Toward a Kernel-Based Uncertainty Decomposition Framework for Data and Models

Rishabh Singh  
rish283@ufl.edu  
Jose C. Principe  
principe@cnel.ufl.edu  
Computational NeuroEngineering Laboratory, Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, U.S.A.

This letter introduces a new framework for quantifying predictive uncertainty for both data and models that relies on projecting the data into a gaussian reproducing kernel Hilbert space (RKHS) and transforming the data probability density function (PDF) in a way that quantifies the flow of its gradient as a topological potential field (quantified at all points in the sample space). This enables the decomposition of the PDF gradient flow by formulating it as a moment decomposition problem using operators from quantum physics, specifically Schrödinger’s formulation. We experimentally show that the higher-order moments systematically cluster the different tail regions of the PDF, thereby providing unprecedented discriminative resolution of data regions having high epistemic uncertainty. In essence, this approach decomposes local realizations of the data PDF in terms of uncertainty moments. We apply this framework as a surrogate tool for predictive uncertainty quantification of point-prediction neural network models, overcoming various limitations of conventional Bayesian-based uncertainty quantification methods. Experimental comparisons with some established methods illustrate performance advantages that our framework exhibits.

1 Introduction

1.1 Information Theory: Physics-Based Perspective. The modern foundation of information theory lies in its ability to quantify uncertainty in a random variable $X$ using entropy (Shannon), which was later generalized by Rényi (1961) among many others. It has become an indispensable tool in communication theory and machine learning as a predictor of value of information to optimize design engineering systems and as a descriptor for density estimation and other statistical evaluations that attempt to characterize the intrinsic generating functions of data (Kullback & Leibler, 1951; Theil & Meisner, 1980; Hahn & Shapiro, 1967). There was an earlier definition of information in statistics proposed by Sir Ronald Fisher in the
context of parameter estimation (Fisher, 1922), which measures the amount of information that an observable random variable $X$ carries about an unknown parameter $\theta$ of a distribution that models $X$. In statistical mechanics, Boltzmann’s (1877) entropy is the logarithm of the number of arrangements that a system can be configured in and it still remains consistent with the thermodynamic observables. Boltzmann and Shannon information share exactly the same mathematical formula. Fisher information (Fisher, 1922) has been well regarded as the cornerstone concept in measuring the gain of information from data and in quantifying the order of a system (Frieden & Hawkins, 2010) instead of disorder, as is done by entropy. This presents a strong physics-based interpretation of data analysis (Frieden, 2004), and it has been used to formulate quantum physical models of stock market data in the field of econophysics (Meng, Zhang, & Guo, 2016; Ahn, Choi, Dai, Sohn, & Yang, 2018). More recently in quantum computing, von Neumann’s quantum entropy (von Neumann, Wigner, & Hofstadter, 1955), which is based on Shannon entropy, is considered to be the cornerstone to design quantum computer algorithms and communication channels. The Belavkin equation (Belavkin, 1989) is a stochastic differential equation describing the dynamics of a quantum system undergoing observation in continuous time, which reduces to Schrödinger’s equation in special conditions. As this brief review shows, there is a conceptual equivalence in modeling information from both the statistical mechanics and the machine learning perspectives that served as inspiration for our framework.

1.2 Structure of This Letter. Because of the novelty of the approach, we start with the motivation and high-level description of the methodology and contributions. The technical part starts in section 2 with the definition of the kernel mean embedding (KME) theory and its extension to the potential field interpretation of data PDF in the sample space. In section 3, we formulate a functional operator on the PDF using Schrödinger’s equation to define the quantum field potential of the data. In section 4, we show how to perform a moment decomposition of the quantum field potential to extract its uncertainty modes. Section 5 presents a step-by-step summary of the entire proposed framework. We first analyze our framework from a completely data-driven perspective (without involving a model) and provide pedagogical examples in section 6 using implementation of the framework on time-series signals, which provides an intuitive understanding of how it characterizes data PDF. We subsequently present the application of the framework for quantifying predictive uncertainties of neural network models in section 7, where we also discuss how the proposed framework differs from existing Bayesian methods in terms of scalability, precision, and computational cost. Related experimental results are provided in section 8, which initially consists of examples illustrating advantages offered by our framework compared to established methods such as Monte Carlo dropout and gaussian process (GP) regression. We also present comparison results
on a transfer learning application to evaluate the scalability of our framework. Further results on some benchmark data sets provide quantified evidence of our framework’s performance advantages.

1.3 Motivation and Contributions.

1.3.1 Data Uncertainty Viewpoint. In most real-world scenarios, observed signals are generated by systems controlled by a multitude of source processes and noise, resulting in complicated data dynamics. Current machine learning models and information-theoretic divergence measures characterize the global data structure but fail to effectively characterize local uncertainties associated with such data. Quantum-based formulations in physics, on the other hand, have been well known for providing high-resolution, multiscale characterizations of system dynamics. This is achieved through a stochastic description of the system in terms of energy modes in a Hilbert space. The key quality of such formulations is their nonparametric nature and a complete multiscale quantification of modes, leading to a description of the system at all points in space. We hypothesize that extending information-theoretic measures in terms of such physics-based formulations could yield similar advantages in the characterization of data and models, thereby providing a multiscale enhanced view of their distributions. This inspires us to develop nonparametric data characterization tools that utilize operators from physics to decompose PDF information while making minimal assumptions. Our conjecture is that the best way to achieve this goal is to begin with nonhomogeneous, local, data-induced metric spaces that would be highly sensitive to signal characterization (Liu, Pokharel, & Principe, 2007). Perhaps the best contender for this metric space is the gaussian reproducing kernel Hilbert space (RKHS), which has been well established to provide universal characterization of data (Parzen, 1970; Williams & Rasmussen, 2006; Bergman, 1970; Vapnik, 2013). Moreover, projecting data into an RKHS transforms them into gaussian functions centered at the data coordinates, which obey the properties of a potential field (Principe, 2010). Hence, the RKHS makes it possible to obtain physical formulations of data properties with simplicity because of the uses of the Hilbert space.

Therefore, toward the goal of effectively quantifying data uncertainty, we introduce an RKHS-based information-theoretic framework that utilizes a physical interpretation of the data space to extract its various uncertainty moments. This methodology has solid foundations because it guarantees, from the kernel mean embedding (KME) theory (Muandet, Fukumizu, Sriperumbudur, & Schölkopf, 2017), accurate estimations of the data PDF as a functional in the RKHS. We interpreted KME as a potential field and called it the information potential field (IPF), where there is a value associated with each data point in the feature space (Principe, 2010). Here we extend this interpretation by taking the normalized Laplacian of the IPF...
over the space of samples to emphasize local contributions to the IPF. The Laplacian of the IPF is inspired by Schrödinger’s equation and represents the gradient flow of density in the Hilbert space—hence, the name quantum information potential field (QIPF). As in quantum physics, one can perform a moment decomposition of the QIPF using Hermite polynomials and obtain multiscale projections of the QIPF at every point in space. Our uncertainty framework is depicted in Figure 1 and can be summarized in terms of the following key steps:

1. **Field-based interpretation of data PDF**: Projection of samples into the gaussian RKHS to obtain a functional description of the data PDF over the space of the samples interpreted as the IPF.

2. **Quantifying the gradient flow of the IPF**: Use Schrödinger’s equation to locally quantify the gradient flow of the IPF over the function space, which we called the QIPF.

3. **PDF mode extraction**: Solve the QIPF moment decomposition problem (formulated by Schrödinger’s equation) using Hermite polynomial projections to extract various moments of the QIPF that provide a high-resolution selective quantification of uncertainty at different regions of the data distribution.

Such a framework presents an entirely new characterization of the data that enables one to quantify the flow across local regions of the PDF by using only a few samples as initial reference. This is not possible by current statistical and Bayesian-based methodologies. It also makes the framework highly suitable for applications on online streaming data. Furthermore, the reliance of the framework on the gaussian RKHS enables it to intrinsically extract a rich statistical representation of the data. The physics-based decomposition of the PDF results in high-resolution, multiscale descriptors of anisotropy, with the higher-order moments being increasingly sensitive to samples in the tail regions of the distribution (where epistemic uncertainty is maximum). This is in stark contrast to the traditional moment decomposition methods of the PDF, which describe central moments.
1.3.2 Model Uncertainty Viewpoint. Deep learning neural network models have made remarkable progress over the past two decades in a large variety of machine learning applications (LeCun, Bengio, & Hinton, 2015). However, despite their success, such models only quantify the expected value of the model output, given the input, and do not provide any information related to their prediction uncertainties. This information is crucial for engineering applications and in sensitive application arenas such as personalized medicine and autonomous driving, especially given how prone neural networks are to overfitting. Moreover, there have been alarming revelations regarding the high susceptibility of such models to adversarial attacks (Su, Vargas, & Sakurai, 2019; Nguyen, Yosinski, & Clune, 2015).

