INTERPRETABLE MACHINE LEARNING

FOR ADDITIVE MANUFACTURING

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Raquel de Souza Borges Ferreira

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

May 2019

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Arman Sabbaghi, Chair

Department of Statistics, Purdue University

Dr. Vinayak Rao

Department of Statistics, Purdue University

Dr. Hao Zhang

Department of Statistics, Purdue University

Dr. Qiang Huang

Daniel J. Epstein Department of Industrial and Systems Engineering, University of Southern California

Approved by:

Dr. Jun Xie

Head of the Graduate Program

To my parents and grandparents.

ACKNOWLEDGMENTS

I have been incredibly lucky in my life to have had such an immense group of fantastic people to be thankful for. I've come to learn that there are such things as good problems. Having too many people to be grateful for, but too little space to be able to cite everyone, is one of those problems. I can only hope to do some justice to everyone in such small space in my dissertation.

First, I would like to express my deepest gratitude to my advisor, Dr. Arman Sabbaghi. His mentoring throughout this process has been greatly appreciated. I am very thankful for his generosity with his time, always helping me in every way. Our discussions about statistics have provided me with a deeper appreciation for the field and for understanding the underlying concepts. I am truly grateful for his confidence in me and his belief in my potential as a fellow statistician.

I would like to thank my committee members for their valuable input, support, and generosity with their time. I have been fortunate to have had some great mentors and professors during my graduate studies. In particularly, I would like to thank Dr. Doerge, Dr. Cayon, Dr. Marcos Prates and Dr. Renato Assunção. Dr. Cayon, your friendship and mentorship throughout my PhD have meant more than I can express in mere words.

I'd like to take a few sentences here to thank all the staff in the Statistics department. Pursuing a PhD is already challenging enough as it is, and having their support and help dealing with bureaucracies and such things was tremendously helpful.

I have made some incredible friends during my time at Purdue. These wonderful friends have helped me keep grounded, and made me take time to laugh and re-energize, besides providing good and helpful discussions. In special, I would like to thank Gui, Dé, Dex, Zoe, Maria, EonYoung, Jessi, Hilda, Barret, Tracy, Kelly-

Ann, Rodrigo, Lu, Hyoeun, Dominique, Ben, Sandy, and Dr. Sabbaghi's supportive research group.

When talking about friends, I must thank and mention the incredible friends in Brazil that stayed with me despite distance, and the constant *saudades*. They have been my safe haven from the PhD. My special thanks to Yan, Isabela, Dani, Sabrina, and the Leste/Oeste team – in particular Luís, Lívia, and Larissa.

I must thank my amazing family. I am deeply grateful for my parents for all they've done for me throughout my entire life. They've been the best parents a person could ask for, and being apart from them these last years has not been easy in any way, shape, or form. They are my heroes, and I can't ever thank them enough. I couldn't have done this without the support, love, and help from my awesome brothers and sisters. I am very grateful for having such amazing grandparents as inspiration, as well as the support and friendship of cousins, aunts, and uncles. In particularly, thank you Gi, Camila, Carole, Rodrigo, tia Vilma, tia Soninha, tia Biá, and tio Ronald! In addition, I must thank the world's cutest goddaughter, Bia, for the all happiness you've brought to my life!

During my years at Purdue, I was tremendously fortunate to have been accepted into the most terrific family. The Amstutz family has taken me in and given me a home so far from home. They have filled my days with love, support, laughter, and let me borrow their ears and shoulders many times. I am extremely grateful for them for taking me in, and letting me be a part of their marvelous family. Thank you Sue, Joel, Molly, Jana, Taylor, Jordan, Finn, Isla, Mildred, Speedo, and everyone in their family!

Lastly, but certainly not least, I would like to acknowledge my wonderful husband Kevin. I am extremely fortunate to have such an incredible, inspiring, understanding, and supportive person in my life. Kev, thank you for taking care of my heart, body, and mind while I focused on taking care of a small part of my brain. I can never thank you enough for all you have done and keep doing for me. You truly are the most incredible person I've known.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

ABBREVIATIONS

AM	Additive manufacturing
ELM	Extreme learning machine
GAME	Generalized average magnitude predictive comparison
GEAR	Generalized average predictive comparison
ICE	Individual conditional expectation
iGAME	Individual generalized average magnitude predictive comparison
iGEAR	Individual generalized average predictive comparison
IM	Interpretable mapper
LIME	Local interpretable model-agnostic explanations
ML	Machine learning
NN	Neural network
TEA	Total equivalent amount

ABSTRACT

Ferreira, R. S. B. Ph.D., Purdue University, May 2019. Interpretable Machine Learning for Additive Manufacturing. Major Professor: Arman Sabbaghi.

This dissertation addresses two significant issues in the effective application of machine learning algorithms and models for the physical and engineering sciences. The first is the broad challenge of automated modeling of data across different processes in a physical system. The second is the dilemma of obtaining insightful interpretations on the relationships between the inputs and outcome of a system as inferred from complex, black box machine learning models.

Automated Geometric Shape Deviation Modeling for Additive Manufacturing Systems

Additive manufacturing systems possess an intrinsic capability for one-of-a-kind manufacturing of a vast variety of shapes across a wide spectrum of processes. One major issue in AM systems is geometric accuracy control for the inevitable shape deviations that arise in AM processes. Current effective approaches for shape deviation control in AM involve the specification of statistical or machine learning deviation models for additively manufactured products. However, this task is challenging due to the constraints on the number of test shapes that can be manufactured in practice, and limitations on user efforts that can be devoted for learning deviation models across different shape classes and processes in an AM system. We develop an automated, Bayesian neural network methodology for comprehensive shape deviation modeling in an AM system. A fundamental innovation in this machine learning method is our new and connectable neural network structures that facilitate the transfer of prior knowledge and models on deviations across different shape classes and AM processes.

Several case studies on in-plane and out-of-plane deviations, regular and free-form shapes, and different settings of lurking variables serve to validate the power and broad scope of our methodology, and its potential to advance high-quality manufacturing in an AM system.

Interpretable Machine Learning

Machine learning algorithms and models constitute the dominant set of predictive methods for a wide range of complex, real-world processes. However, interpreting what such methods effectively infer from data is difficult in general. This is because their typical black box natures possess a limited ability to directly yield insights on the underlying relationships between inputs and the outcome for a process. We develop methodologies based on new predictive comparison estimands that effectively enable one to "mine machine learning models, in the sense of (a) interpreting their inferred associations between inputs and/or functional forms of inputs with the outcome, (b) identifying the inputs that they effectively consider relevant, and (c) interpreting the inferred conditional and two-way associations of the inputs with the outcome. We establish Fisher consistent estimators, and their corresponding standard errors, for our new estimands under a condition on the inputs' distributions. The significance of our predictive comparison methodology is demonstrated with a wide range of simulation and case studies that involve Bayesian additive regression trees, neural networks, and support vector machines. Our extended study of interpretable machine learning for AM systems demonstrates how our method can contribute to smarter advanced manufacturing systems, especially as current machine learning methods for AM are lacking in their ability to yield meaningful engineering knowledge on AM processes.

The issues discussed in this dissertation are addressed in the following three papers. The second paper was awarded the INFORMS 2018 Quality, Statistics, and Reliability Section Best Student Paper Award, and was a Finalist of the INFORMS 2018 Data Mining Section Best Theoretical Paper Competition. All three papers involve machine learning across different shapes and stereolithography process settings

in a real-life additive manufacturing (AM) system.

1. Ferreira R., Sabbaghi A., Huang Q. (2019) Automated geometric shape deviation modeling for additive manufacturing systems via Bayesian neural networks. *Conditionally accepted at IEEE Transactions on Automation Science and Engineering.* (Chapter 2)

2. Ferreira R., Sabbaghi A. (2019) Predictive comparisons for screening and interpreting inputs in machine learning. *Under review.* (Chapter 3)

3. Ferreira R., Sabbaghi A, Prates, M. O. (2019) Generalized predictive comparisons for interpreting complex models *To be submitted.* (Chapter 4)

1. INTRODUCTION

1.1 The necessity of automated shape deviation modeling in an additive manufacturing system

Additive manufacturing (AM) holds the promise of direct digital manufacturing of shapes with highly complex geometries, materials, and functionalities (Bourell et al., 2009; Gibson et al., 2009; Campbell et al., 2011; Huang et al., 2013). A major trajectory of this technology, which constitutes a rapidly evolving domain of cyber-physical systems, is the development of AM systems that seamlessly integrate computer-aided design (CAD) models and connected physical AM processes (Buckholtz et al., 2015; GTAI, 2017; Wu et al., 2015). Several quality control issues exist in AM machines that impede the advancement of AM systems. One such major issue is geometric accuracy control for the inevitable shape deviations that occur due to material phase changes, complicated layer interactions, and process variations inherent in AM (Wang et al., 1996). A crucial requirement for the successful operation of AM systems is the ability to comprehensively control shape deviations across the different CAD inputs and process conditions for the connected AM machines.

One general class of geometric accuracy control strategies in AM is based on the use of statistical deviation models to derive compensation plans, or modifications to CAD models that are predicted to reduce the deviations in manufactured products (Huang et al., 2015b, 2014). Achieving comprehensive deviation control in an AM system with this strategy is complicated by four significant issues that result from the nature and capability of AM for one-of-a-kind manufacturing. First is the wide variety of shapes with varying geometric complexities that are of interest for manufacture. Second is the vast spectrum of AM processes or conditions that can yield fundamentally distinct deviations for products manufactured from the same CAD

model. Third is the fact that only a small sample of test shapes, typically in the single digits, can possibly be manufactured for any AM process (Sabbaghi et al., 2018). Finally, the effort that an AM system operator can devote to learn shape deviation models is typically limited. Comprehensive deviation control via compensation plans in an AM system thus requires a method that can leverage previously developed deviation models for different shapes and processes to automate deviation modeling for new shapes and processes using only a small sample of products.

Fig. 1.1 illustrates this requirement for an AM system with two processes A and B and two shapes 1 and 2. For shape $s \in \{1, 2\}$ manufactured under process $p \in \{\mathrm{A}, \mathrm{B}\}$, we let $f_{s,p} : \mathcal{X} \to \mathcal{Y}$ denote its deviation model that returns the expected deviations (with range $\mathcal{Y}$ denoting the set of deviations for an entire manufactured product) under different compensation plans in domain $\mathcal{X}$. First consider the case in which requests for shape 1 are assigned to both A and B. Suppose $f_{1,\mathrm{A}}$ has previously been specified but $f_{1,\mathrm{B}}$ has not. Given the system's constraints and limited resources for fulfilling the requests, it then becomes important to reduce the effort in specifying $f_{1,\mathrm{B}}$ by automatically adapting $f_{1,\mathrm{A}}$ to B based on a small sample of products manufactured under it. Now consider the case in which a request for the new, more complicated shape 2 is assigned to B. Automated learning of the corresponding deviation model $f_{2,\mathrm{B}}$ can be performed more effectively by leveraging all of the previously specified models for the different shapes and processes that share similar geometric features with shape 2 under B, and using a small sample of new shapes to learn deviation features unique to it.

Current shape deviation modeling techniques cannot address all of the previously described features of AM systems. The methods of Tong et al. (2003, 2008) specify independent polynomial deviation models for each direction of a shape, and their applications are limited to particular shapes under a single process. Huang et al. (2015b) devised a distinct functional method that decouples geometric shape complexity from deviation modeling, but the focus was on individual shapes, with no consideration paid to specifying models for new shapes or processes in an automated manner. Meth-

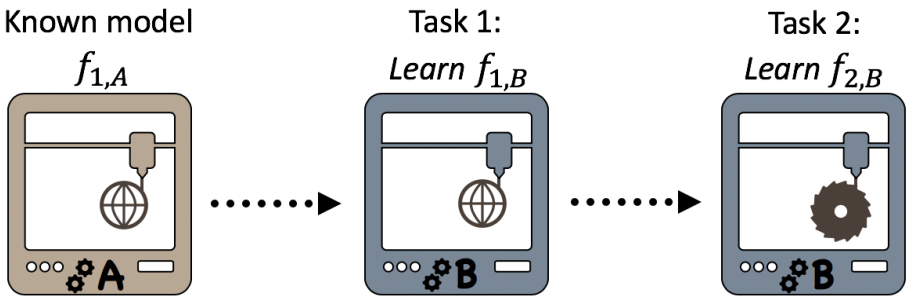


Figure 1.1.: An AM system with two processes A and B and two shapes 1 and 2 for manufacture. The tasks in this system are to learn the deviation model $f_{1,\mathrm{B}}$ for shape 1 under B using knowledge of the deviation model $f_{1,\mathrm{A}}$ for this shape under A, and to specify the deviation model $f_{2,\mathrm{B}}$ of a new shape 2 under B using knowledge of all of the models for the previously manufactured products, in an automated manner.

ods of specifying models for just new classes of shapes based on the combination of the latter approach with the concept of modular deviation features were developed in Huang et al. (2014), Luan and Huang (2017), and Sabbaghi et al. (2018). To address the requirement of deviation modeling across processes, a statistical framework of effect equivalence for model transfer was formulated by Sabbaghi and Huang (2018), and utilized in Jin et al. (2016) and Sabbaghi and Huang (2016) to specify models across distinct shapes and processes. However, all of these methods can incur a great deal of effort to implement and do not readily enable automated modeling for an AM system. Also, existing automatic modeling techniques do not address the first two features of AM systems. For example, the approaches of Schmutzler et al. (2016b,a) to automate deviation modeling of a surrounding cuboid object based on B-splines and the free-form deformation concept is limited to specified shapes and processes. An automated and efficient methodology for comprehensive shape deviation modeling in an AM system remains to be developed.

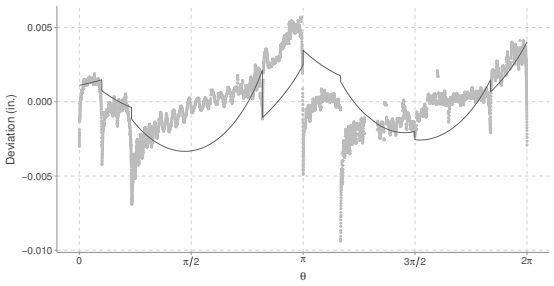
We address this challenge in Chapter 2 via our new methodology that is based on a structured class of Bayesian neural networks (NNs), specifically, Bayesian extreme learning machines (ELMs, Huang et al., 2004), that we developed. Our methodology utilizes point-cloud measurement data collected from a small sample of test shapes,

and does not require detailed knowledge of AM processes. By combining the collected data from all new shapes and process, and previous geometric shape deviation models, we are able to perform automatic model transfer. In comparison to the previously described techniques, it can dramatically facilitate automated modeling of both in-plane and out-of-plane deviations for different shapes under distinct processes in an AM system. This advantage is illustrated by the case study of modeling in-plane deviations for the irregular polygon in Fig. 1.2(a) based on data and models for a small set of cylinders and a single regular pentagon manufactured using stereolithography (all of which are detailed in Section 2.3). Figs. 1.2(b), 1.2(c), and 1.2(d) contain the deviation model fits (represented by black lines) obtained respectively from the methods of Huang et al. (2014), Sabbaghi et al. (2018), and our approach. By inspection of the alignment of the black lines with the shape deviations (represented by gray dots), we see that our method yields better predictive performance than that of Huang et al. (2014), and comparable performance to that of Sabbaghi et al. (2018). Furthermore, in contrast to the other two approaches, our method was automated and required fewer user inputs and efforts for its computational implementation. For example, the method of Sabbaghi et al. (2018) requires the user to specify a complete Bayesian hierarchical model for the irregular polygon, and the computation incurs great efforts.

