving the accuracy and robustness of the predictions, as well as offering an intrinsic method for uncertainty quantification.

6.3 Air Quality Measurement Synthesis

In collaboration with a doctoral student from the Environmental, Earth, and Planetary Sciences (EEPS) lab, we plan to integrate air quality measurements from air sensors and climate models. This synthesis of data sources could improve our understanding of air quality dynamics and create more accurate and comprehensive predictive models. Further studies will be aimed at designing and implementing machine learning algorithms to accomplish this task effectively.

References

[1] H. Babaee, C. Bastidas, M. DeFilippo, C. Chryssostomidis, and G. E. Karniadakis. A Multifidelity Framework and Uncertainty Quantification for Sea Surface Temperature in the Massachusetts and Cape Cod Bays. *Earth and Space Science*, 7(2):e2019EA000954, 2020. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2019EA000954.

[2] Xuhui Meng, Hessam Babaee, and George Em Karniadakis. Multi-fidelity Bayesian Neural Networks: Algorithms and Applications. *Journal of Computational Physics*, 438:110361, August 2021. arXiv:2012.13294 [physics].

[3] Andrei Paleyes, Mark Pullin, Maren Mahsereci, Cliff McCollum, Neil D. Lawrence, and Javier Gonzalez. Emulation of physical processes with Emukit, October 2021. arXiv:2110.13293 [cs].