

## **ABSTRACT**

GUY, HAYLEY CLAIRE. Efficient Dimension Reduction and Uncertainty Quantification for Complex Physical and Biological Systems. (Under the direction of Alen Alexanderian.)

With rapid advances in computing technology, the need for efficient uncertainty quantification (UQ) methods is growing. The goal of UQ methods is to identify, quantify and reduce uncertainties associated with models of real-world systems. For computationally expensive large-scale problems, performing UQ studies can be challenging. Approximating these models via a surrogate model is a common approach to address this challenge. Using surrogate models, one can replace expensive model evaluations by inexpensive surrogate model evaluations. This dissertation provides a number of novel techniques for constructing these surrogate models. To enable efficient surrogate modeling, we consider ideas from active subspaces, reduced order modeling and machine learning. We provide an extension of the active subspace method, for problems with function valued QoIs. We also analyze links between traditional global sensitivity analysis methods and the active subspace method. Theoretical and computational aspects of these methods are considered throughout this dissertation. We demonstrate the effectiveness of the proposed methods with application problems from biology, chemical kinetics and nuclear engineering.

© Copyright 2020 by Hayley Claire Guy

All Rights Reserved

Efficient Dimension Reduction and Uncertainty Quantification for Complex Physical and  
Biological Systems

by  
Hayley Claire Guy

A dissertation submitted to the Graduate Faculty of  
North Carolina State University  
in partial fulfillment of the  
requirements for the Degree of  
Doctor of Philosophy

Applied Mathematics

Raleigh, North Carolina

2020

APPROVED BY:

---

Ralph Smith

---

Pierre Gremaud

---

John Mattingly

---

Alen Alexanderian  
Chair of Advisory Committee

## **DEDICATION**

To Mike, for putting up with me and believing in me even when I was struggling to believe in myself. To my family, for their love and support throughout the years. To my friends, for keeping me grounded, I could not have done it without you all.

## **BIOGRAPHY**

Hayley Claire Guy was born in the suburbs of London, UK where she spent the first 4 years of her life. In 1994 she moved to Co. Meath, Ireland with her family. She completed her Bachelor's degree at Maynooth University, a double major in Applied Mathematics and Experimental Physics in 2011. She then began a 4 year position with KPMG in Dublin, Ireland. Hayley returned to school to pursue her doctorate in Applied Mathematics under the advisement of Dr. Alen Alexanderian in 2015.

## ACKNOWLEDGEMENTS

I would like to thank the following people:

My advisor, Dr. Alen Alexanderian, for his help and guidance over the past five years. I have faced more than a few challenges along the way, but you stuck by me and provided me with the time and encouragement I needed. Thank you for working with me while the outside world interfered. I am truly grateful for your support. My PhD committee, Dr. Ralph Smith, Dr. Pierre Gremaud and Dr. John Mattingly for their feedback and recommendations along the way. My collaborators, Dr. Manav Vohra and Dr. Meilin Yu, who I worked with on ideas, coding, and co-authorship of papers. The numerous teachers who believed in my mathematics abilities since childhood. In particular, my high school math teacher, Ms. Ryan for igniting my passion for math and encouraging me to pursue a degree in mathematics.

My family, for their unconditional love, help and support throughout this journey. My parents, for inspiring me to follow my dreams. None of this would have been possible without the opportunities and experiences you have provided me with over the years. My siblings, my three best friends, for always being there for me whenever I needed someone to laugh with or for someone to listen. My friends, you have all helped in more ways than you could possibly imagine. I am forever grateful for your love and support. My corgi, Cheddars, for his constant company and emotional support as I wrote this dissertation. Finally, my soulmate and partner in life, Mike, who has been a constant source of support and encouragement during the challenges of graduate school and life. I am truly thankful for having you in my life.

# TABLE OF CONTENTS

<b>LIST OF TABLES</b> . . . . .	<b>vii</b>
<b>LIST OF FIGURES</b> . . . . .	<b>viii</b>
<b>Chapter 1 INTRODUCTION</b> . . . . .	<b>1</b>
1.1 Dissertation outline . . . . .	2
1.2 Contributions of this dissertation . . . . .	3
<b>Chapter 2 BACKGROUND</b> . . . . .	<b>5</b>
2.1 Global Sensitivity Analysis (GSA) . . . . .	5
2.1.1 Review of Sobol Indices . . . . .	6
2.1.2 Derivative Based Global Sensitivity Measures (DGSMs) . . . . .	7
2.1.3 Other GSA methods . . . . .	7
2.2 Active Subspaces . . . . .	8
2.3 Machine Learning . . . . .	11
<b>Chapter 3 Surrogate modeling and sensitivity analysis for chemical kinetics ap- plications</b> . . . . .	<b>12</b>
3.1 Introduction . . . . .	12
3.2 GSA measures and their links with active subspaces . . . . .	16
3.3 Computational Method . . . . .	21
3.4 Results and Discussion . . . . .	22
3.5 H <sub>2</sub> /O <sub>2</sub> reaction kinetics: higher-dimensional case . . . . .	26
3.5.1 Computing the active subspace . . . . .	26
3.5.2 Constructing the surrogate model . . . . .	27
3.6 Discussion . . . . .	28
<b>Chapter 4 A distributed active subspace method for scalable surrogate model- ing of function valued outputs</b> . . . . .	<b>31</b>
4.1 Introduction . . . . .	31
4.2 Active subspace-based surrogate models for function-valued QoIs . . . . .	35
4.2.1 Gradient computation . . . . .	39
4.3 Error Analysis . . . . .	40
4.4 Application Problem . . . . .	46
4.4.1 The governing model . . . . .	46
4.4.2 Spectral representation of the model output . . . . .	49
4.5 Results and Discussion . . . . .	49
4.5.1 Active subspace discovery and surrogate model construction . . . . .	49
4.5.2 The accuracy of the surrogate model . . . . .	54
4.6 Conclusions . . . . .	56

<b>Chapter 5</b>	<b>Further insights on active subspaces and activity scores</b>	<b>59</b>
5.1	Activity scores for models with function-valued outputs	59
5.1.1	Functional Activity Scores	60
5.1.2	Results & Discussion	62
5.2	Impact of parameter correlations on active subspaces	65
5.2.1	Quadratic Model	65
5.2.2	Cholera Model	67
<b>Chapter 6</b>	<b>Deep learning for efficient image classification: application to fault detection</b>	<b>73</b>
6.1	Introduction	73
6.2	Neural Networks	74
6.3	Convolutional neural networks	76
6.3.1	Training the CNN	80
6.4	Application Problem: Corrosion detection in spent nuclear fuel canisters	81
6.4.1	Databank Generation	82
6.4.2	Training & Validation	83
6.4.3	Performance Metrics	84
6.4.4	Results	86
6.5	Conclusions	88
<b>Chapter 7</b>	<b>Conclusions</b>	<b>90</b>
<b>APPENDIX</b>		<b>102</b>
Appendix A	Derivation of gradient of the output KL modes	103



## LIST OF TABLES

Table 3.1	Reaction mechanism for $\text{H}_2/\text{O}_2$ from [115]. . . . .	14
Table 5.1	Cholera model parameters from [4, 45]. . . . .	63
Table 5.2	Eigenvalues at different values of $\rho$ . . . . .	72
Table 6.1	Example of entries in a confusion matrix. . . . .	85
Table 6.2	Confusion matrix for tiled images (left) and whole raw images (right)	87
Table 6.3	Performance metrics for optimized network using ResNet-18 architecture. . . . .	87

## LIST OF FIGURES

Figure 3.1	A comparison of the normalized eigenvalue spectrum, $\log(\lambda_i/\lambda_0)$ using $n = \{20, 40, 80, 120\}$ samples with that obtained using a much larger sample size, $N = 1000$ . . . . .	23
Figure 3.2	Left: An illustrative comparison of individual squared components of the converged dominant eigenvector obtained using perturbation and regression strategies using $M = 900$ model runs in each case. Additionally, the dominant eigenvector components obtained using $M = 20000$ model runs (corresponding to $N = 1000$ samples) in the perturbation strategy (test case), used to assess the accuracy of the two strategies are also plotted. Right: An illustrative comparison of the SSPs generated using the perturbation and the regression strategies for computing the active subspace. . . . .	24
Figure 3.3	Left: A bar-graph of normalized activity scores ( $\tilde{\nu}_{i,r}$ 's) for the 19 uncertain pre-exponents ( $A_i$ 's); $r$ denotes the number of eigenvectors in the dominant eigenspace. . . . .	25
Figure 3.4	Left: An illustrative comparison of individual components of the dominant eigenvector in the converged active subspace i.e., at the end of 4 iterations in the perturbation approach and 9 iterations in the regression approach. Right: A comparison of the convergence behavior of the perturbation and the regression approaches. Convergence is accomplished once $\max(\delta \mathbf{W}_{1,j}^{(i)})$ assumes a value smaller than 0.05. . . . .	27
Figure 3.5	A comparison of the PDFs of ignition delay, obtained using model evaluations (solid line) regression-based surrogate (dashed line) and the perturbation-based surrogate (dashed-dotted line). The same set of $10^4$ samples in the cross-validation set were used in each case. . .	28
Figure 4.1	Two realizations of the log-permeability field (left) and the corresponding pressure field (right) using correlation length $\ell = 1 \text{ mm}$ . . .	48
Figure 4.2	Eigenvalue spectrum of $a(\mathbf{x}, \xi)$ versus $p(\mathbf{x}, \xi)$ (top left). Ratio showing saturation of average variance for $p(\mathbf{x}, \xi)$ (top right). First 40 eigenvalues of $C_f$ for different sample sizes $N_s$ (bottom). . . . .	50
Figure 4.3	Top: eigenvalues of the matrix $\hat{\mathbf{S}}_k$ for $k = 1$ (left) $k = 2$ (middle), and $k = 3$ (right); bottom: SSPs for output KL modes $p_1$ (left), $p_2$ (middle), and $p_3$ (right). . . . .	52
Figure 4.4	The normalized eigenvalues of output mode $p_{12}$ (left) and the 2D sufficient summary plot corresponding to $p_{12}$ (right). Note that the first two eigenvalues are very close to each other. . . . .	53

Figure 4.5	Components of the dominant eigenvector $\mathbf{w}$ for the output KL modes, $p_1$ , $p_2$ , and $p_3$ (left, middle, and right images, respectively). The inset plot shows the first 50 components in each case. . . . .	53
Figure 4.6	(Left) standard deviation of $p(\mathbf{x}, \xi)$ using sampling, (middle) standard deviation of $p(\mathbf{x}, \xi)$ using KLE (15 modes), and (right) standard deviation of $p(\mathbf{x}, \xi)$ using KLE (50 modes). . . . .	55
Figure 4.7	Left: The expected value of the error $E_{rel}$ (solid black line) along with the fifth and the ninety fifth percentiles (dashed lines) of $E_{rel}$ . Right: Distribution of $E_{rel}$ computed over the entire domain $\mathcal{D}$ (blue) and over the subdomain $\mathcal{D}'$ (red). . . . .	56
Figure 4.8	Distribution of the error $E_{rel}$ as the number of output KL modes increase from $N = 1$ to $N = 15$ . . . . .	56
Figure 4.9	Distribution of the pressure averaged over concentric circles of various radii. . . . .	57
Figure 5.1	Cholera model: Infected Population . . . . .	63
Figure 5.2	The functional activity scores and functional DGSMs at three different truncation levels, $r = 1$ (top left), $r = 2$ (top right) and $r = 3$ (bottom). . . . .	64
Figure 5.3	Comparison of eigenvalues of $\mathbf{C}$ at different $\rho$ . . . . .	66
Figure 5.4	Angle between $w_{k,0}$ and $w_{k,\rho}$ for various $\rho$ (left) and eigenvalues with eigenvalue derivatives with respect to $\rho$ for various $\rho$ (right). . . . .	68
Figure 5.5	Relative error between the quadratic model and its surrogate approximation at various $\rho$ . . . . .	68
Figure 5.6	Comparison of eigenvalues of $\mathbf{C}$ at different $\rho$ (left) and SSPs for cholera model at various $\rho$ (right). . . . .	69
Figure 5.7	Angle between $w_{1,0}$ and $w_{1,\rho}$ for various $\rho$ (left) and activity scores for cholera model at various values of $\rho$ (right). . . . .	70
Figure 5.8	Relative $L^2$ error at various values of $\rho$ (left) and distribution of surrogate model with actual cholera model (right). . . . .	71
Figure 6.1	schematic representation of an artificial neuron. . . . .	75
Figure 6.2	A simple neural network with one hidden layer. . . . .	75
Figure 6.3	Basic structure of a CNN. . . . .	77
Figure 6.4	An example of convolving a kernel with an input image to produce a feature map. . . . .	77
Figure 6.5	Examples of several activation functions. . . . .	78
Figure 6.6	Example of using max pooling with stride length 1 (move one pixel each time). . . . .	79
Figure 6.7	Lifecycle of nuclear fuel. . . . .	82
Figure 6.8	Nuclear fuel storage site and in-situ inspections. . . . .	83
Figure 6.9	Plot of learning rate vs loss. . . . .	84
Figure 6.10	ROC curve with three examples of models . . . . .	87

Figure 6.11 ROC curve for optimized network . . . . . 88

# CHAPTER

# 1

# INTRODUCTION

Uncertainty quantification (UQ) methods aim to identify, characterize and ultimately reduce uncertainties in models of physical systems. A major challenge facing UQ is the curse of dimensionality. Performing UQ studies on complex physical models may be infeasible due to high-dimensional input or output parameter spaces, and the model may be expensive to evaluate.

Surrogate modeling is used to circumvent these issues by constructing an accurate efficient-to-evaluate approximation to the quantity of interest. Examples of surrogate modeling techniques include polynomial chaos expansions [39, 64], kriging [83, 112], neural networks [48, 103], multivariate adaptive regression splines [34] and active subspaces [18, 23, 86]. A related technique, used to understand and simplify complex computational models is global sensitivity analysis (GSA), which includes variance based [98, 99] and derivative based [58, 101] methods. GSA examines how model outputs are affected by uncertainties in model inputs. These methods are used to identify unimportant model inputs, hence reducing the dimension of the input parameters.

In this dissertation we present a number of surrogate modeling techniques to approximate computationally expensive models. These models may have high-dimensional input or output parameter spaces making uncertainty quantification studies infeasible. In particular we focus on the active subspace based method and machine learning techniques for sensitivity analysis and surrogate modeling of complex physical and biological systems. We will see that there are interesting links between the active subspace method and GSA methods. We apply these methods to application problems from biology, chemical kinetics and nuclear engineering.

## 1.1 Dissertation outline

Chapter 2 reviews relevant background material relating to GSA methods, active subspaces and machine learning. We discuss two common GSA methods in detail, Sobol' indices and derivative based global sensitivity measures (DGSMs) and also briefly reference other methods such as, Morris screening and moment independent importance measures. We introduce the active subspace method for scalar-valued QoIs. Active subspaces can be used to reduce the dimension of the input parameter space by identifying important 'directions' in the space. Active subspaces are related to DGSMs via activity scores. Finally, we briefly introduce machine learning by discussing how these techniques can be used and the differences between supervised and unsupervised learning.

In chapter 3 we apply the active subspace method to an application problem from chemical kinetics [111]. We compute the active subspace using two different approaches, perturbation based where we calculate gradients using finite differences and regression based where we use local linear approximations. We investigate each approach in detail. We also investigate important links between GSA methods and active subspaces, for a general class of input parameter distributions.

Chapter 4 extends the theory of active subspaces, originally defined for scalar-valued QoIs, to function valued QoIs [42]. Theoretical and computational aspects of the method are analyzed. The success of the method is demonstrated in a biotransport application problem.

In chapter 5 we introduce functional activity scores which can be used to screen for unimportant input parameters and can be used to approximate functional DGSMs. We also

investigate the sensitivity of the active subspace method to correlations between input parameters.

In chapter 6 we use convolutional neural networks, a popular tool for image classification, to build a classifier for detecting corrosion from images [79]. We will test the performance of the classifier using a number of metrics including AUC, ROC curve and confusion matrices.

## 1.2 Contributions of this dissertation

The main contributions of this dissertation are as follows:

- We implement the active subspace method to a high-dimensional  $H_2/O_2$  kinetics problem involving 36 uncertain inputs in chapter 3 and to a biotransport problem with high-dimensional inputs and function valued outputs in chapter 4. The results obtained successfully demonstrates that active subspaces can reasonably approximate the uncertainties in the QoIs, indicating immense potential for computational savings.
- We present a mathematical framework and computational method for surrogate modeling for models with high-dimensional inputs and function valued outputs that uses KL expansions for output dimension reduction, active subspaces for input dimension reduction, and adjoint based gradient computation as needed in the active subspace method. Details are in chapter 4.
- We analyze the errors due to active subspace projection and output KL truncation in chapter 4. The presented analysis provides insight on the interplay between these two important sources of errors.
- We provide a complete analysis of the links between total Sobol' indices, DGSMs, and activity scores for a broad class of probability distributions in chapter 3. We also extend the idea of activity scores to function valued QoIs which can be used to approximate functional DGSMs demonstrated through an application from biology in chapter 5.
- We study the impact on the active subspace method to correlations between param-

ters in our input space in chapter 5. We demonstrate this on a simple quadratic model and a cholera model governed by a nonlinear ODE system.

- We develop a machine learning approach, using convolutional neural networks, to efficiently detect and analyze corrosion from images and live video streams of spent nuclear fuel canisters. The classifier achieves accuracies of approximately 96%. Details are given in chapter 6.



## CHAPTER

# 2

## BACKGROUND

In this dissertation we provide different techniques for efficiently approximating computationally expensive models with efficient-to-evaluate surrogate models. This is also closely linked with ideas from sensitivity analysis and dimension reduction methods. In this chapter we provide a brief summary of key tools used in this dissertation. First we discuss two widely used GSA methods, Sobol’ indices in section 2.1.1 and Derivative Based Global Sensitivity Measures (DGSMs) in section 2.1.2. We also briefly mention some other techniques for performing sensitivity analysis and dimension reduction in section 2.1.3. We discuss the active subspace method for scalar valued functions in section 2.2 and finally we introduce machine learning techniques in section 2.3.

### **2.1 Global Sensitivity Analysis (GSA)**

The aim of GSA methods is to quantify how uncertainties in model inputs influence model outputs. [92]. These methods can be used to determine the most important input param-

ters contributing to the output of a mathematical model as well as identifying those input parameters that cause negligible effects on the output. They can also be used to investigate interaction effects between model parameters. In this section we review two commonly used GSA methods, Sobol' indices and derivative-based global sensitivity measures. We will also briefly mention other alternatives.

### 2.1.1 Review of Sobol Indices

Let  $(\Omega, \mathfrak{B}, \mu)$  be a probability space with  $\Omega \subseteq \mathbb{R}^{N_p}$ , the sample space,  $\mathfrak{B}$ , the Borel  $\sigma$ -algebra on  $\Omega$  and  $\mu$ , the law of the uncertain parameter vector  $\xi = (\xi_1, \xi_2, \dots, \xi_{N_p})^T \in \Omega$ . In this section, we assume the components of the random vector  $\xi$  are independent.

Let  $K = \{1, 2, 3, \dots, N_p\}$  be an index set, let  $U = \{j_1, j_2, j_3, \dots, j_m\} \subset K$  and let  $U^c$  be the complement of  $U$  in  $K$ ,  $U^c = K \setminus U$ . The group of variables corresponding to  $U$  is referred to as  $\xi_U = (\xi_{j_1}, \xi_{j_2}, \dots, \xi_{j_m})$ . Consider a scalar-valued random variable  $f : \Omega \rightarrow \mathbb{R}$  where  $f$  is assumed to be square integrable.

Consider the ANOVA decomposition [105] of  $f(\xi)$ :

$$f(\xi) = f_0 + f_1(\xi_U) + f_2(\xi_{U^c}) + f_{12}(\xi), \quad (2.1)$$

where  $f_0$  is the mean of the process,  $f_1(\xi_U) = E\{f(\xi)|\xi_U\} - f_0$ ,  $f_2(\xi_{U^c}) = E\{f(\xi)|\xi_{U^c}\} - f_0$  and  $f_{12}(\xi) = f(\xi) - f_1(\xi_U) - f_2(\xi_{U^c}) - f_0$  is the remainder. This allows us to decompose the total variance of  $f(\xi)$  as follows:

$$\text{Var}(f(\xi)) = \text{Var}_U(f(\xi)) + \text{Var}_{U^c}(f(\xi)) + \text{Var}_{U, U^c}(f(\xi)). \quad (2.2)$$

Using this, we can define the first- and total-order Sobol' indices [97]:

$$S_U = \frac{\text{Var}_U(f(\xi))}{\text{Var}(f(\xi))}, \quad S_U^{\text{tot}} = \frac{\text{Var}_U(f(\xi)) + \text{Var}_{U, U^c}(f(\xi))}{\text{Var}(f(\xi))}. \quad (2.3)$$

The total Sobol' index is used to rank the importance of a parameter. Parameters with small Sobol' indices are deemed unimportant. These indices can be estimated using a Monte Carlo (MC) sampling procedure. In many applications, evaluating  $f$  is computationally intensive

and it may not be feasible to estimate the indices this way. Also, when the dimension of the uncertain parameter vector  $\xi$  is large, computing Sobol' indices' can be very challenging. A cheap-to-evaluate surrogate model  $\hat{f} \approx f$  can be used to accelerate computation of Sobol' indices. Common techniques include Polynomial Chaos expansions [6, 10] and Gaussian processes [74].

### 2.1.2 Derivative Based Global Sensitivity Measures (DGSMs)

An alternative to Sobol' indices is to calculate DGSMs, which can be used to screen for unimportant parameters. DGSMs often obtain sufficient accuracy in identifying unimportant parameters with much less model evaluations than is needed with Sobol' indices. As before, consider a scalar-valued random variable  $f : \Omega \rightarrow \mathbb{R}$  where  $f$  and its partial derivatives are assumed to be square integrable. We consider the following commonly used DGSM [101]:

$$\nu_j(f) = \int_{\Omega} \left( \frac{\partial f}{\partial \xi_j} \right)^2 \mu(d\xi), \quad j = 1, \dots, N_p.$$

Estimation of  $\nu_j$  requires a Monte Carlo (MC) sampling procedure and the ability to estimate derivatives efficiently. As noted above this often requires many less samples than Sobol' indices. The following two theorems given in [101] establish the relation between DGSMs and total Sobol' indices.

**Theorem 2.1.1.** *Assume that  $c \leq \left| \frac{\partial f}{\partial \xi_j} \right| \leq C$ . Then*

$$\frac{c^2}{12 \text{Var}(f(\xi))} \leq S_U^{\text{tot}} \leq \frac{C^2}{12 \text{Var}(f(\xi))}$$

**Theorem 2.1.2.** *Assume that  $\frac{\partial f}{\partial \xi_j} \in L^2$ . Then*

$$S_U^{\text{tot}} \leq \frac{1}{\pi^2 \text{Var}(f(\xi))} \int_{\Omega} \left( \frac{\partial f}{\partial \xi_j} \right)^2 d\xi$$

### 2.1.3 Other GSA methods

Here we briefly introduce some alternatives to Sobol' indices and DGSMs. The Morris screening method [50, 76] can be used to screen for unimportant parameters. It measures

importance by estimating the elementary effects for each input  $\xi_i$ . Using these effects sensitivity measures can be calculated, usually the mean and standard deviation of elementary effect for each input. Morris screening is efficient to compute and is related to DGSMs in that they are capable of ranking the importance of parameters. The difference is that screening methods cannot weigh the importance of one parameter over another [96].

Additionally, using the variance of  $f(\xi)$  in certain applications is not sufficient to screen for unimportant parameters. In [13] the authors discuss moment independent importance measures which look at the effect uncertainties in the inputs has on the entire output distribution rather than to specific moments of the output. As defined in [27], we can measure the influence a given input  $\xi_i$  has on the output  $f(\xi)$  using the following:

$$L_{\xi_i} = \mathbb{E}_{\xi_i}[d(f(\xi), f(\xi)|\xi_i)]$$

where  $d$  denotes some dissimilarity measure between the distribution of  $f(\xi)$  and the distribution of  $f(\xi)|\xi_i$  [9, 80].

Active subspaces provide an alternative to using GSA methods for dimension reduction. A detailed review of this method is given in section 2.2. There exists interesting links between Sobol' indices, DGSMs and active subspaces which we discuss in section 3.2. Activity scores, discussed in Chapter 3 and again in Chapter 5, stem from active subspaces and can be used to measure the importance of input parameters. Active subspaces can also be used to approximate DGSMs which we will investigate in section 5.1.

## 2.2 Active Subspaces

A common approach to input dimension reduction is to identify a subset of input model parameters that are the most important to model variability. This is done typically using a local or global sensitivity analysis approach [51, 58, 82]. The active subspace approach is different; it identifies a set of important directions in the input parameter space rather than giving importance to one input over another. We can rotate the coordinates to align with the directions of strongest variation of the QoI. Each direction can be considered as a set of weights that define a linear combination of all of the inputs. Directions where the

outputs do not vary much are ignored. Consider a model

$$y = g(\xi), \quad \xi \in \mathbb{R}^{N_p}.$$

We assume the uncertain parameters  $\xi$  have an associated probability density function  $\pi(\xi)$  that is supported on  $\Omega \subseteq \mathbb{R}^{N_p}$ . Assume  $g$  is square integrable and has square integrable partial derivatives with respect to  $\xi$ . The following matrix plays a key role in active subspace construction:

$$\mathbf{S} = \int_{\Omega} \nabla g(\xi) \nabla g(\xi)^T \pi(\xi) d\xi. \quad (2.4)$$

The matrix  $\mathbf{S}$  is symmetric and positive semi-definite with spectral decomposition

$$\mathbf{S} = \mathbf{W} \mathbf{\Lambda} \mathbf{W}^T, \quad \mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_{N_p}).$$

The eigenvalues  $\lambda_i$ 's are sorted in descending order  $\lambda_1 \geq \dots \geq \lambda_{N_p} \geq 0$  and  $\mathbf{W}$  contains the orthonormal eigenvectors of  $\mathbf{S}$  as its columns. The active subspace is determined by the dominant eigenvectors of  $\mathbf{S}$ . Specifically, we partition the eigenvalues and eigenvectors according to

$$\mathbf{\Lambda} = \begin{bmatrix} \mathbf{\Lambda}_1 & \\ & \mathbf{\Lambda}_2 \end{bmatrix}, \quad \mathbf{W} = [\mathbf{W}_1 \quad \mathbf{W}_2], \quad (2.5)$$

where  $\mathbf{\Lambda}_1 \in \mathbb{R}^{r \times r}$  is a diagonal matrix with the dominant eigenvalues on its diagonal; and  $\mathbf{W}_1$  contains the corresponding eigenvectors  $\mathbf{w}_1, \dots, \mathbf{w}_r$  as its columns. These columns span the dominant eigenspace of  $\mathbf{S}$ —the *active subspace*. Given  $\xi$  we define  $\mathbf{y} = \mathbf{W}_1^T \xi \in \mathbb{R}^r$  and  $\mathbf{z} = \mathbf{W}_2^T \xi \in \mathbb{R}^{N_p-r}$ , and note that

$$\xi = \mathbf{W}_1 \mathbf{W}_1^T \xi + \mathbf{W}_2 \mathbf{W}_2^T \xi = \mathbf{W}_1 \mathbf{y} + \mathbf{W}_2 \mathbf{z}.$$

The elements of  $\mathbf{y}$  and  $\mathbf{z}$  are the sets of active and inactive variables, respectively. As discussed in detail in [23], the active variables, i.e., elements of  $\mathbf{y}$ , are responsible for most of the variations of the function  $g$ . We can consider approximating  $g$  in the active subspace,  $g(\xi) \approx g(\mathbf{W}_1 \mathbf{W}_1^T \xi)$ . It is convenient to define the function  $G : \mathbb{R}^r \rightarrow \mathbb{R}$  by

$$G(\mathbf{y}) = g(\mathbf{W}_1 \mathbf{y}), \quad \mathbf{y} \in \mathbb{R}^r,$$

and write the approximation to  $g$  as

$$g(\xi) \approx G(\mathbf{W}_1^T \xi). \quad (2.6)$$

In practice, the function  $G$  is typically approximated by a surrogate model. Specifically, we can compute a polynomial regression fit, which we denote by  $\hat{G}(\mathbf{y})$ , for the function  $G(\mathbf{y})$ . Moreover, Monte Carlo sampling is used to approximate the matrix  $\mathbf{S}$  in (2.4):

$$\mathbf{S} \approx \hat{\mathbf{S}} = \frac{1}{N_s} \sum_{i=1}^{N_s} \nabla g(\xi_i) \nabla g(\xi_i)^T. \quad (2.7)$$

Typically, a modest Monte Carlo sample is sufficient for computing reliable approximations to the dominant eigenvalues and eigenvectors of  $\mathbf{S}$ . We provide the steps for computing an active subspace-based surrogate model for the function  $g$  in Algorithm 1. For further details, we refer the readers to [23].

---

**Algorithm 1** Computation of an active subspace-based surrogate model for a scalar-valued function  $g$ .

---

**Input:** A set of  $N_s$  data points  $(\xi_i, g(\xi_i))$ ,  $i = 1, \dots, N_s$ , drawn from the law of  $\xi$ .

**Output:** Surrogate model  $\hat{G}(\mathbf{W}_1^T \xi) \approx g(\xi)$

Compute the gradients  $\mathbf{D}_i = \nabla g(\xi_i)$ ,  $i = 1, \dots, N_s$

Compute

$$\mathbf{S} = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbf{D}_i \mathbf{D}_i^T.$$

Compute spectral decomposition  $\hat{\mathbf{S}} = \mathbf{W} \mathbf{\Lambda} \mathbf{W}^T$ .

Based on decay of the eigenvalues, determine the dimension  $r$  of the active subspace, and partition  $\mathbf{\Lambda}$  and  $\mathbf{W}$  as in (2.5).

Compute  $\mathbf{y}_i = \mathbf{W}_1^T \xi_i$ ,  $i = 1, \dots, N_s$ .

Compute a regression fit  $\hat{G}(\mathbf{y})$  to  $G(\mathbf{y})$  using the data points  $(\mathbf{y}_i, g(\xi_i))$ ,  $i = 1, \dots, N_s$ .

---

## 2.3 Machine Learning

Machine learning methods help us discover patterns in data, build algorithms that can predict outcomes, detect and analyze trends in data and help understand and address complex problems. Machine learning can be divided into two main categories, supervised and unsupervised learning [94].