We are therefore motivated to explore the utility of our proposed framework for the predictive uncertainty quantification of deep learning models. The main idea here is to find the uncertain regions in the input-output mapping distribution that the model learns. In this case, instead of working on data PDF, we propose an uncertainty decomposition of the distribution that the model parameters learn. One way to achieve this on a sample-by-sample basis is by evaluating the cross-entropy between the instantaneous test set outputs (learned distribution realizations) of the internal layers and final layer (model output). The cross-entropy can be directed estimated from the cross-information potential (Principe, 2010), making the entire proposed quantum mode decomposition directly applicable to model uncertainty quantification. Before delving into more details, we first review some established methods of uncertainty quantification (UQ).

Existing UQ methods can be broadly classified into forward UQ and inverse UQ methods from an implementation point of view (Smith, 2013; Sullivan, 2015). In uncertainty propagation (forward UQ), one attempts to directly characterize the model output uncertainty distribution from the implicit uncertainties present in the parameters. Inverse UQ attempts to quantify uncertainty distributions over model parameters. In the context of machine learning, most of the focus has been on the latter category, with a domination of Bayesian-based inferencing methods that offer the most mathematically grounded approach to quantify model uncertainty by learning probability distributions of model weights (MacKay, 1992; Neal, 2012; Bishop, 1995).

Early development of Bayesian-based models revolved around Laplacian approximation (MacKay, 1992), Hamiltonian Monte Carlo (Neal, 2012), and Markov chain Monte Carlo (MCMC)-based Bayesian neural networks (Bishop, 1995). Although such methods offer a principled approach of quantifying model uncertainty mainly by marginalizing over model parameters, they involve prohibitive computational costs and lack scalability to large data and model architectures. Most of the recent work in this field has therefore been related to developing faster variational inference-based approaches that offer more efficient ways of training Bayesian neural
networks (BNNs) (Graves, 2011; Paisley, Blei, & Jordan, 2012; Hoffman, Blei, Wang, & Paisley, 2013). The high-parameter dimensionality and the complexity of weight associations in modern neural networks still make it very difficult for such variational inference approaches to adequately capture parameter dependencies (Pradier, Pan, Yao, Ghosh, & Doshi-Velez, 2018). Other methods involve surrogate modeling techniques that exploit the input-output mapping that the model learns (Nagel, 2017; Fang, Li, & Sudjianto, 2005; Forrester, Sobester, & Keane, 2008). Here, computationally cheap approximations of models are used for easier extraction of the relevant information related to model uncertainty. Forward UQ methods include ensemble-based methods where multiple instances of models with different initializations are trained on noisy data and the result is the aggregation of all model outputs (Tibshirani, 1996; Osband, Blundell, Pritzel, & Van Roy, 2016; Pearce, Zaki, Brintrup, & Neel, 2018). The variations of the results provide the necessary uncertainty information. A notable related work is that of Lakshminarayanan, Pritzel, and Blundell (2017) who use ensemble neural networks to implement forward UQ.

Recent work by Gal and Ghahramani (2016) has gained increased popularity due to its simplicity and effectiveness in quantifying predictive uncertainty. The authors propose Monte Carlo dropout where multiple instantiations of dropout are used during testing of models to obtain the uncertainty intervals associated with the model predictions. A critical disadvantage of this method in real-time applications is the requirement for multiple forward passes during testing, which is not feasible for very large networks used in modern data sets.

We advocate an approach for predictive uncertainty quantification that is nonintrusive to the training process of a traditional deep learning model and relies solely on extracting information from the internal parameter distribution of the trained model with respect to its output. In this regard, we hypothesize that the application of our framework as a forward UQ method could be advantageous. The idea is to create an alternate representation (or an embedding) of the model that quantifies epistemic uncertainty based on the location of its prediction with respect to its learned parameter PDF, with the uncertainty being relatively higher for output locations closer to the tails of the parameter PDF. Toward this end, we utilize our framework as a surrogate uncertainty quantifier of a trained neural network that decomposes the model’s parameter PDF (input-output mapping), realized at every test instance, in terms of uncertainty modes that provide a high-resolution location of the model’s output with respect to its PDF. The implementation is depicted in Figure 2. From a physics-based analogical perspective, one can visualize the framework’s representation of the model as a drum membrane and the output as a drumstick hitting the membrane during each test cycle. The resultant modes of membrane vibration (uncertainty modes of the model PDF), which depend on where exactly the membrane was hit, quantify the uncertainties of the model with respect to the output. We
submit various advantages offered by our framework over traditional model uncertainty quantification paradigms:

- Our framework is nonintrusive to the model’s training process and provides a single-shot estimation of the model’s epistemic uncertainty during testing, unlike variational-inference-based methods that are dependent on random sampling of the model.
- Instead of estimating central moments of uncertainty, as is done by Bayesian methods, the proposed framework extracts high-resolution local uncertainty moments at every point in the sample space that better enhance the quantification of data regions unknown to the model.
- We posit that our framework quantifies epistemic uncertainty with much greater precision and scales better to larger models in modern applications than variational-inference-based methods.

2 Quantifying Data PDF in the RKHS

Functional statistics have a long history, starting with the work of Karhunen (1946), Loeve (1946), Grenander (1950), and Rao (1958). In the early days, the big practical issue was the infinite dimensionality of the space, which was conquered with subspace projections.

2.1 Kernel Mean Embedding. Kernel methods have been very popular and are well established in the field of machine learning (Smola & Schölkopf, 1998). The crux of their success is largely owed to a powerful property of the reproducing kernel Hilbert space (RKHS) associated with positive-definite kernels called the “kernel trick” (Aronszajn, 1950), which allows one to pose any problem in an input set $X$ as a linear-algebraic problem in its RKHS, $H$, with a nonlinear transformation (embedding) of $X$ into $H$ induced by a kernel $k : X \times X \rightarrow R$. In other words, the RKHS, constructed by an appropriate kernel, allows one to simplify any nonlinear relationship
Kernel-Based Uncertainty Decomposition Framework

in an input space as a linear expression in a higher-dimensional space. This property has led to the advent of many popular kernel-based algorithms in machine learning (Hofmann, Schölkopf, & Smola, 2008; Liu, Principe, & Haykin, 2011). Following similar intuition, another elegant property of the RKHS is the theory of kernel mean embedding (KME), which allows one to nonparametrically quantify a data distribution from the input space as an element of its associated RKHS (Berlinet & Thomas-Agnan, 2011). For a detailed explanation of the metric, we refer readers to Muandet et al. (2017). Its definition is summarized as follows.

**Definition 1 (Kernel Mean Embedding).** Suppose that the space \( \mathcal{Z}(\mathcal{X}) \) consists of all probability measures \( \mathbb{P} \) on a measurable space \( (\mathcal{X}, \Sigma) \). The kernel mean embedding of probability measures in \( \mathcal{Z}(\mathcal{X}) \) into an RKHS denoted by \( \mathcal{H} \) and characterized with a reproducing kernel \( k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R} \) is defined by a mapping

\[
\mu : \mathcal{Z}(\mathcal{X}) \rightarrow \mathcal{H}, \quad \mathbb{P} \mapsto \int k(x, )d\mathbb{P}(x).
\]

Hence, the kernel mean embedding (KME) represents the probability distribution in terms of a mean function by utilizing the kernel feature map in the space of the distribution. In other words,

\[
\phi(\mathbb{P}) = \mu = \int k(x, )d\mathbb{P}(x). \tag{2.1}
\]

There are several useful properties associated with the KME. For instance, it is injective for characteristic kernels, meaning that \( \mu_{\mathbb{P}} = \mu_{\mathbb{Q}} \) only when \( \mathbb{P} = \mathbb{Q} \), thus allowing for unique characterizations of data distributions. It also makes minimal assumptions on the data-generating process and enables extensions of most learning algorithms in the space of probability distributions.

In most real-world applications, there is no information available regarding the nature of \( \mathbb{P} \). One must therefore resort to empirical estimation of the KME. The simplest method of empirically computing the KME is by computing its unbiased estimate given by

\[
\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} k(x_{i}, ). \tag{2.2}
\]

Here \( \hat{\mu} \) converges to \( \mu \) for \( n \rightarrow \infty \), in concordance with the law of large numbers. One can intuit that the empirical KME is also a result of the general Dirac formulation assigning a mass of \( 1/n \) to every data sample. This also gives rise to the interpretation of the empirical KME as an instance of a point process (Muandet et al., 2017).
2.2 Potential Field Interpretation. Another important quantity, which extends KME to the space of samples, is the information potential (IP) of the data set (Principe, Xu, Fisher, & Haykin, 2000), which is simply the empirical mean value of the PDF over the space of samples in the RKHS. It appears as the argument of the logarithm, $\Psi(X)$ in Renyi’s quadratic entropy (Rényi, 1961) given by

$$H_2(X) = -\log \int p(x)^2 dx = -\log \Psi(X).$$  \hspace{1cm} (2.3)

One can estimate $\Psi(X)$ by using the Parzen density estimator (Parzen, 1962) for estimating $p(x)$. Hence, assuming a gaussian kernel window of kernel width $\sigma$, one can readily estimate directly from experimental data $x_i, i = 1, \ldots, N$ the information potential as

$$\Psi(X) = \int p(x)^2 dx = \int \left( \frac{1}{N} \sum_{i=1}^{N} G_\sigma (x - x_i) \right)^2 dx$$

$$= \frac{1}{N^2} \int \sum_{i=1}^{N} \sum_{j=1}^{N} G_\sigma / \sqrt{2} (x_j - x_i).$$  \hspace{1cm} (2.4)

Therefore, the IP is a number obtained by the double sum of the gaussian functions centered at differences of samples with a larger kernel size. There is a physical interpretation of $\Psi(X)$ as the total information potential, if we think of the projected samples (the gaussians) as particles in a potential field—hence, the name information potential. The total potential can be written as

$$\Psi(X) = \frac{1}{N} \sum_{j=1}^{N} \psi(x_j),$$  \hspace{1cm} (2.5)

where

$$\psi(x) = \frac{1}{N} \sum_{i=1}^{N} G(x - x_i)$$  \hspace{1cm} (2.6)

represents the field due to the addition of gaussians centered at each sample. We refer to $\psi(x)$ as the information potential field (IPF), and it is the equivalent of the probability measure in RKHS, that is, it gives us the probability in each point of the space induced by the data.