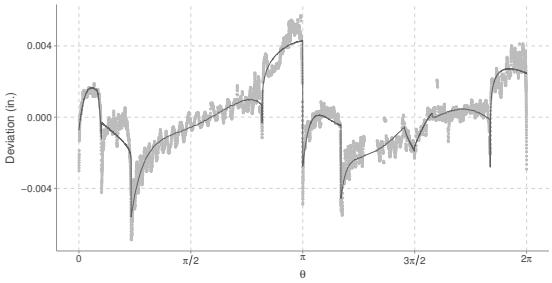
1.2 The necessity of interpretable machine learning

Another significant issue in AM systems is deriving interpretable physical insights on the underlying relationships of the AM inputs with the outcome, or substantive and meaningful engineering knowledge on the system, from machine learning algorithms and models fitted to their data. This issue is related to a broader set of challenges with machine learning algorithms and models, which generally enjoy great renown for their superb predictive capabilities in complex, real-world processes (Deng et al., 2014; Libbrecht and Noble, 2015). Specifically, the continued, strong adoption

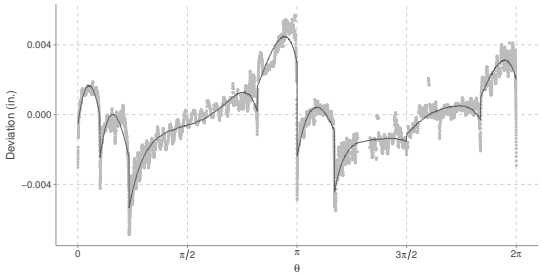
(a) Additively manufactured irregular polygon.



(b) The model obtained from the method of Huang et al. (2014).



(c) The model obtained from the method of Sabbaghi et al. (2018).



(d) The model obtained from our Bayesian ELM method.

Figure 1.2.: (a) An irregular polygon whose in-plane deviations are to be modeled based on the data and models for cylinders and a single regular pentagon. (b) The deviation model obtained from Huang et al. (2014). (c) The deviation model obtained from Sabbaghi et al. (2018). (d) The deviation model obtained from our method.

of machine learning methods has made the issue of their interpretability critically important, because one must generally comprehend an algorithm or model's capabilities beyond prediction. Two requirements for achieving this comprehension are interpreting which inputs a model effectively considers relevant, and the associations it infers between them and the outcome (Lipton, 2016; Doshi-Velez and Kim, 2017). These two interpretability tasks are challenging to accomplish due to the black box nature of machine learning methods. In our particular context of AM systems, our new Bayesian NN methodology can yield accurate predictions of shape deviations in an AM system, but its structure does not immediately yield physical insights or engineering knowledge on the relationships between the AM inputs and shape deviation.

One class of interpretability methods is based on evaluations and comparisons of the predictions obtained from an algorithm (Doshi-Velez and Kim, 2017). For example, the Local Interpretable Model-agnostic Explanations (LIME, Ribeiro et al., 2016) method belongs to this class, and it explains a single prediction by fitting a more interpretable model (e.g., a linear regression or a decision tree) to local perturbations of its inputs. Another such method is the Individual Conditional Expectation (ICE, Goldstein et al., 2015) plot, which builds upon partial dependence plots (Friedman, 2001) to account for interactions and other complicated relationships in black box models. Lundberg and Lee (2016) provided a model-agnostic representation of an input's importance that unifies several of these methods, e.g., the LIME method with those based on expectation Shapley values (Lipovetsky and Conklin, 2001; Štrumbelj and Kononenko, 2014). However, current methods from this class generally have the limitation of being local, in that their focus is on explaining single predictions and small domain regions. Another limitation is their typical assumption that simpler, more interpretable models can be fit to local perturbations. Ribeiro et al. (2018) recently proposed an extension of LIME that combines local methods with if-then decision rules, referred to as "anchors", to address this particular limitation and enable more generalizable interpretations for complex classifiers. However, their method requires one to design distributions for local pertubations that can uncover a machine learning algorithm's behavior, which can be difficult to accomplish. A final limitation of many of these methods that is important to recognize is their inability to quantify inferential uncertainties for any of their derived interpretations.

Gelman and Pardoe (2007) proposed a global interpretability method based on average predictive comparisons that capture expected differences in the outcome associated with unit increases in the inputs. They demonstrated the utility of their approach for interpreting elaborate hierarchical logistic regression models. However, their method is not particularly effective in the case of non-monotonic relationships between inputs and the outcome, and it cannot assess interactions between inputs. Although Gelman and Pardoe (2007, p. 35–36) briefly mentioned possible remedies

for the misleading conclusions that can result from the application of their method in situations with non-monotonic relationships, they did not further explore or extend their approaches for screening relevant inputs in such cases. These issues are crucial for interpreting a complex model that involves many inputs. A final limitation of the current formulation of average predictive comparisons is that it can only handle one input at a time. Thus, it is unable to assess the relationships between several inputs simultaneously with the outcome, or those between functional forms of inputs with the outcome. These relationships are typically of more interest in many contexts. For example, when attempting to interpret an image classifier that takes as its inputs the image's pixels, analyzing a group of pixels is generally of more interest than analyzing each individual pixel.

We first present in Chapter 3 a predictive comparison methodology for global interpretability in machine learning that can be readily applied to identify relevant inputs, infer their interactions as well as non-monotonic relationships with the outcome, and quantify the uncertainty of the inferences. This method extends the work of Gelman and Pardoe (2007) with a supplementary two-step procedure that delves deeper into a model to yield more substantive interpretations of it. Specifically, the first step screens for relevant inputs using new predictive comparisons to capture the magnitude of the inputs' average effects, and the second step infers new conditional and two-way interaction average predictive comparisons between relevant inputs. The new predictive comparisons developed for these two steps were not previously specified or considered by Gelman and Pardoe (2007). In Chapter 4, we extend the previous methodology so as to obtain interpretations of the relationships between multiple inputs simultaneously, and/or functional forms of the inputs, with the outcome. This generalized predictive comparison methodology is particularly significant for AM systems because it can yield more useful insights into the inferences of machine learning algorithms and models for additively manufactured products with complex geometries. Ultimately, both predictive comparison methodologies can yield deeper interpretations of the complex relationships of inputs with the outcome for a large

variety of machine learning algorithms and models compared to those obtained by means of other interpretability methods.

1.3 Outline of dissertation

The rest of this dissertation is organized as follows. We begin in Chapter 2 by addressing the issue of automated shape deviation modeling in AM systems. In Chapter 2.1 we review geometric shape deviations, effect equivalence, modular deviation features, NNs, and ELMs. These concepts underlie the development of our Bayesian ELM methodology in Chapter 2.2. As we describe in Chapter 2.2, a unique innovation in our method is the use of engineering principles to specify new and connectable NN structures for ELMs that eliminate the ad-hoc tuning methods typically associated with NNs (e.g., those in Goodfellow et al. (2016)), and advance automated deviation modeling. The power and broad scope of our method are illustrated in Chapter 2.3 via several case studies involving different shapes and stereolithography processes. A concluding discussion on the broader scope of the results of these case studies is in Chapter 2.4.

The issue of global interpretability of machine learning algorithms and models is addressed in Chapters 3 and 4 via our predictive comparison methodology and generalized predictive comparison methodology, respectively. Chapter 3 is organized in the following manner. Formal descriptions of the notations and assumptions in our methodology, and of standard average predictive comparisons, are contained in Chapter 3.1. Our new predictive comparison methodology is developed in Chapter 3.2. To facilitate the exposition, our formal definitions of the new average predictive comparison estimands, theoretical results on the Fisher (1922) consistency of our estimators for these estimands, and expressions for the standard errors of the estimators are provided in Chapters 3.2.1 and 3.2.2, respectively. The broad scope of our methodology is illustrated in Chapter 3.3 by means of simulation and real-life case studies that involve Bayesian neural networks, the Bayesian additive regression

tree (BART, Chipman et al., 2010) algorithm, and support vector machines (SVM, Vapnik, 1998). An extended case study on the application of our methodology for interpreting the Bayesian NNs used for automated geometric shape deviation modeling in AM systems developed in Chapter 2 is presented in Chapter 3.4. The material for generalized predictive comparisons in Chapter 4 is similarly organized as that of Chapter 3. Specifically, formal descriptions of the notations and assumptions are in Chapter 4.1, the methodology is developed in Chapter 4.2, illustrative case studies are in Section 4.3, and an extended case study on AM systems is in Chapter 4.4. A significant feature of the applications in Chapters 3.4 and 4.4 is that they demonstrate the potential contributions of our methodologies to the effective future use of ML algorithms and models in smart advanced manufacturing systems.

Concluding remarks on the potential impacts of our methodologies to facilitate smart and efficient control of an AM system for a community of connected users, and contribute to the effective and appropriate future use of ML methods in practice, are contained in Chapter 5

2. AUTOMATED GEOMETRIC SHAPE DEVIATION MODELING FOR AM SYSTEMS VIA BAYESIAN NEURAL NETWORKS

2.1 Background

2.1.1 Functional representation of shape deviations

Geometric measurements of an additively manufactured product are collected in the point-cloud format that uses Cartesian coordinates defined with respect to physical axes printed directly on the product. We transform point-cloud data by means of the functional representations of in-plane and out-of-plane deviations first formulated by Huang et al. (2015b) and Huang (2016), respectively. In-plane deviations refer to the two-dimensional, horizontal deviations of a product that has a negligible vertical height. The top and bottom surface deviations are approximately identical in this case, and represented using polar coordinates $(\theta, r(\theta))$ where θ denotes the angle of a point and $r(\theta)$ its radius (Fig. 2.1). Out-of-plane deviations refer to the vertical deviations of fully three-dimensional shapes. For products with negligible lengths/widths, i.e., vertical slices, polar coordinates are used to represent their out-of-plane deviations (Fig. 2.1).

We demonstrate our methodology in this paper on in-plane deviations of shapes with negligible heights, and out-of-plane deviations of shapes with negligible widths. In both cases each point on a product is identified by an angle θ. The CAD model for a shape s is defined under the polar coordinate representation by a nominal radius function $r_s^{\text{nom}} : [0, 2\pi] \to \mathbb{R}_{\geq 0}$ with argument θ. As in Sabbaghi et al. (2018), we assume that each shape s has an associated collection of known parameters γ_s that define r_s^{nom}. For example, γ_s has one entry for a cylinder (namely, its nominal radius), and is a vector for other shapes. The observed radius for a point θ on product s

manufactured under process p is denoted by $r_{s,p}^{\text{obs}}(\theta)$, and its deviation is defined as $\Delta_{s,p}(\theta) = r_{s,p}^{\text{obs}}(\theta) - r_s^{\text{nom}}(\theta)$. An advantage of this representation is that it yields a consistent framework to specify statistical models for in-plane and out-of-plane deviations of different shapes and processes in an AM system. It is important to note that out-of-plane deviations are generated under more complex physical phenomenon (e.g., interlayer bonding effects Jin et al. (2016)) than in-plane deviations.

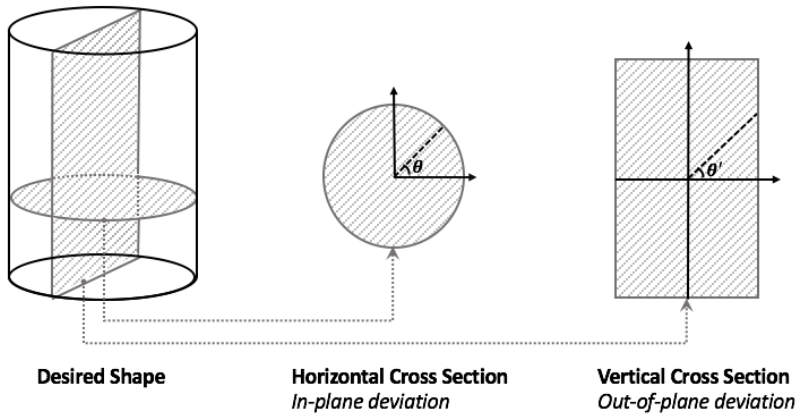


Figure 2.1.: Illustrations of in-plane and out-of-plane geometric shape deviations as defined in Huang et al. (2015b) and Huang (2016), respectively.

As stated in Section 1.1, shape deviations are modeled to derive compensation plans that can enable geometric accuracy control. The compensation factor is defined for each θ as the addition or subtraction of material in the original CAD model at that point. We augment the above notations from $r_{s,p}^{\text{obs}}(\theta)$ to $r_{s,p}^{\text{obs}}(\theta, x)$ and $\Delta_{s,p}(\theta)$ to $\Delta_{s,p}(\theta, x) = r_{s,p}^{\text{obs}}(\theta, x) - r_s^{\text{nom}}(\theta)$, respectively, to account for θ being given compensation $x \in \mathbb{R}$. Compensation also plays an important role for transferring deviation models across processes, which we describe next.

2.1.2 Statistical effect equivalence framework for model transfer across AM processes

Our methodology incorporates aspects of the statistical effect equivalence framework of Sabbaghi and Huang (2018) to perform model transfer across processes in an AM system. The general effect equivalence concept corresponds to the engineering phenomenon in which the effect of a process condition change can be equivalently generated by changing a particular factor (e.g., compensation) under a fixed process condition, and was inspired by the investigation of Wang et al. (2005) on a machining process. To illustrate this framework, consider a shape s with nominal radius function r_s^{nom} manufactured under two processes A and B, with in-plane deviations of interest. Suppose that the deviation model under B has yet to be specified, and that the deviation model under A has been specified as

$$\Delta_{s,\text{A}}(\theta, x) = \delta_{s,\text{A}}(\theta, x) + \epsilon_{s,\text{A}}(\theta), \tag{2.1}$$

where $\delta_{s,\text{A}}(\theta, x)$ is the systematic (or expected) deviation at θ with compensation x, and the $\epsilon_{s,\text{A}}(\theta)$ are random variables representing high-frequency deviation components with expectation $\mathbb{E}\{\epsilon_{s,\text{A}}(\theta)\} = 0$ for all θ Huang et al. (2015b); Sabbaghi et al. (2018). We exclude model parameters in equation (2.1) to simplify the exposition. In terms of the notation from Section 1.1, $\delta_{s,\text{A}}(\theta, x)$ also specifies $f_{s,\text{A}}$ to model the expected in-plane deviation profile for the entire shape under process A. Then under effect equivalence, $\delta_{s,\text{A}}$ is transferred to model deviations for process B via a hypothesized function $T_{s,\text{B}\to\text{A}} : [0, 2\pi] \times \mathbb{R} \to \mathbb{R}$ in the manner

$$\Delta_{s,\text{B}}(\theta, x) = \delta_{s,\text{A}}(\theta, T_{s,\text{B}\to\text{A}}(\theta, x)) + \epsilon_{s,\text{B}}(\theta). \tag{2.2}$$

In equation (2.2), $T_{s,\text{B}\to\text{A}}$ returns a compensation for each θ such that, in expectation, the deviation of θ with compensation x when the shape is manufactured under B is equivalent to the deviation of θ with compensation $T_{s,\text{B}\to\text{A}}(\theta, x)$ when the shape is

manufactured under A. Function $T_{s,\mathrm{B}\to\mathrm{A}}$ is referred to as the total equivalent amount (TEA) of B in terms of compensation with respect to the mean of A for shape s. This general concept of the TEA connects new and old processes in AM systems. The task of model transfer across AM processes is thus reduced by effect equivalence to learning the unknown TEA for a new process from a small sample of shapes manufactured under it and the fitted model for a previous process. Once the TEA is learned, it can be entered into $\delta_{s,\mathrm{A}}$ as in equation (2.2) to perform the model transfer. Bayesian methods for this learning task were developed in Sabbaghi and Huang (2018), but they can require a great deal of effort to implement, and are not automated for applications in AM systems. We describe in Section 2.2 how we utilize the effect equivalence framework to design a new and simple NN architecture in our methodology that directly enables automated learning of TEAs, and hence model transfer, based on small samples of products.