With supervised learning we have a labeled dataset containing a set of input parameters  $x$  and a corresponding output set  $y$ . We want to build a machine learning algorithm to learn the mapping  $f : x \rightarrow y$  from set of inputs  $x$  to the set of outputs  $y$ . The goal is to approximate the function  $f$  well to make predictions accurately on new input data. Supervised learning uses both classification and regression techniques to build predictive models. Algorithms used include linear regression, logistic regression, k-nearest neighbor and neural networks.

With unsupervised learning we have a set of input parameters  $x$  but no labeled output data. The goal is to learn more about the data by modeling the underlying structure of the data. Unsupervised learning can be split into clustering and association type problems.

Machine learning models can be used to approximate complicated input-output relations and thus can be viewed as a surrogate modeling approach. In chapter 6, we discuss using convolutional neural networks, a variant on neural networks, in an application problem from nuclear engineering where we seek to classify defects in nuclear fuel canisters. We provide more background details on deep learning in chapter 6.

## CHAPTER

# 3

# SURROGATE MODELING AND SENSITIVITY ANALYSIS FOR CHEMICAL KINETICS APPLICATIONS

## 3.1 Introduction

Chemical kinetics systems can be complex due to a large number of reactions and a high-dimensional set of input parameters that control the rate of these chemical reactions. In practice there are large uncertainties associated with these rate-controlling parameters. Understanding the impact these parameters have on the model output can be challenging. Several research efforts in recent years address the quantification and propagation of uncertainty in such systems [43, 77, 78, 84, 93, 109, 110]. In this chapter we implement the active subspace method, discussed in section 2.2, to a high-dimensional chemical kinetics system. We recall that the active subspace method identifies *important* directions



in the input parameter space that predominantly capture the variability in the model output. These directions are the dominant eigenvectors of the matrix  $\mathbf{S}$  in (2.4). The active subspace method focuses on reducing the dimensionality of the problem, and hence the computational effort associated with uncertainty propagation. The focus here is on input parameter dimension reduction. This is different from techniques such as Computational Singular Perturbation (CSP) [30, 59, 60, 89, 107] that aim to reduce the complexity of stiff chemical systems by filtering out the fast timescales from the system. The latter is done, for instance, using the eigenvectors of the system Jacobian to decouple the fast and slow processes; see e.g., [30].

The application problem considered here is the  $\text{H}_2/\text{O}_2$  reaction mechanism from [115]. This mechanism has received a lot of attention as a potential source of clean energy for locomotive applications [28], and more recently in fuel cells [26, 70]. The mechanism involves 19 reactions including chain reactions, dissociation/recombination reactions, and formation and consumption of intermediate species; see Table 3.1. For each reaction, the reaction rate is assumed to follow an Arrhenius correlation with temperature:

$$k_i(T) = A_i T^{n_i} \exp(-E_{a,i}/RT), \quad (3.1)$$

where  $A_i$  is the pre-exponent,  $n_i$  is the temperature exponent,  $E_{a,i}$  is the activation energy corresponding to the  $i^{th}$  reaction, and  $R$  is the universal gas constant. The Arrhenius rate law in (3.1) is typically interpreted in a logarithmic form as follows:

$$\log(k_i) = \log(A_i) + n_i \log(T) - E_{a,i}/RT. \quad (3.2)$$

The global reaction associated with the  $\text{H}_2/\text{O}_2$  mechanism can be considered as follows:



The equivalence ratio ( $\Phi$ ) is given by [111]:

$$\Phi = \frac{(M_{\text{H}_2}/M_{\text{O}_2})_{\text{obs}}}{(M_{\text{H}_2}/M_{\text{O}_2})_{\text{st}}}, \quad (3.4)$$

where the numerator on the right-hand-side denotes the ratio of the fuel ( $\text{H}_2$ ) and oxidizer ( $\text{O}_2$ ) at a given condition to the same quantity under stoichiometric conditions. In this study,

Table 3.1: Reaction mechanism for H<sub>2</sub>/O<sub>2</sub> from [115].

Reaction #	Reaction
$\mathcal{R}_1$	$\text{H} + \text{O}_2 \rightleftharpoons \text{O} + \text{OH}$
$\mathcal{R}_2$	$\text{O} + \text{H}_2 \rightleftharpoons \text{H} + \text{OH}$
$\mathcal{R}_3$	$\text{H}_2 + \text{OH} \rightleftharpoons \text{H}_2\text{O} + \text{H}$
$\mathcal{R}_4$	$\text{OH} + \text{OH} \rightleftharpoons \text{O} + \text{H}_2\text{O}$
$\mathcal{R}_5$	$\text{H}_2 + \text{M} \rightleftharpoons \text{H} + \text{H} + \text{M}$
$\mathcal{R}_6$	$\text{O} + \text{O} + \text{M} \rightleftharpoons \text{O}_2 + \text{M}$
$\mathcal{R}_7$	$\text{O} + \text{H} + \text{M} \rightleftharpoons \text{OH} + \text{M}$
$\mathcal{R}_8$	$\text{H} + \text{OH} + \text{M} \rightleftharpoons \text{H}_2\text{O} + \text{M}$
$\mathcal{R}_9$	$\text{H} + \text{O}_2 + \text{M} \rightleftharpoons \text{HO}_2 + \text{M}$
$\mathcal{R}_{10}$	$\text{HO}_2 + \text{H} \rightleftharpoons \text{H}_2 + \text{O}_2$
$\mathcal{R}_{11}$	$\text{HO}_2 + \text{H} \rightleftharpoons \text{OH} + \text{OH}$
$\mathcal{R}_{12}$	$\text{HO}_2 + \text{O} \rightleftharpoons \text{O}_2 + \text{OH}$
$\mathcal{R}_{13}$	$\text{HO}_2 + \text{OH} \rightleftharpoons \text{H}_2\text{O} + \text{O}_2$
$\mathcal{R}_{14}$	$\text{HO}_2 + \text{HO}_2 \rightleftharpoons \text{H}_2\text{O}_2 + \text{O}_2$
$\mathcal{R}_{15}$	$\text{H}_2\text{O}_2 + \text{M} \rightleftharpoons \text{OH} + \text{OH} + \text{M}$
$\mathcal{R}_{16}$	$\text{H}_2\text{O}_2 + \text{H} \rightleftharpoons \text{H}_2\text{O} + \text{OH}$
$\mathcal{R}_{17}$	$\text{H}_2\text{O}_2 + \text{H} \rightleftharpoons \text{HO}_2 + \text{H}_2$
$\mathcal{R}_{18}$	$\text{H}_2\text{O}_2 + \text{O} \rightleftharpoons \text{OH} + \text{HO}_2$
$\mathcal{R}_{19}$	$\text{H}_2\text{O}_2 + \text{OH} \rightleftharpoons \text{HO}_2 + \text{H}_2\text{O}$

computations were performed at fuel-rich conditions,  $\Phi = 2.0$ . Homogeneous ignition at constant pressure is simulated using the TChem software package [88] using an initial pressure,  $P_0 = 1$  atm and initial temperature,  $T_0 = 900$  K. The time required for the rate of temperature increase to exceed a given threshold, regarded as *ignition delay* is recorded.

We seek to understand the impact of uncertainty in the rate-controlling parameters, pre-exponents ( $A_i$ 's) and the activation energies ( $E_{a,i}$ 's) as well as the initial pressure, temperature, and the equivalence ratio on the quantity of interest (QoI), the ignition delay. The  $\log(A_i)$ 's associated with all reactions and the  $E_{a,i}$ 's with non-zero nominal estimates are considered to be uniformly distributed about their nominal estimates provided in [115]. Temperature exponent,  $n_i$  for each reaction is fixed to its nominal value, also provided in [115]. The initial conditions are also considered to be uniformly distributed about their respective nominal values mentioned above. The total number of uncertain inputs is 36, making the present problem computationally challenging. To address this challenge, we focus on reducing the dimensionality of the problem by computing the active subspace. This involves repeated evaluations of the gradient of a model output with respect to the input parameters. Several numerical techniques are available for computing the gradient,

such as finite differences and more advanced methods involving adjoints and sensitivity equations [2, 14, 54]. In this chapter we will compare two different approaches for active subspace discovery. The first approach, which we call the *perturbation* approach, uses finite differences to approximate the gradient of the QoI with respect to the uncertain rate-controlling parameters. The second approach [20], referred to as the *regression* approach here, uses local linear approximations to estimate the active subspace. The regression based approach is computationally less intensive than the perturbation approach. Note the regression based approach is used only in active subspace construction.

Global sensitivity measures provide an alternative approach to input dimension reduction. Sobol' indices [99] rank input parameters based on their contribution to the variance in the output. These indices can be estimated using a Monte Carlo sampling procedure. In many applications, evaluating  $f$  is computationally intensive and estimating the indices this way may not be feasible. Also, when the dimension of the uncertain parameter vector is large, computing Sobol indices' can be very challenging. Multiple efforts have focused on efficient computation of the Sobol' indices [67, 81, 104, 106]. Also derivative based global sensitivity measures (DGSMs) can be used to accurately screen for unimportant parameters often with less model evaluations than is needed when estimating Sobol' indices. DGSMs can also provide approximate upper bounds for Sobol' indices. It was noted in [24, 31] that DGSMs can be approximated by exploiting their links with active subspaces. This led to the definition of the *activity scores*. In Section 3.2, we build on these ideas to provide a complete analysis of links between Sobol indices, DGSMs, and activity scores for functions of independent random inputs whose distribution law belongs to a broad class of probability measures. Computing global sensitivity measures provides important information about a model that go beyond dimension reduction. By identifying parameters with significant impact on the model output, we can assess regimes of validity of the model formulation, and gain critical insight into the underlying physics in many cases.

This chapter is organized as follows. In section 3.2, we study activity scores and their use for approximating DGSMs. We also analyze important links between activity scores and GSA methods, namely DGSMs and the total Sobol' indices. In section 3.3, we outline our approach for active subspace discovery and apply it to a 19-dimensional  $\text{H}_2/\text{O}_2$  reaction kinetics problem. The pre-exponents  $A_i$ 's are considered as the uncertain parameters here. In section 3.5, we apply the approach to a higher dimensional  $\text{H}_2/\text{O}_2$  reaction kinetics

problem where 36 of the input parameters are considered to be uncertain. In both problems we compare the results of the perturbation method (gradient-based) with the regression method (gradient-free approach). In section 3.6, we provide a summary and discussion of the results.

The details of this chapter are published in [111], which is the result of collaborative work with Dr. Manav Vohra and Dr. Sankaran Mahadevan of Vanderbilt University.

## 3.2 GSA measures and their links with active subspaces

Consider a function  $f = f(\xi_1, \xi_2, \dots, \xi_{N_p})$ . While the active subspace framework described above does not make any assumptions about independence of the inputs  $\xi_i$ ,  $i = 1, \dots, N_p$ , the classical framework of variance based sensitivity analysis [91, 99] assumes that the inputs are statistically independent. While extensions to the cases of correlated inputs exist [12, 52, 68, 114], we limit the discussion in this section to the case of random inputs that are statistically independent and are either uniformly distributed or distributed according to the Boltzmann probability distribution. Note that a measure  $\mu$  on  $\mathbb{R}$  is referred to as a Boltzmann measure if it is absolutely continuous with respect to the Lebesgue measure and admits a density of the form  $\pi(x) = C \exp\{-V(x)\}$ , where  $V$  is a continuous function and  $C$  a normalization constant [61]. An important class of Boltzmann distributions are the so called log-concave distributions, which include Normal, Exponential, Beta, Gamma, Gumbel, and Weibull distributions. Note also that the uniform distribution does not fall under the class of Boltzmann distributions [61].

The total-effect Sobol' index ( $T_i(f)$ ) of a model output,  $f(\xi)$  quantifies the total contribution of the input,  $\xi_i$  to the variance of the output [99]. Mathematically, this can be expressed as follows:

$$T_i(f) = 1 - \frac{\mathbb{V}_{\xi_{\sim i}}[\mathbb{E}[f|\xi_{\sim i}]]}{\mathbb{V}(f)}, \quad (3.5)$$

where  $\xi_{\sim i}$  is the input parameter vector with the  $i^{\text{th}}$  entry removed. Here  $\mathbb{E}[f|\xi_{\sim i}]$  denotes the conditional expectation of  $f$  given  $\xi_{\sim i}$  and its variance is computed with respect to  $\xi_{\sim i}$ . The quantity,  $\mathbb{V}(f)$  denotes the total variance of the model output. The total-effect Sobol' index accounts for the contribution of a given input to the variability in the output by itself as

well as due to its interaction or coupling with other inputs. Determining accurate estimates of  $T_i(f)$  typically involves a large number of model runs and is therefore can be prohibitive in the case of compute-intensive applications. Derivative based global sensitivity measures (DGSMs) [100] provide a means for approximating informative upper bounds on  $T_i(f)$  at a lower cost; see also [108].

For  $f : \Omega \rightarrow \mathbb{R}$ , we consider the DGSMs,

$$\nu_i(f) := \mathbb{E} \left\{ \left( \frac{\partial f}{\partial \xi_i} \right)^2 \right\} = \int_{\Omega} \left( \frac{\partial f}{\partial \xi_i} \right)^2 \pi(\xi) d\xi, \quad i = 1, \dots, N_p.$$

Here  $\pi$  is the joint PDF of  $\xi$ . Note that  $\nu_i(f)$  is the  $i^{\text{th}}$  diagonal element of the matrix  $\mathbf{C}$  as defined in (2.4). Consider the spectral decomposition written as  $\mathbf{C} = \sum_{k=1}^{N_p} \lambda_k \mathbf{w}_k \mathbf{w}_k^{\top}$ . Herein, we use the notation  $\langle \cdot, \cdot \rangle$  for the Euclidean inner product. The following result provides a representation of DGSMs in terms of the spectral representation of  $\mathbf{C}$ :

**Lemma 3.2.1.** *We have  $\nu_i(f) = \sum_{k=1}^{N_p} \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2$ .*

*Proof.* Note that  $\nu_i(f) = \mathbf{e}_i^{\top} \mathbf{C} \mathbf{e}_i$ , where  $\mathbf{e}_i$  is the  $i$ th coordinate vector in  $\mathbb{R}^{N_p}$ ,  $i = 1, \dots, N_p$ . Therefore,  $\nu_i(f) = \mathbf{e}_i^{\top} \left( \sum_{k=1}^{N_p} \lambda_k \mathbf{w}_k \mathbf{w}_k^{\top} \right) \mathbf{e}_i = \sum_{k=1}^{N_p} \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2$ .  $\square$

In the case where the eigenvalues decay rapidly to zero, we can obtain accurate approximations of  $\nu_i(f)$  by truncating the summation:

$$\nu_{i,r}(f) = \sum_{k=1}^r \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2, \quad i = 1, \dots, N_p, \quad r \leq N_p.$$

The quantities  $\nu_{i,r}(f)$  are called activity scores in [24, 31], where links between GSA measures and active subspaces is explored. The following result, which can also be found in [24, 31], quantifies the error in this approximation. We provide a short proof for completeness.

**Proposition 3.2.1.** *For  $1 \leq r < N_p$ ,*

$$0 \leq \nu_i(f) - \nu_{i,r}(f) \leq \lambda_{r+1}, \quad i = 1, \dots, N_p.$$

*Proof.* Note that,  $\nu_i(f) - \nu_{i,r}(f) = \sum_{k=r+1}^{N_p} \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2 \geq 0$ , which gives the first inequality.

To see the upper bound, we note,

$$\sum_{k=r+1}^{N_p} \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2 \leq \lambda_{r+1} \sum_{k=r+1}^{N_p} \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2 \leq \lambda_{r+1}.$$

The last inequality holds because  $1 = \|\mathbf{e}_i\|_2^2 = \sum_{k=1}^{N_p} \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2 \geq \sum_{k=r+1}^{N_p} \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2$ .  $\square$

The utility of this result is realized in problems with high-dimensional parameters in which the eigenvalues  $\lambda_i, i = 1, \dots, N_p$ , decay rapidly to zero; in such cases, this result implies that  $\nu_i(f) \approx \nu_{i,r}(f)$ , where  $r$  is the *numerical rank* of  $\mathbf{C}$ . This will be especially effective if there is a large gap in the eigenvalues.

The relations recorded in the following lemma will be useful in the discussion that follows.

**Lemma 3.2.2.** *We have*

1.  $\sum_{i=1}^{N_p} \nu_{i,r}(f) = \sum_{k=1}^r \lambda_k.$
2.  $\sum_{i=1}^{N_p} \nu_i(f) = \sum_{k=1}^{N_p} \lambda_k.$

*Proof.* The first statement of the lemma holds, because

$$\sum_{i=1}^{N_p} \nu_{i,r}(f) = \sum_{i=1}^{N_p} \sum_{k=1}^r \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2 = \sum_{k=1}^r \lambda_k \sum_{i=1}^{N_p} \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2 = \sum_{k=1}^r \lambda_k \|\mathbf{w}_k\|^2 = \sum_{k=1}^r \lambda_k.$$

The statement (b) follows immediately from (a), because  $\nu_i(f) = \nu_{i,N_p}(f)$ .  $\square$

It was shown in [61] that the total-effect Sobol' index  $T_i(f)$  can be bounded in terms of  $\nu_i(f)$ :

$$T_i(f) \leq \frac{C_i}{\mathbb{V}(f)} \nu_i(f), \quad i = 1, \dots, N_p, \quad (3.6)$$

where for each  $i$ ,  $C_i$  is an appropriate *Poincaré* constant that depends on the distribution of  $\xi_i$ . For instance, if  $\xi_i$  is uniformly distributed on  $[-1, 1]$ , then  $C_i = 4/\pi^2$ ; and in the case  $\xi_i$  is normally distributed with variance  $\sigma_i^2$ , then  $C_i = \sigma_i^2$ . Note that (3.6) for the special cases of uniformly distributed or normally distributed inputs was established first in [100]. The bound (3.6) provides a strong theoretical basis for using DGSMs to identify unimportant inputs.

Combining Proposition 5.1.1 and (3.6), shows an interesting link between the activity scores and total-effect Sobol' indices. Specifically, by computing the activity scores, we can identify the unimportant inputs. Subsequently, one can attempt to reduce parameter dimension by fixing unimportant inputs at nominal values.

Suppose activity scores are used to approximate DGSMs, and suppose  $\xi_i$  is deemed unimportant as a result, due to a small activity score. We want to estimate the approximation error that occurs once  $\xi_i$  is fixed at a nominal value. To formalize this process, we proceed as follows. Let  $\xi$  be given and let  $z$  be a nominal value for  $\xi_i$ . Consider the *reduced* model, obtained by fixing  $\xi_i$  at the nominal value:

$$f^{(i)}(\xi; z) = f(\xi_1, \xi_2, \dots, \xi_{i-1}, z, \xi_{i+1}, \dots, \xi_{N_p}),$$

and consider the following relative error indicator:

$$\mathcal{E}(z) = \frac{\int_{\Omega} (f(\xi) - f^{(i)}(\xi; z))^2 \mu(d\xi)}{\int_{\Omega} f(\xi)^2 \mu(d\xi)}.$$

This error indicator is a function of  $z$  with  $z$  distributed according to the distribution of  $\xi_i$ .

**Theorem 3.2.1.** *We have  $E_z \{\mathcal{E}(z)\} \leq 2C_i (\nu_{i,r}(f) + \lambda_{r+1}) / \mathbb{V}(f)$ , for  $1 \leq r < N_p$ .*

*Proof.* Note that, since  $\int_{\Omega} f(\xi)^2 \mu(d\xi) = \mathbb{V}(f) + (\int_{\Omega} f(\xi) \mu(d\xi))^2 \geq \mathbb{V}(f)$ , we have

$$E_z \{\mathcal{E}(z)\} \leq \frac{1}{\mathbb{V}(f)} E_z \left\{ \int_{\Omega} (f(\xi) - f^{(i)}(\xi; z))^2 \mu(d\xi) \right\} = 2T_i(f),$$

where the equality can be shown using arguments similar to the proof of the main result in [102]. Using this, along with (3.6) and Proposition 5.1.1, we have

$$E_z \{\mathcal{E}(z)\} \leq \frac{2C_i}{\mathbb{V}(f)} \nu_i(f) \leq \frac{2C_i}{\mathbb{V}(f)} [\nu_{i,r}(f) + \lambda_{r+1}].$$

□

In [108] the screening metric

$$\tilde{\nu}_i(f) = \frac{C_i \nu_i(f)}{\sum_{i=1}^{N_p} C_i \nu_i(f)}, \quad (3.7)$$

was shown to be useful for detecting unimportant inputs. We can also bound the normalized DGSMs using activity scores as follows. It is straightforward to see that

$$\tilde{v}_i(f) \leq \frac{C_i(v_{i,r}(f) + \lambda_{r+1})}{\sum_{i=1}^{N_p} C_i v_{i,r}(f)} = \frac{C_i v_{i,r}(f)}{\sum_{i=1}^{N_p} C_i v_{i,r}(f)} + \kappa_i \lambda_{r+1},$$

with  $\kappa_i = C_i / (\sum_i C_i v_{i,r}(f))$ . In the case where  $\lambda_{r+1} \approx 0$ , this motivates definition of normalized activity scores

$$\tilde{v}_{i,r}(f) = \frac{C_i v_{i,r}(f)}{\sum_{i=1}^{N_p} C_i v_{i,r}(f)}.$$

**Remark 3.2.1.** *If the random inputs  $\xi_i$ ,  $i = 1, \dots, N_p$ , are iid, then the  $C_i$ 's in the definition of the normalized screening metric will cancel and*

$$\tilde{v}_i(f) = \frac{v_i(f)}{\sum_{i=1}^{N_p} v_i(f)} = \frac{\sum_{k=1}^{N_p} \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2}{\sum_{k=1}^{N_p} \lambda_k}.$$

*The expression for the denominator follows from Lemma 3.2.2(b). Also, in the iid case, using Lemma 3.2.2(a) we can simplify the normalized activity scores as follows.*

$$\tilde{v}_{i,r}(f) = \frac{v_{i,r}(f)}{\sum_{i=1}^{N_p} v_{i,r}(f)} = \frac{\sum_{k=1}^r \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2}{\sum_{k=1}^r \lambda_k}. \quad (3.8)$$

The significance of the developments in this section are as follows. Theorem 3.2.1 provides a theoretical basis for parameter dimension reduction using activity scores. This is done by providing an estimate of the error between the reduced model and the original model. If a precise ranking of parameter importance based on total-effect Sobol' indices is desired, one can first identify unimportant inputs by computing activity scores, and then perform a detailed variance based GSA of the remaining model parameters. This approach will provide great computational savings as variance based GSA will now be performed only for a small number of inputs deemed important based on their activity scores. Moreover, the presented result covers a broad class of input distributions coming from the Boltzmann family of distributions. Additionally, the normalized activity scores discussed above provide practical screening metrics that require only computing the activity scores. This is in contrast to the bound in Theorem 3.2.1 that requires the variance  $\mathbb{V}(f)$  of the model output.



### 3.3 Computational Method

In this section, we outline the methodology for computing the active subspace in an efficient manner. The proposed framework is employed to analyze a 19-dimensional  $\text{H}_2/\text{O}_2$  reaction kinetics problem whereby the logarithm of the pre-exponent ( $A_i$ ) in the rate law associated with individual reactions provided in Table 3.1 is considered to be uniformly distributed in the interval,  $[0.97 \log(A_i^*), 1.03 \log(A_i^*)]$ ;  $A_i^*$  is the nominal estimate provided in [115]. Two approaches are explored for estimating the gradient of ignition delay with respect to  $\log(A_i)$ : a perturbation approach that involves computation of model gradients using finite difference in order to construct the matrix  $\hat{\mathbf{S}}$  in (2.7), and a regression approach that involves a linear regression fit to the available set of model evaluations. The active subspace is computed in an iterative manner to avoid unnecessary model evaluations.

As discussed earlier, gradient estimation using finite differences requires additional model evaluations at the neighboring points in the input domain. Hence, for  $N$  samples in a  $d$ -dimensional parameter space,  $N(d + 1)$  model evaluations are needed. On the other hand, the regression-based approach involves a series of linear regression fits to subsets of available evaluations as discussed in [20]. Hence, the computational effort is reduced by a factor  $(d + 1)$  when using the regression-based approach. In other words, for the same amount of computational effort, the regression approach can afford a sample size that is  $(d+1)$  times larger than that in the case of perturbation approach. The specific sequence of steps for computing the active subspace is discussed as follows.

We begin by evaluating the gradient of the model output,  $\nabla_{\xi} f$ , at an initial set of  $n_0$  samples (generated using Monte Carlo sampling) denoted by  $\xi_i$ ,  $i = 1, \dots, n_0$ . Using the gradient evaluations, the matrix,

$$\hat{\mathbf{S}} = \frac{1}{N_s} \sum_{i=1}^{N_s} \nabla f(\xi_i) \nabla f(\xi_i)^T$$

is computed. Eigenvalue decomposition of  $\hat{\mathbf{S}}$  yields an initial estimate of the dominant eigenspace,  $\mathbf{W}_1$  and the set of corresponding eigenvalues,  $\Lambda_1$ . Note that  $\mathbf{W}_1$  is obtained by partitioning the eigenspace around  $\lambda_j$  such that the ratio of subsequent eigenvalues,  $\left(\frac{\lambda_j}{\lambda_{j+1}}\right) \geq \mathcal{O}(10^1)$ . At each subsequent iteration, model evaluations are generated at a new set of  $n_k$  samples. The new set of gradient evaluations are augmented with the available set to re-construct  $\hat{\mathbf{S}}$  followed by its eigenvalue decomposition. The relative change in the norm

of the difference in squared value of individual components of the dominant eigenvectors between subsequent iterations is evaluated. The process is terminated once the maximum relative change at iteration  $k$ ,  $\max(\delta \mathbf{W}_{1,j}^{(k)})$  ( $j$  is used as an index for the eigenvectors), is smaller than a given tolerance,  $\tau$ , where

$$\delta \mathbf{W}_{1,j}^{(k)} = \frac{\|(\mathbf{W}_{1,j}^{(k)})^2 - (\mathbf{W}_{1,j}^{(k-1)})^2\|_2}{\|(\mathbf{W}_{1,j}^{(k-1)})^2\|}. \quad (3.9)$$

A regression fit to  $G(\mathbf{W}_1^\top \boldsymbol{\xi})$  with  $G$  as in (2.6) is used as a surrogate to characterize and quantify the uncertainty in the model output. Moreover, the components of the eigenvectors in the active subspace are used to compute the activity scores,  $\nu_r(f)$ , which provide an insight into the relative importance of the uncertain inputs. Note that the index,  $r$ , corresponds to the number of eigenvectors in  $\mathbf{W}_1$ . The sequence of steps as discussed are outlined in [111, Algorithm 1].

### 3.4 Results and Discussion

To assess its feasibility and suitability, we implement the proposed strategy to compute the active subspace for the 19-dimensional  $\text{H}_2/\text{O}_2$  reaction kinetics problem by perturbing  $\log(A_i)$  by 3% about its nominal value as discussed earlier. For the purpose of verification,  $\hat{\mathbf{S}}$  was initially constructed using a large set of samples ( $N = 1000$ ) in the input domain. The gradient was estimated using finite difference, and hence, a total of 20,000 model runs were performed. In Figure 3.1, we illustrate the comparison of the resulting normalized eigenvalue spectrum by plotting  $\log(\lambda_i/\lambda_0)$  ( $i = 1, \dots, 19$ ) corresponding to  $N = 1000$  and the same quantity corresponding to a much smaller set of samples,  $n = \{20, 40, 80, 120\}$ . We observe that the dominant eigenvalues,  $\lambda_1, \dots, \lambda_4$ , are approximated reasonably well with just 20 samples. As expected, the accuracy of higher-index eigenvalues is observed to improve with the sample size. Since  $\lambda_1$  is roughly an order of magnitude larger than  $\lambda_2$ , we expect a 1-dimensional active subspace to reasonably approximate the uncertainty in the ignition delay. To further confirm this, we evaluate a relative  $L^2$  norm of the difference ( $\varepsilon_{L^2}^{N-n}$ ) between the squared value of corresponding components of the dominant eigenvector,

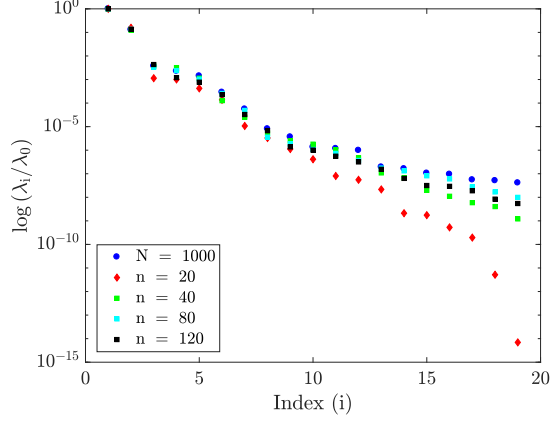


Figure 3.1: A comparison of the normalized eigenvalue spectrum,  $\log(\lambda_i/\lambda_0)$  using  $n = \{20, 40, 80, 120\}$  samples with that obtained using a much larger sample size,  $N = 1000$ .

computed using  $N = 1000$  ( $\mathbf{w}_{1,N}$ ) and  $n = \{20, 40, 80, 120\}$  ( $\mathbf{w}_{1,n}$ ) as follows:

$$\epsilon_{L^2}^{N-n} = \frac{\|\mathbf{w}_{1,N}^2 - \mathbf{w}_{1,n}^2\|_2}{\|\mathbf{w}_{1,N}^2\|_2} \quad (3.10)$$

The quantity,  $\epsilon_{L^2}^{N-n}$ , was found to be  $\mathcal{O}(10^{-2})$  in all cases. Thus, even a small sample size,  $n = 20$ , seems to approximate the dominant eigenspace with reasonable accuracy in this case.

The active subspace for the 19-dimensional problem was also computed using regression-based estimates of the gradient that do not require model evaluations at neighboring points as discussed earlier. The quantity,  $\max(\delta \mathbf{W}_{1,j}^{(r)})$ , defined in (3.9), was used to assess the convergence behavior of the two approaches. Using a set tolerance,  $\tau = 0.05$ , it was observed that both, perturbation and regression approaches took 8 iterations to converge. Note that the computational effort at each iteration was considered to be the same in both cases. More specifically, 5 new random samples were added for the perturbation approach at each iteration. However, as discussed earlier, as total of 100 ( $=5 \times (19+1)$ ) model runs were needed to obtain the model prediction and its gradients at these newly generated samples. Hence, in the case of regression, 100 new random samples were generated at each iteration since gradient computation does not require additional model runs in this case. Thus, including the initial step, a total of 900 model runs were required to obtain a converged active subspace in both cases.