We see here that the main outcome of the RKHS-based formulation of data PDF involves a representation over all the space populated by samples, where each sample becomes a gaussian function. But notice that we still have not quantified the local anisotropy created by the gaussian
bumps, which changes across space and is dependent on the local density of samples. Having such a data description is beyond the concept of PDF. However, it is possible to define local operators in the RKHS to formulate a field-based definition of the IPF that is governed by local interactions between the gaussian bumps. Similar to spectral theory, one can impose a local differential operator (the Laplacian) over the IPF to eventually formulate a decomposition in the gaussian RKHS, which can be an indicator for uncertainty in that particular local region of the space. Indeed, the approach of utilizing the Laplacian operator is the crux for describing many physical phenomena formed as a cumulation of smaller interacting states such as electric and gravitational potentials and wave propagation. For instance, in Gauss’s law formulation, the Laplacian operator is used for describing the local density associated with an electrostatic potential, which subsequently describes the overall charge distribution \( q = -\varepsilon_0 \nabla^2 \varphi \). As another example, the Laplacian is also used for describing the density associated with a gravitational potential to subsequently describe the overall mass distribution. We similarly theorize that one can utilize the Laplacian operator here as well for describing the local sample density associated with the information potential (IPF) around a data sample.

3 Schrödinger’s Eigenvalue Problem over the IPF

We know that in quantum physics, systems are characterized by discrete changes in their stochasticity or uncertainty described by their various eigenmodes (principal components), which describe our analysis goals very well. Moreover, there are well-established mathematical procedures such as the Schrödinger’s eigenvalue problem to decompose the wave function in moments that quantify uncertainty, which fits very well with our goals. Of course, we will have to make appropriate modifications to the theory since we are not working with a physical system but with an abstraction of “particles” as data samples on a potential field defined by the gaussian function in RKHS. To this end, we use a data-equivalent Schrödinger formulation over the IPF given by \( H \psi(x) = E \psi(x) \) (Principe, 2010). Here \( H \) denotes the Hamiltonian over \( \psi(x) \), and \( E \) is the total energy over the sample space. All the physical constants have been lumped on the only variable in the gaussian RKHS, the size of the kernel. Similar to a general quantum system, the Hamiltonian is constructed using two operators: the kinetic and potential energy operators. The kinetic energy operator here consists of a Laplacian function over the IPF, that is, \(-\frac{\sigma^2}{2} \nabla^2 \psi(x)\), which essentially quantifies local sample density in our context, as discussed in section 2.2. The potential energy operator will be denoted by \( V_s \) which we refer to as the quantum information potential field (QIPF). The overall formulation becomes an eigenvalue problem over the data PDF and is given as

\[
H \psi(x) = \left( -\frac{\sigma^2}{2} \nabla^2 + V_s(x) \right) \psi(x) = E \psi(x).
\] (3.1)
Here, the IPF, $\psi(x)$, is the probability measure and the equivalent of the wave function (seen in general quantum systems) that is being decomposed by Schrödinger’s formulas. Note that $\psi(x)$ is the eigenfunction of $H$, and $E$ is the lowest eigenvalue of the operator, which corresponds to the ground state. Since, by convention in quantum systems, the probability measure is defined as the square of the wave function ($p(x) = |\psi(x)|^2$), we rescale $\psi(x)$ in equation 3.1 to be the square root of the IPF, that is, $\psi(x) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} G_{\sigma}(x - x_i)}$. Rearranging the terms in equation 3.1 we get

$$V_s(x) = E + \frac{\sigma^2/2\nabla^2\psi(x)}{\psi(x)}.$$  (3.2)

To determine the value of $V_s(x)$ uniquely, we require that $\min(V_s(x)) = 0$, which makes $E = -\min \frac{\sigma^2/2\nabla^2\psi(x)}{\psi(x)}$.

Hence, in summary, we formulated a Schrödinger’s eigenvalue problem over IPF, equation 3.1, to go beyond the data PDF in the form of $V_s(x)$ in equation 3.2. This is basically a variant of the field-based interpretation of the data PDF. Given the data set, we expect $V_s(x)$ to increase quadratically outside the data region and to exhibit local minima associated with the locations of highest sample density (clusters). This can be interpreted as clustering since the potential function attracts the data density function $\psi(x)$ to its minima, while the Laplacian drives it away, producing a complicated potential function in the space. We should remark that in this framework, $E$ sets the scale at which the minima are observed. This derivation can be easily extended to multidimensional data. We can see that $V_s(x)$ in equation 3.2 is also a potential function that differs from the information potential ($\psi(x)$) formulation of the data PDF because it is now an energy-based formulation associated with a quantum field description of the PDF.

4 Extracting Uncertainty Moments from the QIPF

Unlike the classical interpretation of systems, the quantum interpretation provides a much more detailed decomposition of the system dynamics by assuming it to consist of a large (potentially infinite) number of stochastic features, given by the energy and uncertainty modes. Similarly, when applying this quantum field potential to data, the same interpretation holds. However, this is beyond the conventional description of measure and PDFs. Our results support the concept that the QIPF provides information about the gradient flow of PDF within local space neighborhoods, quantifying anisotropy, but it is still a global measure over the sample space. It is clear that the anisotropy will be higher in regions of the space with a sparser number of samples, which corresponds to the tail regions of the PDF. For practical applications we would like to quantify anisotropy at different densities of samples. Therefore, we further propose a mode decomposition of the QIPF that would further quantify the local gradient flow of the PDF with
high resolution and precision (sensitivity and specificity). Schrödinger’s eigenvalue formulation allows us perform this decomposition with relative ease. Moreover, since such a formulation has been well established in physics to yield uncertainty functionals at all operating points of a physical system, we expect its data-based form, equation 3.1, to yield similar moments in the sample space.

In quantum physics, the solution of a Schrödinger’s equation describing some physical system (e.g., a quantum harmonic oscillator) yields successive eigenstates of the wave function (quantifying system uncertainties) that are related to each other by an orthogonal polynomial sequence called the Hermite polynomials. Following this solution pattern, we extract eigenstates (uncertainty moments) of the data-based Schrödinger’s equation by projecting the wave function (IPF), assumed to be in the ground state, into successively higher-order Hermite polynomial spaces and then computing the corresponding QIPF, equation 3.2.

The generating function of the physicist’s version of Hermite polynomial sequence, first defined by Pierre-Simon Laplace (Stigler, 2005) and later formalized by Charles Hermite (Hermite, 1864), is given as

\[ H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} e^{-x^2}, \] (4.1)

which can be simplified into an explicit form as follows:

\[ H_n(x) = n! \sum_{m=0}^{\lfloor \frac{n}{2} \rfloor} \frac{(-1)^m}{m!(n-2m)!} \frac{x^{n-2m}}{2^m}. \] (4.2)

Apart from occurring in the solutions of eigenvalue problems in physics, the sequence is also seen in various other diverse fields such as combinatorics, numerical analysis (as gaussian quadrature), and probability (Edgeworth series). Interestingly, Hermite functions also form an \( L^2 \) orthonormal basis, which diagonalizes the Fourier transform operator. Following are two of the useful properties of the series:

- **Orthogonality:** Hermite polynomials are orthogonal with respect to the weight function \( w(x) = e^{-x^2} \):

\[ \int_{-\infty}^{\infty} H_m(x)H_n(x)e^{-x^2}dx = \sqrt{\pi} 2^n n! \delta_{nm}. \] (4.3)

- **Recurrence relation:** The physicist’s Hermite polynomial sequence also satisfies the following recursion formula:

\[ H_{n+1}(x) = 2xH_n(x) - H'_n(x). \] (4.4)
Hence we formalize the solution for the moment decomposition of the QIPF equation by using the following conjecture:

**Conjecture 1: Extraction of QIPF uncertainty modes.** Consider the QIPF of the data samples \( x \) as \( V_s(x) = E + \frac{\sigma^2/2\nabla^2\psi(x)}{\psi(x)} \) with the associated ground state wave function given by \( \psi(x) = \sqrt{\frac{1}{\pi}} \sum_{i=1}^{n} k(x_i, t) \). The approximate higher-order uncertainty (or energy) modes of \( \psi(x) \) can be extracted by projecting the ground-state wave function into the corresponding order Hermite polynomial given by \( \psi_k(x) = H_k^*(\psi(x)) \), where \( H_k^* \) denotes the normalized \( k \)th-order Hermite polynomial, normalized so that \( H_k^* = \int_{-\infty}^{\infty} e^{-x^2} [H_k^*(x)]^2 = 1 \). This leads to the evaluation of the higher-order QIPF states as

\[
V_{s}^{k}(x) = E_k + \frac{\sigma^2/2\nabla^2H_k^*(\psi(x))}{H_k^*(\psi(x))} = E_k + \frac{\sigma^2/2\nabla^2\psi_k(x)}{\psi_k(x)}. \tag{4.5}
\]

where \( k \) denotes the order number and \( E_k, \) denotes the corresponding eigenvalues of the various modes and is given by

\[
E_k = -\min \frac{\sigma^2/2\nabla^2\psi_k(x)}{\psi_k(x)}. \tag{4.6}
\]

The extracted modes of the data QIPF given by \( V_{s}^{k}(x) \) are thus stochastic functionals depicting the different moments of uncertainty of the data at any point \( x \). This is different from the IPF formulation of equation 2.6, and it can also be visualized as an energy-based metric resembling the potential energy operator in a quantum harmonic oscillator at various energy levels (eigenstates) depicted by \( E_k \).

5 **Summarizing the Framework**

The implementation details of the framework are illustrated in Figure 3, and the key aspects are briefly summarized as follows:

- **Quantifying data PDF in the gaussian RKHS:** We project data using the gaussian RKHS and provide a nonparametric characterization of the implicit PDF of the data in an incremental manner of individual gaussian bumps associated with data points in the RKHS (quantified as the IPF).
- **Eigenvalue problem formulation of data PDF:** We quantify the local gradient flow of the data PDF by utilizing the Laplacian operator in
Figure 3: Proposed framework for extraction of quantum uncertainty states.

the form of the QIPF expression, a data-based Schrödinger’s eigenvalue formulation.

- **Extraction of data uncertainty modes**: We extract uncertainty moments (or modes) of the local gradient flow of data PDF, now expressed as an eigenvalue problem (QIPF), by implementing orthogonal Hermite polynomial projections of the ground state wavefunction and finding the corresponding QIPF state.

The mathematical foundations of the gaussian RKHS, specifically the kernel trick, which leads to the kernel mean embedding theory, guarantee that our framework (through information potential computation) is able to nonparametrically and universally estimate the PDF associated with any kind of function. This is established by the universal approximation property of kernel methods. In addition to utilizing this property, the QIPF framework, through its unique uncertainty moment decomposition paradigm, is able to point to different regions of the function’s PDF and quantify how well the function is defined at such regions. Therefore, while this serves as a valuable property for quantifying epistemic uncertainty, the QIPF framework can also be utilized as an efficient signal processing tool for decomposing time series signals. The uncertainty modes can be utilized here for clustering different samples of the signal based on how well they fit in the overall PDF. We therefore formalize the utility of the QIPF framework for time-series signals as follows:

**Conjecture 2: Time-series decomposition using QIPF framework.** A given time series signal $x(t)$ can be characterized in terms of its intrinsic
uncertainty modes extracted through the QIPF framework. This is given by:

\[ V^k_s(x_t) = E_k + \frac{\sigma^2/2 \nabla^2 \psi_k(x_t)}{\psi_k(x_t)} \],

where \( \psi_k(x_t) \) is the wave function evaluation of sample at time \( t \) in an RKHS field formed by using all previous samples in time, \( x_0, x_1, \ldots, x_{t-1} \), as centers. \( V^k_s(x_t) \) is the \( k \)th QIPF mode of signal sample at time \( t \) with respect to previous samples. This is equivalent to spectral decomposition of the signal on a sample-by-sample basis.

6 Mode Decomposition of Time Series

We begin our analysis of the proposed framework by studying how it characterizes time-series signals. We used Matlab R2019a to obtain the results shown in this section. For an intuitive understanding of how the different QIPF modes get configured in the space of data, we extracted the first six modes of a simple sine wave signal. We generated 3000 samples of a 50 Hz unweighted sine wave signal sampled at the rate of 6000 samples per second to mimic a continuous signal. The signal was also normalized to zero mean and unit standard deviation. We used all 3000 samples as centers to construct the wave function (square root of the IPF) and then evaluated it at each point in the data space range \( x = (-6, 6) \) using a step size of 0.1. We then evaluated the Hermite projections of the wave function value at each point to subsequently extract six QIPF modes using the formulation given by equation 4.5. This was done for four different kernel widths whose corresponding QIPF plots (represented by solid color lines) are shown in Figure 4. The dashed line represents the empirical KME estimate (or simply the IPF) given by \( p(x) = \psi^2(x) = \frac{1}{N} \sum_{i=1}^{N} \kappa(x, x_i) \), which basically gives an estimate of the data PDF. All plots were normalized for easier visualization. Perhaps the most important property of the extracted QIPF modes that can be observed from the plots is that for all kernel widths, they systematically signify the more uncertain regions of the data space closer to the tails of the data PDF. In fact, one can observe the significant increase in the density (or clustering) of the extracted QIPF modes as one moves farther away from the mean \( (x = 0) \) and toward the PDF tails. Furthermore, we observe here that the modes appear sequentially based on their orders, with the lower-order modes signifying regions closer to the mean and the higher-order modes clustering together at the PDF tails. An interesting observation that must also be noted is that for larger kernel widths (1.8 and 2.6), which exceed the dynamic range of the signal, we can see that some high-order modes begin to emerge in the region around the mean. This behavior is remarkably similar to physical systems. If we consider the same drum membrane analogy
Figure 4: Analysis of mode locations in the data space using different kernel widths. Solid colored lines represent the different QIPF modes. Dashed line represents the empirical KME (IPF).

we used in section 1, one can visualize the space of the samples here as the membrane. If we increase the tension of the membrane and hit it, the drum will vibrate for a long time. In our potential field, the stiffness is controlled by the kernel size. If the kernel size is large, the QIPF becomes stiffer, leading to the energy in the higher QIPF modes to increase. If one decreases the kernel size, the membrane becomes more elastic, leading to many local modes that decay much faster.

As a pedagogical demonstration to understand how the framework characterizes different dynamical data structures, we implement it to compare the extracted uncertainty modes of a simple sine wave oscillator and a Lorenz series. The sine wave represents one of the simplest time series with a single generating function. The Lorenz series is a chaotic deterministic dynamical system with complex state space defined by the following mutually coupled differential equations (with $\sigma$, $\rho$, and $\beta$ as system parameters) governing its dynamics:

$$\frac{dx}{dy} = \sigma (y - x), \quad \frac{dy}{dt} = x(\rho - z) - y, \quad \frac{dz}{dt} = xy - \beta z. \quad (6.1)$$

We generated 3000 samples of Lorenz series after setting the parameters as $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$ and the initial conditions as $x_1 = 0$, $y_1 = 1$, $z_1 = 0$. 

(a) Kernel width = 0.6

(b) Kernel width = 1.2

(c) Kernel width = 1.8

(d) Kernel width = 2.6
and $z_1 = 1.05$. The signal was also normalized to zero mean and unit variance. We generated two sine wave signals, with the first one having a fundamental frequency of 150 Hz and the second one with an added frequency component (150 Hz + 250 Hz). The signals were sampled at a rate of 6000 samples per second and were normalized to zero mean and unit variance. We extracted the first 10 QIPF modes using equations 4.5 and 4.6 to encode the different signals. The kernel width used for doing so was fixed to a moderate value of 1.2 for all signals. Figure 5 shows the signals (top row) and the corresponding histogram plots (bottom row) of the number of times the value of each QIPF mode dominated over the others throughout the durations of the signals. As can be seen in Figure 5a, there are only two dominant modes in case of the single frequency sine wave (modes 2 and 3). Addition of a frequency component leads to an increase in the number of modes contributing to the signal dynamics to four (see Figure 5b). The dominant modes in the case of Lorenz series are more spread out (across all 10 modes), indicating a more complex data dynamical structure. These trends are quite similar to what we would expect from a frequency decomposition of the signals, except that here, we are able to perform this decomposition on a sample-by-sample basis.

7 Model Uncertainty Quantification

7.1 QIPF Framework Implementation. The QIPF mode extraction framework can be naturally extended for implementation on machine learning models. The fundamental idea here is to create a continuous RKHS embedding of the trained model (represented by its internal weights and
Figure 6: Implementation of the proposed framework on a feedforward artificial neural network.