2.1.3 Modular deviation features for model transfer across different shapes

Our methodology also incorporates aspects of the "cookie-cutter" framework of Huang et al. (2014) to perform model transfer across different shapes in an AM system based on their modular deviation features. The key idea of the framework is to capture the carving out of a new shape from an old shape. As the new shape is carved out, a new local deviation feature is introduced, with the old shape capturing a more global deviation feature. To illustrate this idea, consider two shapes 1 and 2 with nominal radius functions r_1^{nom} and r_2^{nom}, respectively, that are manufactured under a single process p without compensation and whose in-plane deviations are of interest. Suppose that the deviation model for shape 2 has yet to be specified, and that the deviation model for shape 1 has been specified as

$$\Delta_{1,p}(\theta) = \delta_{1,p}(\theta) + \epsilon_{1,p}(\theta), \tag{2.3}$$

where $\delta_{1,p}(\theta)$ and $\epsilon_{1,p}(\theta)$ are defined as in equation (2.1). Then a new, hypothesized deviation feature $\delta_{2,p}(\theta)$ is introduced under this framework to specify the model for shape 2 as

$$\Delta_{2,p}(\theta) = \delta_{1,p}(\theta) + \delta_{2,p}(\theta) + \epsilon_{2,p}(\theta). \tag{2.4}$$

Feature $\delta_{2,p}$ is referred to as a cookie-cutter basis function, or local deviation feature, for shape 2, and $\delta_{1,p}$ is the shared global deviation feature that connects the shapes Sabbaghi et al. (2018). Equation (2.4) can also be viewed as specifying the unified model

$$\Delta_{s,p}(\theta) = \delta_{1,p}(\theta) + \mathbb{I}(s = 2)\delta_{2,p}(\theta) + \epsilon_{s,p}(\theta) \tag{2.5}$$

for these shapes, where $\mathbb{I}(\cdot)$ is the indicator function. Additional details for this framework are in (Huang et al., 2014) and Sabbaghi et al. (2018). The task of model transfer across shapes under a single AM process is thus reduced by this framework to learning the unknown $\delta_{2,p}(\theta)$ in equations (2.4) and (2.5) from a small sample of new shapes and the fitted model for shape 1. Huang et al. (2014) considered pre-specified classes of local deviation features that may not yield successful model transfer for complicated shapes in AM systems. Sabbaghi et al. (2018) developed an adaptive Bayesian method to learn local deviation features that are applicable to a wide range of shapes. However, the latter method is not automated and can incur significant effort to learn appropriate models. We describe in Section 2.2 how we incorporate the concept of modular deviation features in our methodology's NN architecture to address the task of automated model transfer across shapes.

2.1.4 Overview of neural networks and extreme learning machines

Our methodology utilizes a new class of single-hidden layer feedforward NNs that we developed to automate and facilitate model transfer across both processes and shapes in an AM system. We briefly review NNs and ELMs here, and describe our new class of NNs in Section 2.2. To simplify this review, we let $\mathbf{y} = (y_1, \ldots, y_N)^\mathsf{T} \in$

$\mathbb{R}^N$ denote the outcomes for N units of analysis, and $\mathbf{z}_i \in \mathbb{R}^K$ the independent variables, or inputs, for unit $i = 1, \ldots, N$. In AM the typical units of analysis are the points θ_i, and $y_i = \Delta_{s,p}(\theta_i, x_i)$ for each point θ_i on a shape s manufactured under process p with compensation x_i. The choice of inputs depends on the product. For example, a useful set for an uncompensated cylinder with nominal radius r_1^{nom} and negligible height is $\mathbf{z}_i = (\theta_i, r_1^{\text{nom}})^\mathsf{T}$ Huang et al. (2015b). Also, a useful set for an uncompensated polygon with nominal radius function r_2^{nom} and negligible height is $\mathbf{z}_i = (\theta_i, r_2^{\text{nom}}(\theta_i), \text{edge}(\theta_i))^\mathsf{T}$, where $\text{edge}(\theta_i)$ denotes the edge containing θ_i Sabbaghi et al. (2018).

Neural networks enjoy the ability to learn complex relationships across a wide range of domains (Deng et al., 2014; Libbrecht and Noble, 2015). Inspired by the brain, NNs involve a composition of "hidden neurons" that are structured via different layers and connections amongst themselves. Although NNs allow great structural freedom in general, in practice it is not clear how a structure should be chosen for a particular data set, and so it is typically specified in an ad-hoc manner. The simplest NN structure is the single-hidden layer feedforward NN with additive hidden neurons and a single activation function $g : \mathbb{R} \to \mathbb{R}$, defined by

$$y_i = \sum_{m=1}^{M} \beta_m \; g\left(\alpha_{m,0} + \mathbf{z}_i^\mathsf{T} \boldsymbol{\alpha}_m\right) + \epsilon_i, \tag{2.6}$$

where the error terms ϵ_i are independent $\text{N}(0, \sigma^2)$ random variables, and the unknown parameters are σ^2, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_M)^\mathsf{T}$, $\alpha_{m,0}$ and $\boldsymbol{\alpha}_m = (\alpha_{m,1}, \ldots, \alpha_{m,K})^\mathsf{T}$ for $m = 1, \ldots, M$. An example of an activation function is the hyperbolic tangent $g(x) = (e^x - e^{-x})/(e^x + e^{-x})$. These NNs can be considered as universal approximators of nonlinear functions Hornik et al. (1989), and possess a wide scope of application due to their flexibility and generality. However, one of their limitations that prevents their immediate application for deviation modeling is that they can incur quite some effort to fit to deviations for complex geometries. For example, the traditional back-propagation algorithm for fitting NNs suffers from both slow convergence and the local

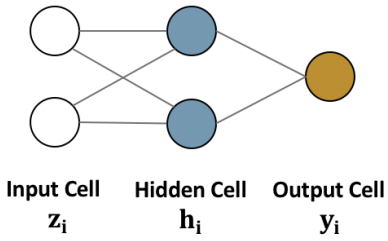


Figure 2.2.: An example of the structure of a standard ELM.

minimum problem (Huang et al., 2015a). Other limitations involve non-identifiability issues (Prieto et al., 2016).

Extreme learning machines are a class of single-hidden layer feedforward NNs that were developed by Huang et al. (2004) to address the limitations of standard NNs. An ELM does not attempt to infer all of the unknown parameters in equation (2.6). Instead, it first randomly sets the parameters in $\alpha_{m,0}$ and $\boldsymbol{\alpha}_m$, and then estimates $\boldsymbol{\beta}$ conditional on this selection (Huang et al., 2004). Thus an ELM reduces the original NN to a linear regression that simplifies model fitting, maintains the universal approximator property (Huang et al., 2006; Huang and Chen, 2007, 2008), and resolves non-identifiability issues. Fig. 2.2 illustrates ELMs, with $h_{i,m} = g(\alpha_{m,0} + \mathbf{z}_i^{\mathsf{T}}\boldsymbol{\alpha}_m)$ denoting hidden neuron m for unit i. In practice, little consideration is paid to the proper tuning of the random parameters (Huang et al., 2011), and they are usually drawn independently from the Uniform$(-1, 1)$ distribution (McDonnell et al., 2015). However, the distribution from which the random parameters are drawn impacts the ELM's generalizability and external validity (Tao et al., 2016). This impact is exacerbated when only small samples are available. Our methodology refines standard ELMs by making use of new techniques we developed to effectively address the requirements for automated deviation modeling in an AM system given small samples.

2.2 Bayesian ELM methodology for automated deviation modeling in an AM system

2.2.1 Outline of methodology

Our Bayesian ELM methodology for automated deviation modeling proceeds via four steps that are outlined below. Details for each step are provided in the following subsections. Fig. 2.3 illustrates our new and connectable ELM structures that enable comprehensive deviation modeling in an AM system.

1: *Model Deviations of Baseline Shape and Process*

Establish a baseline Bayesian ELM deviation model, $\delta_{s,p}$, for shape s from one class of shapes S manufactured under a fixed process p.

2: *Transfer Baseline Deviation Model to a New Process*

Transfer $\delta_{s,p}$ to a new process p' by learning a Bayesian ELM for the TEA $T_{s,p'\to p}$ using the fitted baseline model and data from process p'.

3: *Transfer Baseline Deviation Model to a New Shape*

Transfer $\delta_{s,p}$ to shapes s' from a new class of shapes S' manufactured under p by taking $\delta_{s,p}$ as the global deviation feature and learning a Bayesian ELM for the new local deviation feature $\delta_{s',p}$ using the fitted baseline model and data from shapes in S' manufactured under p.

4: *Transfer Baseline Model to a New Shape and Process* Transfer $\delta_{s,p}$ to shapes s' from class S' manufactured under p' by performing the previous two steps and combining their resulting Bayesian ELM models (Fig. 2.3).

2.2.2 Step One: Model deviations of baseline shape and process

The baseline Bayesian ELM model specified in the first step serves as a building block for subsequent deviation modeling. It is learned from a small number of

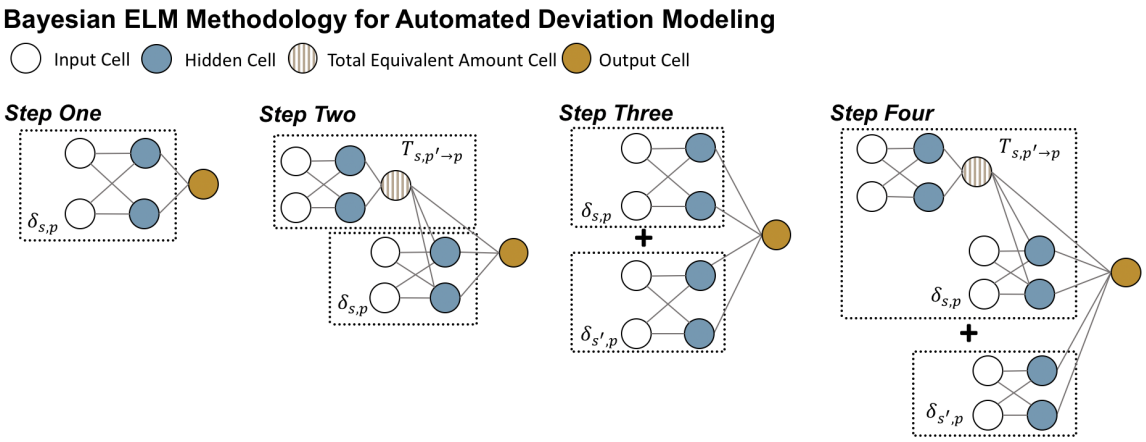


Figure 2.3.: The connectable NN structures in our methodology, where $\delta_{s,p}$ is the baseline deviation model for shape s under process p, $T_{s,p'\to p}$ is the TEA of p' in terms of compensation under p for s, and $\delta_{s',p}$ is the local deviation feature for s' under p.

products in class S manufactured under the fixed process p. To describe it, let $N_{S,p}$ denote the total number of points on all of the products manufactured under p, $M_{S,p}$ the chosen number of hidden neurons for the Bayesian ELM model, and g the activation function. For each point $i \in \{1,\ldots,N_{S,p}\}$, let $\mathbf{z}\left(\theta_i, r_{i,S}^{\text{nom}}, x_i\right) \in \mathbb{R}^K$ denote its vector of inputs that is a function of θ_i, the nominal radius function $r_{i,S}^{\text{nom}}$ for the product on which i resides, and compensation $x_i \in \mathbb{R}$ applied to point i. If compensation was not applied, $x_i \equiv 0$. Finally, for drawn hidden neuron parameters $\alpha_{m,k}^{(S,p)}$ $(m = 1,\ldots,M_{S,p}, k = 0,\ldots,K)$, let $\mathbf{H}_{S,p}$ be the $N_{S,p} \times M_{S,p}$ matrix whose (i,m) entry is $g\left(\alpha_{m,0}^{(S,p)} + \mathbf{z}\left(\theta_i, r_{i,S}^{\text{nom}}, x_i\right)^{\mathsf{T}} \boldsymbol{\alpha}_m^{(S,p)}\right)$ $(i = 1,\ldots,N_{S,p}, m = 1,\ldots,M_{S,p})$. Then the Bayesian ELM deviation model is

$$\mathbf{y}_{S,p} = \mathbf{x}_{S,p} + \mathbf{H}_{S,p}\boldsymbol{\beta}_{S,p} + \boldsymbol{\epsilon}_{S,p}, \tag{2.7}$$

where $\mathbf{y}_{S,p} \in \mathbb{R}^{N_{S,p}}$ is the vector of deviations, $\mathbf{x}_{S,p} \in \mathbb{R}^{N_{S,p}}$ is the vector of compensations, $\boldsymbol{\beta}_{S,p} \in \mathbb{R}^{M_{S,p}}$ is a vector of unknown parameters, and the error terms in $\boldsymbol{\epsilon}_{S,p}$ are independent and identically distributed $\mathrm{N}\left(0, \sigma_{S,p}^2\right)$ random variables with $\sigma_{S,p}^2$ unknown. The addition of compensation in equation (2.7) is informed by the

physical reasoning of Huang et al. (2015b). Our prior probability density function for the parameters is

$$p\left(\boldsymbol{\beta}_{S,p}, \sigma_{S,p}^2 \mid \tau_{S,p}^2\right) \propto \sigma_{S,p}^{-2} \tau_{S,p}^{-M_{S,p}} \exp\left(-0.5 \tau_{S,p}^{-2} \boldsymbol{\beta}_{S,p}^{\mathsf{T}} \boldsymbol{\beta}_{S,p}\right), \tag{2.8}$$

and our hyperprior probability density function for $\tau_{S,p}^2$ is the relatively non-informative Inverse-Gamma distribution $p\left(\tau_{S,p}^2\right) \propto \tau_{S,p}^{-6} \exp\left(-0.01 \tau_{S,p}^{-2}\right)$. The Gibbs algorithm (Geman and Geman, 1984) enables simple and rapid sampling from the posterior distribution of the parameters. Posterior predictions of deviations for S under p immediately follow from these draws, and are also used to transfer the baseline model to new processes.

Example 2.1. Consider in-plane deviations for uncompensated cylinders (shape class 1) manufactured under a process A, with $N_{1,\mathrm{A}} = 1000$ and $M_{1,\mathrm{A}} = 3$. We set $\mathbf{z}\left(\theta_i, r_{i,1}^{\mathrm{nom}}, x_i\right) = \left(\theta_i, r_{i,1}^{\mathrm{nom}}\right)^{\mathsf{T}}$, and $\mathbf{H}_{1,\mathrm{A}}$ is a 1000×3 matrix with entry (i, m) equal to $g\left(\alpha_{m,0}^{(1,\mathrm{A})} + \alpha_{m,1}^{(1,\mathrm{A})} \theta_i + \alpha_{m,2}^{(1,\mathrm{A})} r_{i,1}^{\mathrm{nom}}\right)$. If compensations were applied, we set $\mathbf{z}\left(\theta_i, r_{i,1}^{\mathrm{nom}}, x_i\right) = \left(\theta_i, r_{i,1}^{\mathrm{nom}} + x_i\right)^{\mathsf{T}}$ based on the reasoning of Huang et al. (2015b), and $\mathbf{H}_{1,\mathrm{A}}$ is defined as before.

In contrast to the standard ELM method of McDonnell et al. (2015), we utilize a new mechanism to draw the $\alpha_{m,k}^{(S,p)}$ that we developed to obtain better out-of-sample predictive performance. Specifically, we draw $\alpha_{m,k}^{(S,p)}$ from Uniform $(-a_k, a_k)$ $(m = 1, \ldots, M_{S,p}, k = 1, \ldots, K)$ with the $a_k > 0$ values tuned in an automated manner to avoid saturation of the hidden neurons based on the ratios of the standard deviations of the inputs and knowledge of the non-saturation regions of the activation function. To illustrate, consider Example 2.1 and suppose the hyperbolic tangent is the activation function. The saturation region corresponds to the absolute output of

this function being approximately 1 for inputs greater than 3 in absolute value. We set

$$b_\theta = \sqrt{\frac{1}{N_{S,p}-1}\sum_{i=1}^{N_{S,p}}\left(\theta_i-\bar{\theta}\right)^2}, \qquad b_r = \sqrt{\frac{1}{N_{S,p}-1}\sum_{i=1}^{N_{S,p}}\left\{r_{i,S}^{\text{nom}}\left(\theta_i\right)-\bar{r}\right\}^2},$$

$$a_1 = \frac{5b_\theta}{2\left(b_\theta+b_r\right)\max\left\{\theta_i : i=1,\ldots,N_{S,p}\right\}}, \text{ and } a_2 = \frac{2-a_1\max\left\{\theta_i : i=1,\ldots,N_{S,p}\right\}}{\max\left\{r_{i,S}^{\text{nom}}\left(\theta_i\right) : i=1,\ldots,N_{S,p}\right\}},$$

with $\bar{\theta}$ and $\bar{r}$ the average of the θ_i and $r_{i,S}^{\text{nom}}(\theta_i)$, respectively. Fig. 2.4 illustrates the improved out-of-sample predictions of our mechanism compared to the standard ELM for a $1.5''$ cylinder. In this case, the in-plane deviations of $N_{1,\text{A}} \approx 4000$ points on four cylinders (which are analyzed in Section 2.3) were used to fit the model. The standard ELM yields poor predictions because its use of Uniform$(-1,1)$ saturates its hidden neurons and prevents it from learning the deviation patterns as a function of the inputs. Additional discussions on the critical role of the random assignment of the $\alpha_{m,k}^{(S,p)}$ for the predictive performance of ELM models are in Appendix A.