The accuracy of the two approaches was assessed by estimating  $\epsilon_{L^2}^{N-n}$  using the components of the dominant eigenvector in the converged active subspace in each case in (3.10). The

quantity,  $\varepsilon_{L^2}^{N-n}$  was estimated to be 0.0657 and 0.1050 using perturbation and regression respectively. Squared values of the individual components of the dominant eigenvector from the two approaches and for the case using  $N = 1000$  in the perturbation approach are plotted in Figure 3.2 (left). The set of eigenvector components for the three cases are

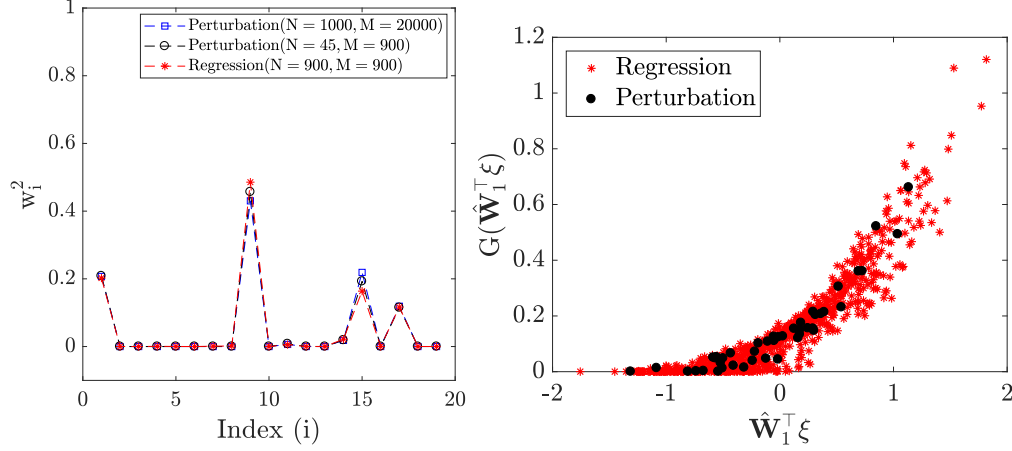


Figure 3.2: Left: An illustrative comparison of individual squared components of the converged dominant eigenvector obtained using perturbation and regression strategies using  $M = 900$  model runs in each case. Additionally, the dominant eigenvector components obtained using  $M = 20000$  model runs (corresponding to  $N = 1000$  samples) in the perturbation strategy (test case), used to assess the accuracy of the two strategies are also plotted. Right: An illustrative comparison of the SSPs generated using the perturbation and the regression strategies for computing the active subspace.

found to be in excellent agreement with each other, indicating that both approaches are sufficiently accurate for this setup.

As mentioned earlier, the model output  $f(\xi)$  i.e. the ignition delay in the  $\text{H}_2/\text{O}_2$  reaction in this case, varies predominantly in a 1-dimensional active subspace. As discussed in Section 2.2 we can approximate  $f(\xi)$  as  $G(\mathbf{W}_1^T \xi)$  where  $\mathbf{W}_1$  has one column containing the eigenvector corresponding to the dominant eigenvalue. The plot of  $G$  versus  $\mathbf{W}_1^T \xi$ , regarded as the *sufficient summary plot* (SSP), obtained using the perturbation-based and regression-based gradient estimates are compared in Figure 3.2 (right). The dominant eigenvector obtained using perturbation is based on  $N = 45$  samples which requires  $M = 900$  model runs. For the same amount of computational effort, we can afford  $N = 900$  samples when

using regression. Hence, the SSP from regression is based on 900 points:  $(\mathbf{W}_1^\top \boldsymbol{\xi}_j, G(\mathbf{W}_1^\top \boldsymbol{\xi}_j))$ ,  $j = 1, \dots, 900$ . On the other hand, the SSP from perturbation is plotted using only 45 points as mentioned earlier. Nevertheless, the illustrative comparison clearly indicates that the two SSPs are in excellent agreement. Moreover, it is interesting to note that the response in ignition delay based on the considered probability distributions for  $\log(A_i)$  although non-linear, can be approximated by a 1-dimensional active subspace.

We further estimate the normalized activity scores for individual uncertain inputs ( $\tilde{\nu}_{i,r}$ ;  $r=1$  since we have a 1-dimensional active subspace) using the components of the dominant eigenvector. The activity scores for the 19 uncertain pre-exponents ( $A_i$ 's), estimated using the perturbation and regression strategies are plotted in Figure 3.3. The activity scores

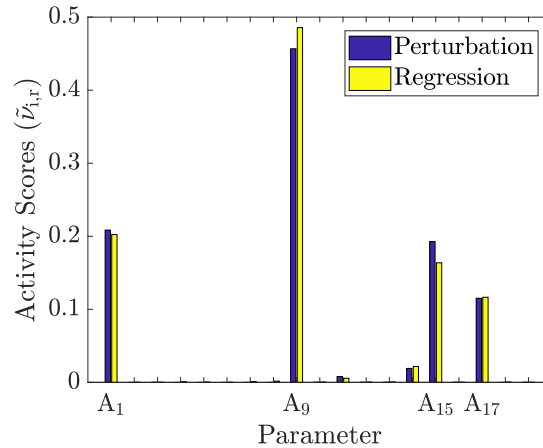


Figure 3.3: Left: A bar-graph of normalized activity scores ( $\tilde{\nu}_{i,r}$ 's) for the 19 uncertain pre-exponents ( $A_i$ 's);  $r$  denotes the number of eigenvectors in the dominant eigenspace.

based on the two approaches for gradient estimation agree favorably with each other as well as those based on the screening metric involving the DGSMs in [108]. It is observed that the uncertainty associated with the ignition delay is largely due to the uncertainty in  $A_9$  while  $A_1$ ,  $A_{15}$ , and  $A_{17}$  are also observed to contribute significantly towards its variance.

The above comparisons indicate that the gradient of the ignition delay with respect to the uncertain  $A_i$ 's is reasonably approximated using both, perturbation and regression approaches in this case. Since both approaches yield consistent results and are comparable in terms of convergence and accuracy, we could use either for the purpose of active subspace

computation for this setting. In the following section, we shift our focus to the higher-dimensional  $\text{H}_2/\text{O}_2$  reaction kinetics application wherein the activation energies in the rate law as well as initial pressure, temperature, and stoichiometric conditions are also considered to be uncertain.

### 3.5 $\text{H}_2/\text{O}_2$ reaction kinetics: higher-dimensional case

Here we aim to investigate the impact of uncertainty in the following problem parameters on the ignition delay associated with the  $\text{H}_2/\text{O}_2$  reaction: (i) pre-exponents ( $A_i$ 's); (ii) the activation energies ( $E_{a,i}$ 's); and (iii) the initial pressure ( $P_0$ ), temperature ( $T_0$ ), and stoichiometry ( $\Phi_0$ ). The  $\log(A_i)$ 's,  $E_{a,i}$ 's for all reactions except  $\mathcal{R}_6 - \mathcal{R}_9$ ,  $\mathcal{R}_{13}$  (due to zero nominal values for  $E_a$ ), and the initial conditions were considered to be uniformly distributed, and perturbed by 2% about their nominal values. Note that the magnitude of the perturbation was selected such that the ignition delay assumes a physically meaningful value in the input domain. The nominal values of the rate parameters,  $A_i$ 's and  $E_{a,i}$ 's were taken from [115]. The nominal values of  $P_0$ ,  $T_0$ , and  $\Phi_0$  were considered to be 1.0 atm, 900 K, and 2.0 respectively.

#### 3.5.1 Computing the active subspace

The active subspace was computed using the iterative procedure outlined earlier. The convergence of the eigenvectors was examined by tracking the quantity ' $\max(\delta \mathbf{W}_{1,j}^{(i)})$ '. In Figure 3.4 (right), we examine  $\max(\delta \mathbf{W}_{1,j}^{(i)})$  with increasing iterations for the perturbation and the regression approaches discussed earlier in Section 3.3. At each iteration, we improve our estimates of the matrix  $\hat{\mathbf{S}}$  by estimating the gradient of the ignition delay at 5 new randomly generated samples in the 36-dimensional input space. However, gradient computation at these 5 samples requires 185 ( $=5 \times (36+1)$ ) model runs when using perturbation. For the same computational effort, the regression approach can afford 185 new samples at each iteration. It is observed that using  $\tau = 0.05$ , the active subspace requires 4 iterations (925 model runs) to converge in the case of perturbation, and 9 iterations (1850 model runs) to converge in the case of regression. Hence, the computational effort required to obtain a

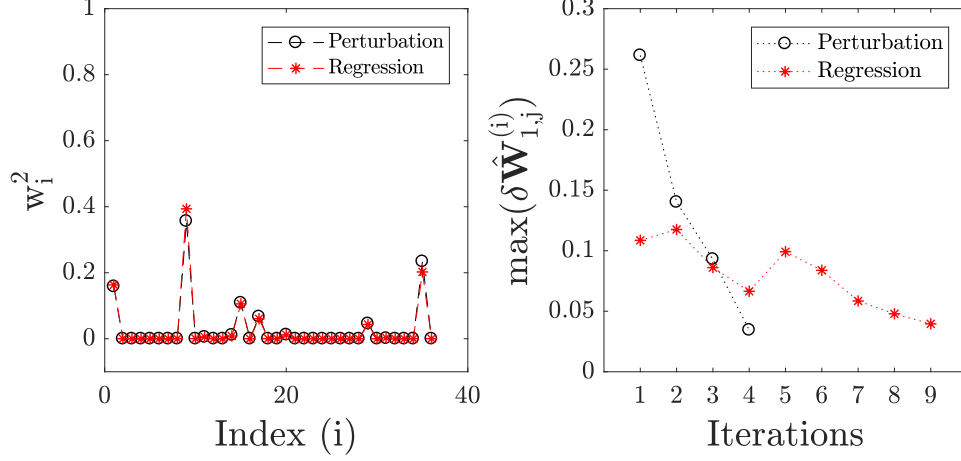


Figure 3.4: Left: An illustrative comparison of individual components of the dominant eigenvector in the converged active subspace i.e., at the end of 4 iterations in the perturbation approach and 9 iterations in the regression approach. Right: A comparison of the convergence behavior of the perturbation and the regression approaches. Convergence is accomplished once  $\max(\delta \hat{\mathbf{W}}_{1,j}^{(i)})$  assumes a value smaller than 0.05.

converged active subspace is doubled when using regression to approximate the gradient. Moreover, gradient estimation in the perturbation approach can be made more efficient by using techniques such as automatic differentiation [57] and adjoint computation [54]. These techniques although not pursued here are promising directions for future efforts pertaining to this work. In Figure 3.4 (right), we compare individual components of the dominant eigenvector in the converged active subspace obtained using the two approaches. The components are observed to be in excellent agreement with each other.

### 3.5.2 Constructing the surrogate model

The surrogate  $\hat{G}$  was constructed using a regression fit [111]. In figure 3.5 we compare the PDFs of the 1-dimensional surrogate for the regression and perturbation approaches with the true model. We also computed the mean and standard deviation for each case and noted the values were in close agreement with the true model  $G$ . The pdfs and the statistics indicate that the uncertainty in ignition delay is accurately captured with the surrogate models.

We computed the normalized activity scores using the active subspace based surrogate and

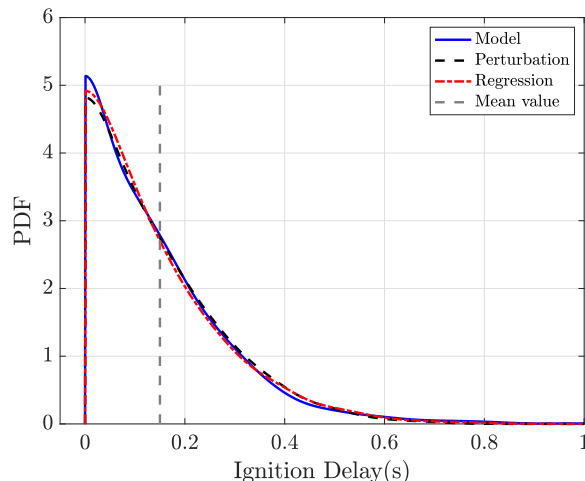


Figure 3.5: A comparison of the PDFs of ignition delay, obtained using model evaluations (solid line) regression-based surrogate (dashed line) and the perturbation-based surrogate (dashed-dotted line). The same set of  $10^4$  samples in the cross-validation set were used in each case.

identified that the following parameters as significant to the variability in ignition delay: pre-exponents  $A_1, A_9, A_{15}$  and  $A_{17}$ , the activation energy  $E_{a,15}$ , and the initial temperature  $T_0$ . The activity scores were found to be consistent with total Sobol' indices. For more details, see [111].

### 3.6 Discussion

In this chapter, we focused on the uncertainty associated with the rate-controlling parameters in the  $H_2/O_2$  reaction mechanism as well as the initial pressure, temperature, and stoichiometry and its impact on ignition delay predictions. The mechanism involves 19 different reactions and in each case, the reaction rate depends upon the choice of a pre-exponent and an activation energy. Hence, in theory, the evolution of the chemical system depends upon 38 rate parameters and three initial conditions. However, we considered uncertainty in all pre-exponents and activation energies with non-zero nominal values in addition to the three initial conditions. To enable efficient UQ we reduced the dimension of the problem using active subspaces, which aligns the input parameters along important directions in the parameter space. Additionally, we demonstrated that the activity scores, computed using the components of the dominant eigenvectors from the active subspace



provide an efficient means for approximating derivative based global sensitivity measures (DGSMs). Furthermore, we investigated links between activity scores, DGSMs, and total Sobol' indices which could be exploited to reduce computational effort associated with global sensitivity analysis.

Active subspace computation requires repeated evaluations of the gradient of the QoI. For this purpose, we explored two approaches, namely, perturbation and regression. Both approaches were shown to yield consistent results for the 19-dimensional problem wherein only the pre-exponents were considered to be uncertain. It was observed that the computational effort required to obtain a converged active subspace was comparable for the two approaches. However, the predictive accuracy of the perturbation approach was found to be relatively higher. Moreover, a 1-dimensional active subspace was shown to reasonably approximate the uncertainty in the ignition delay. Additionally, the activity scores were also shown to be consistent with the screening metric estimates based on DGSMs in [108]. An iterative procedure was adopted to enhance the computational efficiency.

The active subspace was further computed for a 36-dimensional problem wherein all pre-exponents and activation energies with non-zero nominal estimates as well as the initial conditions were considered uncertain. Once again, consistent results were obtained using the two approaches. A 1-dimensional active subspace was shown to reasonably capture the uncertainty in the ignition delay in this case. However, the computational effort required to compute a converged active subspace using perturbation was found to be half of the effort required in the case of regression. Predictive accuracy of the two approaches was found to be comparable. Hence, perturbation seems like a preferred approach for the higher-dimensional problem based on our findings. GSA results indicated that the variability in the ignition delay is predominantly due to the uncertainty in the rate parameters,  $A_1$  and  $A_9$  with significant contributions from  $A_{15}$ ,  $A_{17}$ , and  $E_{a,15}$ . Additionally, the ignition delay was found to be sensitive towards  $T_0$ .

Based on our findings, in the present application, the perturbation approach is preferable for active subspace computation; the computational cost of this approach can be reduced significantly, if more efficient gradient computation techniques (e.g., adjoint-based approaches or automatic differentiation) are feasible. The regression-based approach can be explored in situations involving intensive simulations where gradient computation is very challenging. We must mention however, that the regression based approach may fail

in cases where the sign of the partial derivatives does not remain constant over the entire parameter space, see [18, p.8]. We also mention that alternate regression-based approaches such as ones based on computing a global quadratic model have been proposed and used in the literature; see e.g., [25]. The applicability of such an approach in the context of high-dimensional chemical reaction networks is subject to future work.

The computational framework presented in this work can be easily adapted for other chemical systems as long as the quantity of interest is continuously differentiable in the considered domain of the inputs. We have demonstrated that the active subspace could be exploited for efficient forward propagation of the uncertainty from inputs to the output. The resulting activity scores and the low-dimensional surrogate could further guide optimal allocation of computational resources for calibration of the important rate-controlling parameters and input conditions in a Bayesian setting. Additionally, dimension reduction using active subspaces could assist in developing robust formulations for predicting discrepancy between simulations and measurements due to epistemic uncertainty in the model inputs.

## CHAPTER

## 4

# A DISTRIBUTED ACTIVE SUBSPACE METHOD FOR SCALABLE SURROGATE MODELING OF FUNCTION VALUED OUTPUTS

### **4.1 Introduction**

Models with uncertain input parameters are common in modeling of complex systems. Computational studies such as forward uncertainty propagation, optimization, or parameter estimation require repeated evaluation of the model. These tasks become challenging for expensive-to-evaluate complex models. To address this challenge, surrogate models are often used. By approximating the mapping from the uncertain input parameters to output quantities of interest (QoIs), using a surrogate model, one can replace expensive model

evaluations by inexpensive surrogate model evaluations. Examples of surrogate modeling tools include polynomial chaos expansion [37, 63], multivariate adaptive regression splines [34], and Gaussian processes [83].

In the present work, we consider surrogate construction for models of the form

$$y = f(\mathbf{x}, \boldsymbol{\xi}), \quad (4.1)$$

where  $\mathbf{x}$  belongs to a spatial domain and  $\boldsymbol{\xi} \in \mathbb{R}^{N_p}$  is a vector of uncertain parameters. In our target applications  $f$  is defined in terms of the solution of a partial differential equation (PDE) that is parameterized by a high-dimensional input parameter. Albeit, the proposed framework can be adapted to more general settings.

A simple approach is to construct a surrogate model pointwise in  $\mathbf{x}$ . Specifically, discretizing the spatial domain by grid points  $\{\mathbf{x}_i\}_{i=1}^{N_x}$ , we may consider approximating  $f(\mathbf{x}_i, \boldsymbol{\xi})$ , by constructing surrogate models  $\hat{f}_i(\boldsymbol{\xi}) \approx f(\mathbf{x}_i, \cdot)$ ,  $i = 1, \dots, N_x$ . This straightforward approach can be useful in some cases, however, the power of many of the surrogate modeling approaches can be fully realized if one optimizes them for each  $\mathbf{x}_i$  in the computational grid. This is a computationally expensive task and might become prohibitive if the parameter dimension  $N_p$  is large (e.g.,  $N_p \geq 100$ ), the model exhibits large variations in its response to parametric uncertainties over the spatial domain, or the grid point number  $N_x$  is large, as can happen in two or three-dimensional geometries.

We seek an efficient surrogate modeling approach, for models of the form (4.1), whose complexity in terms of the number of model evaluations does not scale with the dimension  $N_p$  of the input parameters. To enable this, we need a method that is *structure exploiting*. In particular, we seek to discover and utilize informative low-dimensional subspaces in the input space and low-dimensional spectral representations of the model output. The former uses ideas from active subspace methods [18] and the latter uses Karhunen–Loève (KL) expansions [37, 69]. The proposed approach decouples the spatial (i.e.  $\mathbf{x}$ ) dimensions and those of the random variable  $\boldsymbol{\xi}$  and exposes important structures that can be used for building efficient surrogate models.

**Related work.** KL expansions have been used in many works for representing random field parameters in physics models; see e.g., [8, 32, 33, 38, 40, 65, 66, 75, 87, 113]. KL expansions can also be used to represent random field outputs of physics models, as done in the present

work. In models governed by PDEs, the output field often exhibits favorable regularity properties and can be represented by a KL expansion with a relatively small number of KL terms. Examples of this appear for instance in our recent works [5, 17] for models governed by elliptic PDEs.

The active subspace method [18, 23, 86] has become a popular approach in recent years for input parameter dimension reduction and surrogate modeling. This method seeks to identify important linear combinations of the input parameters. A brief summary of the active subspace method, for approximating scalar valued models, has been provided in Chapter 2. The active subspace method has been successfully used for uncertainty analysis in models with scalar valued responses, in a host of engineering applications; examples include scramjet analysis [19], wing shape optimization [71], hydrologic modeling [55], battery modeling [22], and kinetic model uncertainty analysis [56]. Recently, there have been efforts in extending the active subspace method to vector valued functions. In [116] the authors find an upper bound for the error of a ridge function approximation of the original (vector valued) function, and construct an approximating function by minimizing that upper bound. In [56] an interesting approach is introduced for simultaneously approximating multiple outputs using a single low-dimensional shared subspace. The shared subspace is identified by solving a least-squares system to compute an appropriate combination of single-output active subspaces.

We also mention another related and popular class of methods for parameter dimension reduction: variance based [98, 99] and derivative based [58, 101] global sensitivity analysis (GSA). These methods, which were reviewed in Chapter 2, provide means of identifying unimportant model inputs, hence reducing the dimension of the input parameter vector. Derivative based methods are especially attractive, as they can be used to efficiently screen for unimportant input parameters, after which a surrogate model can be computed as a function of a reduced set of parameters. While GSA approaches have traditionally been applied to models with scalar outputs, extensions of GSA methods to vectorial and function valued outputs appear in several recent works [4, 17, 35].

In practice, the active subspace method tends to be very effective in reducing the input dimension as important *directions* in the input parameter space are identified. This is, in contrast to seeking reduced *parameter subsets*, as identified by GSA approaches. We note however that utility of GSA goes beyond input parameter dimension reduction—GSA pro-

vides valuable insight regarding a model by identifying key contributors to model variability. Also, we mention an interesting link between derivative based GSA and active subspaces through the idea of activity scores as detailed in [21] and in Chapter 3; computing active subspaces provides, as a byproduct, approximations to derivative based global sensitivity measures, for scalar valued models.

**Our approach and contributions.** In our proposed approach, we combine active subspaces and KL expansions to enable efficient surrogate modeling for function valued QoIs. Specifically, we consider a suitably truncated KL decomposition

$$f(\mathbf{x}, \xi) \approx \bar{f}(\mathbf{x}) + \sum_{k=1}^N \sqrt{\lambda_k} f_k(\xi) \phi_k(\mathbf{x}),$$

where  $\bar{f}$  is the mean of the process  $f(\mathbf{x}, \xi)$  and  $(\lambda_k, \phi_k)$  are the eigenpairs of the covariance operator of the process. This way, the model uncertainty is encoded in the KL modes,  $f_k$ ,  $k = 1, \dots, N$ . In many cases a small  $N$  can be very effective in approximating  $f$ . These KL modes can then be approximated efficiently using the active subspace approach. The active subspace approach requires the gradients of  $f_k$ , which are scalar functions of the random vector  $\xi$ . For such functions, adjoint state approaches provide efficient means of computing the required gradients. Specifically, gradients can be computed as a cost, in terms of model evaluations, that does not scale with the dimension of the input parameter  $\xi$ . Realizing the proposed approach requires (i) a rigorous functional framework upon which efficient computational algorithms can be built; and (ii) a systematic computational procedure that guides efficient implementations. These have been detailed in Section 3.3.

We deploy our proposed framework in the context of flows in biological tissues. Specifically, in Section 4.5, we tackle biotransport in porous tumors with high-dimensional random inputs (i.e., the permeability field) and random field outputs (i.e., the pressure distribution in tumors). Biological tissues usually have highly heterogeneous, uncertain, material properties [73]. The biotransport process such as drug delivery in tissues thus can exhibit “unpredictable” behaviors due to the uncertainties in tissue material properties [29]. Since the unpredictability can adversely affect the effectiveness of therapy, quantifying variability in biotransport due to uncertain material properties has a high impact on clinical trial protocol design. However, conducting useful uncertainty quantification studies on biotransport is challenging due to the high dimension of the input parameter space, e.g. permeability

and porosity [3]. We find that efficient-to-evaluate and accurate surrogate models can be computed at a modest computational cost, with our proposed framework. The presented computational experiments also indicate that the present method can be used successfully in more general porous medium flow problems, where Darcy flow is a reasonable model. Further investigations of the proposed surrogate modeling method in more complex flow models is subject of our future work.

## 4.2 Active subspace-based surrogate models for function-valued QoIs

In this section, we outline our approach for computing an active subspace-based surrogate model for a function valued QoI. Specifically, we consider models of the form

$$y = f(\mathbf{x}, \boldsymbol{\xi}), \quad \mathbf{x} \in \mathcal{X}, \boldsymbol{\xi} \in \Omega, \quad (4.2)$$

where  $\mathcal{X} \subset \mathbb{R}^n$ , with  $n = 2$  or  $3$ , is a compact set, and  $\Omega \subset \mathbb{R}^{N_p}$  is a sample space. The set  $\mathcal{X}$  will be a (sub-) region of a computational domain, in our target applications. Here  $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_{N_p})^T$  is a vector of uncertain parameters, with probability density function  $\pi(\boldsymbol{\xi})$ . We make the following assumptions about  $f(\mathbf{x}, \boldsymbol{\xi})$ .

**Assumption 1.** *We assume*

- (a)  $f \in L^2(\mathcal{X} \times \Omega)$  and  $f$  is a mean square continuous process.
- (b)  $\frac{\partial f}{\partial \xi_i}(x, \boldsymbol{\xi})$  exists for all  $x \in \mathcal{X}, \boldsymbol{\xi} \in \Omega, i = 1, \dots, N_p$ .
- (c)  $\frac{\partial f}{\partial \xi_i}(x, \boldsymbol{\xi}) \in L^2(\mathcal{X} \times \Omega), i = 1, \dots, N_p$ .

Letting  $\mathbb{E}[\cdot]$  denote expectation with respect to  $\boldsymbol{\xi}$ , the covariance function,  $c : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ , of  $f$  is defined as

$$c(\mathbf{x}, \mathbf{y}) := \mathbb{E}[f(\mathbf{x}, \cdot)f(\mathbf{y}, \cdot)] - \mathbb{E}[f(\mathbf{x}, \cdot)]\mathbb{E}[f(\mathbf{y}, \cdot)]$$

and the associated covariance operator  $C_f : L^2(\mathcal{X}) \rightarrow L^2(\mathcal{X})$  is given by

$$[C_f u](\mathbf{x}) := \int_{\mathcal{X}} c(\mathbf{x}, \mathbf{y}) u(\mathbf{y}) d\mathbf{y}.$$

Assumption 1(a) ensures that the covariance function  $c$  and the process mean

$$\bar{f}(\mathbf{x}) = \mathbb{E}[f(\mathbf{x}, \xi)]$$

are continuous on  $\mathcal{X} \times \mathcal{X}$  and  $\mathcal{X}$ , respectively; see e.g., [49, Theorem 7.3.2].

We reduce the dimension of the output by computing a low-rank approximation using a truncated KL expansion:

$$f(\mathbf{x}, \xi) \approx \hat{f}(\mathbf{x}, \xi) = \bar{f}(\mathbf{x}) + \sum_{k=1}^N \sqrt{\lambda_k(C_f)} f_k(\xi) \phi_k(\mathbf{x}). \quad (4.3)$$

Here  $\lambda_k(C_f)$  and  $\phi_k$  are the eigenvalues and eigenvectors of the covariance operator  $C_f$  and  $f_k$ 's are given by

$$f_k(\xi) = \frac{1}{\sqrt{\lambda_k(C_f)}} \int_{\mathcal{X}} (f(\mathbf{x}, \xi) - \bar{f}(\mathbf{x})) \phi_k(\mathbf{x}) d\mathbf{x}. \quad (4.4)$$

We refer to the coefficients  $f_k$  as the *KL modes* of  $f$ . In the present work, the (generalized) eigenvalue problem,

$$C_f \phi_k = \lambda_k(C_f) \phi_k, \quad \int_{\mathcal{X}} \phi_k(\mathbf{x})^2 d\mathbf{x} = 1, \quad k = 1, 2, \dots, \quad (4.5)$$

is solved using Nyström's method. In practice,  $N$  in (4.3) can be chosen such that

$$\left( \sum_{k=1}^N \lambda_k(C_f) \right) / \left( \sum_{k=1}^{\infty} \lambda_k(C_f) \right) < \text{tol},$$

where tol is a user-specified tolerance. In many applications of interest, where  $f$  is defined in terms of the solution of a differential equation, the eigenvalues  $\lambda_k(C_f)$  exhibit rapid decay, which enables low-rank representations. In particular, this is observed in the application problem considered in the present work.

The KL modes  $f_k(\xi)$  are functions of a potentially high-dimensional parameter  $\xi$  so the next step is to approximate  $f_k(\xi)$  using active subspaces [23]. For each KL mode  $f_k$ ,  $k = 1, \dots, N$ , we compute an active subspace by considering the symmetric positive semidefinite matrix,



$\mathbf{S}_k \in \mathbb{R}^{N_p \times N_p}$ , defined by

$$\mathbf{S}_k = \int_{\mathbb{R}^{N_p}} (\nabla f_k(\xi)) (\nabla f_k(\xi))^T \pi(\xi) d\xi. \quad (4.6)$$

As before, we compute the spectral decomposition  $\mathbf{S}_k = \mathbf{W}_k \mathbf{\Lambda}_k \mathbf{W}_k^T$ , and partition the eigenpairs according to

$$\mathbf{\Lambda}_k = \begin{bmatrix} \mathbf{\Lambda}_{k,1} & \\ & \mathbf{\Lambda}_{k,2} \end{bmatrix}, \quad \mathbf{W}_k = [\mathbf{W}_{k,1} \quad \mathbf{W}_{k,2}],$$

where  $\mathbf{\Lambda}_{k,1} \in \mathbb{R}^{r_k \times r_k}$  is a diagonal matrix with the dominant eigenvalues of  $\mathbf{S}_k$  on its diagonal; and  $\mathbf{W}_k$  contains the corresponding eigenvectors. Defining,

$$G_k(\mathbf{y}) = f_k(\mathbf{W}_{k,1} \mathbf{y}), \quad \mathbf{y} \in \mathbb{R}^{r_k}, k = 1, \dots, N,$$

the KL modes can be approximated by

$$f_k(\xi) \approx G_k(\mathbf{W}_{k,1}^T \xi), \quad k = 1, \dots, N.$$

In practice, the active subspace-based surrogates for the KL modes  $f_k$  are constructed by following Algorithm 1, with  $g$  replaced by  $f_k$ ,  $k = 1, \dots, N$ . Thus, we obtain surrogate models

$$f_k(\xi) \approx \tilde{G}_k(\mathbf{W}_{k,1}^T \xi), \quad k = 1, \dots, N.$$

The overall surrogate model for  $f(\mathbf{x}, \xi)$  is then given by

$$f(\mathbf{x}, \xi) \approx \hat{f}(\mathbf{x}, \xi) = \bar{f}(\mathbf{x}) + \sum_{k=1}^N \sqrt{\lambda_k} \tilde{G}_k(\mathbf{W}_{k,1}^T \xi) \phi_k(\mathbf{x}). \quad (4.7)$$

The computational steps for computing  $\hat{f}(\mathbf{x}, \xi)$  can be divided into three main steps:

1. Compute the truncated KL expansion of  $f(\mathbf{x}, \xi)$ .
2. Compute the active subspace-based approximation to KL modes  $f_k$ ,  $k = 1, \dots, N$ .
3. Form the overall surrogate model as in (4.7).