.activation outputs) that represents its intrinsic distribution. We can then extract the QIPF uncertainty modes associated with the interactions between the model’s embedding and its output. We expect this to quantify the extent to which the model’s test output falls within the scope of its learned predictive distribution. As evidenced in the previous section, we expect higher-order QIPF modes to cluster in data regions where the model has not been trained, represented by the tails of the input-output mapping PDF learned by the network, thereby providing a sensitive uncertainty characterization of data spaces unknown to the model (epistemic uncertainty). In this regard, we specifically focus on neural network models due to a recent surge in their research interest.

The basic implementation strategy of the QIPF decomposition framework on a trained neural network is the same as the data-based implementation, except for the way in which we construct the information potential field (IPF). In this case, we aim to decompose the interactions between the RKHS fields of different pairs of network layers (with one of them typically being the output layer), thereby obtaining a multiscale uncertainty representation of the implicit mapping between them (see Figure 6). Intuitively, this quantifies the probabilistic similarity between two layers of the network during each test cycle. The RKHS field of each layer is represented by the kernel feature map constructed by its corresponding node activation outputs. We evaluate the cross-information potential (CIP), which measures interactions between two information potentials (Principe, 2010). One can represent the cross-information potential between two layers of the
neural network using a generalized form of the kernel mean embedding formulation.

Let us consider two layers of an ANN whose node outputs are represented by the random variables $L_1$ and $L_2$. The kernel feature map of $L_1$ can then be represented in the same form as equation 2.1 as

$$
\mu_{P_{L_1}} = \int k(l_1, .) P_{L_1}(l_1) dl_1.
$$

(7.1)

The mean evaluation of $\mu_{L_1}$ at a point $l_2 \in L_2$ is the value of the information potential field created by $L_1$ at the point $l_2$ and can be represented as

$$
\mu_{P_{L_1}}(l_2) = V_c(P_{L_1}, l_2) = \int k(l_1, .) k(l_2, .) P_{L_1}(l_1) dl_1.
$$

(7.2)

Its empirical evaluation leads to

$$
\hat{V}_c(P_{L_1}, l_2) = \frac{1}{n} \sum_{i=1}^{n} k(l_1(i) - l_2),
$$

(7.3)

which we refer to as the cross-information potential field at $l_2$. The quantum decomposition can then be performed in the same way as before (summarized in Figure 6), thus leading to extraction of multiscale uncertainty features associated with the layer-layer interactions of the neural network. We will typically consider $L_2$ to be the output layer in most empirical evaluations, thus measuring the probabilistic interactions between the model’s output and one or more of the hidden layers. One can either consider the model weights or its activation outputs to create the RKHS embedding of the model. Projection of weights in the RKHS provides a stable quantification of the model’s PDF. The projections of activation outputs, on the other hand, quantify instantaneous realizations of the model’s PDF at each test input and often provides a more computationally feasible approach. In the experiments (see section 9), we show examples of both approaches. We expect the QIPF modes (resulting from the information potential interactions between the hidden layers weights and activations and the prediction output) to be more densely clustered in test data regions unknown to the model, thereby characterizing its epistemic uncertainty at each test iteration.

7.2 Comparison with Bayesian Approaches.

7.2.1 Bayesian Uncertainty Extraction. The basic approach of Bayesian-based predictive uncertainty quantification methods involves probabilistic descriptions of model parameters given a fixed set of data by using variational inference. For instance, consider a set of model variables $\omega = [\omega_i]_{i=1}^{L}$
for a model with $L$ layers. One can then formulate the predictive distribution for a new input point $x^*$ as

$$p(y^*|x^*, X, Y) = \int p(y^*|x^*, \omega)p(\omega|X, Y)d\omega. \quad (7.4)$$

Here $y^* \in \mathbb{R}^D$. To approximate $p(\omega|X, Y)$, variational inference methods work by defining a distribution $q(\omega)$ and minimizing the Kullback-Leibler (KL) divergence between the two distributions, $KL(q(\omega)||p(\omega|X, Y))$, which results in the following formulation of approximate predictive distribution:

$$q(y^*|x^*) = \int p(y^*|x^*, \omega)q(\omega)d\omega. \quad (7.5)$$

In recent years, Monte Carlo dropout (Gal & Ghahramani, 2016) has become a very popular method to approximate the model’s predictive distribution. The authors showed that regularizing any network using dropout and $L_2$ regularization is equivalent to performing variational inference to determine $q(y^*|x^*)$, and Monte Carlo sampling of the network during testing leads to an approximation of its first two moments summarized by the following expectations:

$$E_q(y^*) \approx \frac{1}{T} \sum_{t=1}^{T} \hat{y}(x^*, \hat{M}_1^t, \hat{M}_2^t \ldots \hat{M}_L^t), \quad (7.6)$$

$$E_q((y^*)^T(y^*)) \approx \tau^{-1}I_D + \frac{1}{T} \sum_{t=1}^{T} \hat{y}(x^*, \hat{M}_1^t, \hat{M}_2^t \ldots \hat{M}_L^t)^T \times \hat{y}(x^*, \hat{M}_1^t, \hat{M}_2^t \ldots \hat{M}_L^t), \quad (7.7)$$

with the total number of stochastic forward passes (Monte Carlo samples) being equal to $T$, $\tau$ defining the model precision and $\hat{y}$ being the current output of the $t$th forward pass parameterized by the current input and the model variable realizations in different layers at that particular pass. Hence, quantifying predictive uncertainty using MC dropout becomes equivalent to performing multiple forward passes through the network and averaging the results.

7.2.2 Scalability, Precision, and Computational Cost. Hence, Bayesian-based inferencing methods, which involve posterior estimation of the model’s predictive distribution, require a large number of model realizations to quantify the predictive uncertainty with respect to its output. As is seen from equation 7.5, such methods intrinsically require marginalization
over all model variables to give the ideal estimates of uncertainty. Considering a neural network with \( L \) layers and \( n \) neurons in each layer, this would translate to a complexity of \( \mathbf{O}(n^L) \), assuming we are marginalizing over each neuron (considered as a random variable) to obtain the complete predictive distribution of the model. This is intractable for modern applications involving large data sets and networks. Monte Carlo dropout considerably reduces this cost by marginalizing over only a few neurons selected through random sampling during testing and hence reduces the cost to approximately \( \mathbf{O}(d \times n^L) \) \( \forall 0 < d < 1 \), where \( d \) is the dropout rate. Although the extracted predictive distribution \( q(\omega) \) through variational inference contains multimodal information, as pointed out in Gal and Ghahramani (2016), in practice, one is limited to extracting only the first two central moments given by equations 7.6 and 7.7 to maintain a reasonably low requirement for the number of forward passes \( (T) \). In big data applications, however, where the number of model layers and neurons increases considerably, this too becomes intractable. The proposed QIPF framework differs from the variational inference approaches in a major way by relying on prior estimation of the model’s predictive distribution in the RKHS, thereby offering a single-shot estimation of the model’s predictive distribution with respect to its output. This is enabled by the kernel trick and the mathematical guarantees provided by the kernel mean embedding theory and therefore makes the QIPF framework a more feasible solution for real-time quantification of model uncertainty while testing. The computational bottleneck in the QIPF framework comes from the computation of the information potential field, \( \Psi(X) = \frac{1}{n} \sum_{j=1}^{N} \psi(x_j) \), for data-based implementation or, equivalently, from the computation of the cross-information potential (CIP) given by equation 7.3 for model implementation. Considering the same network \( (L \) layers and \( n \) neurons per layer), the computational complexity of the CIP becomes \( \mathbf{O}(mnL) \), where \( m \) is the number of output-layer neurons. Here we consider the CIP evaluation at \( m \) outputs from the last layer in an RKHS field created by \( nL \) activation outputs as centers. The framework therefore is significantly cheaper in terms of computation when compared with variational inference methods, but its cost is still dependent on the size of the model network (in terms of \( n \) and \( L \)). Several ways of curbing the computational growth of the information potential with respect to the increasing data size are summarized in Principe (2010). One notable method of achieving this is by using the fast Gauss transform (FGT) (Greengard & Strain, 1991), which leverages the shifting property of the gaussian kernel to make the computation of weighted sums of gaussians more efficient. It reduces the computation of CIP from \( \mathbf{O}(mnL) \) to \( \mathbf{O}(m) + \mathbf{O}(nL) \). Another effective approach is by implementing incomplete Cholesky decomposition (Van Loan & Golub, 1983) on the CIP gram matrix (augmented to make it symmetric). This reduces the computational complexity of the CIP from \( \mathbf{O}(mnL) \) to \( \mathbf{O}(mD^2) \) where \( D \) represents the reduced form of the gram matrix associated with \( nL \) centers.
Other methods such as quantization (Chen, Zhao, Zhu, & Príncipe, 2011) involve efficient reduction of the number of RKHS centers \( n_L \) in this example for computing any kernel-based metric such as the CIP in a more computational friendly manner.

A major property that makes the QIPF framework significantly advantageous over Bayesian approaches is the way in which the uncertainty moments are computed. Instead of computing central moments of predictive uncertainty as is done in variational inference methods, we compute the local moments associated with the gradient flow of the data PDF at every point in the sample space. The formulation of the eigenfunction problem through Schrödinger’s equation allows us to potentially extract infinite moments without any sampling restrictions (as is seen in Bayesian approaches) because we simply use successive Hermite polynomial projections in the RKHS for moment decomposition. This enables a much higher resolution of quantified uncertainty than is possible through variational inference. Moreover, Schrödinger’s equation naturally tailors each moment to only quantify the successively more uncertain data regions with respect to its PDF.