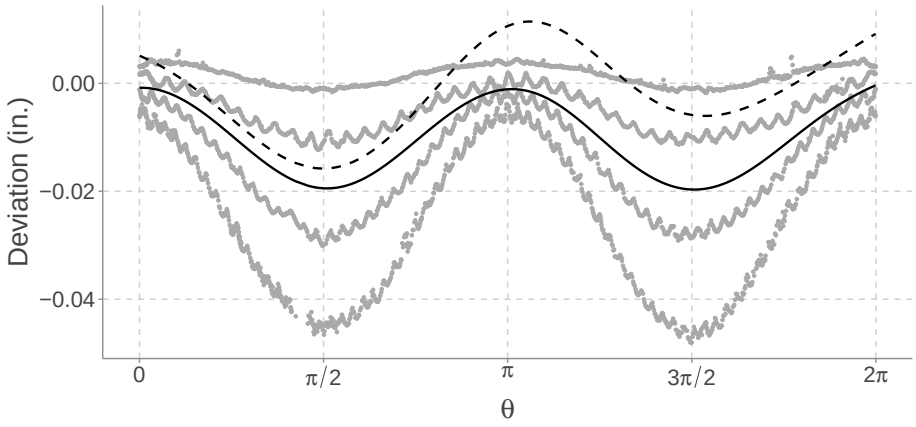


Figure 2.4.: In-plane deviations (gray dots) for four cylinders manufactured under process A, out-of-sample predictions for a $1.5''$ cylinder obtained using the Uniform$(-1,1)$ distribution for all $\alpha_{m,k}^{(1,\text{A})}$ (dashed line), and out-of-sample predictions obtained using our tuned Uniform distributions (solid line).

An important consideration in this step is the choice of shape class and process. Shapes and processes whose deviations are convenient to describe are preferable, as

they generally require fewer test shapes to fit and can be more usefully applied in the remaining steps. For example, we observe in Section 2.3 that as few as four shapes are sufficient to specify a baseline Bayesian ELM deviation model with both good predictive performance and capability for model transfer.

2.2.3 Step Two: Transfer baseline model to new process

We learn the Bayesian ELM for the $T_{s,p'\to p}$ via a discrepancy measure (Rubin, 1984; Meng, 1994) that combines posterior predictions from the baseline model with data from process p' to extract information on the TEAs. Our measure is defined for each point $i \in \{1, \ldots, N_{S,p'}\}$ on the products under p' as

$$T_{i,S,p'\to p} = \underset{t\geq -r_{i,S}^{\text{nom}}(\theta_i)}{\operatorname{argmin}} \left\{ y_i - t - x_i - \mathbf{h}_i^{\mathsf{T}} \widetilde{\boldsymbol{\beta}}_{S,p} \right\}^2, \tag{2.9}$$

where y_i is the deviation for point i, $\widetilde{\boldsymbol{\beta}}_{S,p}$ is a random variable distributed according to the posterior of $\boldsymbol{\beta}_{S,p}$ from the previous step, and $\mathbf{h}_i \in \mathbb{R}^{M_{S,p}}$ has entry m equal to $g\left(\alpha_{m,0}^{(S,p)} + \mathbf{z}\left(\theta_i, r_{i,S}^{\text{nom}}, x_i + t\right)^{\mathsf{T}} \boldsymbol{\alpha}_m^{(S,p)}\right)$. We summarize the discrepancy measure distribution for each i by its expectation $\widehat{T}_{i,S,p'\to p}$, and form the vector $\widehat{\mathbf{T}}_{S,p'\to p} \in \mathbb{R}^{N_{S,p'}}$ containing them. The Bayesian ELM for the TEAs is then specified as

$$\widehat{\mathbf{T}}_{S,p'\to p} = \mathbf{H}_{S,p'\to p}\boldsymbol{\beta}_{S,p'\to p} + \boldsymbol{\epsilon}_{S,p'\to p}, \tag{2.10}$$

where $\mathbf{H}_{S,p'\to p}$ is the $N_{S,p'} \times M_{S,p'\to p}$ matrix whose (i,m) entry is $g\left(\alpha_{m,0}^{(S,p')} + \mathbf{z}\left(\theta_i, r_{i,S}^{\text{nom}}, x_i\right)^{\mathsf{T}} \boldsymbol{\alpha}_m^{(S,p')}\right)$ for random $\alpha_{m,k}^{(S,p')}$ ($M_{S,p'\to p}$ is the selected number of hidden neurons), $\boldsymbol{\beta}_{S,p'\to p} \in \mathbb{R}^{M_{S,p'\to p}}$ is a vector of unknown parameters, and the error terms in $\boldsymbol{\epsilon}_{S,p'\to p}$ are independent and identically distributed $\mathrm{N}\left(0, \sigma_{S,p'\to p}^2\right)$ random variables with $\sigma_{S,p'\to p}^2$ unknown. Our prior probability density function is

$$p\left(\boldsymbol{\beta}_{S,p'\to p}, \sigma_{S,p'\to p}^2 \mid \tau_{S,p'\to p}^2\right) \propto \sigma_{S,p'\to p}^{-2} \tau_{S,p'\to p}^{-M_{S,p'\to p}} \times \exp\left(-0.5\tau_{S,p'\to p}^{-2} \boldsymbol{\beta}_{S,p'\to p}^{\mathsf{T}} \boldsymbol{\beta}_{S,p'\to p}\right), \tag{2.11}$$

and our hyperprior probability density function for $\tau^2_{S,p'\to p}$ is $p\left(\tau^2_{S,p'\to p}\right) \propto \tau^{-6}_{S,p'\to p}$ $\exp\left(-0.01\tau^{-2}_{S,p'\to p}\right)$. The Gibbs algorithm can again be used to derive the posterior.

As illustrated in Fig. 2.3, the Bayesian ELM TEA model is connected to the baseline Bayesian ELM deviation model in an immediate and simple manner. To formally describe this connection, let $\mathbf{h}_{i,S,p'\to p}$ denote row i of $\mathbf{H}_{S,p'\to p}$, and $\widehat{\boldsymbol{\beta}}_{S,p'\to p}$ contain the posterior modes of each entry in $\boldsymbol{\beta}_{S,p'\to p}$. Also, define $\mathbf{H}_{S,p'}$ as the $N_{S,p'} \times M_{S,p}$ matrix whose (i,m) entry is $g\left(\alpha^{(S,p)}_{m,0} + \mathbf{z}\left(\theta_i, r^{\text{nom}}_{i,S}, x_i + \mathbf{h}_{i,S,p'\to p}\widehat{\boldsymbol{\beta}}_{S,p'\to p}\right)^{\mathsf{T}} \boldsymbol{\alpha}^{(S,p)}_m\right)$, where $\alpha^{(S,p)}_{m,0}$ and $\boldsymbol{\alpha}^{(S,p)}_m$ are from step one. Then the comprehensive Bayesian ELM deviation model that can be fitted to products from S manufactured under both p and p' is

$$\begin{aligned}
\mathbf{y}_{S,p} &= \mathbf{x}_{S,p} + \mathbf{H}_{S,p}\boldsymbol{\beta}_{S,p} + \boldsymbol{\epsilon}_{S,p}, \\
\mathbf{y}_{S,p'} &= \mathbf{x}_{S,p'} + \mathbf{H}_{S,p'\to p}\widehat{\boldsymbol{\beta}}_{S,p'\to p} + \mathbf{H}_{S,p'}\boldsymbol{\beta}_{S,p} + \boldsymbol{\epsilon}_{S,p'},
\end{aligned} \tag{2.12}$$

where the error terms in $\boldsymbol{\epsilon}_{S,p'}$ are independent and identically distributed $\text{N}\left(0, \sigma^2_{S,p'}\right)$ random variables with $\sigma^2_{S,p'}$ unknown, and independent of those in $\boldsymbol{\epsilon}_{S,p}$. Note that we utilize posterior modes in equation (2.12) to facilitate deviation modeling under p' for general, nonlinear activation functions. This corresponds to concepts of Meng (2010) for practical machine learning. Our use of computationally tractable discrepancy measures and TEA parameter estimates distinguishes our method from that of Sabbaghi and Huang (2018), which used intensive Bayesian calculations for all unknowns.

Example 2.2. Consider deviations for cylinders under a new process B. With $N_{1,\text{B}} = 100$, $M_{1,\text{B}\to\text{A}} = 2$, $\mathbf{H}_{1,\text{B}\to\text{A}}$ is 100×2, and has (i,m) entry $g\left(\alpha^{(1,\text{B})}_{m,0} + \alpha^{(1,\text{B})}_{m,1}\theta_i + \alpha^{(1,\text{B})}_{m,2} r^{\text{nom}}_{i,1}\right)$. Also, $\mathbf{H}_{1,\text{B}}$ has (i,m) entry $g\left(\alpha^{(1,\text{A})}_{m,0} + \alpha^{(1,\text{A})}_{m,1}\theta_i, +\alpha^{(1,\text{A})}_{m,2}\left(r^{\text{nom}}_{i,1} + \mathbf{h}_{i,1,\text{B}\to\text{A}}\widehat{\boldsymbol{\beta}}_{1,\text{B}\to\text{A}}\right)\right)$.

2.2.4 Step Three: Transfer baseline model to new shape

The global deviation feature for shapes from a new class S' manufactured under process p is specified according to $\mathbf{H}_{S,p}\boldsymbol{\beta}_{S,p}$ as in equation (2.7), and a Bayesian

ELM for their local deviation feature $\delta_{s',p}$ is then learned in a principled and automated manner by leveraging the global deviation feature model with data from the new shapes. To formally describe this, let $N_{S',p}$ denote the total number of points on products from S' manufactured under p, and $M_{S',p}$ the chosen number of hidden neurons for the Bayesian ELM of the local deviation feature. For each point $i \in \{1, \ldots, N_{S',p}\}$ on these products, let $\mathbf{w}\left(\theta_i, r_{i,S'}^{\text{nom}}, x_i\right) \in \mathbb{R}^J$ denote its vector of inputs that is a function of θ_i, the nominal radius function $r_{i,S'}^{\text{nom}}$ for the specific product from S' on which i resides, and the applied compensation $x_i \in \mathbb{R}$. Also, let $\mathbf{z}\left(\theta_i, r_{i,S}^{\text{nom}}, x_i\right) \in \mathbb{R}^K$ denote the vector of inputs for the corresponding shape from S whose deviation model captures the global deviation feature for i. We define the $N_{S',p} \times M_{S,p}$ matrix $\mathbf{H}_{S',p,\text{G}}$ whose (i,m) entry is $g\left(\alpha_{m,0}^{(S,p)} + \mathbf{z}\left(\theta_i, r_{i,S}^{\text{nom}}, x_i\right)^{\mathsf{T}} \boldsymbol{\alpha}_m^{(S,p)}\right)$, so that $\mathbf{H}_{S',p,\text{G}}\boldsymbol{\beta}_{S,p}$ captures the global deviation feature of products from S'. To specify the local deviation feature, we draw $\alpha_{m,j}^{(S',p)}$ as independent Uniform random variables $(m = 1, \ldots, M_{S',p}, j = 0, \ldots, J)$, and define the $N_{S',p} \times M_{S',p}$ matrix $\mathbf{H}_{S',p,\text{L}}$ whose (i,m) entry is $g\left(\alpha_{m,0}^{(S',p)} + \mathbf{w}\left(\theta_i, r_{i,S'}^{\text{nom}}, x_i\right)^{\mathsf{T}} \boldsymbol{\alpha}_m^{(S',p)}\right)$. Then the Bayesian ELM deviation model for S' is specified as

$$\mathbf{y}_{S',p} = \mathbf{x}_{S',p} + \mathbf{H}_{S',p,\text{G}}\boldsymbol{\beta}_{S,p} + \mathbf{H}_{S',p,\text{L}}\boldsymbol{\beta}_{S',p} + \boldsymbol{\epsilon}_{S',p}, \tag{2.13}$$

where $\mathbf{H}_{S',p,\text{L}}\boldsymbol{\beta}_{S',p}$ captures the local deviation feature. As before, $\boldsymbol{\beta}_{S',p} \in \mathbb{R}^{M_{S',p}}$ is a vector of unknown parameters, and the error terms in $\boldsymbol{\epsilon}_{S',p}$ are independent and identically distributed $\text{N}\left(0, \sigma_{S',p}^2\right)$ random variables with $\sigma_{S',p}^2$ unknown. Our prior probability density function for the parameters is

$$p\left(\boldsymbol{\beta}_{S',p}, \sigma_{S',p}^2 \mid \tau_{S',p}^2\right) \propto \sigma_{S',p}^{-2} \tau_{S',p}^{-M_{S',p}} \exp\left(-0.5\tau_{S',p}^2 \boldsymbol{\beta}_{S',p}^{\mathsf{T}} \boldsymbol{\beta}_{S',p}\right), \tag{2.14}$$

and our hyperprior probability density function for $\tau_{S',p}^2$ is $p\left(\tau_{S',p}^2\right) \propto \tau_{S',p}^{-6} \exp\left(-0.01\tau_{S',p}^2\right)$. We derive the posterior of the parameters, and hence predictions of deviations for shapes from S' under process p, via the Gibbs algorithm. By the same reasoning as

before, the comprehensive Bayesian ELM deviation model involving equations (2.7) and (2.13) can be fitted to products from both S and S' manufactured under p.

Example 2.3. Consider uncompensated squares under A, which belong to the shape class 2 of polygons. Suppose $N_{2,\mathrm{A}} = 500$. One useful set of inputs that can capture the local deviation feature is $\mathbf{w}\left(\theta_i, r_{i,2}^{\mathrm{nom}}\right) = \left(\theta_i, r_{i,2}^{\mathrm{nom}}(\theta_i), \mathrm{edge}(\theta_i)\right)^{\mathsf{T}}$, where $\mathrm{edge}(\theta_i) \in \{1, 2, 3, 4\}$ indicates the edge containing i. Also, the global deviation feature for a square is captured by the cylinder with radius $r_{i,1}^{\mathrm{nom}}$ equal to the square's circumradius Huang et al. (2014), and so $\mathbf{z}\left(\theta_i, r_{i,1}^{\mathrm{nom}}\right)$ is accordingly defined for each i.

In contrast to the work of Sabbaghi et al. (2018), our Bayesian ELM method reduces the effort for learning local deviation features, from specifying and fitting an entire nonlinear model for it to the simpler task of selecting the number of hidden neurons. Also, our modular ELM components for the local deviation features are immediately connectable for specifying comprehensive models across shape classes. Ultimately, our use of Bayesian statistics in this and the previous steps of our methodology plays a key role in enabling automated deviation modeling across an AM system based on sequential updates of prior deviation models with data from different shapes and processes.

2.3 Case studies

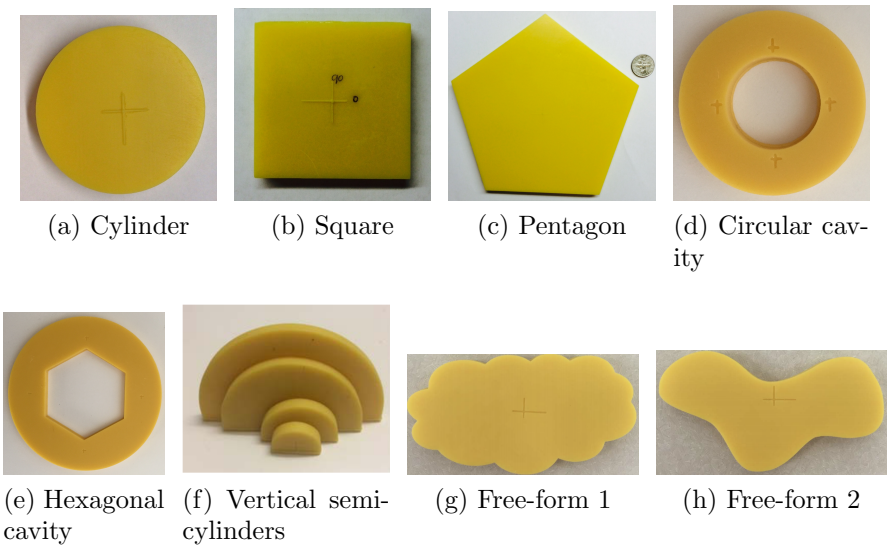
2.3.1 Overview

We present case studies of our Bayesian ELM method for automated deviation modeling of several shape classes (Fig. 2.5) and stereolithography conditions. The products were manufactured under different settings of an ULTRA machine, which is a commercial mask image projection stereolithography (Zhou and Chen, 2012) platform by EnvisionTec. The observed settings for two processes A and B considered throughout are in Table 2.1. All deviations were measured by a Micro-Vu Vertex system.