The most computationally challenging part of the above process is the first step, in which

we require an ensemble of model evaluations  $f(\cdot, \xi_i)$ ,  $i = 1, \dots, N_s$ . These model evaluations will be used to compute the KL expansion of the model  $f(\mathbf{x}, \xi)$ . For this, we use Algorithm 1 in [5], that uses Nyström's method to compute  $\lambda_k(C_f)$  and the corresponding eigenvectors  $\phi_k(\cdot)$ ,  $k = 1, \dots, N$ . This process also provides the evaluations of the KL modes,  $f_k(\xi_i)$ ,  $k = 1, \dots, N$ ,  $i = 1, \dots, N_s$ . As shown in our numerical results, often a modest choice of  $N_s$  is sufficient for obtaining reliable approximations to (i) the dominant eigenpairs, and (ii) KL modes  $f_k$ .

Notice that for implementing the proposed method, differentiability of the KL modes is required. Moreover, as seen below, for the purposes of error analysis, Lipschitz continuity of the output KL modes is needed. These requirements can be satisfied through suitable boundedness assumptions on the partial derivatives of  $f(\mathbf{x}, \xi)$ , as we now explain. For convenience, and with no loss of generality, we consider the case where  $\bar{f}(\mathbf{x}) \equiv 0$  and consider

$$F(\xi) = \int_{\mathcal{X}} f(\mathbf{x}, \xi) v(\mathbf{x}) d\mathbf{x}, \quad (4.8)$$

where we use a generic element  $v \in L^2(\mathcal{X})$ , with  $\|v\|_{L^2(\mathcal{X})} = 1$ , in place of the eigenvectors  $\phi_k$ . This  $F$  can be thought of as a generic *unnormalized* KL mode. In addition to the earlier assumptions on  $f$ , we also require

$$\left| \frac{\partial f}{\partial \xi_j}(\mathbf{x}, \xi) \right| \leq b_j(\mathbf{x}), \quad \text{for all } \mathbf{x} \in \mathcal{X}, \xi \in \Omega,$$

where  $b_j$ ,  $j = 1, \dots, N$ , are square integrable.

**Lemma 4.2.1.** *Suppose the process  $f(\mathbf{x}, \xi)$  satisfies Assumption 1 and (4.8). Then,  $F$  is differentiable and is Lipschitz continuous.*

*Proof.* Differentiability of  $F$  can be shown using the standard arguments of differentiating under the integral sign; see e.g., what where it is shown that under the present set of assumptions,  $\frac{\partial F}{\partial \xi_j} = \int_{\mathcal{X}} \frac{\partial f}{\partial \xi_j}(\mathbf{x}, \xi) v(\mathbf{x}) d\mathbf{x}$ . We also have

$$|\partial_j F(\xi)| \leq \int_{\mathcal{X}} |\partial_j f(\mathbf{x}, \xi) v(\mathbf{x})| d\mathbf{x} \leq \int_{\mathcal{X}} |b_j(\mathbf{x})| |v(\mathbf{x})| d\mathbf{x} \leq \|b_j\|_{L^2(\mathcal{X})} \|v\|_{L^2(\mathcal{X})},$$

for every  $\mathbf{x} \in \mathcal{X}$ . Using this, it is straightforward to show,  $|F(\xi_1) - F(\xi_2)| \leq L \|\xi_1 - \xi_2\|$ , for all  $\xi_1, \xi_2 \in \Omega$ , with  $L = \left( \sum_{j=1}^N \|b_j\|_{L^2(\mathcal{X})}^2 \right)^{1/2}$ .  $\square$

### 4.2.1 Gradient computation

The present active subspace-based surrogate modeling approach requires computing the gradient of the output KL modes. Here, it is more convenient to consider the “unnormalized KL modes”,  $F_k$ ,  $k = 1, \dots, N$ .

$$F_k(\xi) = \int_{\mathcal{X}} (f(\mathbf{x}, \xi) - \bar{f}(\mathbf{x})) \phi_k(\mathbf{x}) d\mathbf{x}. \quad (4.9)$$

Finite-difference approximations provide a simple approach, but will be prohibitive when the input dimension  $N_p$  is large and model evaluations are expensive. For field quantities defined in terms of the solution  $u$  of a PDE parameterized by  $\xi$ ,  $F_k$  is a functional of  $u$ . This is an ideal situation for deploying adjoint based gradient computation. To illustrate this, we assume the model  $f(\mathbf{x}, \xi)$  is a function of the solution  $u(\mathbf{x}, \xi)$  of a PDE. For instance  $f$  can be the restriction of  $u$  to a sub-domain, or  $f$  can be flux of  $u$  through a boundary. For illustration, we consider the case where the PDE is of the form  $\mathcal{A}(\xi)u = b$ , where  $\mathcal{A}$  denotes a differential operator that is parameterized by the uncertain parameter vector  $\xi$  and  $b$  is a source term, and assume  $f(\mathbf{x}, \xi) = u(\mathbf{x}, \xi)$ . To simplify the presentation further, we consider the discretized problem, where the discretized KL mode is defined by

$$\mathbf{F}_k(\xi) = (\mathbf{u} - \bar{\mathbf{u}})^T \mathbf{W} \boldsymbol{\phi}_k, \quad \text{where} \quad \mathbf{A}(\xi) \mathbf{u} = \mathbf{b}.$$

Here  $\mathbf{u}$  is the discretized state variable,  $\mathbf{W}$  is a diagonal matrix with quadrature weights on diagonal,  $\boldsymbol{\phi}_k$  is the discretized  $k$ th eigenvector of output covariance,  $\mathbf{A}$  is the discretized PDE operator, and  $\mathbf{b}$  is the discretized source term. Computing the gradient of  $\mathbf{F}_k$  with respect to  $\xi$  can be done using a standard Lagrangian formalism [41]. Namely, we define the Lagrangian

$$\mathcal{L}(\mathbf{u}, \xi, \mathbf{q}) = (\mathbf{u} - \bar{\mathbf{u}})^T \mathbf{W} \boldsymbol{\phi}_k + \mathbf{q}^T (\mathbf{A}(\xi) \mathbf{u} - \mathbf{b}),$$

where  $\mathbf{q}$  is a Lagrange multiplier. Setting  $\frac{\partial \mathcal{L}}{\partial \mathbf{q}} = 0$  recovers the state equation, and setting  $\frac{\partial \mathcal{L}}{\partial \mathbf{u}} = 0$  gives the adjoint equation

$$\mathbf{A}^T(\xi) \mathbf{q} = -\mathbf{W} \boldsymbol{\phi}_k.$$

Then, the gradient of  $F_k$  is given by

$$\nabla F_k(\xi)^T = \frac{\partial \mathcal{L}}{\partial \xi} = \mathbf{q}^T \frac{\partial \mathbf{A}(\xi)}{\partial \xi} \mathbf{u}.$$

Note that evaluation of  $\nabla F_k$  requires one state (forward) equation solve and one adjoint equation solve, independently of the dimension of the uncertain parameter vector  $\xi$ . (The forward solves can be reused across the output KL modes.) Thus, to compute  $\nabla F_k$ ,  $k = 1, \dots, N$ , we need one solution of the state equation, and  $N$  adjoint solves.

Computing the active subspaces for the output KL modes can be done with a set  $N_s$  of model evaluations, used across all KL modes. Thus, the computational cost of active subspace discovery for the output KL modes is  $N_s(1 + N)$ , independent of the parameter dimension  $N_p$ . Typically a modest  $N_s$  is sufficient, as seen in our numerical results. Furthermore, the same set of model evaluations can be used for surrogate model construction for the output KL modes. This enables efficient computation of active subspaces for individual KL modes, at a cost that does not scale with the dimension of  $\xi$ .

Notice that the present illustration uses an equation  $\mathbf{A}(\xi)\mathbf{u} = \mathbf{b}$ , which is linear in the state variable and nonlinear in the uncertain parameter  $\xi$ . Such an equation can result from discretizing linear (in state) PDEs that are parameterized by uncertain parameters. The adjoint approach can more generally be applied to nonlinear PDE models; see e.g., [41].

In the present work, we use adjoint based gradient computation for computing the gradient of the output KL modes for models governed by elliptic PDEs with a random coefficient function; see section 4.5.

### 4.3 Error Analysis

The presented computational strategy involves several approximations for computing the active subspace-based surrogate for  $f(\mathbf{x}, \xi)$ . In this section, we analyze the errors incurred due to (i) truncation of the output KL expansion and (ii) active subspace approximation of the KL modes. For the purposes of the presented analysis, it is more convenient to work

with unnormalized KL modes (4.9), and consider

$$f_N(\mathbf{x}, \boldsymbol{\xi}) = \bar{f}(\mathbf{x}) + \sum_{k=1}^N F_k(\boldsymbol{\xi}) \phi_k(\mathbf{x}). \quad (4.10)$$

The active subspace strategy is flexible regarding the distribution law of the random vector  $\boldsymbol{\xi}$ ; see e.g., [18]. However, in the present work, where we consider models with uncertain coefficient functions that are modeled using log-Gaussian random fields,  $\boldsymbol{\xi}$  is a standard Gaussian random vector, i.e.,  $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ .

As detailed in [18], while Algorithm 1 provides a practical surrogate modeling framework, it is not directly amenable to theoretical analysis. To enable error analysis, following [18], we define the functions  $G_k$ , used for active subspace projection of the KL modes in the following way:

$$G_k(\mathbf{y}) = \int f_k(\mathbf{W}_{k,1}\mathbf{y} + \mathbf{W}_{k,2}\mathbf{z}) \pi_{Y|Z}(\mathbf{z}|\mathbf{y}) d\mathbf{z}, \quad (4.11)$$

where  $\pi_{Y|Z}$  is the conditional density,

$$\pi_{Y|Z}(\mathbf{z}|\mathbf{y}) := \frac{\pi(\mathbf{W}_{k,1}\mathbf{y} + \mathbf{W}_{k,2}\mathbf{z})}{\pi_Z(\mathbf{z})}, \quad \text{with } \pi_Z(\mathbf{z}) = \int \pi(\mathbf{W}_{k,1}\mathbf{y} + \mathbf{W}_{k,2}\mathbf{z}) d\mathbf{y}.$$

In practice, the *marginalized*  $G_k$ , which for convenience we can denote by

$$G_k(\mathbf{y}) = \mathbb{E}_{\mathbf{z}} \{f_k(\mathbf{W}_{k,1}\mathbf{y} + \mathbf{W}_{k,2}\mathbf{z})\},$$

can be approximated by Monte Carlo sampling,

$$G_k(\mathbf{y}) \approx \hat{G}_k(\mathbf{y}) := \frac{1}{N_{\text{AS}}} \sum_{i=1}^{N_{\text{AS}}} f_k(\mathbf{W}_{k,1}\mathbf{y} + \mathbf{W}_{k,2}\mathbf{z}_i). \quad (4.12)$$

However, as seen below, even a very small Monte Carlo Sample (even with  $N_{\text{AS}} = 1$ ) can be acceptable. This partly justifies and explains the effectiveness of Algorithm 1, which can be seen as a special case of (4.12), with  $N_{\text{AS}} = 1$  and  $\mathbf{z}_1 = \mathbf{0}$ .

Recall the approximation of the KL modes  $F_k(\boldsymbol{\xi}) \approx G_k(\mathbf{W}_{k,1}^T \boldsymbol{\xi})$ . As a first step in our error

analysis, we quantify the error in this approximation. For each  $k \in \{1, \dots, N\}$ , we define

$$\delta_k := \left( \sum_{j=r_k+1}^{N_p} \lambda_j(\mathbf{S}_k) \right)^{1/2}, \quad (4.13)$$

where  $\{\lambda_j(\mathbf{S}_k)\}_{j=1}^{N_p}$  are the eigenvalues of  $\mathbf{S}_k$  defined in (4.6) and  $r_k$  is the dimension of the active subspace for the  $k$ th KL mode  $F_k(\xi)$ . The following result bounds the error in approximating the individual KL modes.

**Lemma 4.3.1.** *Let  $\delta_k$  be as in (4.13), and let  $G_k$  and  $\hat{G}_k$  be as in (4.11) and (4.12). Then,*

1.  $\int_{\Omega} (F_k(\xi) - G_k(\mathbf{W}_{k,1}^T \xi))^2 \pi(\xi) d\xi \leq \delta_k^2.$
2.  $\int_{\Omega} (F_k(\xi) - \hat{G}_k(\mathbf{W}_{k,1}^T \xi))^2 \pi(\xi) d\xi \leq (1 + N_{AS}^{-1/2}) \delta_k^2.$

*Proof.* The KL modes  $F_k$  are square integrable, mean zero, and by Lemma 4.2.1, they are differentiable and Lipschitz continuous. Thus, the first statement follows from using [18, Theorem 4.3] and the second one follows from [18, Theorem 4.4].  $\square$

Next, we consider the error, due to active subspace projection (for individual KL modes), in approximating the KL expansion:

$$e(\mathbf{x}, \xi) = |f_N(\mathbf{x}, \xi) - \hat{f}_N(\mathbf{x}, \xi)|$$

where  $f_N$  is as in (4.10), and

$$\hat{f}_N(\mathbf{x}, \xi) := \sum_{k=1}^N G_k(\mathbf{W}_{k,1}^T \xi) \phi_k(\mathbf{x}) \quad (4.14)$$

its active subspace-based approximation where  $G_k(\mathbf{W}_{k,1}^T \xi)$  is our approximation of the KL modes  $F_k(\xi)$ , as defined before. Then let

$$\bar{e}(\xi) = \int_{\mathcal{X}} e(\mathbf{x}, \xi) d\mathbf{x}.$$

**Theorem 4.3.1.**  $\mathbb{E}\{\bar{e}\} \leq |\mathcal{X}|^{\frac{1}{2}} \sum_{k=1}^N \delta_k$ , where  $\delta_k$  is as in (4.13).

*Proof.* Note that,

$$e(\mathbf{x}, \xi) = \left| \sum_{k=1}^N (F_k(\xi) - G_k(\mathbf{W}_{k,1}^T \xi)) \phi_k(\mathbf{x}) \right| \leq \sum_{k=1}^N |F_k(\xi) - G_k(\mathbf{W}_{k,1}^T \xi)| |\phi_k(\mathbf{x})|.$$

Then,

$$\begin{aligned} \mathbb{E}\{\bar{e}\} &= \mathbb{E} \left\{ \int_{\mathcal{X}} \sum_{k=1}^N |F_k(\xi) - G_k(\mathbf{W}_{k,1}^T \xi)| |\phi_k(\mathbf{x})| d\mathbf{x} \right\} \\ &= \sum_{k=1}^N \left( \int_{\mathcal{X}} |\phi_k(\mathbf{x})| d\mathbf{x} \right) \mathbb{E}\{|F_k(\xi) - G_k(\mathbf{W}_{k,1}^T \xi)|\} \\ &\leq \sum_{k=1}^N \left[ \int_{\mathcal{X}} |\phi_k(\mathbf{x})|^2 d\mathbf{x} \right]^{\frac{1}{2}} \left[ \int_{\mathcal{X}} 1^2 d\mathbf{x} \right]^{\frac{1}{2}} \mathbb{E}\{|F_k(\xi) - G_k(\mathbf{W}_{k,1}^T \xi)|^2\}^{1/2} \leq |\mathcal{X}|^{\frac{1}{2}} \sum_{k=1}^N \delta_k \end{aligned}$$

where we have used Cauchy–Schwarz inequality and Lemma 4.3.1(a). □

**Remark 4.3.1.** Note that in view of Lemma 4.3.1(b), if we use  $\hat{G}_k$  defined in (4.12), instead of  $G_k$  in (4.14), we can repeat the argument in proof of Theorem 4.3.1 to get the following estimate:

$$\mathbb{E}\{\bar{e}\} \leq |\mathcal{X}|^{\frac{1}{2}} (1 + N_{AS}^{-1/2}) \sum_{k=1}^N \delta_k.$$

Finally, we consider the overall error of approximating  $f(\mathbf{x}, \xi)$ , due to KL truncation and active subspace projection:

$$E(\mathbf{x}, \xi) = |f(\mathbf{x}, \xi) - \hat{f}_N(\mathbf{x}, \xi)|$$

where  $f$  is the original QoI and  $\hat{f}_N$  is the active subspace-based approximation as before. We consider,

$$\bar{E}(\xi) := \int_{\mathcal{X}} E(\mathbf{x}, \xi) d\mathbf{x}.$$

We have the following result:

**Theorem 4.3.2.**

$$\mathbb{E}\{\bar{E}(\xi)\} \leq |\mathcal{X}|^{\frac{1}{2}} \left[ \left( \sum_{k=N+1}^{\infty} \lambda_k(C_f) \right)^{1/2} + \sum_{k=1}^N \left( \sum_{j=r_k+1}^{N_p} \lambda_j(\mathbf{S}_k) \right)^{1/2} \right]. \quad (4.15)$$

*Proof.* We have

$$\begin{aligned} \bar{E}(\xi) &= \int_{\mathcal{X}} E(\mathbf{x}, \xi) d\mathbf{x} = \int_{\mathcal{X}} |f(\mathbf{x}, \xi) - f_N(\mathbf{x}, \xi) + f_N(\mathbf{x}, \xi) - \hat{f}_N(\mathbf{x}, \xi)| \\ &\leq \int_{\mathcal{X}} |f(\mathbf{x}, \xi) - f_N(\mathbf{x}, \xi)| d\mathbf{x} + \int_{\mathcal{X}} |f_N(\mathbf{x}, \xi) - \hat{f}_N(\mathbf{x}, \xi)| d\mathbf{x} \\ &\leq \left\{ \int_{\mathcal{X}} |f(\mathbf{x}, \xi) - f_N(\mathbf{x}, \xi)|^2 d\mathbf{x} \right\}^{1/2} |\mathcal{X}|^{\frac{1}{2}} + \bar{e}(\xi) \\ &= \left\{ \sum_{k=N+1}^{\infty} F_k(\xi)^2 \right\}^{1/2} |\mathcal{X}|^{\frac{1}{2}} + \bar{e}(\xi). \end{aligned}$$

Now we consider

$$\begin{aligned} \mathbb{E}\{\bar{E}(\xi)\} &\leq \mathbb{E}\left( \left\{ \sum_{k=N+1}^{\infty} F_k(\xi)^2 \right\}^{1/2} \right) |\mathcal{X}|^{\frac{1}{2}} + \mathbb{E}\{\bar{e}(\xi)\} \\ &\leq |\mathcal{X}|^{\frac{1}{2}} \left[ \mathbb{E}\left( \left\{ \sum_{k=N+1}^{\infty} F_k(\xi)^2 \right\}^{1/2} \right) + \sum_{k=1}^N \delta_k \right] \leq |\mathcal{X}|^{\frac{1}{2}} \left[ \mathbb{E}\left( \sum_{k=N+1}^{\infty} F_k(\xi)^2 \right)^{1/2} + \sum_{k=1}^N \delta_k \right]. \end{aligned}$$

Note that  $\mathbb{E}\left( \sum_{k=N+1}^{\infty} F_k(\xi)^2 \right) = \sum_{k=N+1}^{\infty} \mathbb{E}(F_k(\xi)^2) = \sum_{k=N+1}^{\infty} \lambda_k(C_f)$ . Therefore, we have our desired result

$$\begin{aligned} \mathbb{E}\{\bar{E}(\xi)\} &\leq |\mathcal{X}|^{\frac{1}{2}} \left[ \left( \sum_{k=N+1}^{\infty} \lambda_k(C_f) \right)^{1/2} + \sum_{k=1}^N \delta_k \right] \\ &= |\mathcal{X}|^{\frac{1}{2}} \left[ \left( \sum_{k=N+1}^{\infty} \lambda_k(C_f) \right)^{1/2} + \sum_{k=1}^N \left( \sum_{j=r_k+1}^{N_p} \lambda_j(\mathbf{S}_k) \right)^{1/2} \right]. \end{aligned}$$

□

We note that the first term in (4.15) indicates error due to truncation of the output KL



expansion. Recall that

$$\sum_{k=1}^{\infty} \lambda_k(C_f) = \int_{\mathcal{X}} c_f(\mathbf{x}, \mathbf{x}) d\mathbf{x} < \infty,$$

where the equality is due to Mercer's Theorem, and the finiteness of the integral is due to continuity of the covariance operator, which is a consequence of the mean square continuity assumption on the process. Therefore, the first term in (4.15) can be made arbitrarily small by taking  $N$  sufficiently large. However, choosing a large  $N$  could entail accumulation of error due to active subspace projection error in the second term in (4.15); this error, however, can be controlled by increasing  $r_k$ . In practice, in many applications, a small number of output KL modes (i.e., a small  $N$ ), can be used to obtain an accurate KL representation for the output. Moreover, typically low-dimensional (in many cases one- or two-dimensional) active subspaces can be afforded for approximating the dominant KL modes. We demonstrate these issues numerically in Section 4.5.

**Remark 4.3.2.** *In view of Remark 4.3.1, if we use  $\hat{G}_k$  defined in (4.12), instead of  $G_k$  in (4.14), we can repeat the argument in proof of Theorem 4.3.2 to get the following estimate:*

$$\mathbb{E}\{\bar{E}(\xi)\} \leq |\mathcal{X}|^{\frac{1}{2}} \left[ \left( \sum_{k=N+1}^{\infty} \lambda_k(C_f) \right)^{1/2} + (1 + N_{AS}^{-1/2}) \sum_{k=1}^N \left( \sum_{j=r_k+1}^{N_p} \lambda_j(\mathbf{S}_k) \right)^{1/2} \right].$$

**Additional sources of error.** The above error analysis only concerns errors associated with active subspace projection and output KL truncation. In practical computations there are a number of other errors. Most of these errors are related to the active subspace approach used for approximating the KL modes  $F_k$ . These include errors in approximating the eigenvalues and eigenvectors of  $\mathbf{S}_k$ 's, incurred due to sample average approximation to these matrices and surrogate modeling errors incurred in approximating  $G_k$ 's. Errors in approximating  $\Lambda_k$  and  $\mathbf{W}_k$  are analyzed for instance in [18]. Errors due to surrogate modeling of  $G_k$ 's are difficult to quantify in general, as these errors depend on the choice of surrogate modeling framework.

There are also further errors related to KL approximation of the output field. Namely, we need to approximate the mean field  $\mathbb{E}\{f(\mathbf{x}, \cdot)\}$ . If simple Monte Carlo sampling is used, approximations of the mean field exhibit the usual Monte Carlo convergence behavior. However, quasi-Monte Carlo approaches can provide efficient means of obtaining more accurate estimates. Finally, there will be errors in computing the eigenvalues  $\lambda_k(C_f)$  and the

corresponding eigenvectors. These errors are due to (i) sample average approximation to the output covariance function, and (ii) errors due to discretizing the generalized eigenvalue problem (4.5). The discretization errors of course depend on the numerical method used for solving the eigenvalue problem. For example, if Nyström’s method is used, as done in the present work, the discretization errors can be controlled by the resolution of the computational grid, and the quadrature method used.

We demonstrate numerically that once a suitable output dimension reduction is determined, the active subspace approach can be deployed to obtain approximations to the output KL modes and an overall surrogate model that captures the statistical properties of  $f$  reliably. We find that this can be accomplished with an ensemble of function evaluations  $\{f(\cdot, \xi_j)\}_{j=1}^{N_s}$ , with a modest  $N_s$ .

## 4.4 Application Problem

In this section, we present our computational results in the context of a biotransport application problem. We begin by describing the governing model in Section 4.4.1. Next, we discuss computation of the KL expansion of the output in Section 4.4.2. This is followed by our results on active subspace discovery and surrogate model construction in Section 4.5.1. We test the accuracy of the computed surrogate models in Section 4.5.2.

### 4.4.1 The governing model

In this section we describe the biotransport problem we seek to investigate using the proposed method. We are interested in understanding the impact of uncertainty in the material properties of cancerous tumors. Specifically we seek to characterize the uncertainties in the pressure field when a single needle injection occurs at the center of a spherical tumor with uncertain heterogeneous structure. We focus on a 2D cross-section, and consider Darcy’s law constrained by mass conservation in a domain  $\mathcal{D} \subset \mathbb{R}^2$  given by a circle of radius  $R_{\text{tumor}} = 5$  mm, centered at the origin, with an inner circle of radius  $R_{\text{needle}} = 0.25$  mm, modeling the injection site, removed. We denote the inner and outer boundaries of the

domain by  $\Gamma_N$  and  $\Gamma_D$ , respectively. The following elliptic PDE governs the fluid pressure  $p$ :

$$\begin{aligned} -\nabla \cdot \left( \frac{\kappa}{\mu} \nabla p \right) &= 0 \quad \text{in } \mathcal{D}, \\ p &= 0 \quad \text{on } \Gamma_D, \\ \frac{\kappa}{\mu} \nabla p \cdot \mathbf{n} &= \frac{Q}{2\pi R_{\text{needle}}} \quad \text{on } \Gamma_N. \end{aligned} \tag{4.16}$$

In this equation,  $\kappa$  denotes the absolute permeability field,  $\mu$  is the fluid dynamic viscosity,  $Q$  is the volume flow rate per unit length, and  $\mathbf{n}$  is the outward-pointing normal vector. The nominal values for the above parameters are  $\kappa = 0.5 \text{ md}$ ,  $\mu = 8.9 \times 10^{-4} \text{ Pa} \cdot \text{s}$ , and  $Q = 1 \text{ mm}^2/\text{min}$ . These values are chosen according to previous investigations of fluid transport in tumors [16, 72, 90]. As noted in a number of previous works, tumors exhibit complex structures due to their invasive nature. Generally, tumors consist of loosely organized abnormal cells, fibers, vasculature, and lymphatics [15], resulting in disordered tissues with complex heterogeneous structures.

In the present work, we model the permeability field  $\kappa$  by a log-Gaussian random field as follows. Let  $z(\mathbf{x}, \omega)$  be a centered Gaussian process; here  $\omega \in \Omega$  where  $\Omega$  is an appropriate sample space. We assume  $z$  has unit pointwise variance and has correlation function

$$c_z(\mathbf{x}, \mathbf{y}) = \exp \left\{ -\frac{1}{\ell} \|\mathbf{x} - \mathbf{y}\|_1 \right\}, \quad \mathbf{x}, \mathbf{y} \in \mathcal{D}, \tag{4.17}$$

where  $\ell > 0$  is the correlation length. In the present study we set the correlation length  $\ell = 1 \text{ mm}$ . Then, we define the log-permeability field  $a = \log \kappa$  according to

$$a(\mathbf{x}, \omega) = a_0(\mathbf{x}) + \sigma_a z(\mathbf{x}, \omega), \quad \mathbf{x} \in \mathcal{D}, \omega \in \Omega,$$

where  $a_0$  and  $\sigma_a^2$  represent the pointwise mean and variance, respectively. We can represent  $a(\mathbf{x}, \omega)$  using a truncated KL expansion:

$$a(\mathbf{x}, \omega) \approx \hat{a}(\mathbf{x}, \omega) := a_0(\mathbf{x}) + \sum_{j=1}^{N_p} \sqrt{\lambda_j(C_a)} \xi_j(\omega) e_j(\mathbf{x}), \tag{4.18}$$

where  $a_0$  is the mean field,  $(\lambda_j(C_a), e_j)$  are the eigenpairs of the covariance operator  $C_a$  of  $a(\mathbf{x}, \omega)$ ,  $N_p$  is the input parameter dimension, and  $\xi_j$  are independent standard normal ran-

dom variables. With this parameterization, the uncertainty in the (approximate) log permeability field  $\hat{a} = \log \kappa$  is completely characterized by the random vector  $\xi = (\xi_1, \xi_2, \dots, \xi_{N_p})^T$ . That is,  $\hat{a}(\mathbf{x}, \omega) = \hat{a}(\mathbf{x}, \xi(\omega))$ , and thus, we can consider the (approximate) log-permeability field as a random process  $\hat{a} : \mathcal{D} \times \Omega \rightarrow \mathbb{R}$ . The QoI under study here is the pressure field  $p(\mathbf{x}, \xi)$ .

For illustration, two sets of realizations of the permeability field and the corresponding pressure field with correlation length  $\ell = 1 \text{ mm}$  are presented in Figure 4.1. We observe large fluctuations in the permeability field and relatively mild fluctuations in the pressure field.

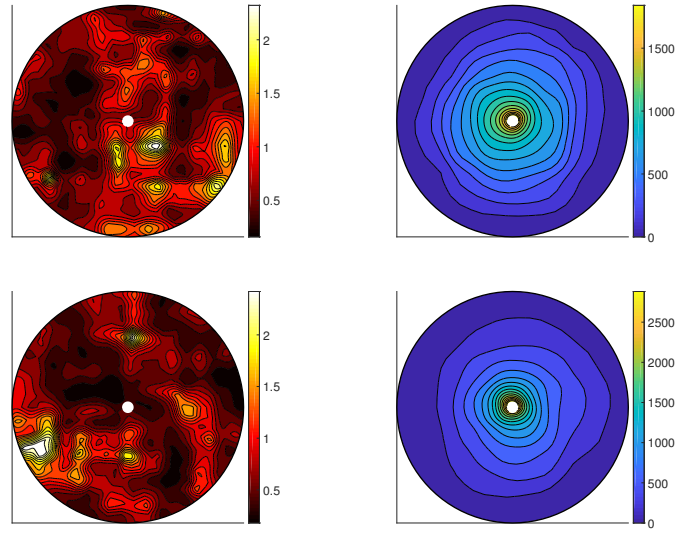


Figure 4.1: Two realizations of the log-permeability field (left) and the corresponding pressure field (right) using correlation length  $\ell = 1 \text{ mm}$ .

