Journal of Computational Physics 375 (2018) 565–588

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Deep UQ: Learning deep neural network surrogate models for high dimensional uncertainty quantification

Rohit K. Tripathy, Ilias Bilionis *

Predictive Science Laboratory, School of Mechanical Engineering, 585 Purdue Mall, Purdue University, West Lafayette, IN 47907-2088, USA

ARTICLE INFO

Article history:
Received 26 February 2018
Received in revised form 20 August 2018
Accepted 22 August 2018
Available online 27 August 2018

Keywords:
Deep Neural Networks
Dimensionality reduction
Stochastic elliptic PDE
Uncertainty quantification

ABSTRACT

State-of-the-art computer codes for simulating real physical systems are often characterized by vast number of input parameters. Performing uncertainty quantification (UQ) tasks with Monte Carlo (MC) methods is almost always infeasible because of the need to perform hundreds of thousands or even millions of forward model evaluations in order to obtain convergent statistics. One, thus, tries to construct a cheap-to-evaluate surrogate model to replace the forward model solver. For systems with large numbers of input parameters, one has to address the curse of dimensionality through suitable dimensionality reduction techniques. A popular class of dimensionality reduction methods are those that attempt to recover a low-dimensional representation of the high-dimensional feature space. However, such methods often tend to overestimate the intrinsic dimensionality of the input feature space. In this work, we demonstrate the use of deep neural networks (DNN) to construct surrogate models for numerical simulators. We parameterize the structure of the DNN in a manner that lends the DNN surrogate the interpretation of recovering a low-dimensional nonlinear manifold. The model response is a parameterized nonlinear function of the low-dimensional projections of the input. We think of this low-dimensional manifold as a nonlinear generalization of the notion of the *active subspace*. Our approach is demonstrated with a problem on uncertainty propagation in a stochastic elliptic partial differential equation (SPDE) with uncertain diffusion coefficient. We deviate from traditional formulations of the SPDE problem by lifting the assumption of fixed lengthscales of the uncertain diffusion field. Instead we attempt to solve a more challenging problem of learning a map between an arbitrary snapshot of the diffusion field and the response.

1. Introduction

With the tremendous increase in the availability of computational resources, computer codes which simulate physical systems have become highly sophisticated. Today, state-of-the-art numerical simulators are parameterized by a very large number of quantities which are used to describe material properties, initial conditions, boundary conditions, constitutive laws, etc. It is often the case, that many of the inputs to a numerical simulator are not known exactly. This raises the question – how defensible are the predictions from such numerical simulators? How do we objectively assess the effects

* Corresponding author.
E-mail addresses: rtripath@purdue.edu (R.K. Tripathy), ibilion@purdue.edu (I. Bilionis).
URL: https://www.predictivesciencelab.org/ (I. Bilionis).

https://doi.org/10.1016/j.jcp.2018.08.036
0021-9991/

of the uncertainties in model inputs on the quality of the model predictions? Answering such questions are at the core of the field of uncertainty quantification (UQ) [1,2]. Specifically, the task of rigorously quantifying the effect of input parameter uncertainty on the model outputs is known as the forward UQ or uncertainty propagation (UP) problem.

The simplest method for tackling the UP problem is the Monte Carlo (MC) method [3–5]. The basic idea of MC is that one can compute empirical estimates of the statistics of some quantity of interest (QoI) by sampling averages. The MC method is guaranteed to converge in the limit of infinite samples. MC methods, and its advanced variants, are routinely applied to UQ tasks such as UP [6], inverse problems [7,8], model calibration [9] and stochastic optimization [10]. The computational time to convergence of MC methods is independent of the number of the stochastic dimensions. However, the number of samples needed by MC methods, to obtain convergent statistics is large, typically being of the order of hundreds of thousands or millions. This makes MC methods unsuitable for UQ tasks involving expensive computer codes.