The progression of the case studies is in Fig. 2.6. Section 2.3.2 contains the implementation of the first step for the shape class 1 of cylinders with small heights manufactured under process A. Three cases for the second step are in Section 2.3.3: cylinders with small heights manufactured under process B, circular cavities in cylinders with small heights, and out-of-plane deviations for vertical semi-cylinders. The latter two sets of products were manufactured under process B, but their deviations are effectively generated under new processes C and D, respectively, due to the distinct physics involved in deviations of cavities and interlayer bonding effects for vertical products (Sabbaghi and Huang, 2016; Jin et al., 2016). Section 2.3.4 contains the implementation of the third step for the shape class 2 of polygons, with the specific products being squares with small heights manufactured under process A. The models from the previous three steps are combined in the implementation of the fourth step in Section 2.3.5 for new polygons and freeform products with small heights manufactured under process B. Comparisons between the results obtained from our methodology and those obtained using an existing Bayesian ELM method of Soria-Olivas et al. (2011) are provided in Appendix B to further demonstrate the power and advantages of our approach. In all of these case studies, deviations from approximately 1000 equally-spaced points on each individual product were used to form training data sets for the models, and the activation function was the hyperbolic tangent.

Table 2.1.: Observed settings for the ULTRA processes.

Variable	**Process A**	**Process B**
Product height	$0.5''$	$0.25''$
Layer thickness	$0.004''$	$0.00197''$
Mask resolution	1920×1200	1920×1200
Pixel dimension	$0.005''$	$0.005''$
Illuminating time/layer	9 s	7 s
Waiting time/layer	15 s	15 s
Resin type	SI500	SI500



(a) Cylinder (b) Square (c) Pentagon (d) Circular cavity

(e) Hexagonal cavity (f) Vertical semi-cylinders (g) Free-form 1 (h) Free-form 2

Figure 2.5.: The manufactured products considered in our case studies.

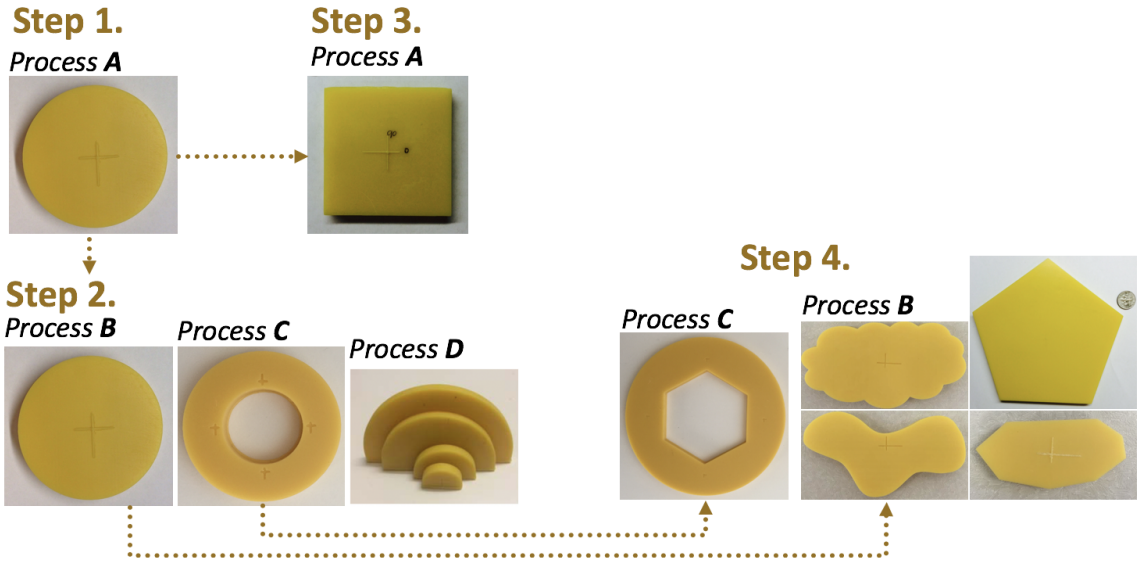
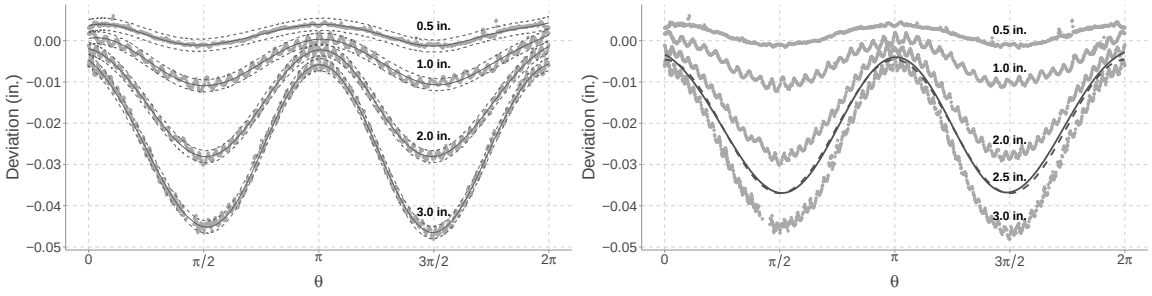


Figure 2.6.: Overview of the case studies for our methodology.

2.3.2 Baseline model for in-plane cylinder deviations

The baseline Bayesian ELM model for in-plane deviations under process A was fitted using four uncompensated cylinders of nominal radii $0.5''$, $1''$, $2''$, and $3''$, respectively. We set $M_{1,\mathrm{A}} = 40$ and $\mathbf{z}\left(\theta_i, r_{i,1}^{\mathrm{nom}}\right) = \left(\theta_i, r_{i,1}^{\mathrm{nom}}\right)^{\mathsf{T}}$. The posterior predictions of deviations from our model are summarized in Fig. 2.7(a). By inspection, our model provides a good fit to the deviations. Huang et al. (2015b) previously specified a Bayesian nonlinear regression model for these deviations that was informed by their domain knowledge of the stereolithography process A. In contrast, our Bayesian ELM model was specified without any such knowledge and yields equivalent predictive performances (e.g., as illustrated in Fig. 2.7(b) for a $2.5''$-radius cylinder), which serves to illustrate the effective reductions in user efforts and inputs afforded by our method. Another demonstration of the effectiveness of our method compared to an existing Bayesian ELM is in Appendix B. We conclude that the deviation model specified by our method will enable the same level of in-plane deviation control for cylinders under process A as that of the model of Huang et al. (2015b), which was an order of magnitude reduction in validation experiments.



(a) (b)

Figure 2.7.: (a) In-plane deviations (dots) for four cylinders under process A, and the posterior predictive means (solid lines) and 95% central posterior predictive intervals (dashed lines) from our Bayesian ELM model. (b) Comparison of the posterior predictive means for a $2.5''$-radius cylinder obtained from our approach (solid) with those obtained from Huang et al. (2015b) (dashed).

2.3.3 Transfer of baseline cylinder model to new processes

Three uncompensated cylinders of nominal radii $0.5'', 1.5''$, and $3''$ manufactured under process B are considered in the first case study of the second step of our Bayesian ELM methodology. It is important to note that, besides the observed differences in product height, layer thickness, and illuminating time per layer, process B also differs from A in terms of new lurking variable settings that induce overcompensation (Huang et al., 2014). The distinct and complicated nature of process B is clear upon inspection of these cylinders' deviations in Fig. 2.8. Specifically, in contrast to A, in-plane cylinder deviations under B increase on average as a function of the nominal radius, and are asymmetrical. Our methodology effectively learns these complex features in the transfer of the baseline model to B by means of its model for $T_{1,\mathrm{B}\to\mathrm{A}}$. It also facilitates model transfer by involving only the single user input of $M_{1,\mathrm{B}\to\mathrm{A}}$ (which we set to 40) instead of the traditional specification of an entirely new model. The summary of our transferred model's posterior predictions in Fig. 2.8 demonstrates our successful modeling of in-plane cylinder deviations under B. Sabbaghi and Huang (2018) and Sabbaghi et al. (2018) previously specified Bayesian nonlinear regression models for these products' deviations. However, our Bayesian ELM model is preferable to theirs because it is fitted in a much simpler manner using the Gibbs algorithm compared to their computationally demanding Hamiltonian Monte Carlo (Duane et al., 1987) implementation, with no loss of predictive performance. Our comprehensive Bayesian ELM model for processes A and B is also preferable to fitting a standard NN or ELM model just on the data from B because our method yields smaller predictive uncertainties due to its incorporation of more data.

The second step of our method can also accommodate new deviation processes that arise in complex geometries. Three circular cavities of nominal radii $0.5''$, $1''$, and $1.5''$ contained in cylinders of nominal radii $1''$, $2''$, and $3''$, respectively, are considered to demonstrate this fact. The posterior predictions obtained from our model for these

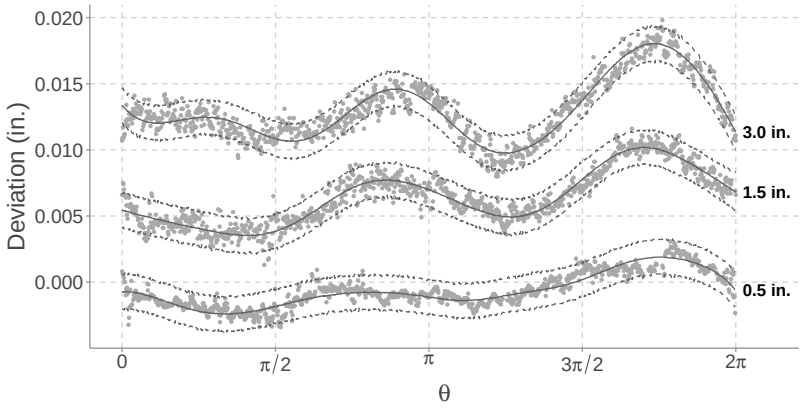


Figure 2.8.: In-plane deviations (dots) for three cylinders under process B, and the posterior predictive means (solid lines) and 95% central posterior predictive intervals (dashed lines) obtained by the transfer of the baseline deviation model to B.

products with $M_{1,\mathrm{C}\to\mathrm{A}} = 40$ (Fig. 2.9) indicate that our transferred model performs well in fitting this data. The broader consequence of this case is that our Bayesian ELM methodology can enable comprehensive and automated deviation modeling for both cavity and boundary components in geometrically complex products. These products were also modeled of Sabbaghi and Huang (2016), but our method is more advantageous because it greatly reduces the effort in learning the TEA and fitting the transferred model.

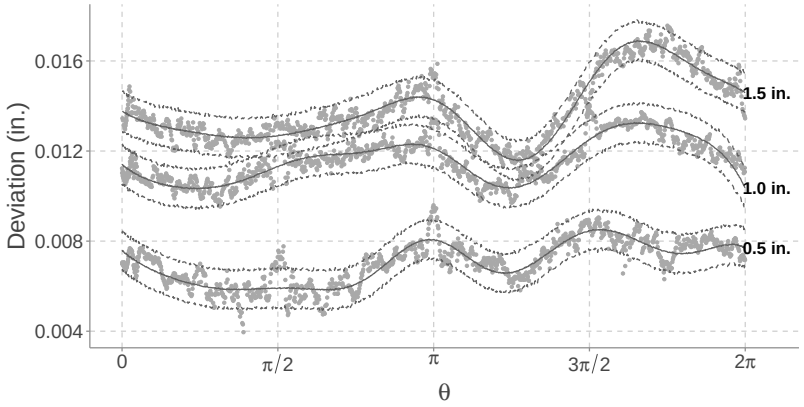


Figure 2.9.: In-plane deviations (dots) for three circular cavities, and the posterior predictive means (solid lines) and 95% central posterior predictive intervals (dashed lines) obtained by our transfer of the baseline deviation model.

The final case study here involves specifying out-of-plane deviation models. This is an especially challenging task because interlayer bonding effects yield complicated vertical deviations Jin et al. (2016). Four vertical semi-cylinders of nominal radii $0.5'', 0.8'', 1.5''$, and $2''$ are considered to demonstrate how the second step effectively addresses this challenge. Fig. 2.10 summarizes the posterior predictions from our transferred model with $M_{1,\mathrm{D}\to\mathrm{A}} = 40$. Jin et al. (2016) previously modeled out-of-plane deviations using nonlinear regression. Again, our approach is preferable to that of Jin et al. (2016) because it automates the specification of a deviation model with good predictive performance, and reduces the user's efforts in leveraging both in-plane and out-of-plane deviation data to perform the model transfer. The advantages of our model compared to existing ELM models are established in Appendix B.

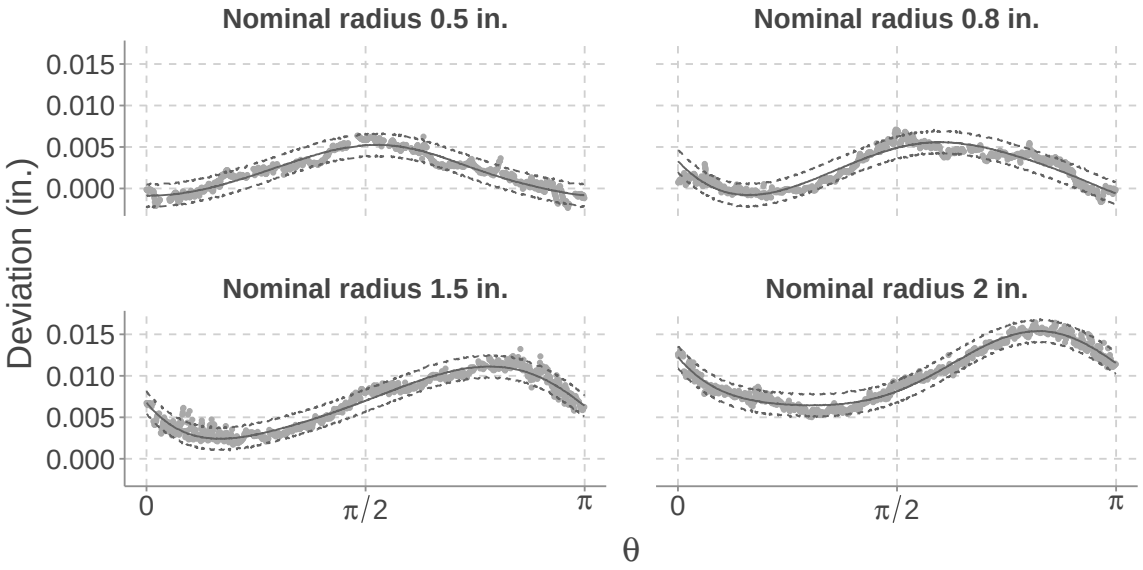


Figure 2.10.: Out-of-plane deviations (dots) for four vertical semi-cylinders, and the posterior predictive means (solid lines) and 95% central posterior predictive intervals (dashed lines) obtained by our transfer of the baseline deviation model.