### 4.4.2 Spectral representation of the model output

We represent the QoI,  $p$ , using a truncated KL expansion

$$p(\mathbf{x}, \xi) \approx \bar{p}(\mathbf{x}) + \sum_{k=1}^N \sqrt{\lambda_k(C_f)} p_k(\xi) \phi_k(\mathbf{x}), \quad p_k(\xi) = \frac{1}{\sqrt{\lambda_k(C_f)}} \int_{\mathcal{D}} (p(\mathbf{x}, \xi) - \bar{p}(\mathbf{x})) \phi_k(\mathbf{x}) d\mathbf{x}. \quad (4.19)$$

Here  $\lambda_k(C_f)$ ,  $\phi_k(\mathbf{x})$  are the eigenvalues and corresponding eigenfunctions of the covariance operator  $C_f$  of  $p(\mathbf{x}, \xi)$ .

In Figure 4.2 (top left) we see that the eigenvalues of the covariance operator  $C_f$  show faster decay than those of the log-permeability field  $a(x, \xi)$ . In Figure 4.2 (top right) we compute the ratio  $\rho_k = (\sum_{k=1}^N \lambda_k(C_f)) / \text{Tr}(C_f)$ , where  $N$  is the number of KL modes retained. Using just  $N = 10$  KL modes gives us  $\rho_k \approx 0.9$  indicating that 90% of the variance is captured. If we add five more modes so that  $N = 15$  then we capture nearly 95% of the average variance in the model output  $p$ . Figure 4.2 (bottom) shows the first 40 eigenvalues of the covariance operator  $C_f$ , for a number of different sample sizes  $N_s$  used to approximate the covariance function of  $p$ . Note that using  $N_s = 300$  samples we can approximate the dominant eigenvalues reasonably well. In the current study we set  $N_p$  the input parameter dimension to be 200 and to retain  $N = 15$  KL modes. From the results that follow we will see that we can achieve significant dimension reduction for the input parameter.

## 4.5 Results and Discussion

### 4.5.1 Active subspace discovery and surrogate model construction

To construct the active subspace-based surrogate model for  $p(\mathbf{x}, \xi)$ , we need the gradients of the output KL modes defined in (4.19). For this, we use the adjoint method. The adjoint based expression for  $p_k(\xi)$  can be derived using a formal Lagrange approach; see, e.g., [41]. For a basic derivation of the gradient of the output KL modes see appendix A. The adjoint

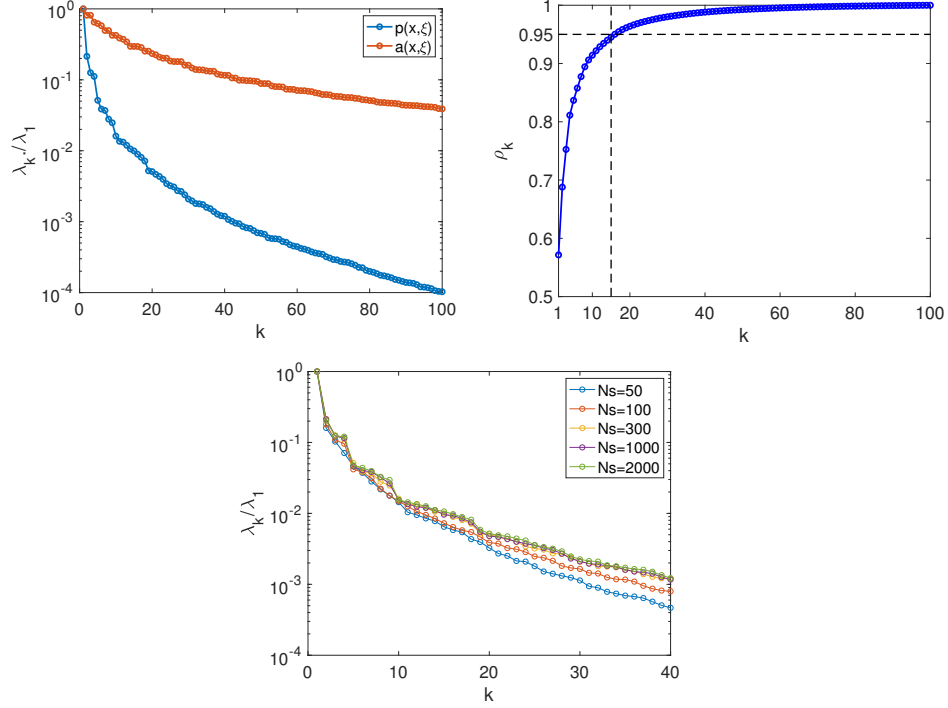


Figure 4.2: Eigenvalue spectrum of  $a(\mathbf{x}, \xi)$  versus  $p(\mathbf{x}, \xi)$  (top left). Ratio showing saturation of average variance for  $p(\mathbf{x}, \xi)$  (top right). First 40 eigenvalues of  $C_f$  for different sample sizes  $N_s$  (bottom).

based expression for the partial derivatives of  $p_k$ 's are given by,

$$\frac{\partial p_k}{\partial \xi_j} = \frac{\sqrt{\lambda_j(C_a)}}{\mu} \int_{\mathcal{D}} e_j(\mathbf{x}) e^{\hat{a}(\mathbf{x}, \xi)} \nabla p(\mathbf{x}) \cdot \nabla q(\mathbf{x}) d\mathbf{x}. \quad (4.20)$$

where  $p$  is the solution of the (forward) PDE (4.16) and  $q$  is the solution of the adjoint equation:

$$\begin{aligned} -\nabla \cdot \left( \frac{\kappa}{\mu} \nabla q \right) &= -\frac{1}{\sqrt{\lambda_k(C_f)}} \phi_k \quad \text{in } \mathcal{D}, \\ q &= 0 \quad \text{on } \Gamma_D, \\ \frac{\kappa}{\mu} \nabla q \cdot \mathbf{n} &= 0 \quad \text{on } \Gamma_N. \end{aligned} \quad (4.21)$$

Note that in the forward and adjoint equation, we let the permeability field be  $\kappa(\mathbf{x}, \xi) = e^{\hat{a}(\mathbf{x}, \xi)}$ .

Guided by the results in the previous subsection, we focus on the first  $N = 15$  output KL

modes. For each  $k \in \{1, \dots, N\}$ , we generate a sample  $\{\nabla p_k(\xi_i)\}_{i=1}^{N_s}$ , with  $N_s = 300$ . For  $\{\xi_i\}_{i=1}^{N_s}$ , we used the same set of 300 parameter samples used in computing the KL expansion of the output. Using (A.3),

$$\nabla p_k(\xi_i) = \frac{1}{\mu} \begin{bmatrix} \sqrt{\lambda_1(C_a)} \int_{\mathcal{D}} e_1(\mathbf{x}) e^{\hat{a}(\mathbf{x}, \xi_i)} \nabla p(\mathbf{x}, \xi_i) \cdot \nabla q(\mathbf{x}, \xi_i) d\mathbf{x} \\ \sqrt{\lambda_2(C_a)} \int_{\mathcal{D}} e_2(\mathbf{x}) e^{\hat{a}(\mathbf{x}, \xi_i)} \nabla p(\mathbf{x}, \xi_i) \cdot \nabla q(\mathbf{x}, \xi_i) d\mathbf{x} \\ \vdots \\ \sqrt{\lambda_{N_p}(C_a)} \int_{\mathcal{D}} e_{N_p}(\mathbf{x}) e^{\hat{a}(\mathbf{x}, \xi_i)} \nabla p(\mathbf{x}, \xi_i) \cdot \nabla q(\mathbf{x}, \xi_i) d\mathbf{x} \end{bmatrix}, \quad i = 1, \dots, N_s.$$

To compute these, we reuse the model evaluations  $\{p(\mathbf{x}, \xi_i)\}_{i=1}^{N_s}$ , from the computation of the output KL expansion earlier; the adjoint variables  $q(\cdot, \xi_i)$  are computed by solving the adjoint equation (A.1), with  $\kappa = e^{\hat{a}(\mathbf{x}, \xi_i)}$ ,  $i = 1, \dots, N_s$ .

Using the gradient samples, we approximate the matrix  $\mathbf{S}_k$  defined in (4.6) for each KL mode,  $k = 1, \dots, N$ :

$$\hat{\mathbf{S}}_k = \frac{1}{N_s} \sum_{i=1}^{N_s} \nabla p_k(\xi_i) \nabla p_k(\xi_i)^T.$$

In each case, we consider the corresponding spectral decomposition  $\hat{\mathbf{S}}_k = \mathbf{W}_k \hat{\Lambda}_k \mathbf{W}_k^T$ . To identify the active subspace and where to partition the eigenpairs we examine the spectrum of  $\hat{\mathbf{S}}_k$ . As an illustration, in Figure 4.3 (top) we present the spectrum for the first three output KL modes. Visually we observe a gap between the first and second eigenvalue for each of the modes indicating one-dimensional active subspaces. In Figure 4.3 (bottom), we show the corresponding sufficient summary plots (SSPs) for the corresponding modes. A sufficient summary plot here is a scatter plot of the active variables  $y = \mathbf{W}_{k,1}^T \xi$  versus the output KL modes  $p_k(\xi)$ . We observe a strong univariate trend which further indicates the presence of one-dimensional active subspaces. To provide a consistent truncation approach, we can use a threshold  $\gamma$  and choose the dimension  $r_k$  of the active subspace according to  $\lambda_{k,1}/\lambda_{k,r_{k+1}} > \gamma$ . In the current study we use  $\gamma = 10$ . Using this approach, we identified a one-dimensional active subspace for each of the first 15 output modes, except modes 11 and 12 where two-dimensional active subspaces were identified. To illustrate, we report the spectrum of  $\mathbf{S}_{12}$  and the SSP for  $p_{12}$  in Figure 4.4. Based on the determined values of  $r_k$ ,

we partition

$$\mathbf{\Lambda}_k = \begin{bmatrix} \mathbf{\Lambda}_{k,1} \\ \mathbf{\Lambda}_{k,2} \end{bmatrix}, \quad \mathbf{W}_k = [\mathbf{W}_{k,1} \quad \mathbf{W}_{k,2}], \quad k = 1, \dots, N.$$

$\mathbf{\Lambda}_{k,1}$  contains the dominant eigenvalues and  $\mathbf{W}_{k,1}$  the corresponding eigenvectors.

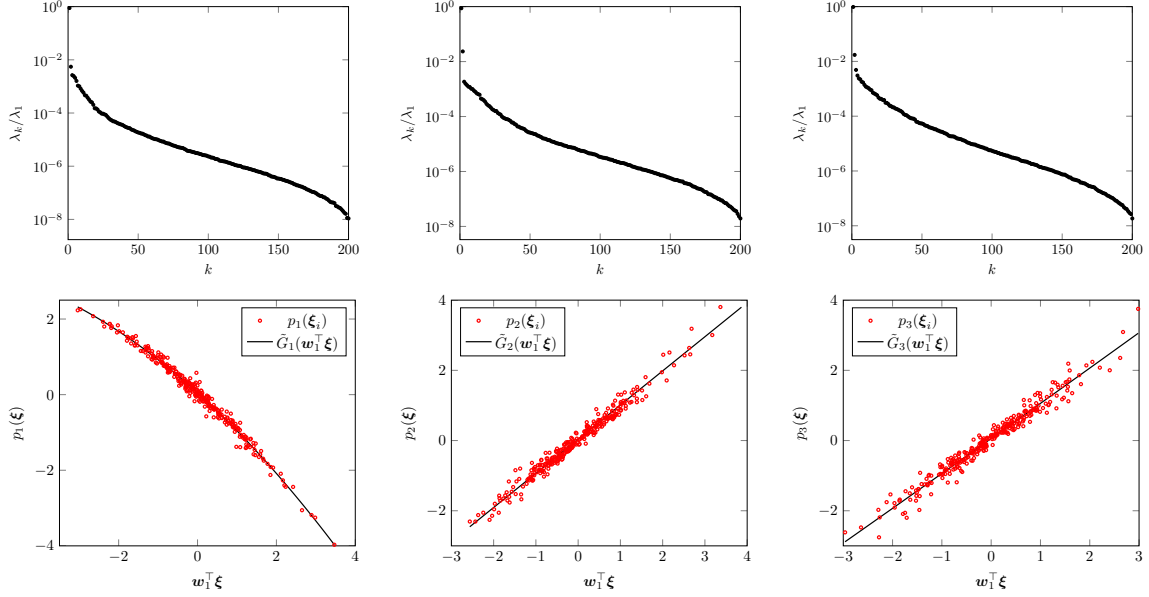


Figure 4.3: Top: eigenvalues of the matrix  $\hat{\mathbf{S}}_k$  for  $k = 1$  (left)  $k = 2$  (middle), and  $k = 3$  (right); bottom: SSPs for output KL modes  $p_1$  (left),  $p_2$  (middle), and  $p_3$  (right).

Recall that the active subspace approach essentially seeks “important linear combinations” of the input parameters. To illustrate this, in Figure 4.5, we present the components of the dominant eigenvector for the first three output KL modes  $p_1, p_2$  and  $p_3$ . The smaller inset plot shows the first 50 components of the dominant eigenvector for  $p_1, p_2$  and  $p_3$ . The magnitude of the components give us a sensitivity measure for each of the input parameters. A large component value indicates that that particular input is important in defining the direction of most variation in our function  $p$ . We note that the first output KL mode is sensitive to a few components of the input parameter vector. In contrast, the second and especially the third output KL modes show sensitivity to a larger number of components of  $\xi$ .



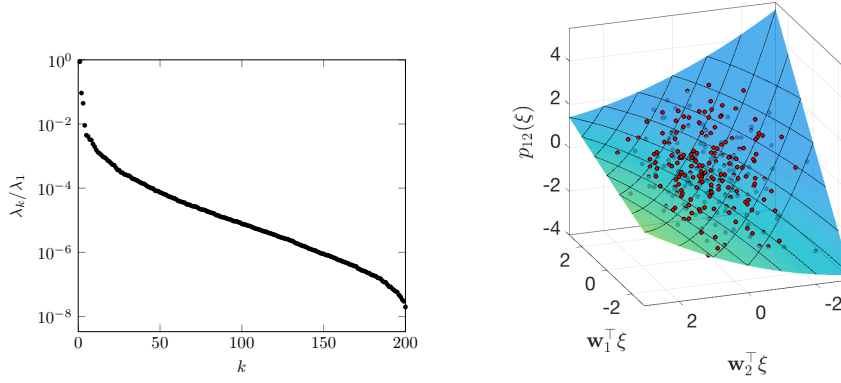


Figure 4.4: The normalized eigenvalues of output mode  $p_{12}$  (left) and the 2D sufficient summary plot corresponding to  $p_{12}$  (right). Note that the first two eigenvalues are very close to each other.

Finally, we note that the present results indicate a significant dimension reduction. The dominant output KL modes, each a function of 200 parameters, can be approximated in one or two dimensional active subspaces.

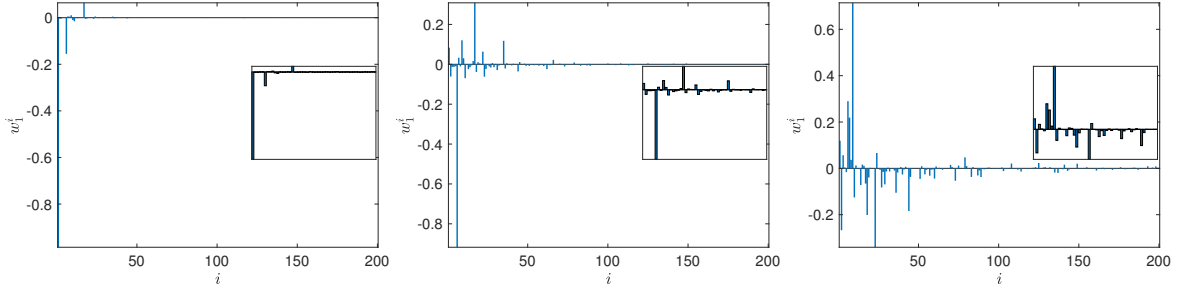


Figure 4.5: Components of the dominant eigenvector  $\mathbf{w}$  for the output KL modes,  $p_1$ ,  $p_2$ , and  $p_3$  (left, middle, and right images, respectively). The inset plot shows the first 50 components in each case.

Next, we compute the surrogate models for the output KL modes, following the strategy described in Section 3.3. The surrogate models  $\tilde{G}_k(\mathbf{W}_{k,1}^T \boldsymbol{\xi}) \approx p_k(\boldsymbol{\xi})$ ,  $k = 1, \dots, N$  are constructed by regression fit. Specifically, letting  $p_{i,k} = p_k(\boldsymbol{\xi}_i)$  and  $y_i = \mathbf{W}_{k,1}^T \boldsymbol{\xi}_i$ ,  $i = 1, \dots, N_s$ , we compute

least squares approximating polynomials  $\tilde{G}_k$  that minimize

$$\sum_{i=1}^{N_s} (\tilde{G}_k(y_i) - p_{i,k})^2, \quad k = 1, \dots, N.$$

In our computations, we found that linear regressions fits were suitable for the dominant output KL modes, except modes 1, 11, and 12, for which we used a quadratic fit. For illustration, we report the computed surrogate models for the first three output modes in Figure 4.3 (bottom) and for the mode  $p_{12}$  in Figure 4.4 (right). We now have all the pieces to form the surrogate model for the pressure field  $p(\mathbf{x}, \xi)$ :

$$p(\mathbf{x}, \xi) \approx \hat{p}(\mathbf{x}, \xi) = \bar{p}(\mathbf{x}) + \sum_{k=1}^N \sqrt{\lambda_k} \tilde{G}_k(\mathbf{W}_{k,1}^T \xi) \phi_k(\mathbf{x}). \quad (4.22)$$

Below, we examine the accuracy of the computed surrogate model and show its effectiveness in capturing the statistical properties of the pressure field.

### 4.5.2 The accuracy of the surrogate model

In this section, we provide various tests of accuracy that examine different aspects of the proposed method. To provide a baseline for comparison, we computed 10,000 realizations of the exact pressure field and its surrogate model approximation.

We begin by examining the success of the low-rank KL approximation of the pressure field in capturing the variance of the process. The variance of the pressure field can be obtained from  $\text{Var}(p(\mathbf{x}, \xi)) = \mathbb{E}\{p(\mathbf{x}, \xi)^2\} - \mathbb{E}\{p(\mathbf{x}, \xi)\}^2$ , which we approximate using the computed samples of  $p(\mathbf{x}, \xi)$ . The variance field for the truncated KL expansion of  $p$  is determined completely by the spectral decomposition of its (approximate) covariance operator:

$$\text{Var}(\hat{p}(\mathbf{x}, \xi)) = \sum_{k=1}^N \lambda_k \phi_k(\mathbf{x})^2,$$

where  $\lambda_k$  and  $\phi_k(\mathbf{x})$  are the eigenvalues and eigenvectors of the covariance operator  $C_p$ . Taking the square root we obtain the standard deviation of both the exact pressure field and its approximation; results are shown in Figure 4.6. We note that for both, the standard

deviation is highest at the center of the tumor and decreases to zero as we move away from the center. We also observe that even a low-rank approximation to  $p(\mathbf{x}, \xi)$  captures the standard deviation of the pressure field well.

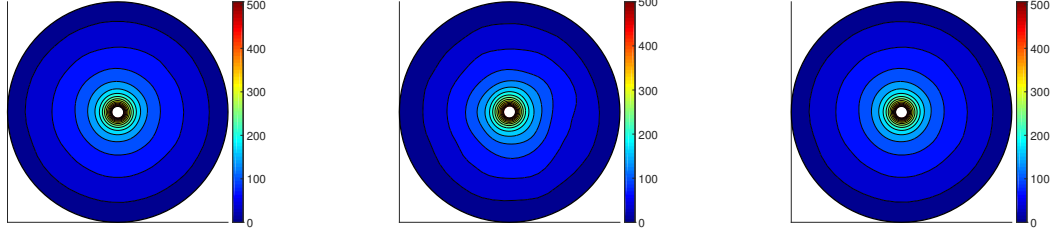


Figure 4.6: (Left) standard deviation of  $p(\mathbf{x}, \xi)$  using sampling, (middle) standard deviation of  $p(\mathbf{x}, \xi)$  using KLE (15 modes), and (right) standard deviation of  $p(\mathbf{x}, \xi)$  using KLE (50 modes).

Next, we study the accuracy of the computed surrogate model by focusing on the relative error indicator

$$E_{rel}(\xi) = \frac{\int_{\mathcal{D}} |p(\mathbf{x}, \xi) - \hat{p}(\mathbf{x}, \xi)| d\mathbf{x}}{\int_{\mathcal{D}} |p(\mathbf{x}, \xi)| d\mathbf{x}},$$

where  $\hat{p}(\mathbf{x}, \xi)$  is computed according to (4.22). In Figure 4.7 (left), we plot the expected value of  $E_{rel}$  as the number of output KL modes increases; the dashed lines indicate the fifth and ninety fifth percentiles. We note that with  $N = 15$  output KL modes, the relative error is about 5% on average. To better understand the behavior of the relative error, we report its distribution in Figure 4.7 (right), where we used  $N = 15$  output KL modes in computing  $\hat{p}$ . We also show the distribution of the relative error computed over a subdomain  $\mathcal{D}' = \{\mathbf{x} \in \mathcal{D} : \|\mathbf{x}\| \leq 2\}$  in Figure 4.7 (right). This is done to quantify the approximation errors near the injection site. We observe that the distribution of the error is shifted to the left, indicating smaller approximation errors in  $\mathcal{D}'$  (with high probability). To see more clearly how the distribution of the relative error evolves as the number of KL modes increase, we show the probability density function of  $E_{rel}$  corresponding to different number of output KL modes in Figure 4.8. Note that not only does the mode of the distribution get smaller, the spread of the error also decreases, which can be inferred from Figure 4.7 (left) as well.

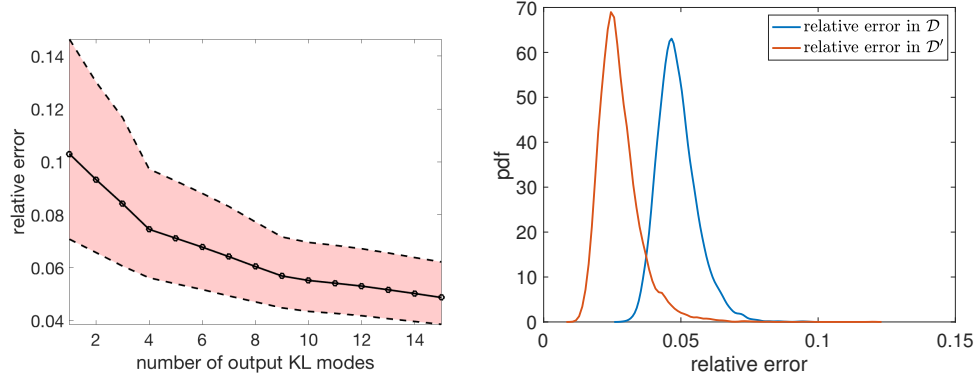


Figure 4.7: Left: The expected value of the error  $E_{rel}$  (solid black line) along with the fifth and the ninety fifth percentiles (dashed lines) of  $E_{rel}$ . Right: Distribution of  $E_{rel}$  computed over the entire domain  $\mathcal{D}$  (blue) and over the subdomain  $\mathcal{D}'$  (red).

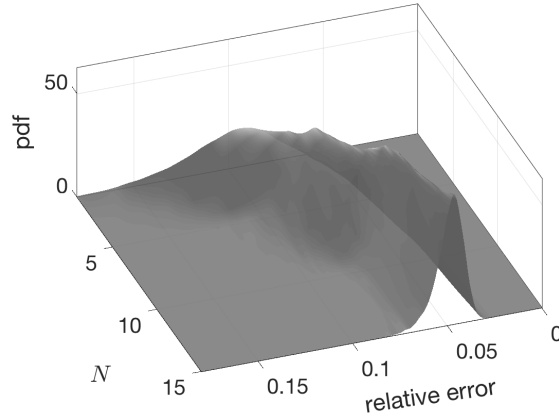


Figure 4.8: Distribution of the error  $E_{rel}$  as the number of output KL modes increase from  $N = 1$  to  $N = 15$ .

In Figure 4.9, we report the distribution of the average pressure along concentric circles of various radii. This shows that the computed surrogate captures statistical properties of the pressure well in different parts of the domain.

## 4.6 Conclusions

We have presented a distributed active subspace method for scalable surrogate modeling of PDE-governed physical processes with high-dimensional inputs and function valued outputs. To save the modeling efforts spent on function valued outputs, we employ the

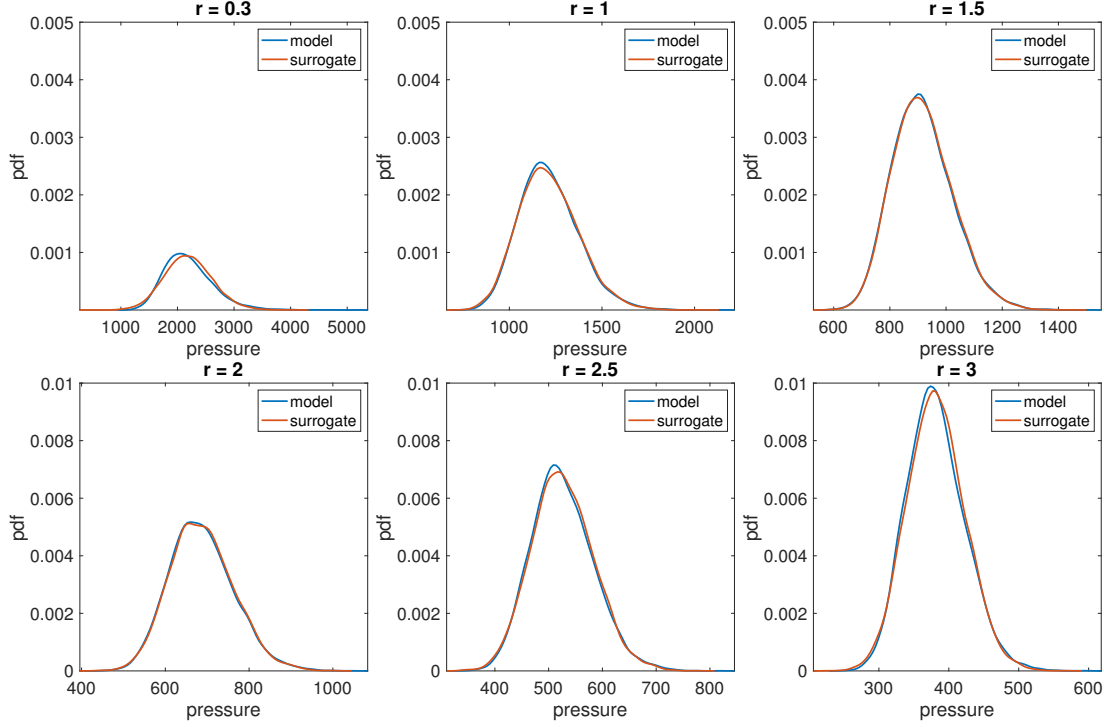


Figure 4.9: Distribution of the pressure averaged over concentric circles of various radii.

truncated KL expansion to decouple the spatial dimensions with those of the random variables. As a result, the randomness in outputs is fully represented by scalar valued KL modes. For elliptic PDEs, as observed in the work, a low-rank KL representation of the model output is usually sufficient for an accurate representation. To reduce the dimension of inputs, we construct active subspaces for each of the dominant output KL modes. Since output gradients with respect to the random variables need to be calculated when constructing active subspaces, we develop an adjoint-based framework to ensure that the computational cost does not scale with the input dimension. The method development is complemented by a rigorous mathematical formulation as well as theoretical analysis of errors due to active subspace projection and output KL representation.

We then deploy the distributed active subspace method to conduct surrogate modeling of the pressure field in a biotransport model in tumors with an uncertain permeability field. We demonstrate that a low-rank representation of the pressure field (i.e., output) can be achieved with the truncated KL expansion. The input (i.e., random variables used to represent the uncertain log-permeability field) dimension can be very high (e.g., several hundreds) when the correlation length of the log-permeability field is small. However,

we observe that dominant output KL modes admit low (one or two) dimensional active subspaces. We observe that the average relative error between the surrogate model and the PDE solution can be controlled under 5% even when a very low-rank representation of the outputs is employed. We also show that the surrogate models can capture statistical properties of the pressure well in different parts of the domain.

In future work, we will investigate extensions of the distributed active subspace method to surrogate modeling of the time-dependent diffusion and convection-diffusion processes in biological and geological flows. We envision that the truncated KL expansion can be used to extract spatiotemporal coherent structures from the physical process, and the low-dimensional active subspaces can then be constructed for the scalar valued KL modes associated with these structures. This surrogate modeling approach will contribute to cost-effective forward uncertainty quantification, and facilitate solution of inverse problems under uncertainty, such as Bayesian inversion of tissue material properties from medical images.

## CHAPTER

# 5

## FURTHER INSIGHTS ON ACTIVE SUBSPACES AND ACTIVITY SCORES

In section 5.1 we will extend the idea of activity scores, originally defined for scalar valued functions to models with function valued outputs. Activity scores can be used to approximate DGSMs which we will demonstrate through an application problem from biology. In section 5.2 of this chapter we will investigate the sensitivity of the active subspace method to correlations among input parameters. We would like to understand how these correlations might influence the dimension of our active subspace.

### **5.1 Activity scores for models with function-valued outputs**

In this section we extend the idea of activity scores, defined originally for scalar-valued QoIs, to the case of function-valued QoIs which we will call the functional activity scores. Activity scores provide an alternate point of view for derivative based GSA, for function valued QoIs.

We will use functional activity scores to approximate functional DGSMs defined in (5.3)[17] and apply it to a cholera model.

To briefly review, activity scores are sensitivity metrics that stem from active subspaces. Let  $\mathbf{W}_1$  and  $\mathbf{\Lambda}_1$  be as in (2.5). The activity score for the  $i$ th input variable is defined as:

$$\alpha_i(r) = \sum_{j=1}^r \lambda_j \langle e_i, w_j \rangle^2, \quad (5.1)$$

where  $r$  is the dimension of the active subspace and  $e_i$  is the  $i$ th coordinate vector in  $\mathbb{R}^{N_p}$ .

It is important to note that activity scores come for free, as a by product of active subspace analysis. No further model evaluations are needed, beyond the model evaluations used for the purposes of active subspace discovery.