8 Experiments and Analysis

We present simulation results to illustrate and compare the performance of the QIPF framework with respect to currently popular approaches for the problem of predictive uncertainty quantification. All simulations described in this section were performed using Python 3.6. Being a kernel-based approach, we compare the QIPF framework’s performance with that of gaussian process regression (GPR) (Williams & Rasmussen, 2006), a widely acclaimed kernel method for machine learning that is known for providing reliable uncertainty estimates associated with its predictions. We also provide comparisons with Monte Carlo dropout (Gal & Ghahramani, 2016) that has gained recent popularity as an approximate variational-inference-based method for uncertainty quantification of neural networks.

8.1 Regression.

8.1.1 Data Sets. We generate two different regression data sets as didactic examples for experimental comparisons and analysis. The idea of such data sets is to simulate real-world scenarios where tasks in machine learning encounter test data from outside the training domain or have to face external noise or outliers in their training set. Indeed these synthesized data sets have also been used in the uncertainty quantification literature for demonstration of methods (Osband et al., 2016; Gal & Ghahramani, 2016). The first data set consists of 60 regression pairs \( x_i, y_i \) from the following weighted sine signal:

\[ y_i = x_i \sin(x_i). \] (8.1)
Here, the training inputs $x_i$ are drawn uniformly from $(-5, 5)$. The data set is shown in Figure 7a as synthesized data set I. The blue circles represent the training samples, and the red dotted line represents its underlying governing function in the region $(-15, 15)$. Although the training pairs are sampled only from a specific region, testing (for all algorithms) is performed in the entire data region by sampling 120 test data pairs uniformly from the region $(-15, 15)$. The pink bands represent test data regions for which training data have not been provided. We therefore expect high predictive uncertainties in these regions. As part of the analysis, we also add six widely varying outlier samples (not lying on the governing function) to the training set of synthesized data set I as shown in Figure 7b. Synthesized data II consists of noisy regression pairs $x_i, y_i$ sampled from the following signal:

$$y_i = x_i + \sin(\alpha (x_i + w_i)) + \sin(\beta (x_i + w_i)) + w_i.$$  \hfill (8.2)

Here, we set $\alpha = 4, \beta = 13$ and $w_i \sim N(\mu = 0, \sigma^2 = 0.03^2)$. We draw 40 input samples for training uniformly from $(-1, 0.2)$ and 10 from $(0.7, 1)$, leaving the region $(0.2, 0.7)$ as blank. For model testing, we draw 120 test sample pairs uniformly from $(-2, 2)$. The data set is depicted in Figure 7c. In addition to these data sets, we also perform model extrapolation experiments on the Mauna Loa CO$_2$ data set, which consists of atmospheric CO$_2$ concentrations measured from in situ air samples collected at the Mauna Loa Observatory, Hawaii (Keeling & Whorf, 1991).

8.1.2 Model Implementations. For implementing the MC dropout and QIPF framework on synthesized data I, we use a small, fully connected and ReLu-activated neural network with three hidden layers containing 20 neurons each. We train the network on the given training samples in the region $(-5, 5)$. Since the network is very small, the dropout rate for training was set to 0.05 (similar to that recommended in Gal and Ghahramani (2016), for
a similar network size), and we used 100 epochs with the batch size equal to the number of training samples. Thereafter, we tested the network on 120 input points sampled uniformly from the entire data region \((-15, 15)\). For implementing MC dropout, the test dropout rate was set to 0.2, and we used 100 forward stochastic runs to quantify the uncertainty interval at each test point. We implemented the QIPF framework by extracting five cross-QIPF states of the prediction point with respect to the activation outputs of each hidden layer. We used a kernel width based on the values of hidden-layer activation outputs and fixed it to 20 times Silverman’s thumb rule for bandwidth estimation (Silverman, 2018). The criteria for kernel width depend on the range of data space on which the QIPF framework is to be implemented, as well as the desired resolution of modes. Since we are operating on a relatively small neural network over a limited data span, we speculated that five QIPF modes would be enough for our implementation. Thereafter, we quantified the uncertainty interval by measuring the standard deviation of the extracted cross-QIPF states at each point of prediction. For synthesized data II, we used a two-layer network with 50 neurons each for training. The training dropout rate was set to 0.1, and we used 100 training epochs with the batch size equal to the total training data. We implemented MC dropout and QIPF frameworks in the same manner as before on the entire data region \((-2, 2)\) using 120 uniformly generated test samples. In the case of the CO$_2$ data, we used a relatively larger fully connected ReLu network of five layers with 100 neurons each. The training dropout rate in this case was set at 0.2. We test the network over the entire training region as well as an extrapolated region outside it. We also fitted the gaussian process regression model on all data sets. In each case, the parameters associated with the covariance kernel function were chosen using grid search so as to maximize the log marginal likelihood of the data.

8.1.3 Analysis of Results. The results of the different uncertainty quantifiers are shown in Figure 8. The blue line represents the predictions, and the blue shaded regions depict the uncertainty ranges (standard deviation) at the prediction points quantified by the different methods (GPR, MC, dropout and QIPF). The red dotted line indicates the test set prediction errors with respect to the generating function values at those points, and the pink bands represent regions in the input space where no training samples were generated. For synthesized data set I in Figure 8a, it can be observed that the GP regression model is able to better identify the generating signal dynamics than the neural network, producing low predictive errors for some distance outside of the training set. This is expected for small non-linear data sets where kernel methods outperform ANNs. It also produces well-calibrated uncertainty ranges that are seen to be roughly proportional to the predictive errors. The QIPF framework can also be seen to produce uncertainty ranges that scale more proportionally with respect to the test
Figure 8: Comparison of predictive UQ methods. Blue lines are model predictions. Blue shaded areas are their associated uncertainty ranges. Blue dots are training samples. Red dot lines are test prediction errors. Pink bands are untrained regions.
error. MC dropout produces comparatively fewer realistic uncertainties that soon converge to a constant level showing no change with respect to prediction errors thereafter. We observe wide disparities between the uncertainty quantification of the three algorithms in Figure 8b when the outliers are added to the training data (synthesized data set I). We observe here that all models converge to the center of the outlier data when making predictions at those points. This is expected since the mean of such widely varying data points would represent the lowest error region for most learning models. However, we also notice that MC dropout becomes very overconfident with low uncertainties associated with its predictions at the outlier regions, which is opposite of what we would ideally expect. This is also reported in Hernández-Lobato and Adams (2015). The GP regression model also shows unrealistically low uncertainties in its predictions at the particular outlier points, though it still produces increased uncertainty range around them. Also, unlike before, the GP model can be observed to converge to an unrealistic constant level of uncertainty as one goes outside the training region, regardless of the increase in predictive errors. The QIPF framework, on the other hand, shows a remarkable property of increasing its uncertainty range at the outlier regions, which is the ideal behavior. It is also seen to maintain its property of increasing proportionally with the predictive errors in all outside data regions. This indicates the sensitivity of the QIPF framework toward data variances and outliers in the training set. For synthesized data II (which consists of added normal noise), similar observations can be made related to the nature of uncertainties quantified by the different methods outside the training domain. As before, the QIPF framework is seen to produce uncertainty estimates that increase more realistically outside the training domain and more proportional to the predictive errors when compared to MC dropout and GP regression. Both QIPF framework and MC dropout can be seen to be sensitive in their characterizations of uncertainty in the thin, middle, untrained band, given by the region (0.2, 0.7). However, we also observe here that, like before, MC dropout becomes unrealistically overconfident due to large variances in the training data pairs (on the left side of its corresponding graph in Figure 8c). Similar analysis on the CO₂ data (see Figure 8d) reveals GP regression to fit better than the other two models in the training region with very little error. However, it continues to be insensitive to predictive error outside the training domain by exhibiting a constant level of uncertainty when extrapolated during testing. The same trend of GP regression is reported in Gal and Ghahramani (2016). Both MC dropout and QIPF framework extrapolate more realistically in terms of uncertainty. We summarize the observations related to the QIPF framework from these results as follows:

• The framework is observed to be robust toward training set outliers and is able to effectively capture the model’s associated uncertainties with respect to them during testing. This can be attributed to the
ability of the gaussian RKHS in better capturing the true data distributions.
- For all data sets, the QIPF framework is observed to produce uncertainty estimates that are more consistent with predictive errors in all regions of the data domain, consequently exhibiting realistic uncertainty ranges for both model interpolation and extrapolation applications.
- The framework also exhibits increased sensitivity to inherent data variances.