2.3.4 Transfer of baseline cylinder model to a new shape

Three uncompensated squares of circumradius $1''$, $2''$, and $3''$ manufactured under A are considered in this case study of the transfer of the baseline deviation model to the new shape class 2 of polygons via the third step of our methodology. Additional, more complicated products from this class are in the next subsection. The straight edges and sharp corners in these polygons introduce complex local deviation features that can be difficult to model (Fig. 2.11). The third step effectively learns the complex local deviation features with the previously specified $\delta_{1,\mathrm{A}}$ as the global deviation feature. We set $M_{2,\mathrm{A}} = 50$, $\mathbf{w}\left(\theta_i, r_{i,2}^{\mathrm{nom}}\right) = \left(\theta_i, r_{i,2}^{\mathrm{nom}}\left(\theta_i\right), \mathrm{edge}\left(\theta_i\right)\right)^{\mathsf{T}}$, and $\mathbf{z}\left(\theta_i, r_{i,1}^{\mathrm{nom}}\right) = \left(\theta_i, r_{i,1}^{\mathrm{nom}}\right)^{\mathsf{T}}$ for each point i on a square, and fit the comprehensive Bayesian ELM model to the in-plane deviations of both cylinders and squares under A. The summary of our fit for squares in Fig. 2.11 indicates the success in our model transfer. A comparison of our model's posterior predictions for these products with those obtained by the approach of Huang et al. (2014) further highlights the high predictive performance that can result from the application of our methodology compared to other methods that pre-specify local deviation features.

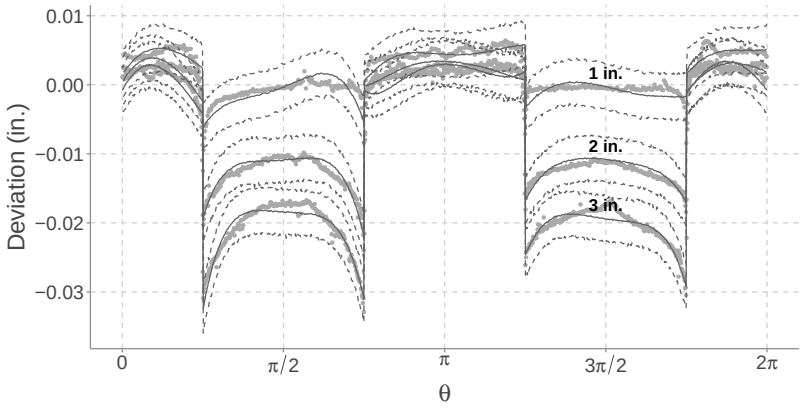


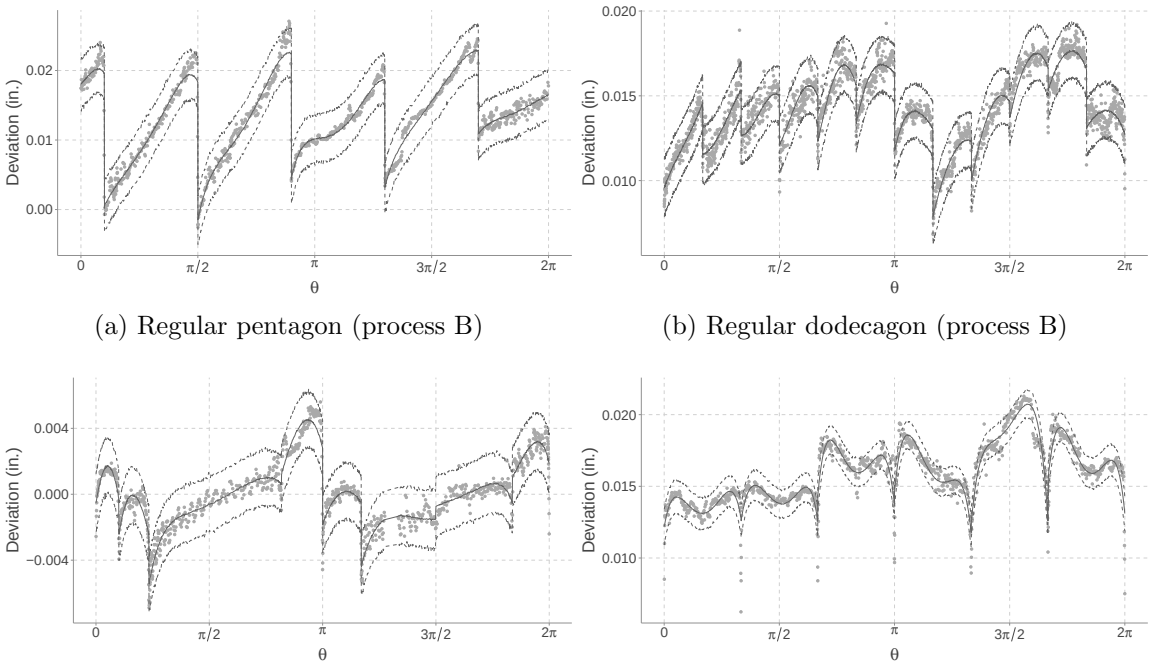
Figure 2.11.: In-plane deviations (dots) for three squares, and the posterior predictive means (solid lines) and 95% central posterior predictive intervals (dashed lines) obtained by our transfer of the cylinder deviation model to these polygons.

2.3.5 Transfer of baseline model to new shapes and processes

An uncompensated regular pentagon of circumradius 3″, regular dodecagon of circumradius 3″, and irregular polygon with smallest bounding circle of radius 1″ (Fig. 1.2) manufactured under B are considered in this first case study of the fourth step of our method. The in-plane deviations of these products were modeled of Huang et al. (2014) and Sabbaghi et al. (2018) by complicated nonlinear regression models that took quite some effort to specify and fit. In contrast, the simpler combination of our connectable ELM structures for $T_{1,\mathrm{B}\to\mathrm{A}}$ and the polygon local deviation feature, which we learned in the previous steps, enables us to specify and fit deviation models for these products in a more automated and effortless manner. The sole user input is the number of hidden neurons for the various ELM structures, which is clearly simpler than learning entirely new models as under current deviation modeling methods. Our models' posterior predictions (Figs. 2.12(a),(b) and 1.2(d)) demonstrate their high predictive performance compared to the more complicated models in previous work.

A hexagonal cavity of circumradius 1.8″ contained in a 3″ cylinder is considered as another case to further demonstrate how the fourth step can accommodate polygons under a new process. Following the reasoning in Section 2.3.1, the deviations for this product are generated under process C, and we immediately connect $T_{1,\mathrm{C}\to\mathrm{A}}$ with the local deviation feature for polygons to transfer the baseline model to this product. The fit of the transferred model is summarized in Fig. 2.12(d). Sabbaghi and Huang (2016) modeled this product's deviations using Bayesian nonlinear regression, but our approach is preferable because it yields a model with better predictive performance that requires less effort to specify and fit.

We conclude with two free-form shapes under B (corresponding to the free-form products in Fig. 2.5). Our connectable ELM structures possess a sufficiently broad scope so as to account for the new, complicated deviation features that arise in the complex geometries of these products. Fig. 2.13 summarizes the posterior predictions of the transferred models obtained from the fourth step. The ability of our method



(a) Regular pentagon (process B)

(b) Regular dodecagon (process B)

(c) Irregular polygon (process B)

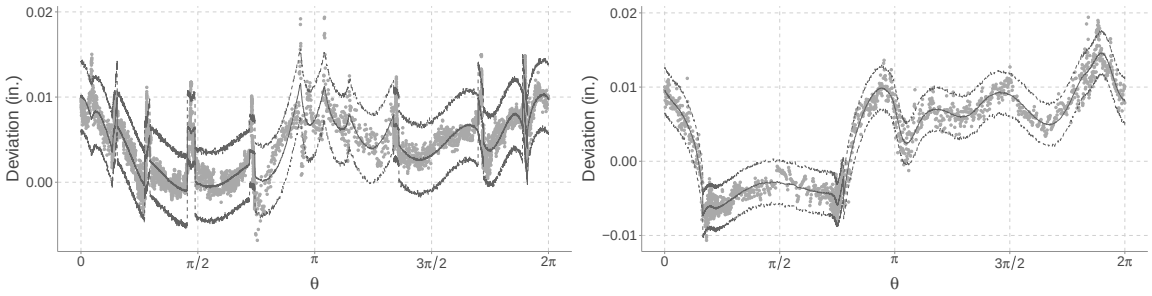
(d) Hexagonal cavity (process C)

Figure 2.12.: In-plane deviations (dots) for polygons under different processes, and the posterior predictive means (solid lines) and 95% central posterior predictive intervals (dashed lines).

to automate modeling of these free-form shapes is a significant demonstration of its effectiveness for AM systems, especially as current deviation modeling methods cannot accommodate free-form shapes with the same ease as our method.

2.4 Discussion

Three broad results about general properties of our methodology can be drawn based on our wide-ranging case studies on solid and hollow products, regular and free-form shapes, and in-plane and out-of-plane deviations. First, in contrast to existing methods, our use of ELMs eliminates identifiability issues in fitting a baseline deviation model. Second, our structured approach to leveraging data and models across different processes and shapes eliminates identifiability issues in learning TEAs and local deviation features. Third, our new random mechanism for ELMs effectively



(a) Free-form 1 (b) Free-form 2

Figure 2.13.: In-plane deviations (dots) for two free-form shapes under B, and the posterior predictive means (solid lines) and 95% central posterior predictive intervals (dashed lines).

prevents saturation of hidden neurons. Thus, we can conclude that our Bayesian ELM methodology can yield deviation models with good predictive performance in an automated manner, involving negligible effort, for a wide variety of shape classes and processes in AM systems. It is important to note that no specific domain knowledge for the stereolithography machine involved in these case studies was employed in our method. This fact further demonstrates the general scope of our method.

Although we demonstrated the generality of our methodology as well as its good predictive performance, as previously mentioned, its complicated structure impedes further interpretations or insights. We address the issue of interpretability of our Bayesian neural networks, and complex machine learning models in general, in our next chapters.

3. PREDICTIVE COMPARISONS FOR SCREENING AND INTERPRETING INPUTS IN MACHINE LEARNING MODELS

3.1 Average predictive comparisons

3.1.1 Notations and assumptions

We let $p(y \mid x, \beta)$ denote the probability density function that corresponds to a machine learning algorithm or model for data (x_i, y_i), where $i = 1, \ldots, n$ indexes the observations, $y_i \in \mathbb{R}$ is the (continuous) outcome, $x_i \in \mathbb{R}^K$ is the input vector, and $\beta \in \mathbb{R}^L$ is the (unknown) parameter vector. Inputs differ from predictors here in that a single input can enter into the model via multiple predictors (Gelman and Pardoe, 2007, p. 26). The sampled inputs $x_1, \ldots, x_n$ are assumed throughout to be independent and identically distributed according to a probability density function $p(x)$ that is independent of β. In our descriptions of average predictive comparisons, we will partition x as $x = (u, v)$, where $u \in \mathbb{R}$ is of interest and considered continuous (the case of discrete u is discussed later), and $v \in \mathbb{R}^{K-1}$. We assume that a distribution $p(\beta)$ for β exists that captures the uncertainty associated with β, and from which samples $\beta^{(s)}$ $(s = 1, \ldots, S)$ can be drawn. Examples include a Bayesian posterior distribution or a bootstrap distribution of β (Efron, 1979; Efron and Tibshirani, 1994). The method by which the $\beta^{(s)}$ are drawn is generally not of concern in the predictive comparison methodology so long as it captures inferential uncertainty (Gelman and Pardoe, 2007, p. 31).

3.1.2 Definitions and estimators of standard average predictive comparisons

For a particular change in u from $u^{(1)}$ to $u^{(2)}$, Gelman and Pardoe (2007, p. 24) define the predictive comparison $\delta_u\left(u^{(1)} \to u^{(2)}\right)$ of u on y, given β and fixed v, using the expectation $\mathbb{E}(\cdot)$ of the outcome variable derived according to the specified $p\left(y \mid x, \beta\right)$. Specifically,

$$\delta_u\left(u^{(1)} \to u^{(2)}\right) = \frac{\mathbb{E}\left(y \mid u^{(2)}, v, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, v, \beta\right)}{u^{(2)} - u^{(1)}}.$$

This quantity is interpreted as the expected change in the outcome corresponding to this change in u. To illustrate, for a logistic regression model when $u^{(2)} - u^{(1)} = 1$, $\delta_u\left(u^{(1)} \to u^{(2)}\right)$ corresponds to a predicted change in probability (Gelman and Pardoe, 2007, p. 25). The $\delta_u\left(u^{(1)} \to u^{(2)}\right)$ are summarized by their weighted mean over all possible positive changes $u^{(1)}$ to $u^{(2)}$, with the changes as the weights. This weighted mean is referred to as an average predictive comparison, and denoted by Δ_u. Gelman and Pardoe (2007, p. 32) describe how such a summary is reasonable from an estimation perspective, as the predictive comparison $\delta_u\left(u^{(1)} \to u^{(2)}\right)$ may be unstable for small differences between $u^{(2)}$ and $u^{(1)}$. To simplify the formal presentation of Δ_u, let $q = \left(u^{(1)}, u^{(2)}, v, \beta\right), p(q) = p\left(u^{(1)} \mid v\right) p\left(u^{(2)} \mid v\right) p(v)p(\beta)$, and $U^+ = \left\{u^{(1)}, u^{(2)} \in \mathbb{R},\ v \in \mathbb{R}^{K-1},\ \beta \in \mathbb{R}^L : u^{(1)} < u^{(2)}\right\}$. Then Δ_u is defined as

$$\Delta_u = \frac{\displaystyle\int_{U^+} \left\{\mathbb{E}\left(y \mid u^{(2)}, v, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, v, \beta\right)\right\} p(q)\mathrm{d}q}{\displaystyle\int_{U^+} \left(u^{(2)} - u^{(1)}\right) p(q)\mathrm{d}q}. \tag{3.1}$$

Gelman and Pardoe (2007) estimate the unknown estimands Δ_u for each input u in x by $\widehat{\Delta}_u = S^{-1}\sum_{s=1}^{S}\widehat{\Delta}_u^{(s)}$, where

$$\widehat{\Delta}_u^{(s)} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\left\{\mathbb{E}\left(y \mid u_j, v_i, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta^{(s)}\right)\right\} w_{ij}\text{sign}(u_j - u_i)}{\sum_{i=1}^{n}\sum_{j=1}^{n}(u_j - u_i)w_{ij}\text{sign}(u_j - u_i)},$$

and the weights $w_{ij} = \{1+M(v_i, v_j)\}^{-1}$ for a selected metric $M : \mathbb{R}^{K-1}\times\mathbb{R}^{K-1} \to \mathbb{R}_{\geq 0}$ on $\mathbb{R}^{K-1}$ are meant to approximate $p(u_i \mid v)p(u_j \mid v)$ according to the reasoning that the likelihood of a transition from u_i to u_j is inversely related to the distance between v_i and v_j (Gelman and Pardoe, 2007, p. 37). In general, standard errors of the estimators $\widehat{\Delta}_u$ are derived as $\text{SE}\left(\widehat{\Delta}_u\right) = (S-1)^{-1/2}\left\{\sum_{s=1}^{S}\left(\widehat{\Delta}_u^{(s)} - \widehat{\Delta}_u\right)^2\right\}^{1/2}$ by viewing β as random and x as fixed (Gelman and Pardoe, 2007, p. 38–39). We utilize the Mahalanobis (1927) metric throughout this dissertation to specify the weights, and provide a formal definition of it in the appendix. This particular metric was also extensively utilized by Gelman and Pardoe (2007). It is important to note that our predictive comparison methodology is not limited in its consideration of metrics for the weights to only the Mahalanobis metric.

Gelman and Pardoe (2007) did not investigate the conditions under which the estimators $\widehat{\Delta}_u$ would be consistent. We proceed to address this issue here by identifying the following condition on the inputs' distributions and the chosen metric under which the estimators will be Fisher consistent. In Condition 3.1 the selected metric M is fixed, and the function $f_{v^{(1)}} : \mathbb{R}^{K-1} \to [0, 1]$ for any $v^{(1)} \in \mathbb{R}^{K-1}$ is defined with respect to M as

$$f_{v^{(1)}}(v) = \left\{1 + M\left(v^{(1)}, v\right)\right\}^{-1}. \tag{3.2}$$

Fisher consistency is formally defined in the appendix.