### 5.1.1 Functional Activity Scores

We consider mathematical models of the form

$$y = f(\mathbf{x}, \xi), \quad (5.2)$$

where  $\mathbf{x}$  belongs to a compact set  $\mathcal{X} \subset \mathbb{R}^d$  with  $d = 1, 2$ , or  $3$ , and  $\xi$  is an element of an uncertain parameter space  $\Omega \subseteq \mathbb{R}^{N_p}$ . We have the following definition for a functional derivative based global sensitivity measure (DGSM) from [17]:

$$\mathbb{N}_j(f; \mathcal{X}) = \int_{\mathcal{X}} \int_{\Omega} \left( \frac{\partial f}{\partial \xi_j}(x, \xi) \right)^2 \pi(d\xi) dx = \int_{\mathcal{X}} v_j(f(x, \cdot)) dx, \quad (5.3)$$

where  $\pi$  is the law of the uncertain parameter vector  $\xi$  and

$$v_j(f) = \int_{\Omega} \left( \frac{\partial f}{\partial \xi_j} \right)^2 \pi(d\xi).$$



We will compute an approximation to the functional DGSMs using the idea of activity scores. We consider a discretization of the integral over  $\mathcal{X}$ :

$$\mathbb{N}_j^{(n)}(f; \mathcal{X}) = \sum_{i=1}^n w_i \int_{\Omega} \left( \frac{\partial f}{\partial \xi_j}(x_i, \xi) \right)^2 \pi(d\xi)$$

where  $n$  is the number of discretization points,  $x_i$  are the nodes and  $w_i$  are the weights,  $i = 1, \dots, n$ . We define the following symmetric positive semi-definite matrix  $N_p \times N_p$  matrix  $\mathbf{S}$ :

$$\mathbf{S} = \int_{\Omega} \mathbf{J}(\xi)^T \mathbf{W} \mathbf{J}(\xi) \pi(d\xi) \quad (5.4)$$

where  $J_{ij} = \frac{\partial f}{\partial \xi_j}(x_i, \xi)$  and  $\mathbf{W} = \text{diag}(w_1, \dots, w_n)$  are the associated weights. Since  $\mathbf{S}$  is symmetric positive semi-definite it has spectral decomposition:

$$\mathbf{S} = \sum_{k=1}^{N_p} \lambda_k \mathbf{u}_k \mathbf{u}_k^T, \quad (5.5)$$

where  $\lambda_k \geq 0$  are sorted in descending order and  $\mathbf{u}_k$  are the corresponding orthonormal eigenvectors. We can see that the functional DGSM is given by:

$$\mathbb{N}_j^{(n)}(f; \mathcal{X}) = \mathbf{e}_j^T \mathbf{S} \mathbf{e}_j \quad (5.6)$$

where  $\mathbf{e}_j$  is the  $j$ th coordinate vector in  $\mathbb{R}^{N_p}$ . Substituting in the spectral decomposition of  $\mathbf{S}$  we have:

$$\mathbb{N}_j^{(n)}(f; \mathcal{X}) = \mathbf{e}_j^T \sum_{k=1}^{N_p} \lambda_k \mathbf{u}_k \mathbf{u}_k^T \mathbf{e}_j = \sum_{k=1}^{N_p} \lambda_k \langle \mathbf{e}_j, \mathbf{u}_k \rangle^2 \quad (5.7)$$

Truncating the summation up to  $r$  with  $1 \leq r \leq N_p$  we define the approximation:

$$\mathbf{a}_j(f; r) = \sum_{k=1}^r \lambda_k \langle \mathbf{e}_j, \mathbf{u}_k \rangle^2.$$

We call these the *functional activity scores*. The following result, quantifies the error in approximating functional DGSMs with the functional activity scores.

**Proposition 5.1.1.** For  $1 \leq r < N_p$ ,

$$0 \leq \mathbb{N}_j^{(n)}(f; \mathcal{X}) - \mathbf{a}_j(f; r) \leq \lambda_{r+1}, \quad i = 1, \dots, N_p.$$

*Proof.* Note that,  $\mathbb{N}_j^{(n)}(f; \mathcal{X}) - \mathfrak{a}_j(f; r) = \sum_{k=r+1}^{N_p} \lambda_k \langle \mathbf{e}_j, \mathbf{u}_k \rangle^2 \geq 0$ , which gives the first inequality. For the second inequality note,

$$\sum_{k=r+1}^{N_p} \lambda_k \langle \mathbf{e}_j, \mathbf{u}_k \rangle^2 \leq \lambda_{r+1} \sum_{k=r+1}^{N_p} \langle \mathbf{e}_j, \mathbf{u}_k \rangle^2 \leq \lambda_{r+1}.$$

The last inequality holds because  $1 = \|\mathbf{e}_j\|_2^2 = \sum_{k=1}^{N_p} \langle \mathbf{e}_j, \mathbf{u}_k \rangle^2 \geq \sum_{k=r+1}^{N_p} \langle \mathbf{e}_j, \mathbf{u}_k \rangle^2$ .  $\square$

We note that  $\mathbb{N}_j^{(n)}(f; \mathcal{X}) \approx \mathfrak{a}_j(f; r)$  in problems where the eigenvalues decay rapidly to zero.

### 5.1.2 Results & Discussion

Here we demonstrate the use of functional activity scores to approximate the functional DGSMs for a cholera model [45], [4]. We consider a population of  $N_{\text{pop}}$  individuals who are split into three groups: susceptible  $S$ , infectious  $I$ , and recovered individuals  $R$ . We also consider the concentrations of highly-infectious bacteria,  $B_H$  and low infectious bacteria,  $B_L$ . The quantities  $B_H$  and  $B_L$  are measured in cells per milliliter. The following non-linear system of ODEs, developed in [45], represents the the time-evolution of the state variables.

$$\begin{aligned} \frac{dS}{dt} &= bN_{\text{pop}} - \beta_L S \frac{B_L}{\kappa_L + B_L} - \beta_H S \frac{B_H}{\kappa_H + B_H} - bS, \\ \frac{dI}{dt} &= \beta_L S \frac{B_L}{\kappa_L + B_L} + \beta_H S \frac{B_H}{\kappa_H + B_H} - (\gamma + b)I, \\ \frac{dR}{dt} &= \gamma I - bR, \\ \frac{dB_H}{dt} &= \zeta I - \chi B_H, \\ \frac{dB_L}{dt} &= \chi B_H - \delta B_L, \end{aligned} \tag{5.8}$$

with initial conditions  $(S(0), I(0), R(0), B_H(0), B_L(0)) = (S_0, I_0, R_0, B_{H_0}, B_{L_0})$ . We use the parameter units and nominal values from [45] which are summarized in table 5.1.

We consider a total population of  $N_{\text{pop}} = 10,000$  and let the initial states be as follows:  $S_0 = N_{\text{pop}} - 1$ ,  $I_0 = 1$ ,  $R_0 = 0$ , and  $B_{H_0} = B_{L_0} = 0$ . Figure 5.1 shows the number of cases of cholera as a function of time. We see that the number of people infected with cholera peaks

Table 5.1: Cholera model parameters from [4, 45].

Model Parameter	Symbol	Units	Values
Rate of drinking $B_L$ cholera	$\beta_L$	$\frac{1}{\text{week}}$	1.5
Rate of drinking $B_H$ cholera	$\beta_H$	$\frac{1}{\text{week}}$	7.5
$B_L$ cholera carrying capacity	$\kappa_L$	$\frac{\# \text{ bacteria}}{\text{ml}}$	$10^6$
$B_H$ cholera carrying capacity	$\kappa_H$	$\frac{\# \text{ bacteria}}{\text{ml}}$	$\frac{\kappa_L}{700}$
Human birth and death rate	$b$	$\frac{1}{\text{week}}$	$\frac{1}{1560}$
Rate of decay from $B_H$ to $B_L$	$\chi$	$\frac{1}{\text{week}}$	$\frac{168}{5}$
Rate at which infectious individuals spread $B_H$ bacteria to water	$\zeta$	$\frac{\# \text{ bacteria}}{\# \text{ individuals} \cdot \text{ml} \cdot \text{week}}$	70
Death rate of $B_L$ cholera	$\delta$	$\frac{1}{\text{week}}$	$\frac{7}{30}$
Rate of recovery from cholera	$\gamma$	$\frac{1}{\text{week}}$	$\frac{7}{5}$

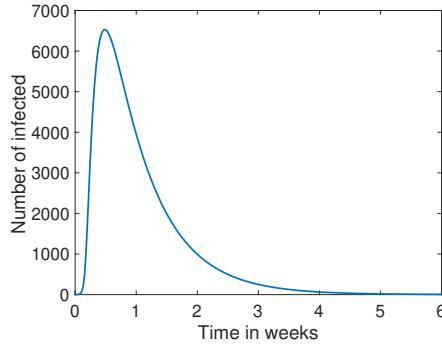


Figure 5.1: Cholera model: Infected Population

at approximately 6,500 people less than 1 week after exposure to the disease. That number declines rapidly to near 0 approximately 5 weeks later.

We computed the matrix  $\mathbf{S}$  using Monte Carlo (MC) sample sizes of  $10^4$ . In figure 5.2 we computed the functional activity scores at various truncation levels  $r = 1, 2$  and 3. With  $r = 2$  we accurately capture the magnitude of the actual functional DGSMs computed as in [17]. These results are consistent with those reported in [4, 17]. Using  $r = 1$  fails to capture the magnitude but it does successfully identify important parameters from unimportant ones. Using a cut-off of 0.05 we determined that parameters 1, 4, 5, 7 are unimportant while 2, 3, 6, 8 would be considered the important parameters.

In [116] the authors use the matrix constructed in (5.4) to perform dimension reduction

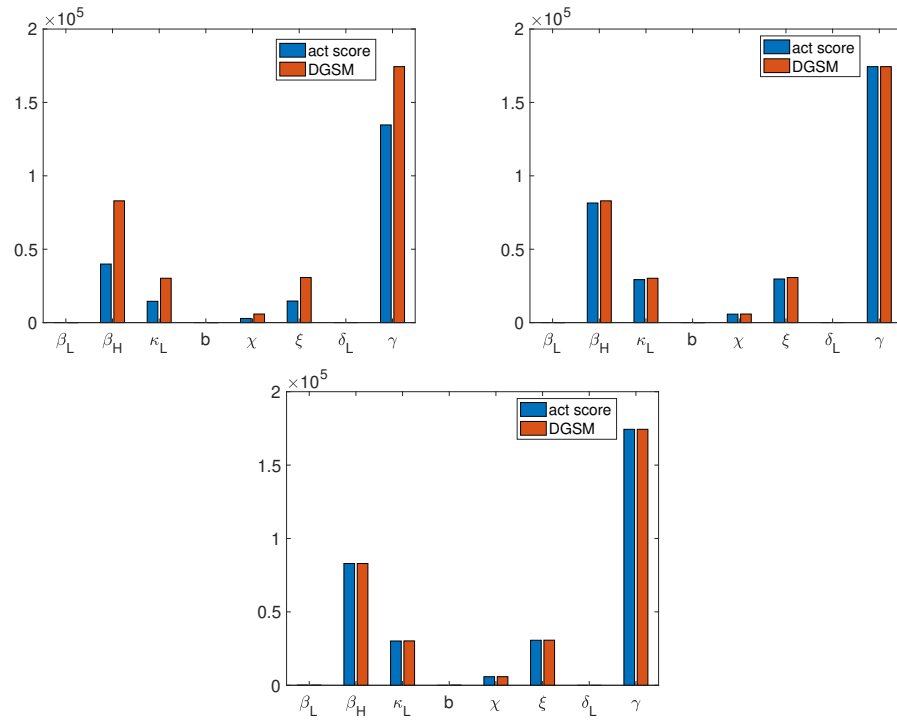


Figure 5.2: The functional activity scores and functional DGSMs at three different truncation levels,  $r = 1$  (top left),  $r = 2$  (top right) and  $r = 3$  (bottom).

on the input parameter space. Their approach is to use a ridge function to approximate the original function and then construct an upper bound for the error between the two. Since the matrix  $\mathbf{S}$  (5.4) has already been constructed we note as a post-processing step one could calculate the functional activity scores for free.

In this section we have discussed how to extend activity scores to function valued QoIs. We have illustrated using a cholera model how functional activity scores can be used to screen for unimportant parameters. We have also shown that one can use the functional activity scores to accurately approximate the functional DGSMs.

## 5.2 Impact of parameter correlations on active subspaces

The active subspace method does not use any specific assumptions on the correlation structure of the input parameters. In our work so far, the input parameters have been assumed to be uncorrelated. In particular, this assumption was justified in Chapter 4, where the input parameter vector consisted of the coefficients in the KL expansion of the input, which are independent standard normal random variables by construction. Here we consider two different problems, a simple quadratic model and a cholera model governed by a nonlinear ODE system with correlations in the input parameters. We investigate the impact of correlations on active subspace calculations and find that the impact of correlations is minimal.

### 5.2.1 Quadratic Model

We consider a vector  $\xi \in \mathbb{R}^{N_p}$  of uncertain input parameters with  $\xi \sim N(0, \Sigma)$ . We use the following quadratic function for our investigations:

$$f(\xi) = \frac{1}{2} \xi^T \mathbf{A} \xi, \quad (5.9)$$

where  $\mathbf{A} \in \mathbb{R}^{N_p}$  is a symmetric positive definite matrix. To construct this matrix  $\mathbf{A}$  we first construct a square matrix  $\mathbf{B} \in \mathbb{R}^{N_p \times N_p}$  of normally distributed random variables. Taking a QR factorization of  $\mathbf{B}$  we obtain an orthogonal matrix  $\mathbf{Q}$  and an upper triangular matrix

**R.** We use  $\mathbf{Q}$  to form our matrix  $\mathbf{A}$  via  $\mathbf{A} = \mathbf{Q}\mathbf{D}\mathbf{Q}^T$  with  $\mathbf{D}$  a diagonal matrix with the desired eigenvalues of  $\mathbf{A}$ . The gradient of this function  $\nabla f = \mathbf{A}\xi$  is used to form the matrix  $\mathbf{C}$  for active subspace construction:

$$\mathbf{C} = \int (\nabla f)(\nabla f)^T \pi d\xi. \quad (5.10)$$

We can compute the matrix  $\mathbf{C}$  analytically as follows:

$$\mathbf{C} = E[\mathbf{A}\xi(\mathbf{A}\xi)^T] = E[\mathbf{A}\xi\xi^T\mathbf{A}] = \mathbf{A}E[\xi\xi^T]\mathbf{A} = \mathbf{A}\Sigma\mathbf{A}, \quad (5.11)$$

where  $\Sigma$  is the covariance matrix. For the purposes of this study we chose  $N_p = 5$  and  $\Sigma$  to be:

$$\Sigma = \begin{bmatrix} 1 & 0.5\rho & 0.5\rho & 0 & 0.8\rho \\ 0.5\rho & 1 & 0 & 0 & 0 \\ 0.5\rho & 0 & 1 & 0 & 0.3\rho \\ 0 & 0 & 0 & 1 & 0 \\ 0.8\rho & 0 & 0.3\rho & 0 & 1 \end{bmatrix}, \quad 0 \leq \rho \leq 1. \quad (5.12)$$

This choice of  $\Sigma$  was motivated by the study in [44]. We compute the eigendecomposition  $\mathbf{C} = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^T$ , where  $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_{N_p})$  and  $\mathbf{W}$  contains the orthonormal eigenvectors of  $\mathbf{C}$ . We computed this eigendecomposition for various values of  $\rho$  to investigate the effect of correlations between input parameters.

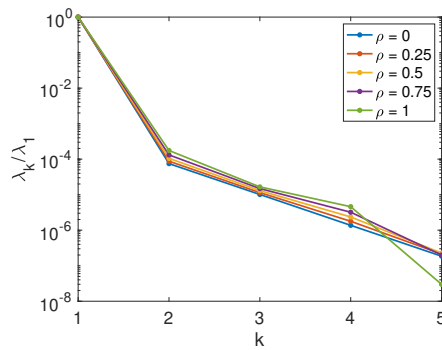


Figure 5.3: Comparison of eigenvalues of  $\mathbf{C}$  at different  $\rho$ .

In figure 5.3 we compare the eigenvalues for different values of  $\rho$  ranging from zero to 1. We can see that correlation has very little effect on the eigenvalues and no effect on the dimension of the active subspace indicating the method is not sensitive to input parameter correlations. Since we have identified that the dimension of the active subspace has not changed, stayed dimension one, we projected the original parameter vector  $\xi$  onto the active subspace  $y = \mathbf{W}_1^T \xi$  where  $\mathbf{W}_1$  contains one eigenvector corresponding to the largest eigenvalue. From there we constructed a surrogate model approximation  $f(\xi) \approx \hat{f}(\mathbf{W}_1^T \xi)$  using a linear regression fit for each  $\rho$ .

In order to get a quantitative study on the impact  $\rho$  has on the eigenvalues we look at the eigenvalue derivatives [62] given by:

$$\frac{d\lambda_k}{d\rho} = \mathbf{w}_k^T \frac{d\mathbf{A}}{d\rho} \mathbf{w}_k, \quad k = 1, \dots, N_p \quad (5.13)$$

where the  $\mathbf{w}_k$ 's are the columns of  $\mathbf{W}$ . In Figure 5.4 (left) we observe that the derivatives are much smaller in comparison to the eigenvalues. For the eigenvalue of interest, the largest one, the derivative looks to be almost constant as the value of  $\rho$  changes.

Next, we calculated the angles between the eigenvectors given by:

$$\theta = \arccos(\mathbf{w}_{k,0} \cdot \mathbf{w}_{k,\rho}), \quad k = 1, \dots, N_p \quad (5.14)$$

where  $\mathbf{w}_{k,0}$  are the columns of  $\mathbf{W}$  with  $\rho = 0$ . In Figure 5.4 (right) we plot these angles for increasing values of  $\rho$ . We observe that the angle for the dominant mode is near zero for all  $\rho$ . The angles increase for higher order modes but since we have a 1D active subspace we are only interested in the dominant mode anyway.

Finally we looked at the relative error between the the model  $f$  and its surrogate model approximation  $\hat{f}$ . From figure 5.5 we observe as the value of  $\rho$  increases the relative error also increases. Although we see an increase, we note that the relative error remains small.

## 5.2.2 Cholera Model

Here we investigate the effects of correlation between parameters for the cholera model described in section 5.1.2. For the purposes of this study we focus on the first week of

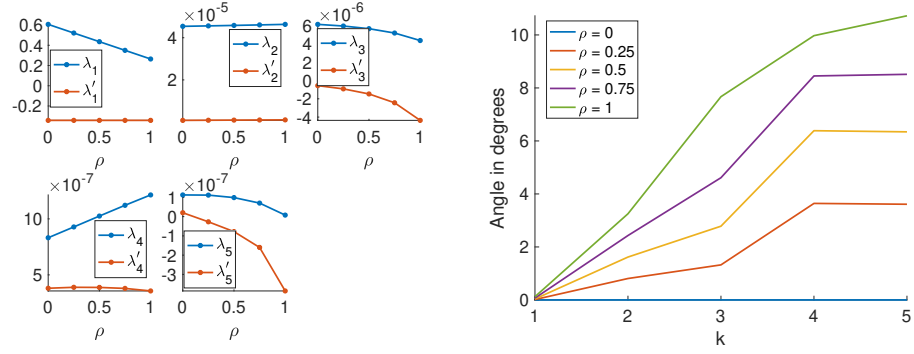


Figure 5.4: Angle between  $w_{k,0}$  and  $w_{k,\rho}$  for various  $\rho$  (left) and eigenvalues with eigenvalue derivatives with respect to  $\rho$  for various  $\rho$  (right).

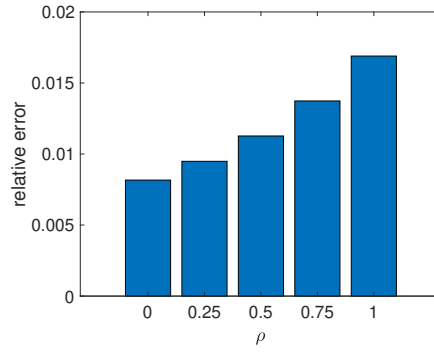


Figure 5.5: Relative error between the quadratic model and its surrogate approximation at various  $\rho$ .



exposure to cholera disease. Our quantity of interest (QoI) here is:

$$f(\xi) = \int_0^1 I(t, \xi) dt. \quad (5.15)$$

We solve the problem up to time  $t = 1$  using the ode45 solver provided in MATLAB. We consider correlations between  $\beta_L$ , rate of drinking low infectious cholera  $B_L$  and  $\kappa_L$ , the  $B_L$  cholera carrying capacity. We used a gaussian copula to introduce dependence between these two parameters.

As in section 5.2.1 to construct the active subspace we compute the matrix  $\mathbf{C}$  for each value of  $\rho$  and its eigendecomposition  $\mathbf{C} = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^T$ , where  $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_{N_p})$  and  $\mathbf{W}$  contains the orthonormal eigenvectors of  $\mathbf{C}$ .

In figure 5.6 (left) we see the eigenvalues of  $\mathbf{C}$  for each  $\rho$ . As with the quadratic model we see no discernable difference as we increase the correlation between the parameters. There appears to be a gap between the 1st and 2nd eigenvalue indicating the presence of 1-dimensional active subspace. The SSP plots in figure 5.6 (right) confirm that we have identified 1-dimensional active subspaces for each  $\rho$ . We see that each of the plots show a strong linear univariate trend. Since the shape is linear we choose to use a linear regression model as our surrogates.

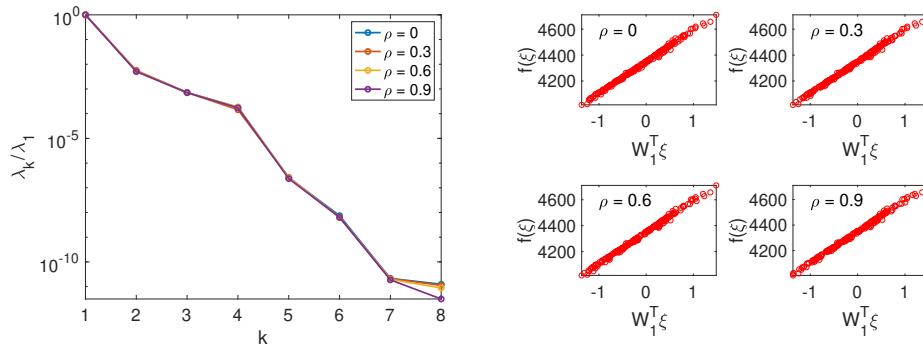


Figure 5.6: Comparison of eigenvalues of  $\mathbf{C}$  at different  $\rho$  (left) and SSPs for cholera model at various  $\rho$  (right).

In figure 5.7 (left) we plot the angle between the dominant eigenvectors for increasing values of  $\rho$ . We observe that the angle for the dominant mode remains close to zero for each value of  $\rho$ . We also report the activity scores for each of the parameters in the cholera model in figure 5.7 (right). We recall that activity scores are defined by the formula:

$$v_{i,r}(f) = \sum_{k=1}^r \lambda_k \langle \mathbf{e}_i, \mathbf{w}_k \rangle^2, \quad i = 1, \dots, N_p, \quad r \leq N_p.$$

where  $r$  is the dimension of the active subspace,  $N_p$  the dimension of our input parameter space,  $\lambda_k$ ,  $\mathbf{w}_k$  are the eigenvalues and corresponding eigenvectors of  $\mathbf{C}$  and  $\mathbf{e}_i$  is the  $i$ th coordinate vector in  $\mathbb{R}^{N_p}$ . Since we identified a 1-dimensional active subspace regardless of the value of  $\rho$  the formula for activity scores reduces to:

$$v_{i,1}(f) = \lambda_1 \langle \mathbf{e}_i, \mathbf{w}_1 \rangle^2, \quad i = 1, \dots, N_p.$$

where  $N_p = 8$  since we have 8 input parameters for the cholera model. We note that the 8th parameter,  $\gamma$ , rate of recovery from cholera, is identified as the most influential parameter in our space of inputs. We study the accuracy of the computed surrogate model by focusing

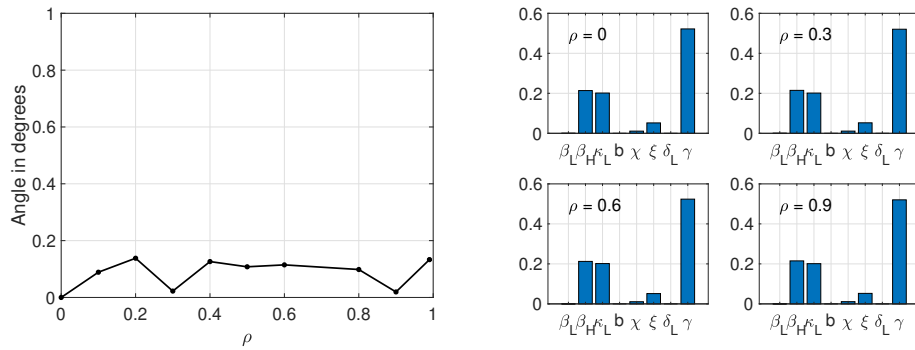


Figure 5.7: Angle between  $w_{1,0}$  and  $w_{1,\rho}$  for various  $\rho$  (left) and activity scores for cholera model at various values of  $\rho$  (right).

on the relative  $L^2$  error indicator

$$E = \frac{\|f - \tilde{f}\|_2}{\|f\|_2} \quad (5.16)$$

where  $\tilde{f}$  is our linear surrogate model. In figure 5.8 (left) we note that the relative  $L^2$  error is very small and does not vary much as the value of  $\rho$  changes. Finally we look at the pdf of the surrogate model with actual model evaluations for each  $\rho$ . As shown in figure 5.8 (right) the surrogate model does well at capturing the overall distribution of the model.

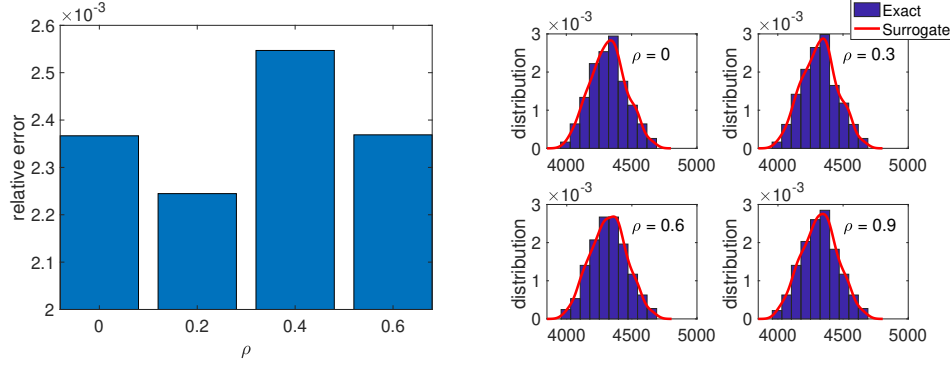


Figure 5.8: Relative  $L^2$  error at various values of  $\rho$  (left) and distribution of surrogate model with actual cholera model (right).

In this section we have considered correlations between input parameters. We have observed in the two examples, a simple quadratic model and a cholera model, that introducing dependence between parameters has had little influence on the active subspace method. We did notice some difference between the angles of the higher order eigenvectors in figure 5.4 but the dimension of the active subspace and the overall approximation were not affected. We note however that in certain cases correlations will have an impact. The following example illustrates that introducing correlations between input parameters can actually help us identify a lower dimensional active subspace.

Consider a vector  $\xi \in \mathbb{R}^3$  with  $\xi \sim N(0, \Sigma)$  and the quadratic function from section 5.2.1. As in section 5.2.1, we can construct the matrix  $\mathbf{C}$  in (5.10) analytically:  $\mathbf{C} = \mathbf{A}\Sigma\mathbf{A}$ . In this

example, we let  $\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 11 \end{bmatrix}$  and  $\Sigma = \begin{bmatrix} 1 & \rho & \rho \\ \rho & 1 & \rho \\ \rho & \rho & 1 \end{bmatrix}$  with  $0 \leq \rho \leq 1$ .

In table 5.2 we present the eigenvalues for different values of  $\rho$ . We note that when  $\rho = 0$  the values of the eigenvalues indicate we have a 2D active subspace since we have a significant

Table 5.2: Eigenvalues at different values of  $\rho$ .

Eigenvalues	$\rho = 0$	$\rho = 0.8$	$\rho = 0.9$	$\rho = 1$
$\lambda_1$	121.00	199.84	210.91	222
$\lambda_2$	100.00	21.88	10.95	0.00
$\lambda_3$	1.00	0.29	0.15	0.00

gap between the 2nd and 3rd eigenvalue. However, as  $\rho$  approaches 1, clearly we approach a 1D active subspace. This simple example illustrates that correlations between inputs can have an influence on the dimension of the active subspace method and can be useful in further reducing the dimension of the input parameter space.

## CHAPTER

# 6

# DEEP LEARNING FOR EFFICIENT IMAGE CLASSIFICATION: APPLICATION TO FAULT DETECTION

## 6.1 Introduction

In this chapter we will use convolutional neural networks (CNNs) to perform image classification. The application problem outlined below focuses on developing a machine learning approach using CNNs. CNNs will be used to detect faults in steel canisters containing spent nuclear fuel. The CNN classifier is able to detect corrosion and pitting with accuracies of approximately 96%. The approach will assist with remote and real-time detection, interpretation and prediction of corrosion in welded stainless-steel canisters storing spent nuclear fuel and high-level radioactive waste. The framework would allow scanning a larger area, and hence larger data sets, faster than currently possible. The capabilities of the approach

would be a big step forward to reduce radiation exposure to operators, minimize inspection costs, and ensure long-term safety of spent nuclear fuel canisters.

In section 6.2 we give a brief overview of neural networks and in section 6.3 we provide background material on CNNs, a particular type of neural network used in image classification problems. Section 6.4 discusses the application problem in detail. We provide details on how the dataset was generated and how we trained and validated the CNN classifier. We provide performance metrics and results from testing the classifier on unseen data. Finally, in section 6.5 we provide some closing remarks on the machine learning approach and highlight some areas for future work.

The work in this chapter was made possible in part by the Artificial Intelligence Summer Institute, which is part of Oak Ridge National Laboratory's AI Initiative, and is facilitated through the Oak Ridge Institute for Science and Education. Oak Ridge National Laboratory is a multiprogram Department of Energy Laboratory operated by UT-Battelle.