8.2 Classification.

8.2.1 Toy Example: MNIST digit rotation. We also demonstrate the ability of the QIPF framework quantify uncertainties related to classification problems. We start with a toy example of training a network on the MNIST data set (LeCun, Cortes, & Burges, 1998). The network consists of a ReLu-activated and fully connected MLP with three hidden layers with 512, 256, 128 neurons, respectively (from first to last hidden layers). We train the network without the implementation of dropout for 10 epochs using a batch size of 100. During testing, we rotated a single digit of 1 gradually (60 times uniformly) and fed each rotated version to the trained NN. During each test instance, we extracted the first 10 cross-QIPF modes of the average node input value of the last layer (before thresholding) with respect to the activation outputs of hidden layer 1. In this application, we extracted the QIPF modes twice using different kernel widths (20 and 30 times the Silverman bandwidth of the hidden layer 1 activation outputs) and considered the average of the two runs. Figure 9a shows samples of the gradually rotated test sequence, and Figure 9b shows the graph of the standard deviation of the 10 average QIPF modes at each test input. The pink bands represent the rotations at which the network produced incorrect classification results. One can observe the sharp rise of the standard deviation of the uncertainty modes at the misclassified test regions, thus indicating that the framework produces uncertainty results consistent with the model classification errors.

8.2.2 Transfer Learning Using VGG-16. In recent years, there has been a surge of research interest in transfer learning, where large networks, pre-trained on benchmark data sets, are used for efficient feature extraction from other data sets. They present a promising paradigm for increasing the generalizability of deep learning models while requiring very low training to adapt to new data sets. This therefore presents a good venue for the utilization of epistemic model uncertainty quantification methods since we effectively want to explore how well our framework quantifies predictive uncertainty in new data regions outside the model’s training system. To this end, we implement our framework on a VGG-16 CNN architecture (Simonyan & Zisserman, 2014) pretrained on the ImageNet data set (Deng
etal., 2009) and fine-tuned for a classification task on the Kuzushiji-MNIST data set (Clanuwat et al., 2018), which was recently introduced as a more challenging replacement to the MNIST data set and consists of 10 classes of historical cursive Japanese letters (a subset of the full Kuzushiji-49 data set). Some samples are shown in Figure 10.

The K-MNIST data set consists of 10 classes, a training set of 60,000 samples, and a test set of 10,000 samples. We used CNN blocks of VGG-16 network pretrained on the ImageNet data set and froze all of its layers to training except for the last convolution block consisting of three CNN layers. For fine-tuning to K-MNIST, we further connected two dense layers with the first one consisting of 1024 neurons and the second one being the output layer with 10 neurons. Additionally we added a dropout layer in between them with the rate set at 0.5. The overall architecture is depicted in Figure 11. The total number of parameters of the network was 15,250,250 with the total number of trainable parameters being 7,614,986 (for fine-tuning). We fine-tuned the network on the training set of K-MNIST. We used the Adam optimizer and categorical cross-entropy loss function and trained the network for only 12 epochs with the batch size set as 128. We obtained a training accuracy of 99.50% and a test accuracy of 96.96%.

(a) Testing sequence obtained by rotating MNIST digit of one.

(b) Uncertainty results of QIPF framework tested on rotated sequence of a digit.

Figure 9: Behavior of QIPF framework toward classification errors (pink bands).
Figure 10: Samples of the Kuzushiji-MNIST data set. Each row corresponds to a different class, and columns show samples corresponding to each class.

Figure 11: Uncertainty quantification for transfer learning: QIPF framework and MC dropout are implemented on a pretrained and fine-tuned VGG-16 model while testing. The uncertainty estimates are then thresholded to detect classification errors.
Figure 12: Receiver operating characteristics (ROC) for detection of classification errors using QIPF and MC dropout uncertainty estimates. The QIPF framework is seen to have greater sensitivity and specificity, with the area under the curve as 0.89 compared to MC dropout’s 0.79.

We tested the network twice, first using MC dropout and then using the QIPF framework to obtain the predictive uncertainty associated with each test output. For MC dropout implementation, we inserted a dropout layer after each network layer. We used a small dropout rate of 0.1 for all CNN layers and a rate of 0.2 for both dense layers and implemented 500 forward passes at each test iteration. The standard deviation of the 500 network predictions was used as quantified predictive uncertainty value at each test iteration. For QIPF implementation, we used the weights (instead of activation outputs) of the each layer of the network, except the last, as centers for computing the CIP associated with the last layer node output corresponding to the predicted class label (before softmax). We used a kernel width of 1 and slightly downsized the number of centers by using the average pooling of weights at the layers so that the total number of centers were equal to 33,736. Finally, we extracted eight QIPF modes and measured their average to quantify uncertainty. For the uncertainties quantified by each method (MC dropout and QIPF framework), we performed thresholding (depicted in Figure 11) to classify the obtained uncertainty values as either a classification error detection or no error detection. We then measured the true-positive and false-positive rates corresponding to the detections (for different threshold values) by comparing them with the actual classification errors of the network and plotted the receiver operating characteristics for both MC dropout and QIPF (see Figure 12).
Hence we can see from Figure 12 that the QIPF framework significantly outperforms the Monte Carlo dropout in terms of sensitivity and specificity in the detection of classification errors of the K-MNIST data for all threshold values. Moreover, the computation time of the QIPF framework was observed to be considerably faster (0.204 seconds for 33,736 RKHS centers) compared to that of MC dropout (4.755 seconds for 500 runs) per test iteration. This shows that the QIPF framework is able to scale better to larger networks and data sets (in terms of both time complexity and performance) when compared to MC dropout. Furthermore, the results also reveal a potential utility of the QIPF framework in the domain of transfer learning. We intend to explore this application in the future.

8.3 Benchmark Data Sets. We quantify and compare the quality of uncertainty estimates of our method with MC dropout over UCI data sets that are typically used as benchmarking data in various uncertainty quantification literature. We measure the quality of uncertainty estimate by quantifying how calibrated the uncertainty estimates are with respect to prediction errors. We chose data sets with diverse numbers of samples in order to test the framework on different forms of nonlinearities. We train neural networks with 50 neurons in each hidden layer on 20 randomly generated train-test splits of the normalized UCI data sets (similar to the experimental framework of Gal and Ghahramani (2016) and Hernández-Lobato and Adams (2015). A single kernel width of 1 was used in this case for extracting 10 QIPF states. We measured the RMSE of the uncertainty range (standard deviation of the QIPF modes at each test sample) with respect to the test error in each train-test split. This was done to measure how calibrated and scaled the estimated uncertainty range was with respect to the error. The average RMSE and its standard deviation for the 20 test splits are presented in Table 1 for the framework’s implementation on neural network architectures consisting of one, two, and three hidden layers, respectively. It can be observed that the QIPF framework has lower RMSE values than MC dropout for most data sets in all network configurations, thereby indicating that the estimated uncertainty using QIPF is more realistic.

9 Conclusion

In this letter, we introduced a new information-theoretic approach for quantifying uncertainty that is inspired by quantum physical principles and concepts. We formulated a new uncertainty moment decomposition framework for data and models through a systematic and effective quantification gradient flow of data PDF by utilizing the structure and mathematical guarantees provided by the RKHS. The key advantages offered by our framework include its significantly high precision in detecting epistemic uncertainty at all points in the data and model space, the ability to provide a single-shot quantification of uncertainty, and the ability to scale
Table 1: Normalized RMSE between the Standard Deviation of Quantified Uncertainty and the Test Error.

| UCI Data Sets                  | N   | Q | MC Dropout    | QIPF     | Significance |
|--------------------------------|-----|---|---------------|----------|--------------|
| **1-hidden-layer NN**          |     |   |               |          |              |
| Yatch Hydrodynamics            | 308 | 6 | 0.332 ± 0.051 | 0.204 ± 0.050 | 0.128        |
| Boston housing                 | 506 | 13| 0.246 ± 0.038 | 0.234 ± 0.042 | 0.012        |
| Power Plant                    | 9568| 4 | 0.170 ± 0.035 | 0.124 ± 0.035 | 0.046        |
| Concrete Strength              | 1030| 8 | 0.234 ± 0.035 | 0.221 ± 0.044 | 0.013        |
| Energy Efficiency              | 768 | 8 | 0.238 ± 0.028 | 0.268 ± 0.061 | 0.030        |
| **2-hidden layer NN**          |     |   |               |          |              |
| Boston housing                 | 506 | 13| 0.222 ± 0.041 | 0.234 ± 0.049 | 0.012        |
| Power Plant                    | 9568| 4 | 0.151 ± 0.050 | 0.150 ± 0.057 | 0.001        |
| Concrete Strength              | 1030| 8 | 0.218 ± 0.036 | 0.211 ± 0.043 | 0.007        |
| Energy Efficiency              | 768 | 8 | 0.235 ± 0.032 | 0.274 ± 0.046 | 0.039        |
| **3-hidden layer NN**          |     |   |               |          |              |
| Boston housing                 | 506 | 13| 0.223 ± 0.023 | 0.234 ± 0.041 | 0.011        |
| Power Plant                    | 9568| 4 | 0.146 ± 0.038 | 0.144 ± 0.050 | 0.002        |
| Concrete Strength              | 1030| 8 | 0.220 ± 0.030 | 0.204 ± 0.035 | 0.016        |
| Energy Efficiency              | 768 | 8 | 0.263 ± 0.034 | 0.240 ± 0.048 | 0.023        |

Note: N denotes the number of samples, and Q denotes data dimensionality.

better to larger networks and models in terms of performance and computational cost. We gave pedagogical examples to show how our model provides a multiscale characterization of the tails of the data PDF where epistemic uncertainty is maximum. We demonstrated how our framework can be utilized as a powerful tool for the predictive uncertainty quantification of models in both regression and classification tasks involving diverse and differently sized networks and data sets. In the future, we intend to explore the performance of the framework on larger model architectures in the domain of uncertainty quantification and transferability. We also intend to conduct more data-based implementations of the framework in order to explore its utility in the signal processing domain.