Condition 3.1 (Stable metric-input distribution assumption)**.** Let $M : \mathbb{R}^{K-1} \times \mathbb{R}^{K-1} \to \mathbb{R}_{\geq 0}$ be a metric on the v inputs. Then M provides a stable metric approximation for the marginal probability density function $p(v)$ and the conditional probability density function $p\left(v \mid u\right)$ if, for any $v^{(1)} \in \mathbb{R}^{K-1}$ and $u^{(2)} \in \mathbb{R}$,

$$\frac{p\left(v^{(1)}\right)}{\displaystyle\int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v)p(v)dv} = \frac{p\left(v^{(1)} \mid u^{(2)}\right)}{\displaystyle\int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v)p\left(v \mid u^{(2)}\right)dv},$$

and

$$\int_{\mathbb{R}^{K-1}} \left\{ \frac{p\left(\tilde{v}\right)}{\displaystyle\int_{\mathbb{R}^{K-1}} f_{\tilde{v}}(v)p(v)dv} \right\} p\left(\tilde{v}\right) d\tilde{v} < \infty,$$

where functions $f_{v^{(1)}}, f_{\tilde{v}} : \mathbb{R}^{K-1} \to [0,1]$ correspond to M, and $v^{(1)}$ and $\tilde{v}$ respectively, as in equation (3.2).

This condition is similar to the reasoning used by Gelman and Pardoe (2007, p. 37) in their introduction of the function $f_{v^{(1)}}\left(v^{(2)}\right)$ to approximate $p\left(u^{(1)} \mid v^{(1)}\right) p\left(u^{(2)} \mid v^{(1)}\right)$ for $u^{(1)}, u^{(2)} \in \mathbb{R}$ and $v^{(1)} \in \mathbb{R}^{K-1}$, where $\left(u^{(1)}, v^{(1)}\right)$ and $\left(u^{(2)}, v^{(1)}\right)$ are two input vectors with $u^{(1)} \neq u^{(2)}$. The term "stable metric-input distribution assumption" refers to the approximation $\displaystyle\int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v)p(v)dv$ for $p\left(v^{(1)}\right)$, which is centered at $v^{(1)}$, having the same accuracy, in terms of the ratio of the actual density value with its approximation, as the approximation $\displaystyle\int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v)\ p\left(v \mid u^{(2)}\right) dv$ for $p\left(v^{(1)} \mid u^{(2)}\right)$, which again is centered at $v^{(1)}$. The second part of this condition describes the approximation inaccuracies as finite in expectation over the distribution of v inputs. Condition 3.1 is reasonable for spherically symmetric inputs and typical choices of metrics (such as the Mahalanobis metric). As elliptical distributions can be transformed to be spherically symmetric, this condition is also applicable when x has an elliptical distribution. Under Condition 3.1, we obtain in Theorem 3.1 that $\widehat{\Delta}_u$ is a Fisher consistent estimator of Δ_u. The proof of this result is in Appendix B.1.

Theorem 3.1. If the metric M used in $\widehat{\Delta}_u$ satisfies the stable metric-input distribution assumption, then $\widehat{\Delta}_u$ is a Fisher consistent estimator for Δ_u.

Gelman and Pardoe (2007, p. 34) define an average predictive comparison for discrete u as

$$\frac{\sum_{u^{(1)}}\sum_{u^{(2)}}\left[\left\{\Delta_u\left(u^{(1)}\to u^{(2)}\right)\right\}^2\int p(q_{-\beta})\mathrm{d}q_{-\beta}\right]}{\sum_{u^{(1)}}\sum_{u^{(2)}}\left\{\int p(q_{-\beta})\mathrm{d}q_{-\beta}\right\}}, \tag{3.3}$$

where $\Delta_u\left(u^{(1)}\to u^{(2)}\right)$ is defined similarly to equation (3.1) with the difference that the integral in its denominator does not include $u^{(2)}-u^{(1)}$, and $q_{-\beta}$ corresponds to vector q excluding β. Further details for this case are in (Gelman and Pardoe, 2007, p. 33, 38, 40). Gelman and Pardoe (2007, p. 36–37) briefly suggested the use of either equation (3.3) or the average absolute predictive comparison estimand in the case that $\Delta_u = 0$ due to non-monotonic relationships between the inputs and the outcome. However, they never implemented or explored these ideas.

3.2 Predictive comparison methodology for global interpretability in machine learning

3.2.1 Step One: Screen relevant inputs

The first step in our methodology involves screening relevant inputs based on new predictive comparison estimands we define in Definitions 3.1 and 3.2 that capture the magnitudes of the inputs' average effects under different perspectives. This step corresponds to a suggestion noted, but never implemented or explored, by Gelman and Pardoe (2007, p. 33–34). Our new estimands are distinct from those of Gelman and Pardoe (2007). For example, the second involves $\lambda_u^2\left(u^{(1)}\right)=\left\{\mathbb{E}\left(y\mid u^{(1)},v,\beta\right)-\overline{E_{u|v}\left(y\mid u,v,\beta\right)}\right\}^2$, where we define $\overline{E_{u|v}(y\mid u,v,\beta)}=\int_{\mathbb{R}}\mathbb{E}\left(y\mid u^{(2)},v,\beta\right)p(u^{(2)}\mid v)\mathrm{d}u^{(2)}$, which was never considered in (Gelman and Pardoe, 2007).

Definition 3.1. The average magnitude predictive comparison estimand is

$$\Delta_{\mathrm{mag}(u)}=\left[\int\left\{\mathbb{E}\left(y\mid u^{(2)},v,\beta\right)-\mathbb{E}\left(y\mid u^{(1)},v,\beta\right)\right\}^2p(q)\mathrm{d}q\Big/\int\left(u^{(2)}-u^{(1)}\right)^2p(q)\mathrm{d}q\right]^{1/2}.$$

Definition 3.2. The average root integral of squares predictive comparison estimand is

$$\Lambda_u = \left\{ \int_{\mathbb{R}^L} \int_{\mathbb{R}^{K-1}} \int_{\mathbb{R}} \lambda_u^2 \left(u^{(1)}\right) p\left(u^{(1)} \mid v\right) p(v) p(\beta) \mathrm{d}u^{(1)} \mathrm{d}v \mathrm{d}\beta \right\}^{1/2}.$$

The estimand in Definition 3.1 is interpreted as the average difference in the expected y across different pairs of u values. It can be viewed as the square root of the weighted mean of the $\delta_u^2\left(u^{(1)} \to u^{(2)}\right)$, with the weights based on the squared changes in u. This estimand is simpler and more interpretable than that in equation (3.3) because it does not involve nested integration, and it directly works with the squared predictive comparison rather than squaring the average predictive comparison. An alternative average magnitude predictive comparison estimand that can be robust to outliers is constructed using the absolute predictive comparisons $\left|\delta_u\left(u^{(1)} \to u^{(2)}\right)\right|$ instead of the $\delta_u^2\left(u^{(1)} \to u^{(2)}\right)$.

Our estimators for these new estimands are specified in a natural manner by

$$\widehat{\Delta}_{\mathrm{mag}(u)} = \left\{ \frac{\sum_{s=1}^{S} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ \mathbb{E}\left(y \mid u_j, v_i, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta^{(s)}\right) \right\}^2 w_{ij}}{S \sum_{i=1}^{n} \sum_{j=1}^{n} (u_j - u_i)^2 w_{ij}} \right\}^{1/2},$$

$$\widehat{\Lambda}_u = \left[\frac{1}{S} \sum_{s=1}^{S} \sum_{i=1}^{n} \left\{ \mathbb{E}\left(y \mid x_i, \beta^{(s)}\right) - \overline{E_{u|v_i}\left(y \mid u, v_i, \beta^{(s)}\right)} \right\}^2 \right]^{1/2},$$

respectively, where we define

$$\overline{E_{u|v_i}\left(y \mid u, v_i, \beta^{(s)}\right)} = \frac{\sum_{j=1}^{n} w_{ij} \mathbb{E}\left(y \mid u_j, v_i, \beta^{(s)}\right)}{\sum_{j=1}^{n} w_{ij}}.$$

The standard error of the estimator $A_u \in \left\{\widehat{\Delta}_{\text{mag}(u)}, \widehat{\Lambda}_u\right\}$ is derived in the supplement as

$$\text{SE}\left(A_u\right) = \frac{1}{2A_u}\left[\frac{1}{S-1}\sum_{s=1}^{S}\left\{\left(A_u^{(s)}\right)^2 - A_u^2\right\}^2\right]^{1/2}.$$

A result on the Fisher consistencies of $\widehat{\Delta}_{\text{mag}(u)}$ and $\widehat{\Lambda}_u$ for the corresponding estimands is in Theorem 3.2, and proven in the appendix.

Theorem 3.2. If the metric M in $\widehat{\Delta}_{\text{mag}(u)}$ and $\widehat{\Lambda}_u$ satisfies the stable metric-input distribution assumption, then $\widehat{\Delta}_{\text{mag}(u)}$ and $\widehat{\Lambda}_u$ are Fisher consistent for $\Delta_{\text{mag}(u)}$ and Λ_u, respectively.

We screen relevant inputs using the relative predictive comparison $R\left(A_u\right) = A_u / \sum_{k=1}^{K} A_k$, where A_k is either $\widehat{\Delta}_{\text{mag}(k)}$ or $\widehat{\Lambda}_k$ for all inputs k in x according to the choice of A_u. The ultimate choice of $\widehat{\Delta}_{\text{mag}(u)}$ or $\widehat{\Lambda}_u$ is context dependent. Under $R\left(\widehat{\Delta}_{\text{mag}(u)}\right)$, inputs whose changes are associated with larger changes in the outcome are judged more relevant. Under $R\left(\widehat{\Lambda}_u\right)$, increased relevance is attributed to inputs that exhibit higher variation between predicted outcomes and an average baseline. We derive their standard errors in the supplement as

$$\text{SE}\left\{R(A_u)\right\} = \frac{\text{SE}(A_u)\sum_{k\neq u} A_k}{\left(\sum_{k=1}^{K} A_k\right)^2}.$$

3.2.2 Step Two: Infer conditional effects and two-way interactions of inputs

After screening for relevant inputs in the first step, we proceed in the second step to infer new conditional and two-way interaction average predictive comparisons that can yield interpretable insights into the joint relationships between inputs and the outcome. The two new estimands in this step are in Definitions 3.5 and 3.6. These definitions involve a partition of v as $v = (z, v_{-z})$ for a selected input $z \in \mathbb{R}$ of interest

in the conditional and two-way interaction average predictive comparisons, and the new terms of $q_{-z} = \left(u^{(1)}, u^{(2)}, v_{-z}, \beta\right)$ and $p\left(q_{-z}\right) = p\left(u^{(1)} \mid v\right) p\left(u^{(2)} \mid v\right) p\left(v_{-z}\right) p(\beta)$. Our selection of these estimands is motivated by the effect hierarchy principle that low-order input effects are more likely to be active than high-order effects (Wu and Hamada, 2009, p. 172).

Definition 3.3. The conditional predictive comparison of input values $u^{(1)}, u^{(2)}$ given z is

$$\delta_{u|z}\left(u^{(1)} \to u^{(2)}, z\right) = \left\{\mathbb{E}\left(y \mid u^{(2)}, z, v_{-z}, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, z, v_{-z}, \beta\right)\right\} \Big/ \left(u^{(2)} - u^{(1)}\right).$$

Definition 3.4. The two-way interaction predictive comparison of input values $u^{(1)}, u^{(2)}$, and $z^{(1)}, z^{(2)}$ is

$$\delta_{u\times z}\left(u^{(1)} \to u^{(2)},\ z^{(1)} \to z^{(2)}\right) = D\left(\left(u^{(1)}, u^{(2)}\right) \times \left(z^{(1)}, z^{(2)}\right)\right) \Big/ \left(u^{(2)} - u^{(1)}\right)\left(z^{(2)} - z^{(1)}\right),$$

where

$$\begin{aligned} D\left(\left(u^{(1)}, u^{(2)}\right) \times \left(z^{(1)}, z^{(2)}\right)\right) = {} & \mathbb{E}\left(y \mid u^{(2)}, z^{(2)}, v_{-z}, \beta\right) - \mathbb{E}\left(y \mid u^{(2)}, z^{(1)}, v_{-z}, \beta\right) \\ & - \mathbb{E}\left(y \mid u^{(1)}, z^{(2)}, v_{-z}, \beta\right) + \mathbb{E}\left(y \mid u^{(1)}, z^{(1)}, v_{-z}, \beta\right). \end{aligned}$$

Definition 3.5. The average conditional predictive comparison estimand of u given z is

$$\Delta_{u|z} = \frac{\displaystyle\int_{U^+_{-z}} \left\{\mathbb{E}\left(y \mid u^{(2)}, z, v_{-z}, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, z, v_{-z}, \beta\right)\right\} p\left(q_{-z}\right) \mathrm{d}q_{-z}}{\displaystyle\int_{U^+_{-z}} \left(u^{(2)} - u^{(1)}\right) p\left(q_{-z}\right) \mathrm{d}q_{-z}},$$

where $U^+_{-z} = \left\{u^{(1)}, u^{(2)} \in \mathbb{R},\ v_{-z} \in \mathbb{R}^{K-2},\ \beta \in \mathbb{R}^L : u^{(1)} < u^{(2)}\right\}$.

Definition 3.6. The average two-way interaction predictive comparison estimand of (u, z) is

$$\Delta_{u\times z} = \frac{\displaystyle\int_{(U,Z)^+} D\left(\left(u^{(1)}, u^{(2)}\right) \times \left(z^{(1)}, z^{(2)}\right)\right) p\left(u^{(1)}, u^{(2)}, z^{(1)}, z^{(2)} \mid v_{-z}\right) p\left(v_{-z}\right) p(\beta)\mathrm{d}q}{\displaystyle\int_{(U,Z)^+} \left(u^{(2)} - u^{(1)}\right)\left(z^{(2)} - z^{(1)}\right) p\left(u^{(1)}, u^{(2)}, z^{(1)}, z^{(2)} \mid v_{-z}\right) p\left(v_{-z}\right) p(\beta)\mathrm{d}q},$$

where $(U, Z)^+ = \left\{u^{(1)}, u^{(2)}, z^{(1)}, z^{(2)} \in \mathbb{R},\ v_{-z} \in \mathbb{R}^{K-2},\ \beta \in \mathbb{R}^L : u^{(1)} < u^{(2)},\ z^{(1)} < z^{(2)}\right\}$.

Both estimands in Definitions 3.5 and 3.6 capture the dependencies of the predictive comparisons of one input on the level of another. The first involves separate averages of the numerator and denominator of $\delta_{u|z}\left(u^{(1)} \to u^{(2)}, z\right)$ over $u^{(1)}, u^{(2)}, v_{-z}$, and β for all increasing transitions of u. The second is similarly specified using $\delta_{u\times z}\left(u^{(1)} \to u^{(2)}, z^{(1)} \to z^{(2)}\right)$.