We typically deal with expensive computer codes, by building a cheap-to-evaluate surrogate of the response surface. To do this, a set of locations in the uncertain parameter-space are carefully selected and the forward model is evaluated at these locations. This produces a set of independent observations of the model response. The total number of such simulations to be performed is determined by one's computational budget and desired accuracy. Because the surrogate model can be queried very cheaply, one can use it as a replacement of the original simulator and perform UQ tasks using MC techniques. Popular choices for surrogate models in the literature include, Gaussian processes [11–15], polynomial chaos expansions [16–19], radial basis functions [20] and relevance vector machines [21]. Despite their success, these methods become intractable for problems in which the number of input stochastic dimensions is large.

In order to construct a surrogate response surface for a multivariate function with a large number of uncertain parameters, one has to overcome the phenomenon known as the *curse of dimensionality*, a term coined by the mathematician Richard Bellman [22] in the context of dynamic programming. In the context of statistical sampling and machine learning the implication of the curse of dimensionality is that to sufficiently explore a high dimensional space, one must visit an exponentially large number of points in that space. As a concrete example, suppose the task of approximating a surrogate model for a univariate function can be done by visiting 10 locations in the input space and evaluating the forward model at those input locations. For a bivariate function of similar lengthscale, one would need to visit roughly $10 \times 10 = 100$ points in the input space to maintain a similar level of accuracy of the constructed surrogate. Generalizing, a d-variate function requires visiting $\mathcal{O}(10^d)$ locations in the input space and evaluating the forward solver at those locations. Even if the forward model is inexpensive to evaluate, attempting to naively construct surrogate response surfaces for high dimensional functions is a futile task.

Suitable *dimensionality reduction* techniques have to be employed in order to deal with the curse of dimensionality. The simplest way of doing so is to rank the input dimensions in order of their "importance", and rejecting those input dimensions which contribute the least to the outcome of the numerical simulation. Techniques that adopt this approach include sensitivity analysis [23] and automatic relevance determination (ARD) [24]. These methods, while effective for problems with a small number of uncorrelated input dimensions, are not useful for problems involving functional uncertainties, such as stochastic partial differential equations (SPDE).

Many common dimensionality reduction techniques follow a simple idea: project the high dimensional inputs, onto a low-dimensional subspace which captures most of the information contained in the original input. In the UQ community, the most common dimensionality reduction method used is the *Karhunen–Loève expansion* (KLE) [25,26]. The KLE involves computing an eigendecomposition of the covariance function associated with the uncertain functional input. The eigenfunctions represent a set of orthogonal basis functions and the decay of the eigenvalues determine the set of basis functions to be retained for the purpose of constructing a low dimensional approximation of the high dimensional random field. In the machine learning (ML) community, this is more popularly known as the *principal component analysis* (PCA) [27,28], whose goal is to produce a low-rank approximation of the empirical centered covariance matrix of the available input data. The result of such a computation is an orthogonal matrix, which maps a point in the high dimensional input space to a point on a low dimensional manifold, such that there is minimal reconstruction error. The obvious drawback of PCA is the fact that one is constrained to discover only linear projections. Furthermore, PCA is an unsupervised technique, which means that it only looks at samples of the input and does not consider information contained in the model outputs. As a result, PCA tends to overestimate the intrinsic dimensionality of the system. Thus, inspite of a reduction in the total number of input dimensions, the reduced representation remains very high dimensional and thereby unsuitable for surrogate model construction. One can alleviate the linearity constraint by using the kernel PCA (KPCA) [29,30], which uses the kernel trick to discover nonlinear manifolds. Nevertheless, KPCA is also an unsupervised technique that ignores model output information.

A recent advancement in dimensionality reduction is *active subspaces* (AS) [31–36]. An active subspace is a low dimensional linear manifold embedded in the input space which captures most of the model output variation. It does so by recovering an orthogonal projection matrix obtained through an eigendecomposition of an empirical covariance matrix of the model output gradients. In the absence of model output gradients (a scenario typical in engineering applications), one can estimate the projection matrix by posing it as a hyperparameter in a Gaussian process regression model and learning it from the available data through a maximum likelihood (MLE) computation [12]. While the upshot of the AS method is that one utilizes the information contained in the model outputs along with the model inputs, one is still constrained to discover only linear manifolds of the data.

In this work, we propose a systematic approach for constructing surrogate models using *deep neural networks* (DNNs) [37–40]. Neural networks (NNs) (or artificial neural networks (ANNs)) are a class of function approximators that have shot