## 6.2 Neural Networks

Neural networks are complex computational models that are structured in a way that mimic the human brain. From a numerical analysis point of view, the models are used to approximate complicated input-output relations. Neural networks are designed to complete specific tasks including clustering, classification and prediction. Neural networks thus provide a form of surrogate modeling approach suited for complex input-output relations that are not amenable to traditional approaches.

Each artificial neuron is a function that takes a set of inputs and maps them to a single output. The output from each artificial neuron is represented as follows:

$$y = \phi \left( \sum_{i=1}^n w_i x_i + b \right), \quad (6.1)$$

where  $\phi$  the activation function,  $w_i$  are the connection weights,  $x_i$  the input data and  $b$  the bias. In figure 6.1 we show a schematic representation of an artificial neuron.

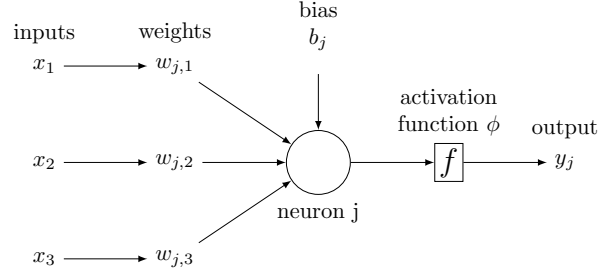


Figure 6.1: schematic representation of an artificial neuron.

Neural networks typically contain a large number of these artificial neurons arranged in a series of fully connected layers; that is, each neuron of a layer is linked to all the neurons of the next layer but has no link with the neurons of the same layer, see Figure 6.2.

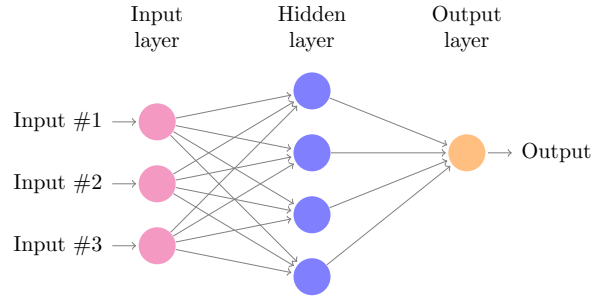


Figure 6.2: A simple neural network with one hidden layer.

Given a set of inputs  $\mathcal{X}$ , a set of outputs  $\mathcal{Y}$  and a function  $f : \mathcal{X} \rightarrow \mathcal{Y}$  which maps the set of inputs  $\mathcal{X}$  to the set of outputs  $\mathcal{Y}$ , the goal is to use a surrogate model to approximate the function  $f$ . Given a finite data set  $\{(x_i, y_i)\}_{i=1}^N$  with  $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$  and a set  $H$  of hypothesis functions, we want to compute:

$$h_{min} = \arg \min_{h \in H} \frac{1}{N} \sum_{i=1}^N (h(x_i) - y_i)^2 \quad (6.2)$$

such that  $h_{min} \approx f$  ; see [1].

The number of parameters in a neural network increase with the number of layers in the network. A challenge with networks with multiple layers is the amount of time required to successfully train the network. In computer vision, for example, it would take a huge amount of weights and biases to characterize the network because images themselves are high-dimensional vectors. To illustrate, the number of weights in a fully connected layer with 100 neurons for a 64x64x3 image is over 1 million, which makes traditional neural networks an infeasible option for this type of application.

A variant on the neural network is the convolutional neural network (CNN), a popular method for large data problems in image processing/analysis. The main difference of a CNN and a traditional neural network is that the initial input matrix is reduced through an operation called convolution. [118] [53] [47] [95] [36] [117]

In our work, we focus on using convolutional neural networks for image classification, assigning a label to an input image from a fixed set of categories.

## 6.3 Convolutional neural networks

The structure of a CNN is as follows: first, we have a convolution layer, followed by an activation layer, and then a pooling layer. The purpose of these three layers is to reduce the number of parameters to be learned during the training stage. The output of the pooling layer is sent to a traditional fully connected neural net for final classification. A description of each of the layers is outlined below. In Figure 6.3 we present an overview of the structure of a basic CNN.

In most applications the network will consist of many convolution, pooling and fully connected layers. Due to the dimension reduction in the convolution and pooling layers, CNNs are a popular method for large data problems in image processing/analysis. Using the traditional neural network would be prohibitively expensive for such application problems.

### Convolution Layer

The convolution involves passing a kernel or filter over an image multiple times to produce a feature map. Given an  $m \times n$  filter  $\mathbf{K}$  and an  $s \times t$  image  $\mathbf{I}$ , the convolution of  $\mathbf{I}$  with  $\mathbf{K}$  is



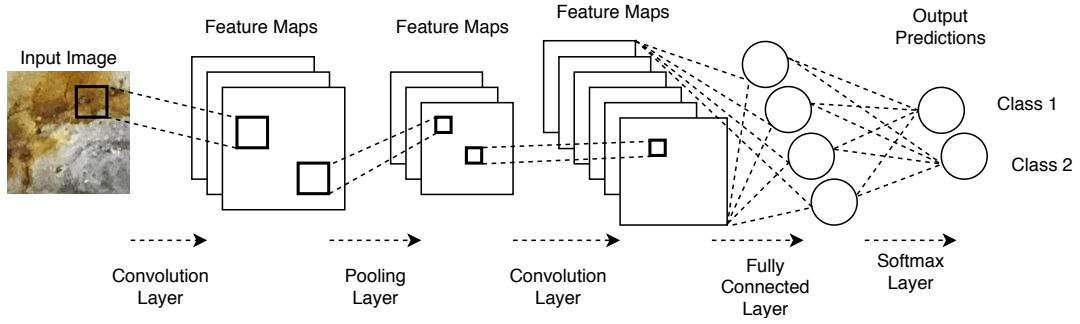


Figure 6.3: Basic structure of a CNN.

given by the following:

$$\mathbf{F}_{i,j} = (I * K)_{i,j} = \sum_{k=1}^m \sum_{l=1}^n K_{k,l} I_{i+k-1,j+l-1}, \quad i = 1, \dots, s-m+1, j = 1, \dots, t-n+1. \quad (6.3)$$

The convolution process is demonstrated in Figure 6.4 on a small example. For example to

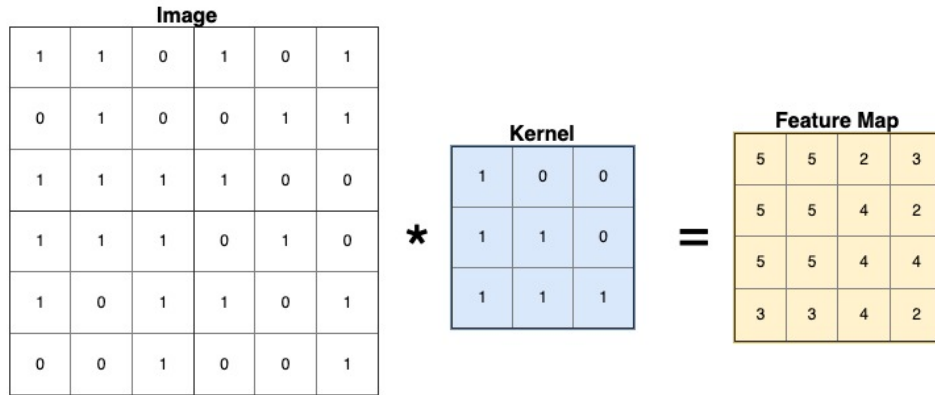


Figure 6.4: An example of convolving a kernel with an input image to produce a feature map.

get the first entry in the feature map in figure 6.4 we do the following:

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} = 1(1) + 1(0) + 0(0) + 0(1) + 1(1) + 0(0) + 1(1) + 1(1) + 1(1) = 5 \quad (6.4)$$

The first convolutional layers in the network capture low level features of an image such as edges and color. Subsequent layers learn more high level features such as objects and labels [7] .

### Activation Layer

After each convolution layer, we apply activation layers to the feature map obtained through convolution. The purpose of this layer is to help the model account for interaction effects and nonlinear effects. There are many nonlinear functions including tanh and sigmoid functions, but the most popular is the Rectified Linear Unit (ReLU):

$$\text{ReLU}(x) = \begin{cases} 0 & \text{if } x < 0, \\ x & \text{otherwise.} \end{cases} \quad (6.5)$$

Figure 6.5 shows plots of these activation functions.

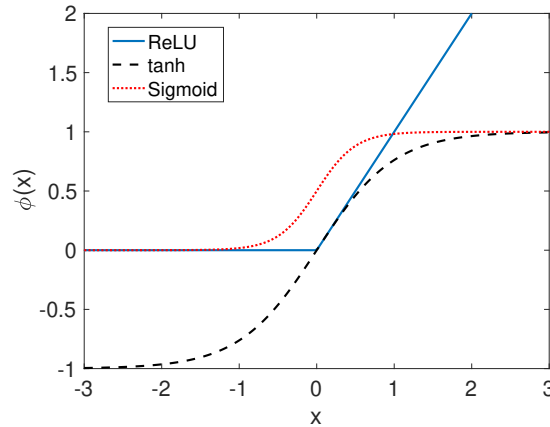


Figure 6.5: Examples of several activation functions.

### Pooling Layer

The feature map obtained in the convolution and activation layers is passed through a pooling layer where the dimension of the map is reduced further. The pooling layer reduces the number of parameters the network needs to learn and reduces computation time. Two

common functions used in the pooling operation are: average pooling, where the average value for each patch (usually  $2 \times 2$  pixels) on the feature map is calculated, and maximum pooling, where the maximum value for each patch of the feature map is calculated. We use max pooling in our work. This approach is demonstrated in figure 6.6.

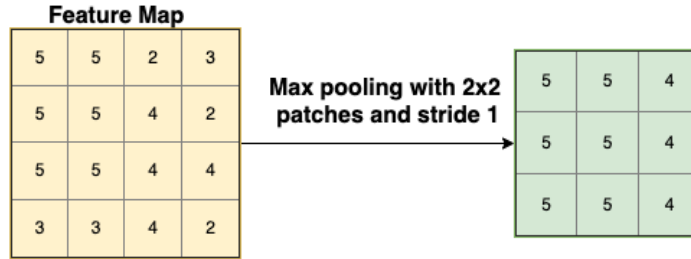


Figure 6.6: Example of using max pooling with stride length 1 (move one pixel each time).

Once the image has passed through the convolution, activation and pooling layers we vectorize the output (a matrix) to a column vector. We pass this vector through multiple fully connected layers which we discussed in section 6.2. The final layer in a CNN is usually the softmax layer. It takes the vector of values (suppose size  $K$ ) from the fully connected layers as input and outputs a probability distribution, where each value represents a probability that a feature belongs to a particular class. The softmax function is defined as follows:

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^N e^{z_k}}, \quad j = 1 \dots, K. \quad (6.6)$$

where  $N$  is the number of classes,  $\mathbf{z}$  is the input vector to the layer and  $\sigma(\mathbf{z})_j$  is the output class probability. For example, suppose the output from a fully connected layer is the vector

$$\mathbf{z} = [1.2 \quad 3.1 \quad 2.8 \quad 2.3]^T.$$

Passing  $\mathbf{z}$  through the softmax layer gives us the following output vector of probabilities

$$\sigma(\mathbf{z}) = [0.0639 \quad 0.4274 \quad 0.3166 \quad 0.1920]^T.$$

This indicates the network is about 43% confident that the input image belongs to class 2.

### 6.3.1 Training the CNN

In this section we describe a common approach used for training a CNN. We follow the outline and derivations in [47]. Consider a CNN with  $L$  layers, where layer 1 represents the input layer and layer  $L$ , the output layer. Let  $n_\ell$  denote the number of neurons in layer  $\ell$  where  $n_1$  and  $n_L$  represent the dimension of the input and output layers, respectively.

Let  $\mathbf{W}^\ell$  represent the matrix of weights at layer  $\ell$  where  $w_{j,k}^\ell$  denotes the weight that neuron  $j$  in layer  $i$  applies to the output from neuron  $k$ . Let  $\mathbf{x} \in \mathbb{R}^{n_1}$  be the input data to the network. We can summarize the outputs from the network as follows:

$$\begin{aligned} \mathbf{h}^1 &= \mathbf{x}, \quad \text{and} \\ \mathbf{h}^\ell &= \phi(\mathbf{W}^\ell \mathbf{h}^{\ell-1} + \mathbf{b}^\ell), \quad \ell = 2, \dots, L, \end{aligned} \tag{6.7}$$

where  $\phi$  is the activation function,  $\mathbf{b}^\ell$ , the bias at layer  $\ell$  and  $\mathbf{h}^\ell$ , the output at layer  $\ell$ .

Suppose we have  $N$  input data points  $\{x_i\}_{i=1}^N$  with  $N$  known output/classification labels  $\{y_i\}_{i=1}^N$ . We would like to minimize the number of incorrect classifications made. We do this by minimizing the mean squared error:

$$E = \frac{1}{N} \sum_{i=1}^N E_{x_i} = \frac{1}{N} \sum_{i=1}^N [h^L(x_i) - y_i]^2, \tag{6.8}$$

where  $h^L(x_i)$  indicates the output at the final layer  $L$  for the input  $x_i$ . The goal is to learn the parameters  $\mathbf{W}$  and  $\mathbf{b}$  for all layers that minimize  $E$ . It is useful to imagine the weights and biases stored as one single vector  $\boldsymbol{\theta}$ . We use the stochastic gradient descent (SGD) method in our work [85]. A generic one step of the method is given below:

1. Randomly choose  $i \in 1, \dots, N$
2. Update  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla E_{x_i}(\boldsymbol{\theta})$

where  $\eta$  is the learning rate. Back-propagation is used to calculate the gradient of the objective function (see (6.8)) with respect to the network's parameters,  $\mathbf{W}$  and  $\mathbf{b}$ . We take the derivatives with respect to every  $w_{j,k}^\ell$  and  $b_j^\ell$ . Let:

$$\mathbf{z}^\ell = \mathbf{W}^\ell \mathbf{h}^{\ell-1} + \mathbf{b}^\ell, \quad \ell = 2, \dots, L \tag{6.9}$$

so that  $\mathbf{h}^\ell = \sigma(\mathbf{z}^\ell)$  for  $\ell = 2, \dots, L$ . The error in the  $j$ th neuron of layer  $\ell$  is defined as:

$$\delta_j^\ell = \frac{\partial E}{\partial w_j^\ell}. \quad (6.10)$$

Let  $\circ$  denote the Hadamard product such that  $(\mathbf{x} \circ \mathbf{y})_i = x_i y_i$  for vectors and  $(\mathbf{A} \circ \mathbf{B})_{ij} = A_{ij} B_{ij}$ . It then follows that [47]:

$$\begin{aligned} \delta^L &= \sigma(\mathbf{z}^L) \circ (\mathbf{h}^L - \mathbf{y}), \\ \delta^\ell &= \sigma(\mathbf{z}^\ell) \circ (\mathbf{W}^{\ell+1})^T \delta^{\ell+1}, \quad \ell = 2, \dots, L-1, \\ \frac{\partial E}{\partial b_j^\ell} &= \delta_j^\ell, \quad \ell = 2, \dots, L, \\ \frac{\partial E}{\partial w_{j,k}^\ell} &, \quad \ell = 2, \dots, L. \end{aligned} \quad (6.11)$$

One forward pass through the neural network gives an initial estimate of the output  $h_L$  at layer  $L$ . This estimate is used to calculate all the derivatives with respect to weights and biases moving backwards through the network. This is the back propagation process. The weights are then updated according to an optimization rule using these derivatives, and the process is repeated.

It is worth noting that back-propagation is essentially adjoint based gradient computation, a widely used technique in PDE optimization. Moreover, what we call learning rate here, is what optimizers call step length, in line search algorithms.

## 6.4 Application Problem: Corrosion detection in spent nuclear fuel canisters

After spent nuclear fuel is removed from a reactor core, it is placed in a wet storage assembly for an extended period of time to manage radioactivity and heat. As a result of an increased reliance on nuclear power, further intermittent storage is required to prevent the storage pools from filling up. Dry cask canisters have been used to store spent fuel rods after they have been cooled down in wet storage. The lifecycle of nuclear fuel [11] is shown in figure 6.7.

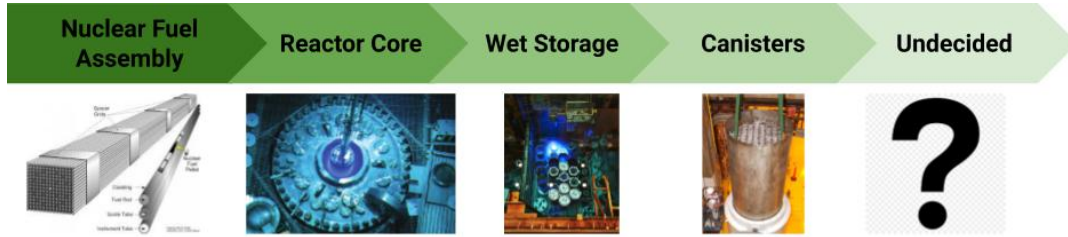


Figure 6.7: Lifecycle of nuclear fuel.

With a lack of a long-term disposal plan for spent nuclear fuel, these dry cask canisters, which have a planned utility of approximately twenty years, are going to be pushed further than initial predictions and may be vulnerable to physical degradation. Undetected defects in spent nuclear fuel canisters have the potential to irradiate the surrounding environment and personnel. This necessitates the development of remotely operated systems for real-time fast detection of defect type, location and size to support future repair activities, reduce operator errors and optimize repair quality. An accurate and safe inspection system is needed to analyze the current conditions of dry cask canisters and to ensure their continued integrity and use.

While current techniques rely on in-situ manual inspections, (see figure 6.8) they are subject to human error, pose an increased risk to personnel, and take an unreasonable amount of time to execute, approximately a year to evaluate just one canister. Thus, the current method is not a practical, long-term solution to analyzing dry cask integrity. New methods of assessing defects in canisters are needed to automate the process and ensure the continued sustainability of dry cask storage.

We propose a deep learning approach that will analyze images taken inside dry cask storage canisters and detect corrosion and pitting in real time, eliminating human exposure to radiation and minimizing human error.

#### 6.4.1 Databank Generation

We use 168 raw images, consisting of 84 corroded canister images and 84 intact canister images. The images were taken with a 16MP phone camera from a variety of locations, angles and lighting conditions to ensure high variability in the dataset. It includes images taken from inside the canisters as well as outside. The images were provided by the Electric

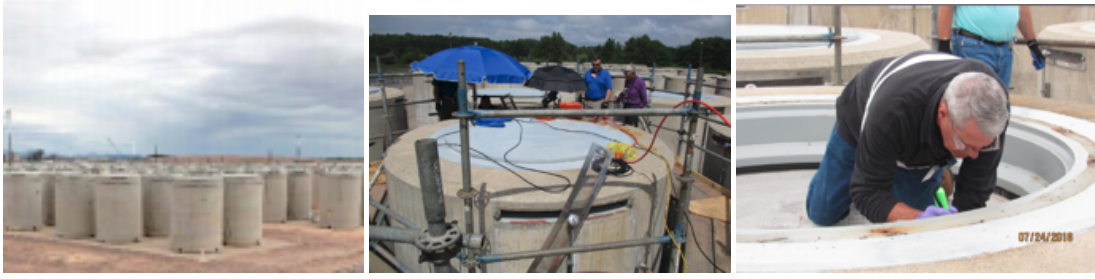


Figure 6.8: Nuclear fuel storage site and in-situ inspections.

Power Research Institute (EPRI).

The 168 images were split into smaller images (tiled image) of a 256x256 pixel resolution, which produced approximately 38,000 images. After manually labeling each tiled image as either an intact or corroded image, the dataset was split randomly to generate the training, validation, and test sets. The dataset was split 60% for training purposes and 20% each for validation and testing. To keep the sets independent, tiles from a particular image were included in only one set. The total number of images in the training set is approximately 22,000.

A number of data augmentation techniques were implemented to widen the domain of applicability for the models predictive power. Data preprocessing included random rotations, flips, and color shifting of the images.

### 6.4.2 Training & Validation

CNNs learn features automatically when training by updating the weights after every iteration of the SGD algorithm so no feature extraction techniques are needed, which is not the case for other image-based classification problems. The ratio between the training to validation set was 3:1 with training accuracy calculated from approximately 22,000 tiles and validation accuracy out of approximately 7,500 tiles.

ResNets [46] are a specific type of CNN architecture which can help increase the number of hidden layers in the network. We chose to use the ResNet-18 architecture in this work, a CNN which is 18 layers deep. We use SGD method as the optimizer and take our loss function to be the mean squared error.

A parameter that is needed to train the network is the batch size which is the number of samples that will be propagated through the network at any one time. We experimented with many different batch sizes and settled on using a batch size of 128. This means that the algorithm takes the first 128 images from the training set and trains the network, then takes the second 128 images and trains again and so on. One Epoch is when the entire dataset has been trained on the network just once.

To train the network we needed to select an appropriate learning rate. To help select the learning rate we plot the learning rate vs the loss. We selected the point where the decrease in the loss was the fastest. In figure 6.9 (left) this corresponded to a learning rate of  $10^{-2}$ .

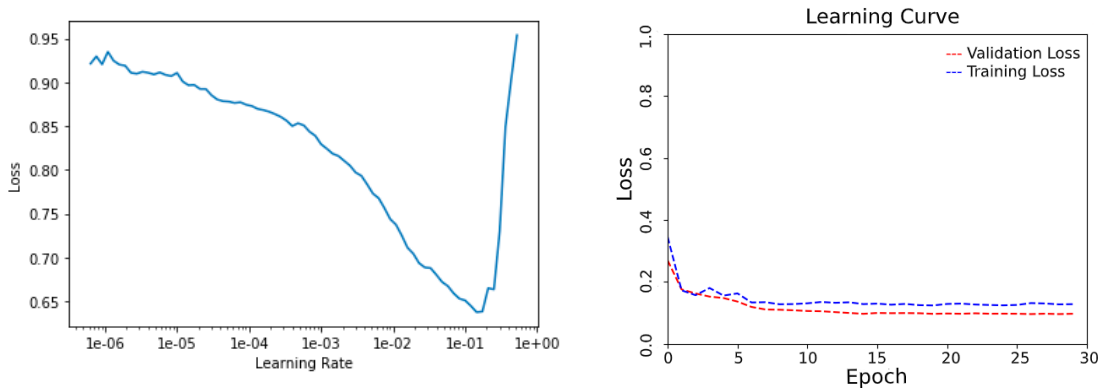


Figure 6.9: Plot of learning rate vs loss.

In figure 6.9 (right) we see how the network is improving/learning with each epoch. The goal is to continue training until the training loss and validation loss level off, and there is a small gap between the two. Continuing to train beyond this point usually results in overfitting. The issue with overfitting is the network becomes more specialized towards the training data and may struggle to adapt to new data.

### 6.4.3 Performance Metrics

We have trained a CNN to classify images of steel canisters as intact or corroded. The next step is to assess the performance of the CNN on unseen image data. Performance is



Table 6.1: Example of entries in a confusion matrix.

		Actual	
Predicted	Positive	Positive	Negative
	Negative	TP FN	FP TN

assessed using a number of different tests and measures. A common way to summarize classification accuracy is a confusion matrix. A confusion matrix is a table of the number of correct and incorrect predictions summarized with count values and broken down by each class. There are 4 important results from the confusion matrix:

1. True Positive (TP): Number of cases predicted as positive and the actual result was positive.
2. True Negative (TN) : Number of cases predicted as negative and the actual output was negative.
3. False Positive (FP): Number of cases predicted positive and the actual output was negative.
4. False Negative (FN) : Number of cases predicted negative and the actual output was positive.

In table 6.1 we show how the entries of the confusion matrix are presented. The matrix also gives us insight into the types of errors the model is making. FP are classed as Type I errors and FN, Type 2 errors.

We can calculate a number of other performance metrics from the results of the confusion matrix. *Precision* is given by the relation:  $TP/(TP+FP)$ . It tells us what proportion of positive predictions were actually correct. *Recall* is given by the relation  $TP/(TP+FN)$ . It tell us what proportion of actual positives were correctly identified. There is a trade off between precision and recall. If we try to increase one of the metrics then the other metric will decrease in response. For example if we have high recall and low precision then we have a low number for FN but a high FP. In contrast, when we have low recall and high precision we have a high FN but low FP.

The *F1 Score* is another measure of the model's accuracy. It is defined as the harmonic mean

between precision and recall:

$$F_1 = 2 \times \frac{(\text{precision} \times \text{recall})}{\text{precision} + \text{recall}}.$$

This score considers both false positives and false negatives. The F1 score can be more useful than accuracy in cases where one has an unbalanced class distribution. The range for the F1 Score is [0, 1]. It tells us how precise the model is, how many cases it classifies correctly, as well as how robust it is.

Another performance metric is *specificity* given by the relation:  $TN/(TN + FP)$ . Specificity is the true negative rate or the proportion of negatives that are identified correctly. A receiver operating characteristic curve (ROC curve) is created by plotting the true positive rate (recall) versus the false positive rate (1-specificity) at different threshold values. This curve gives us another way to visualize the performance of the classifier. The ROC curve demonstrates the tradeoff between sensitivity and specificity (any increase in sensitivity will be accompanied by a decrease in specificity). The closer the ROC curve is to the upper left corner, the higher the overall accuracy of the test. In figure 6.10 we can see the curves for three different models. A straight line curve through the center of the plot represents a bad model which is only 50% accurate. The other curves show better models with the red curve being the best. Notice as the model gets better the curve approaches the upper left corner of the plot. The area under the curve (AUC) is a measure of a model's accuracy. AUC is a measure of how well a model can distinguish between two different groups.

#### 6.4.4 Results

Now that we have defined each of the performance metrics we used in our work, we report the performance results from our optimized CNN classifier using ResNet-18 architecture. The performance of the CNN model was tested on 7,966 unseen tiled images from our test set, 819 corroded images and 7,147 intact images. The classification accuracy was found to be ~96% which means the CNN correctly identified 7,647 of the test images. In table 6.2 we present the confusion matrices. The first shows the results of testing on the tiled images. We also checked the results when testing on the raw images. Of particular interest were the number of FP and FN results. False positives in this case represents cases where the

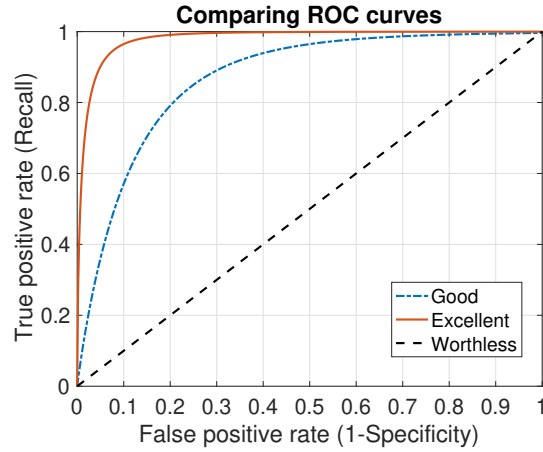


Figure 6.10: ROC curve with three examples of models

Table 6.2: Confusion matrix for tiled images (left) and whole raw images (right)

Tiled Images: Actual				Raw Images: Actual			
Predicted	Corroded	Corroded	Intact	Predicted	Corroded	Corroded	Intact
	Intact	677	142		Intact	17	2
		148	6999			14	0

network predicted images as corroded when they were in fact intact. False negatives are the cases where the network classified the images as intact when they were actually corroded images. From this table we find that the accuracy for corroded images is approximately 83% and for the intact images approximately 98%. Table 6.3 summarizes the performance metrics for our optimized CNN. The optimized network was then tested on the 168 large original images and resulted in an accuracy of 94% as presented in table 6.2.

Table 6.3: Performance metrics for optimized network using ResNet-18 architecture.

Metric	Result
Accuracy	96.31%
F1 Score	0.980
Precision	0.972
Sensitivity (Recall)	0.987
Specificity	0.830
AUC	0.980

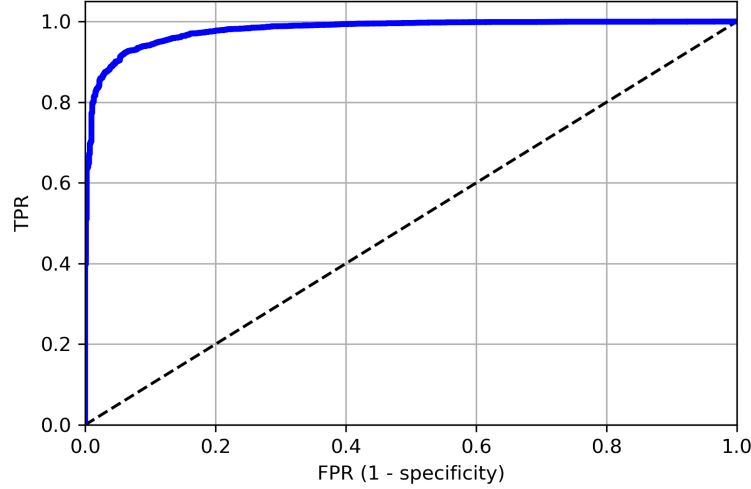


Figure 6.11: ROC curve for optimized network

In figure 6.11 we present the ROC curve for the optimized network. This figure allows us to visually assess the performance of the CNN model. The ROC curve shows us the trade-off between the True Positive Rate (Recall) and the False Positive Rate (FPR) at different classification thresholds. We note that the CNN model has performed well since the ROC curve is close to the top left corner of the figure. The area under a ROC curve (AUC) given in table 6.3 quantifies the overall ability of the model to differentiate between those images with corrosion and those without. With a value of 0.98 we can say that the model has performed very well.

## 6.5 Conclusions

In this chapter, we have proposed a machine learning approach for image classification using CNNs. Images are high-dimensional objects which makes traditional neural networks an infeasible option. We have proposed a deep learning approach using a deep architecture of convolutional neural networks (CNNs) for detecting corrosion without calculating the defect features. The designed CNN demonstrates an accuracy of approximately 96%. The results demonstrate that a deep learning approach is effective in analyzing canister integrity,

an approach that has the potential to significantly increase the speed of canister inspections, minimize inspection costs, and eliminate radiation doses to personnel and can indeed find canister defects in realistic situations.

Future work includes more exhaustive testing by acquiring a larger dataset of images from a diverse collection of nuclear canisters. We would also like to investigate using pixel-wise labeling for the images. Pixel-wise labeling would greatly increase the resolution of the heatmaps, enabling full semantic segmentation of the entire input image. Such an approach would require of an upfront investment to annotate images pixel-wise.