Acknowledgments

We acknowledge the support provided for this work by ONR under agreement N00014-18-1-2306 and DARPA under agreement FA9453-18-1-0039.

References

Ahn, K., Choi, M., Dai, B., Sohn, S., & Yang, B. (2018). Modeling stock return distributions with a quantum harmonic oscillator. Europhysics Letters, 120(3), 38003.
Aronszajn, N. (1950). Theory of reproducing kernels. Transactions of the American Mathematical Society, 68(3), 337–404.
Belavkin, V. (1989). A new wave equation for a continuous nondemolition measurement. *Physics Letters A*, 140(7–8), 355–358.

Bergman, S. (1970). *The kernel function and conformal mapping*. Providence, RI: American Mathematical Society.

Berlinet, A., & Thomas-Agnan, C. (2011). *Reproducing kernel Hilbert spaces in probability and statistics*. New York: Springer Science & Business Media.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. New York: Oxford University Press.

Boltzmann, L. (1877). Über die beziehung zwischen dem zweiten hauptsatzen des mechanischen wärmetheorie und der wahrscheinlichkeitsrechnung, respective den sätzen über das wärmegleichgewicht. Kk Hof-und Staatsdruckerei.

Chen, B., Zhao, S., Zhu, P., & Príncipe, J. C. (2011). Quantized kernel least mean square algorithm. *IEEE Transactions on Neural Networks and Learning Systems*, 23(1), 22–32.

Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., & Ha, D. (2018). *Deep learning for classical Japanese literature*. arXiv:1812.01718.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248–255). Piscataway, NJ: IEEE.

Fang, K.-T., Li, R., & Sudjianto, A. (2005). *Design and modeling for computer experiments*. London: Chapman and Hall/CRC.

Frieden, B. R. (2004). *Science from Fisher information: A unification*. Cambridge: Cambridge University Press.

Frieden, B. R., & Hawkins, R. J. (2010). Quantifying system order for full and partial coarse graining. *Physical Review E*, 82(6), 066117.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the International Conference on Machine Learning* (pp. 1050–1059).

Graves, A. (2011). Practical variational inference for neural networks. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems, 24* (pp. 2348–2356). Red Hook, NY: Curran.

Greengard, L., & Strain, J. (1991). The fast gauss transform. *SIAM Journal on Scientific and Statistical Computing*, 12(1), 79–94.

Grenander, U. (1950). Stochastic processes and statistical inference. *Arkiv för matematik*, 1(3), 195–277.

Hahn, G. J., & Shapiro, S. S. (1967). *Statistical models in engineering*. New York: Wiley.

Hermite, M. (1864). *Sur un nouveau développement en série des fonctions*. Imprimerie de Gauthier-Villars.

Hernández-Lobato, J. M., & Adams, R. (2015). Probabilistic backpropagation for scalable learning of Bayesian neural networks. In *Proceedings of the International Conference on Machine Learning* (pp. 1861–1869).
Hoffman, M. D., Blei, D. M., Wang, C., & Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research, 14*(1), 1303–1347.

Hofmann, T., Schölkopf, B., & Smola, A. J. (2008). Kernel methods in machine learning. *Annals of Statistics, 36*, 1171–1220.

Karhunen, K. (1946). Zur spektraltheorie stochastischer processe. *Ann. Acad. Sci. Fennicae, AI*, 34.

Keeling, C., & Whorf, T. (1991). Mauna Loa atmospheric CO₂ modern record. *Trends, 91*, 12–15.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics, 22*(1), 79–86.

Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In I. Guyon, Y. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems, 30* (pp. 6402–6413). Red Hook, NY: Curran.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436.

LeCun, Y., Cortes, C., & Burges, C. J. (1998). The MNIST database of handwritten digits, 1998. http://yann.lecun.com/exdb/mnist.

Liu, W., Pokharel, P. P., & Principe, J. C. (2007). Correntropy: Properties and applications in non-gaussian signal processing. *IEEE Transactions on Signal Processing, 55*(11), 5286–5298. doi:10.1109/TSP.2007.896065

Liu, W., Principe, J. C., & Haykin, S. (2011). *Kernel adaptive filtering: A comprehensive introduction*. Hoboken, NJ: Wiley.

Loève, M. (1946). Fonctions aléatoires à décomposition orthogonale exponentielle. *La Revue Scientifique, 84*, 159–162.

MacKay, D. J. (1992). A practical Bayesian framework for backpropagation networks. *Neural Computation, 4*(3), 448–472.

Meng, X., Zhang, J.-W., & Guo, H. (2016). Quantum Brownian motion model for the stock market. *Physica A: Statistical Mechanics and Its Applications, 452*, 281–288.

Muandet, K., Fukumizu, K., Sriperumbudur, B., & Schölkopf, B. (2017). Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends in Machine Learning, 10*(1–2), 1–141.

Nagel, J. B. (2017). *Bayesian techniques for inverse uncertainty quantification*. PhD diss., ETH Zurich.

Neal, R. M. (2012). *Bayesian learning for neural networks*. New York: Springer Science & Business Media.

Nguyen, A., Yosinski, J., & Clune, J. (2015). Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 427–436). Piscataway, NJ: IEEE.

Osband, I., Blundell, C., Pritzel, A., & Van Roy, B. (2016). Deep exploration via bootstrapped DQN. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems, 29* (pp. 4026–4034). Red Hook, NY: Curran.

Paisley, J., Blei, D., & Jordan, M. (2012). *Variational Bayesian inference with stochastic search*. arXiv:1206.6430.

Parzen, E. (1962). On estimation of a probability density function and mode. *Annals of Mathematical Statistics, 33*(3), 1065–1076.
Parzen, E. (1970). *Statistical inference on time series by RKHS methods* (Technical Report). Department of Statistics, Stanford University.

Pearce, T., Zaki, M., Brintrup, A., & Neel, A. (2018). *Uncertainty in neural networks: Bayesian ensembling*. arXiv:1810.05546.

Pradier, M. F., Pan, W., Yao, J., Ghosh, S., & Doshi-Velez, F. (2018). *Latent projection BNNs: Avoiding weight-space pathologies by learning latent representations of neural network weights*. arXiv:1811.07006.

Principe, J. C. (2010). *Information theoretic learning: Rényi’s entropy and kernel perspectives*. New York: Springer Science & Business Media.

Principe, J. C., Xu, D., Fisher, J., & Haykin, S. (2000). Information theoretic learning. *Unsupervised Adaptive Filtering*, 1, 265–319.

Rao, C. R. (1958). Some statistical methods for comparison of growth curves. *Biometrics*, 14(1), 1–17.

Rényi, A. (1961). On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1: Contributions to the theory of statistics. Berkeley: University of California Press.

Silverman, B. W. (2018). *Density estimation for statistics and data analysis*. London: Routledge.

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv:1409.1556.

Smith, R. C. (2013). *Uncertainty quantification: Theory, implementation, and applications*. Philadelphia: SIAM.

Smola, A. J., & Schölkopf, B. (1998). *Learning with kernels*. Citeseer.

Stigler, S. M. (2005). Ps Laplace, théorie analytique des probabilités, (1812); essai philosophique sur les probabilités (1814). In I. Grattan-Guinness, Roger Cooke, Leo Corry, Pierre Crépel, & Niccolo Guicciardini (Eds.), *Landmark writings in Western mathematics*, 1640–1940 (pp. 329–340). Amsterdam: Elsevier.

Su, J., Vargas, D. V., & Sakurai, K. (2019). One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23, 828–841.

Sullivan, T. J. (2015). *Introduction to uncertainty quantification*. Berlin: Springer.

Theil, H., & Meisner, J. F. (1980). Simultaneous equation estimation based on maximum entropy moments. *Economics Letters*, 5(4), 339–344.

Tibshirani, R. (1996). A comparison of some error estimates for neural network models. *Neural Computation*, 8(1), 152–163.

Van Loan, C. F., & Golub, G. H. (1983). *Matrix computations*. Baltimore: Johns Hopkins University Press.

Vapnik, V. (2013). *The nature of statistical learning theory*. New York: Springer Science & Business Media.

von Neumann, J., Wigner, E. P., & Hofstadter, R. (1955). *Mathematical foundations of quantum mechanics*. Princeton, NJ: Princeton University Press.

Williams, C. K., & Rasmussen, C. E. (2006). *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.

Received February 21, 2020; accepted November 16, 2020.