## CHAPTER

# 7

## CONCLUSIONS

In this dissertation we have explored various techniques to effectively reduce the dimension of large-scale problems to allow efficient UQ studies. In chapter 3, we successfully applied the active subspace method to a high-dimensional chemical kinetics application problem. We compared the performance of the perturbation approach to the regression based approach. In some applications, the regression-based approach is useful where gradient computation is very challenging. We computed and used local linear models to approximate gradients but computing a global quadratic model has been proposed and used in [25]. Applying a global quadratic model to this application or other high-dimensional chemical kinetics problems is subject of future work. We also explored links between traditional GSA methods and active subspaces. In particular we investigated the links between total Sobol' indices, DGSMs and activity scores, the latter which is associated with active subspaces. We quantified the approximation error of using activity scores to approximate DGSMs.

In chapter 4, we developed a surrogate modeling approach for complex models with high-dimensional inputs and function valued outputs that uses KL expansions for output di-

mension reduction, active subspaces for input dimension reduction, and adjoint based gradient computation in the active subspace method. We demonstrated the effectiveness of the approach using a biotransport problem with high-dimensional random inputs and random field outputs. We have also analyzed errors due to truncating the KL expansion and due to active subspace construction. Topics for future work include extending the approach to surrogate modeling of the time-dependent diffusion and convection-diffusion processes in biological and geological flows.

In chapter 5, we introduced functional activity scores which we use to approximate the functional DGSMs. Functional activity scores come for free, as a by product of active subspace analysis. No further model evaluations are needed, beyond the model evaluations used for the purposes of active subspace discovery. We computed the functional activity scores for a biotransport application problem and compared the results to the functional DGSMs. In that chapter we also investigated the effects of input parameter correlations on the active subspace method. We noted, in some cases, introducing correlations between parameters has little to no effect on the dimension of the active subspace or the surrogate model approximation. In other problems, we found that parameter correlations can actually be useful in identifying a lower dimensional active subspace.

In chapter 6, we implemented a machine learning approach, using CNNs, to detect faults in steel canisters containing spent nuclear fuel. Despite the small sample size of 168 whole images, the trained CNN classifier achieves high detection accuracy. The approach shows potential to automate and speed up inspections, to minimize inspection costs, and replace human-conducted onsite inspections, thus reducing radiation doses to personnel. Topics of future work include more testing on a much larger dataset of images to ensure the robustness of the CNN classifier. We would also like to label the images at a pixel-wise level which will require a significant investment of time.

## BIBLIOGRAPHY

- [1] S. Aleshin-Guendel and S. Alvarez. “Examining the Structure of Convolutional Neural Networks”. In: (2017).
- [2] A. Alexanderian, N. Petra, G. Stadler, and O. Ghattas. “Mean-variance risk-averse optimal control of systems governed by PDEs with random parameter fields using quadratic approximations”. In: *SIAM/ASA J. Uncertainty Quantif.* 5 (2017), pp. 1166–1192.
- [3] A. Alexanderian, L. Zhu, M. Salloum, R. Ma, and M. Yu. “Investigation of biotransport in a tumor with uncertain material properties using a non-intrusive spectral uncertainty quantification method”. In: *J. Biomech. Eng.* (2017), pp. 091006–1–091006–11.
- [4] A. Alexanderian, P. A. Gremaud, and R. C. Smith. “Variance-based sensitivity analysis for time-dependent processes”. In: *Reliability Engineering & System Safety* 196 (2020), p. 106722.
- [5] A. Alexanderian, W. Reese, R. C. Smith, and M. Yu. “Model Input and Output Dimension Reduction Using Karhunen–Loève Expansions With Application to Biotransport”. In: *ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg* 5.4 (2019).
- [6] A. Alexanderian, J. Winokur, I. Sraj, A. Srinivasan, M. Iskandarani, W. C. Thacker, and O. M. Knio. “Global sensitivity analysis in an ocean general circulation model: a sparse spectral projection approach”. In: *Computational Geosciences* 16.3 (2012), pp. 757–778.
- [7] V. Andrearczyk and P. F. Whelan. “Deep learning in texture analysis and its application to tissue image classification”. In: *Biomedical Texture Analysis*. Elsevier, 2017, pp. 95–129.
- [8] I. Babuška, F. Nobile, and R. Tempone. “A stochastic collocation method for elliptic partial differential equations with random input data”. In: *SIAM Journal on Numerical Analysis* 45.3 (2007), pp. 1005–1034.
- [9] M. Baucells and E. Borgonovo. “Invariant probabilistic sensitivity analysis”. In: *Management Science* 59.11 (2013), pp. 2536–2549.



- [10] G. Blatman and B. Sudret. “Efficient computation of global sensitivity indices using sparse polynomial chaos expansions”. In: *Reliability Engineering & System Safety* 95.11 (2010), pp. 1216–1229.
- [11] U. N.W.T. R. BOARD. *Chloride-induced stress corrosion cracking potential in dry storage canisters for spent nuclear fuel*. 2017.
- [12] E. Borgonovo. “A new uncertainty importance measure”. In: *Reliab. Eng. Syst. Saf.* 92 (2007), pp. 771–784.
- [13] E. Borgonovo and B. Iooss. “Moment independent and reliability-based importance measures”. In: (2016).
- [14] A. Borzì and V. Schulz. *Computational optimization of systems governed by partial differential equations*. Vol. 8. SIAM, 2011.
- [15] W. H. Clark. “Tumour progression and the nature of cancer”. In: *Br J Cancer* 64 (1991), pp. 631–44.
- [16] W. H. Clark. “Biphasic finite element model of solute transport for direct infusion into nervous tissue”. In: *Annals of Biomedical Engineering* 35 (2007), 2145—2158.
- [17] H. L. Cleaves, A. Alexanderian, H. Guy, R. C. Smith, and M. Yu. “Derivative-based global sensitivity analysis for models with high-dimensional inputs and functional outputs”. In: *SIAM Journal on Scientific Computing* 41.6 (2019), A3524–A3551.
- [18] P. Constantine. *Active Subspaces: Emerging Ideas in Dimension Reduction for Parameter Studies*. SIAM, Philadelphia, 2015.
- [19] P. G. Constantine, M. Emory, J. Larsson, and G. Iaccarino. “Exploiting active subspaces to quantify uncertainty in the numerical simulation of the HyShot II scramjet”. In: *Journal of Computational Physics* 302 (2015), pp. 1–20.
- [20] P. G. Constantine. *Active subspaces: Emerging ideas for dimension reduction in parameter studies*. Vol. 2. SIAM, 2015.
- [21] P. G. Constantine and P. Diaz. “Global sensitivity metrics from active subspaces”. In: *Reliability Engineering & System Safety* 162 (2017), pp. 1–13.
- [22] P. G. Constantine and A. Doostan. “Time-dependent global sensitivity analysis with active subspaces for a lithium ion battery model”. In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* 10 (2017), pp. 243–262.
- [23] P. G. Constantine, E. Dow, and Q. Wang. “Active subspace methods in theory and practice: applications to kriging surfaces”. In: *SIAM Journal on Scientific Computing* 36.4 (2014), A1500–A1524.

- [24] P. Constantine and P. Diaz. “Global sensitivity metrics from active subspaces”. In: *Reliab. Eng. Syst. Saf.* 162 (2017), pp. 1–13.
- [25] P. Constantine and A. Doostan. “Time-dependent global sensitivity analysis with active subspaces for a lithium ion battery model”. In: *Stat. Anal. Data Min.* 10 (2017), pp. 243–262.
- [26] S. Cosnier, A. Gross, A. Le Goff, and M. Holzinger. “Recent advances on enzymatic glucose/oxygen and hydrogen/oxygen biofuel cells: Achievements and limitations”. In: *J. Power Sources* 325 (2016), pp. 252–263.
- [27] S. Da Veiga. “Global sensitivity analysis with dependence measures”. In: *Journal of Statistical Computation and Simulation* 85.7 (2015), pp. 1283–1305.
- [28] L. Das. “Hydrogen-oxygen reaction mechanism and its implication to hydrogen engine combustion”. In: *Int. J. Hydrogen Energy* 21 (1996), pp. 703–715.
- [29] P. Debbage. “Targeted drugs and nanomedicine: present and future”. In: *Current Pharmaceutical Design* 15 (2009), pp. 153–72.
- [30] B. Debusschere, Y. Marzouk, H. Najm, B. Rhoads, D. Goussis, and M. Valorani. “Computational singular perturbation with non-parametric tabulation of slow manifolds for time integration of stiff chemical kinetics”. In: *Combust. Theor. Model.* 16.1 (2012), pp. 173–198.
- [31] P. Diaz. “Global sensitivity metrics from active subspaces with applications”. MA thesis. Colorado School of Mines, 2016.
- [32] A. Doostan, R. G. Ghanem, and J. Red-Horse. “Stochastic model reduction for chaos representations”. In: *Computer Methods in Applied Mechanics and Engineering* 196.37-40 (2007), pp. 3951–3966.
- [33] H. Elman. “Solution Algorithms for Stochastic Galerkin Discretizations of Differential Equations with Random Data”. In: *Handbook of Uncertainty Quantification* (2017), pp. 1–16.
- [34] J. Friedman. *Fast MARS*. Tech. rep. 110. Laboratory for Computational Statistics, Department of Statistics, Stanford University, 1993.
- [35] F. Gamboa, A. Janon, T. Klein, A. Lagnoux, et al. “Sensitivity analysis for multidimensional and functional outputs”. In: *Electronic Journal of Statistics* 8.1 (2014), pp. 575–603.
- [36] C. Gershenson. “Artificial neural networks for beginners”. In: *arXiv preprint cs/0308031* (2003).

- [37] R. G. Ghanem and P. D. Spanos. *Stochastic finite elements: a spectral approach*. Springer-Verlag, New York, 1991, pp. x+214.
- [38] R. Ghanem. “Probabilistic characterization of transport in heterogeneous media”. In: *Computer Methods in Applied Mechanics and Engineering* 158.3 (1998), pp. 199–220.
- [39] R. G. Ghanem and P. D. Spanos. *Stochastic finite elements: a spectral approach*. Courier Corporation, 2003.
- [40] I. G. Graham, F. Y. Kuo, J. A. Nichols, R. Scheichl, C. Schwab, and I. H. Sloan. “Quasi-Monte Carlo finite element methods for elliptic PDEs with lognormal random coefficients”. In: *Numerische Mathematik* 131.2 (2015), pp. 329–368.
- [41] M. Gunzburger. *Perspectives in flow control and optimization*. Vol. 5. Siam, 2003.
- [42] H. Guy, A. Alexanderian, and M. Yu. “A distributed active subspace method for scalable surrogate modeling of function valued outputs”. In: *arXiv preprint arXiv:1908.02694* (2019).
- [43] M. Hantouche, S. Sarathy, and O. Knio. “Global sensitivity analysis of n-butanol reaction kinetics using rate rules”. In: *Combust. Flame* 196 (2018), pp. 452–465.
- [44] J. Hart and P. Gremaud. “An approximation theoretic perspective of Sobol’ indices with dependent variables”. In: *International Journal for Uncertainty Quantification* 8.6 (2018).
- [45] D. M. Hartley, J. G. Morris Jr, and D. L. Smith. “Hyperinfectivity: a critical element in the ability of *V. cholerae* to cause epidemics?” In: *PLoS medicine* 3.1 (2005), e7.
- [46] K. He, X. Zhang, S. Ren, and J. Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.
- [47] C. F. Higham and D. J. Higham. “Deep learning: An introduction for applied mathematicians”. In: *SIAM Review* 61.4 (2019), pp. 860–891.
- [48] M. Holeňa, D. Linke, U. Rodemerck, and L. Bajer. “Neural networks as surrogate models for measurements in optimization algorithms”. In: *International Conference on Analytical and Stochastic Modeling Techniques and Applications*. Springer. 2010, pp. 351–366.
- [49] T. Hsing and R. Eubank. *Theoretical foundations of functional data analysis, with an introduction to linear operators*. John Wiley & Sons, 2015.

- [50] B. Iooss and P. Lemaître. “A review on global sensitivity analysis methods”. In: *Uncertainty management in simulation-optimization of complex systems*. Springer, 2015, pp. 101–122.
- [51] B. Iooss and A. Saltelli. “Introduction to sensitivity analysis”. In: *Handbook of uncertainty quantification*. Ed. by R. Ghanem, D. Higdon, and H. Owhadi. Springer, 2017, pp. 1103–1122.
- [52] J. Jacques, C. Lavergne, and N. Devictor. “Sensitivity analysis in presence of model uncertainty and correlated inputs”. In: *Reliab. Eng. Syst. Saf.* 91 (2006), pp. 1126–1134.
- [53] A. K. Jain, J. Mao, and K. M. Mohiuddin. “Artificial neural networks: A tutorial”. In: *Computer* 29.3 (1996), pp. 31–44.
- [54] A. Jameson. “Aerodynamic design via control theory”. In: *J. Sci. Comput.* 3 (1988), pp. 233–260.
- [55] J. Jefferson, J. Gilbert, P. Constantine, and R. Maxwell. “Active subspaces for sensitivity analysis and dimension reduction of an integrated hydrologic model”. In: *Computers & Geosciences* 83 (2015), pp. 127–138.
- [56] W. Ji, J. Wang, O. Zahm, Y. M. Marzouk, B. Yang, Z. Ren, and C. K. Law. “Shared low-dimensional subspaces for propagating kinetic uncertainty to multiple outputs”. In: *Combustion and Flame* 190 (2018), pp. 146–157.
- [57] A. Kiparissides, S. Kucherenko, A. Mantalaris, and E. Pistikopoulos. “Global sensitivity analysis challenges in biological systems modeling”. In: *Ind. Eng. Chem. Res.* 48 (2009), pp. 7168–7180.
- [58] S. Kucherenko and B. Iooss. “Derivative-based global sensitivity measures”. In: *Handbook of Uncertainty Quantification*. Ed. by R. Ghanem, D. Higdon, and H. Owhadi. Springer, 2017.
- [59] S. Lam. “Singular perturbation for stiff equations using numerical methods”. In: *Recent advances in the aerospace sciences*. Springer, 1985, pp. 3–19.
- [60] S. Lam and D. Goussis. “Understanding complex chemical kinetics with computational singular perturbation”. In: *Symposium (International) on Combustion*. Vol. 22. Elsevier, 1989, pp. 931–941.
- [61] M. Lamboni, B. Iooss, A.-L. Popelin, and F. Gamboa. “Derivative-based global sensitivity measures: General links with Sobol’ indices and numerical tests”. In: *Math. Comput. Simul.* 87 (2013), pp. 45–54.

- [62] P. Lancaster. “On eigenvalues of matrices dependent on a parameter”. In: *Numerische Mathematik* 6.1 (1964), pp. 377–387.
- [63] O. P. Le Maître and O. M. Knio. *Spectral methods for uncertainty quantification*. Scientific Computation. With applications to computational fluid dynamics. Springer, New York, 2010, pp. xvi+536.
- [64] O. Le Maître and O. M. Knio. *Spectral methods for uncertainty quantification: with applications to computational fluid dynamics*. Springer Science & Business Media, 2010.
- [65] O. P. Le Maître, M. T. Reagan, H. N. Najm, R. G. Ghanem, and O. M. Knio. “A stochastic projection method for fluid flow: II. Random process”. In: *Journal of computational Physics* 181.1 (2002), pp. 9–44.
- [66] O. Le Maître, O. Knio, H. Najm, and R. Ghanem. “Uncertainty propagation using Wiener–Haar expansions”. In: *Journal of computational Physics* 197.1 (2004), pp. 28–57.
- [67] C. Li and S. Mahadevan. “An efficient modularized sample-based method to estimate the first-order Sobol index”. In: *Reliab. Eng. Syst. Saf.* 153 (2016), pp. 110–121.
- [68] G. Li, H. Rabitz, P. Yelvington, O. Oluwole, F. Bacon, C. Kolb, and J. Schoendorf. “Global sensitivity analysis for systems with independent and/or correlated inputs”. In: *J. Phys. Chem. A* 114 (2010), pp. 6022–6032.
- [69] M. Loève. *Probability theory. I*. Fourth. Graduate Texts in Mathematics, Vol. 45. Springer-Verlag, New York-Heidelberg, 1977, pp. xvii+425.
- [70] B. Loges, A. Boddien, H. Junge, and M. Beller. “Controlled generation of hydrogen from formic acid amine adducts at room temperature and application in H<sub>2</sub>/O<sub>2</sub> fuel cells”. In: *Angew. Chem. Int. Ed.* 47 (2008), pp. 3962–3965.
- [71] T. W. Lukaczyk, F. Palacios, J. J. Alonso, and P. Constantine. “Active Subspaces for Shape Optimization”. In: *the 10th AIAA Multidisciplinary Design Optimization Conference*. AIAA-2014-1171. National Harbor, Maryland, 2014.
- [72] R. Ma, D. Su, and L. Zhu. “Multiscale simulation of nanoparticle transport in deformable tissue during an infusion process in hyperthermia treatments of cancers.” In: *Nanoparticle Heat Transfer and Fluid Flow, Computational & Physical Processes in Mechanics & Thermal Science Series*. Ed. by W. J. Minkowycz, E. Sparrow, and J. P. Abraham. Vol. 4. CRC Press, Taylor & Francis Group, 2012.

- [73] N. Mangado, G. Piella, J. Noailly, J. Pons-Prats, and M. Ángel González Ballester. “Analysis of Uncertainty and Variability in Finite Element Computational Models for Biomedical Engineering: Characterization and Propagation”. In: *Front Bioeng Biotechnol.* 4 (2016), p. 85.
- [74] A. Marrel, B. Iooss, B. Laurent, and O. Roustant. “Calculations of sobol indices for the gaussian process metamodel”. In: *Reliability Engineering & System Safety* 94.3 (2009), pp. 742–751.
- [75] H. G. Matthies and A. Keese. “Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations”. In: *Computer methods in applied mechanics and engineering* 194.12-16 (2005), pp. 1295–1331.
- [76] M. D. Morris. “Factorial sampling plans for preliminary computational experiments”. In: *Technometrics* 33.2 (1991), pp. 161–174.
- [77] R. Morrison, T. Oliver, and R. Moser. “Representing model inadequacy: A stochastic operator approach”. In: *SIAM/ASA J. Uncertainty Quantif.* 6 (2018), pp. 457–496.
- [78] S. Nannapaneni and S. Mahadevan. “Reliability analysis under epistemic uncertainty”. In: *Reliab. Eng. Syst. Saf.* 155 (2016), pp. 9–20.
- [79] T. Papamarkou, H. Guy, B. Kroencke, J. Miller, P. Robinette, D. Schultz, J. Hinkle, L. Pullum, C. Schuman, J. Renshaw, et al. “Automated detection of pitting and stress corrosion cracks in used nuclear fuel dry storage canisters using residual neural networks”. In: *arXiv preprint arXiv:2003.03241* (2020).
- [80] E. Pianosi and T. Wagener. “A simple and efficient method for global sensitivity analysis based on cumulative distribution functions”. In: *Environmental Modelling & Software* 67 (2015), pp. 1–11.
- [81] E. Plischke, E. Borgonovo, and C. Smith. “Global sensitivity measures from given data”. In: *Eur. J. Oper. Res.* 226 (2013), pp. 536–550.
- [82] C. Prieur and S. Tarantola. “Variance-based sensitivity analysis: Theory and estimation algorithms”. In: *Handbook of Uncertainty Quantification*. Ed. by R. Ghanem, D. Higdon, and H. Owhadi. Springer, 2017, pp. 1217–1239.
- [83] C. E. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning Cambridge*. 2006.
- [84] M. Reagana, H. Najm, R. Ghanem, and O. Knio. “Uncertainty quantification in reacting-flow simulations through non-intrusive spectral projection”. In: *Combust. Flame* 132 (2003), pp. 545–555.

- [85] S. Ruder. “An overview of gradient descent optimization algorithms”. In: *arXiv preprint arXiv:1609.04747* (2016).
- [86] T. M. Russi. “Uncertainty Quantification with Experimental Data and Complex System Models”. PhD thesis. University of California, Berkeley, 2010.
- [87] G. Saad and R. Ghanem. “Characterization of reservoir simulation models using a polynomial chaos-based ensemble Kalman filter”. In: *Water Resources Research* 45.4 (2009).
- [88] C. Safta, H. Najm, and O. Knio. “TChem-a software toolkit for the analysis of complex kinetic models”. In: *Report No. SAND2011-3282, Sandia National Laboratories, CA, USA, 2011* (2011).
- [89] M. Salloum, A. Alexanderian, O. Le Maître, H. Najm, and O. Knio. “Simplified CSP analysis of a stiff stochastic ODE system”. In: *Comput. Methods Appl. Mech. Eng.* 217–220 (2012), pp. 121–138.
- [90] M. Salloum, R. Ma, D. Weeks, and L. Zhu. “Controlling nanoparticle delivery in magnetic nanoparticle hyperthermia for cancer treatment: experimental study in agarose gel”. In: *Int. J. Hyperthermia* 24 (2008), pp. 337–345.
- [91] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola. “Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index”. In: *Comput. Phys. Commun.* 181 (2010), pp. 259–270.
- [92] A. Saltelli, K. Chan, E. M. Scott, et al. *Sensitivity analysis*. Vol. 1. Wiley New York, 2000.
- [93] S. Sankararaman and S. Mahadevan. “Integration of model verification, validation, and calibration for uncertainty quantification in engineering systems”. In: *Reliab. Eng. Syst. Saf.* 138 (2015), pp. 194–209.
- [94] S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.
- [95] V. Sharma, S. Rai, and A. Dev. “A comprehensive study of artificial neural networks”. In: *International Journal of Advanced research in computer science and software engineering* 2.10 (2012).
- [96] R. C. Smith. *Uncertainty quantification: theory, implementation, and applications*. Vol. 12. Siam, 2013.

- [97] I. M. Sobol. "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates". In: *Mathematics and computers in simulation* 55.1-3 (2001), pp. 271–280.
- [98] I. Sobol. "Estimation of the sensitivity of nonlinear mathematical models". In: *Matematicheskoe Modelirovanie* 2.1 (1990), pp. 112–118.
- [99] I. Sobol. "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates". In: *Mathematics and Computers in Simulation* 55.1-3 (2001), pp. 271 –280.
- [100] I. Sobol' and S. Kucherenko. "Derivative based global sensitivity measures and the link with global sensitivity indices". In: *Math. Comput. Simul.* 79 (2009), pp. 3009–30017.
- [101] I. Sobol' and S. Kucherenko. "Derivative based global sensitivity measures and their link with global sensitivity indices". In: *Mathematics and Computers in Simulation* 79.10 (2009), pp. 3009–3017.
- [102] I. Sobol', S. Tarantola, D. Gatelli, S. Kucherenko, and W. Mauntz. "Estimating the approximation error when fixing unessential factors in global sensitivity analysis". In: *Reliab. Eng. Syst. Saf.* 92 (2007), pp. 957–960.
- [103] O. Sudakov, D. Koroteev, B. Belozarov, and E. Burnaev. "Artificial Neural Network Surrogate Modeling of Oil Reservoir: A Case Study". In: *International Symposium on Neural Networks*. Springer. 2019, pp. 232–241.
- [104] B. Sudret. "Global sensitivity analysis using polynomial chaos expansions". In: *Reliability Engineering & System Safety* 93.7 (2008), pp. 964 –979.
- [105] S Tarantola, D Gatelli, S. Kucherenko, W Mauntz, et al. "Estimating the approximation error when fixing unessential factors in global sensitivity analysis". In: *Reliability Engineering & System Safety* 92.7 (2007), pp. 957–960.
- [106] J.-Y. Tissot and C. Prieur. "A randomized orthogonal array-based procedure for the estimation of first-and second-order Sobol'indices". In: *J. Stat. Comput. Simul.* 85 (2015), pp. 1358–1381.
- [107] M. Valorani, F. Creta, D. Goussis, J. Lee, and H. Najm. "An automatic procedure for the simplification of chemical kinetic mechanisms based on CSP". In: *Combust. Flame* 146.1 (2006), pp. 29–51.
- [108] M. Vohra, A. Alexanderian, C. Safta, and S. Mahadevan. "Sensitivity-driven adaptive construction of reduced-space surrogates". In: *J Sci. Comput. (article in press)* ().



- [109] M. Vohra, X. Huan, T. Weihs, and O. Knio. “Design analysis for optimal calibration of diffusivity in reactive multilayers”. In: *Combust. Theor. Model.* 21 (2017), pp. 1023–1049.
- [110] M. Vohra, J. Winokur, K. Overdeep, P. Marcello, T. Weihs, and O. Knio. “Development of a reduced model of formation reactions in Zr-Al nanolaminates”. In: *J. Appl. Phys.* 116 (2014), p. 233501.
- [111] M. Vohra, A. Alexanderian, H. Guy, and S. Mahadevan. “Active subspace-based dimension reduction for chemical kinetics applications with epistemic uncertainty”. In: *Combustion and Flame* 204 (2019), pp. 152–161.
- [112] X Wu, C Wang, and T Kozłowski. “Kriging-based surrogate models for uncertainty quantification and sensitivity analysis”. In: *Proceedings of the MC-2017, International Conference on Mathematics Computational Methods Applied to Nuclear Science Engineering*. 2017.
- [113] D. Xiu and G. E. Karniadakis. “Modeling uncertainty in flow simulations via generalized polynomial chaos”. In: *Journal of Computational Physics* 187.1 (2003), pp. 137–167.
- [114] C. Xu and G. Gertner. “Extending a global sensitivity analysis technique to models with correlated parameters”. In: *Comput. Stat. Data Anal.* 51 (2007), pp. 5579–5590.
- [115] R. Yetter, F. Dryer, and H. Rabitz. “A comprehensive reaction mechanism for carbon monoxide/hydrogen/oxygen kinetics”. In: *Combust. Sci. Technol.* 79 (1991), pp. 97–128.
- [116] O. Zahm, P. G. Constantine, C. Prieur, and Y. M. Marzouk. “Gradient-based dimension reduction of multivariate vector-valued functions”. In: *SIAM Journal on Scientific Computing* 42.1 (2020), A534–A558.
- [117] Z. Zhang. “A gentle introduction to artificial neural networks”. In: *Annals of translational medicine* 4.19 (2016).
- [118] J. M. Zurada. *Introduction to artificial neural systems*. Vol. 8. West publishing company St. Paul, 1992.

## **APPENDIX**

## APPENDIX

### A

# DERIVATION OF GRADIENT OF THE OUTPUT KL MODES

Note that the weak form of the forward problem (4.16) is given by

$$\left\langle \frac{e^{\hat{a}(\mathbf{x}, \xi)}}{\mu} \nabla p, \nabla \tilde{p} \right\rangle - \langle h, \tilde{p} \rangle_{\Gamma_N} = 0 \quad \text{for all } \tilde{p} \in V$$

where

$$V = \{v \in H^1(\mathcal{D}) : v|_{\Gamma_N} = 0\},$$

$h = \frac{Q}{2\pi R_{needle}}$ ,  $\langle \cdot, \cdot \rangle$  is the  $L^2(\mathcal{D})$  inner product, and  $\langle \cdot, \cdot \rangle_{\Gamma_N}$  is the  $L^2(\Gamma_N)$  inner product. Note that we let the permeability field be  $\kappa(\mathbf{x}, \xi) = e^{\hat{a}(\mathbf{x}, \xi)}$ .

**Computing gradient of  $p_k$ 's.** To compute the gradient we follow a formal Lagrange ap-

proach. We consider the Lagrangian

$$\mathcal{L}(p, \xi, q) = \frac{1}{\sqrt{\lambda_k(C_f)}} \int_{\mathcal{D}} (p - \tilde{p}) \phi_k dx + \left\langle \frac{e^{\hat{a}(x, \xi)}}{\mu} \nabla p, \nabla q \right\rangle - \langle h, q \rangle_{\Gamma_N}.$$

Here  $q$  is a Lagrange multiplier, which in the present context is referred to as the adjoint variable. Taking variational derivatives of  $\mathcal{L}$  with respect to  $q$ , and  $p$ , give the state and the adjoint equations, respectively. In particular, the adjoint equation is found by considering

$$\frac{d}{d\epsilon} \mathcal{L}(p + \epsilon \tilde{p}, \xi, q) |_{\epsilon=0} = 0, \quad \text{for all } \tilde{p} \in V.$$

This gives the weak form of the adjoint equation,

$$\frac{1}{\sqrt{\lambda_k(C_f)}} \langle \tilde{p}, \phi_k \rangle + \left\langle \frac{e^{\hat{a}(x, \xi)}}{\mu} \nabla \tilde{p}, \nabla q \right\rangle = 0, \quad \text{for all } \tilde{p} \in V.$$

The strong form of the adjoint equation is

$$\begin{aligned} -\nabla \cdot \left( \frac{e^{\hat{a}(x, \xi)}}{\mu} \nabla q \right) &= -\frac{1}{\sqrt{\lambda_k(C_f)}} \phi_k \quad \text{in } \mathcal{D}, \\ q &= 0 \quad \text{on } \Gamma_D, \\ \frac{e^{\hat{a}(x, \xi)}}{\mu} \nabla q \cdot n &= 0 \quad \text{on } \Gamma_N. \end{aligned} \tag{A.1}$$

Letting  $p$  and  $q$  be the solutions of the state and adjoint equations respectively, for the gradient, we note that in a direction  $\tilde{\xi} \in \mathbb{R}^{N_p}$

$$(\nabla_{\tilde{\xi}} p_k) \cdot \tilde{\xi} = \frac{d}{d\epsilon} \mathcal{L}(p, \xi + \epsilon \tilde{\xi}, q) |_{\epsilon=0} = \left\langle (\hat{a}(x, \tilde{\xi}) - a_0(x)) \frac{e^{\hat{a}(x, \xi)}}{\mu} \nabla p, \nabla q \right\rangle, \quad \tilde{\xi} \in \mathbb{R}^{N_p}. \tag{A.2}$$

In particular, letting  $\tilde{\xi}$  be the  $j$ th coordinate direction in  $\mathbb{R}^{N_p}$ , we get the adjoint based expression for the partial derivatives of  $p_k$ 's:

$$\frac{\partial p_k}{\partial \xi_j} = \frac{\sqrt{\lambda_j(C_a)}}{\mu} \int_{\mathcal{D}} e_j(\mathbf{x}) e^{\hat{a}(\mathbf{x}, \xi)} \nabla p(\mathbf{x}) \cdot \nabla q(\mathbf{x}) d\mathbf{x}. \tag{A.3}$$