Similar to the first step, we construct estimators for these new estimands as

$$\widehat{\Delta}_{u|z} = \frac{\displaystyle\sum_{s=1}^{S}\sum_{i=1}^{n}\sum_{j=1}^{n}\left\{\mathbb{E}\left(y \mid u_j, z, v_{-z,i}, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, z, v_{-z,i}, \beta^{(s)}\right)\right\} w_{ij}\mathrm{sign}(u_j - u_i)}{S\displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n}(u_j - u_i)w_{ij}\mathrm{sign}(u_j - u_i)}, \tag{3.4}$$

$$\widehat{\Delta}_{u\times z} = \frac{\displaystyle\sum_{s=1}^{S}\sum_{i=1}^{n}\sum_{j=1}^{n} d((u_i, u_j) \times (z_i, z_j))w_{ij}\mathrm{sign}\{(u_j - u_i)(z_j - z_i)\}}{S\displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n}(u_j - u_i)(z_j - z_i)w_{ij}\mathrm{sign}\{(u_j - u_i)(z_j - z_i)\}}, \tag{3.5}$$

with

$$\begin{aligned} d((u_i, u_j) \times (z_i, z_j)) = {} & \mathbb{E}\left(y \mid u_j, z_j, v_{-z,i}, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_j, z_i, v_{-z,i}, \beta^{(s)}\right) \\ & - \mathbb{E}\left(y \mid u_i, z_j, v_{-z,i}, \beta^{(s)}\right) + \mathbb{E}\left(y \mid u_i, z_i, v_{-z,i}, \beta^{(s)}\right). \end{aligned}$$

The standard error of estimator $V_{(u,z)} \in \left\{\widehat{\Delta}_{u|z}, \widehat{\Delta}_{u\times z}\right\}$ is derived as $\mathrm{SE}\left(V_{(u,z)}\right) = \left\{\sum_{s=1}^{S}\left(V_{(u,z)}^{(s)} - V_{(u,z)}\right)^2 \Big/ (S-1)\right\}^{1/2}$ in the supplement. Estimators for the magnitudes of the two-way interaction are

$$\widehat{\Delta}_{\mathrm{mag}(u\times z)} = \left[\frac{\sum_{s=1}^{S}\sum_{i=1}^{n}\sum_{j=1}^{n} d((u_i,u_j)\times(z_i,z_j))^2 w_{ij}}{S\sum_{i=1}^{n}\sum_{j=1}^{n}\{(u_j-u_i)(z_j-z_i)\}^2 w_{ij}}\right]^{1/2}, \tag{3.6}$$

$$\widehat{\Lambda}_{u\times z} = \Bigg[\frac{1}{S}\sum_{s=1}^{S}\sum_{i=1}^{n}\Bigg\{\mathbb{E}\left(y \mid u_i, z_i, v_{-z,i}, \beta^{(s)}\right) + \overline{E\left(y \mid x, \beta^{(s)}\right)} - \overline{E_{u|v_i}\left(y \mid u, v_i, \beta^{(s)}\right)} - \overline{E_{z|u_i,v_{-z_i}}\left(y \mid u_i, z, v_{-z,i}, \beta^{(s)}\right)}\Bigg\}^2\Bigg]^{1/2}, \tag{3.7}$$

where we define $\overline{E\left(y \mid x, \beta^{(s)}\right)} = n^{-1}\sum_{i=1}^{n}\mathbb{E}\left(y \mid x_i, \beta^{(s)}\right)$ as the overall predicted mean. Theorem 3.3 summarizes a result on the Fisher consistencies of these estimators for the corresponding estimands in the second step of our methodology.

Theorem 3.3. If the metric M in $\widehat{\Delta}_{u|z}$, $\widehat{\Delta}_{u\times z}$, $\widehat{\Delta}_{\mathrm{mag}(u\times z)}$, and $\widehat{\Lambda}_{u\times z}$ satisfies the stable metric-input distribution assumption, then these estimators are Fisher consistent for their corresponding estimands.

3.3 Illustrative studies

3.3.1 Simulation study on a Bayesian neural network

Our first illustration of the new predictive comparison methodology involves a simulation study based on an example from Oakley and O'Hagan (2004) and Surjanovic and Bingham (2013). In this study, we have 250 observations and 15 inputs

in which inputs 1 to 5 have negligible effects, 6 to 10 have moderate effects, and 11 to 15 have significant effects. The data are simulated as

$$y_i = x_i^\mathsf{T} a_1 + s(x_i)^\mathsf{T} a_2 + c(x_i)^\mathsf{T} a_3 + x_i^\mathsf{T} A x_i + \epsilon_i,$$

where (1) $\epsilon_i \sim \mathrm{N}(0,1)$ independently, (2) each $x_i \in \mathbb{R}^{15}$ has as it entries independent and identically distributed $\mathrm{N}(0,1)$ random variables, (3) each $s(x_i), c(x_i) \in \mathbb{R}^{15}$ are set by evaluating the sine and cosine functions, respectively, for each entry in x_i, (4) $a_1, a_2, a_3 \in \mathbb{R}^{15}$ are fixed, and (5) A is a fixed 15×15 matrix. The following Bayesian neural network with one hidden layer and three hidden neurons,

$$y_i = \beta_0 + \sum_{m=1}^{3} \beta_m \tanh\left(\alpha_{0,m} + \sum_{k=1}^{15} \alpha_{k,m} x_{i,k}\right) + \epsilon_i,$$

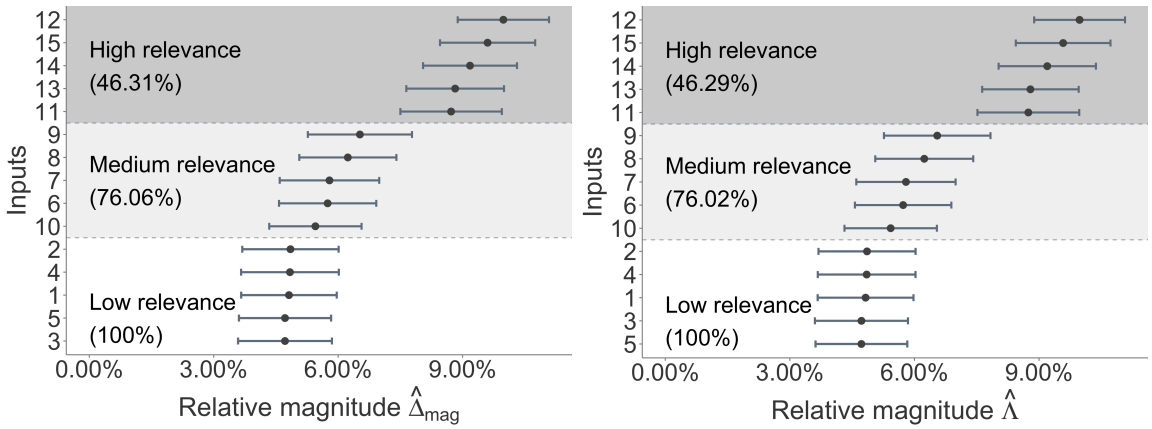
provides a good fit here, where $\epsilon_i \sim N(0, \sigma^2)$ and the joint prior probability density function for $\boldsymbol{\beta}, \boldsymbol{\alpha}$, and $\log \sigma$ is

$$p(\boldsymbol{\alpha}, \boldsymbol{\beta}, \log \sigma) \propto \exp\left(\frac{-\boldsymbol{\alpha}^\mathsf{T}\boldsymbol{\alpha}}{20}\right) \exp\left(\frac{-\boldsymbol{\beta}^\mathsf{T}\boldsymbol{\beta}}{20}\right).$$

Figure 3.1 summarizes the results of 1000 simulations for the two types of relevance measures from the first step of the methodology. Inputs 11 to 15 are effectively classified as the five most relevant inputs under both. The average of the root mean square errors for the training data across the simulations was 4.37, and the standard deviation was 0.32. The corresponding values for the test data (each of which involved an additional 250 observations) were 1.16 and 0.58, respectively.

3.3.2 Understanding a BART model for student performance

We next consider the real-life case study in (Cortez and Silva, 2008) of constructing a predictive model for student performance. Cortez and Silva (2008) applied a variety of machine learning algorithms in their study of student performance. Their data



(a) Screening inputs with $R\left(\widehat{\Delta}_{\text{mag}(u)}\right)$.

(b) Screening inputs with $R\left(\widehat{\Lambda}_u\right)$.

Figure 3.1.: Summaries of model interpretations over 1000 simulated datasets. In this simulation study, inputs 11 to 15 are most relevant, 6 to 10 are of medium relevance, and 1 to 5 are least relevant. Cumulative relevance is presented in parentheses for each set of five inputs. Dots represent mean values, and bars represents one standard deviation, over the simulations.

consist of 649 secondary school students and 30 inputs, four of which are categorical. Descriptions of the inputs are in the supplement. Performance is the final grade, on a $0-20$ scale, in a Portuguese class. We implement our new predictive comparison methodology to interpret the associations inferred by BART (McCulloch et al., 2018) in this study.

Computation of the weights in this case requires care because few students belong to precisely the same set of categories. For a quantitative input u, v is partitioned as $v = (v^{\text{cat}}, v^{\text{quant}})$, where $v^{\text{cat}} = (v^{\text{cat}_1}, \ldots, v^{\text{cat}_4})$ contains the categorical inputs and v^{quant} the remaining quantitative inputs. Weights w_{ij} are defined as $w_{ij} = g\left(v_i^{\text{cat}}, v_j^{\text{cat}}\right) / \left\{1 + M\left(v_i^{\text{quant}}, \quad v_j^{\text{quant}}\right)\right\}$, where

$$g\left(v_i^{\text{cat}_k}, v_j^{\text{cat}_k}\right) = \begin{cases} 1 & \text{if } v_i^{\text{cat}} = v_j^{\text{cat}}, \\ 1/n_{\text{cat}_k} & \text{if } v_i^{\text{cat}} \neq v_j^{\text{cat}}, \end{cases}$$

and n_{cat_k} is the number of categories in v^{cat_k} for $k = 1, \ldots, 4$.

The results for interpreting the inputs are summarized in Figure 3.2(a), and for identifying relevant inputs are in Figures 3.2(b) and 3.2(c). The gray areas in the latter figures correspond to approximately 80% cumulative relevance. Seventeen inputs capture approximately 80% of the total relevance under $R\left(\widehat{\Delta}_{\text{mag}(u)}\right)$, and twenty inputs capture approximately 80% of the total relevance under $R\left(\widehat{\Lambda}_u\right)$. The top four inputs in terms of $R\left(\widehat{\Delta}_{\text{mag}(u)}\right)$ are among the top five most relevant inputs under $R\left(\widehat{\Lambda}_u\right)$, and were also identified as the most important by Cortez and Silva (2008, p. 7). One relevant input under either measure is the desire to take higher education, which is associated with an increase of 1.42 points in final grade. Another is attending the school Mousinho da Silveira instead of Gabriel Pereira, which is associated with a 1.07 point decrease in final grade. The standard average predictive comparison suggests that receiving extra educational support from a school is associated with a decrease in final grade, which may appear contradictory at first. However, a better understanding can be obtained by considering the conditional effects of school support given school. Specifically, Gabriel Pereira has an average predictive comparison estimate of -1.06 (with standard error 0.32), while Mousinho da Silveira has an average predictive comparison estimate of -1.09 (with standard error of 0.30), and these consistent negative conditional associations between school support and final grade is likely due to lurking confounders for both (e.g., school support only being offered to poorly performing students).

To further evaluate these results, the full model was compared against reduced models containing those inputs that cumulatively represent approximately 80% of total relevance based on either $R\left(\widehat{\Delta}_{\text{mag}(u)}\right)$ or $R\left(\widehat{\Lambda}_u\right)$. Two additional scenarios were also considered: (1) using all data to fit the models, and (2) splitting the data into training and test sets and obtaining the metrics from the test set. Mean absolute deviation, root mean squared error, and root relative squared error of predictions

were used for the comparisons, and are summarized in Figure 3.2(d). These three quantities are defined as, respectively,

$$\text{MAD}(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|,$$

$$\text{RMSE}(y, \hat{y}) = \left\{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2\right\}^{1/2},$$

$$\text{RRSE}(y, \hat{y}) = \left\{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}\right\}^{1/2},$$

where $y = (y_1, \ldots, y_n)$ denotes the observed set of outcomes, $\bar{y}$ is their average, and $\hat{y} = (\hat{y}_1, \ldots, \hat{y}_n)$ denotes their predictions under a specified model. By inspection, the reduced and full models behave similarly.

3.3.3 Interpreting a SVM for wine preferences

Our final illustration is based on the case study in (Cortez et al., 2009) of predicting wine preferences. Cortez et al. (2009) compared the predictive performances of neural networks, SVMs, and linear models in this context, and concluded that the second algorithm yields the highest predictive accuracy. Understanding the possible physicochemical effects of inputs on preference is also important for improving production and advancing certification processes. We apply our new predictive comparison methodology to perform this interpretation from the SVM for the red wine preference data. For this data, preference is calculated as the median of at least three scores on a 0 to 10 scale.

The SVM algorithm is implemented using a Gaussian kernel and standardized inputs (Pedregosa et al., 2011). Hyperparameters are fixed at precision $\gamma = 0.125$, penalty $C = 1$, and $\alpha \approx 0.008$ in the insensitive-loss function. Both γ and C are de-

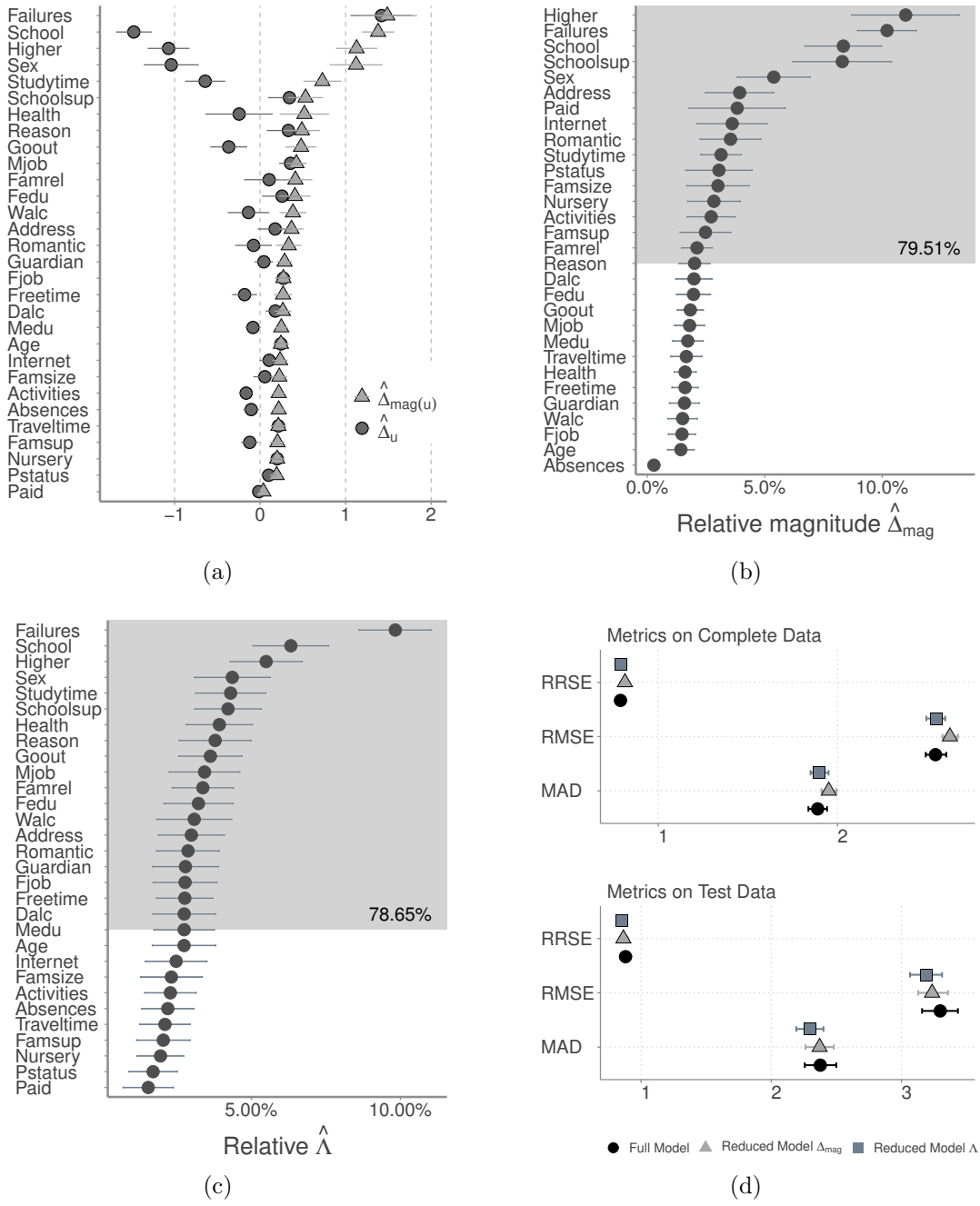


Figure 3.2.: Interpretations of the model fit for student performance. The grey areas represent approximately 80% of the total relative relevance. (a) Summaries of the average predictive comparisons $\widehat{\Delta}_u$ and average magnitude predictive comparisons $\widehat{\Delta}_{\mathrm{mag}(u)}$. (b) Screening relevant inputs based on $R\left(\widehat{\Delta}_{\mathrm{mag}(u)}\right)$. (c) Screening relevance based on $R\left(\widehat{\Lambda}_u\right)$. (d) Performance measurements calculated using either the entire data or only the test set for the full model (circle), and for the models with selected inputs corresponding to approximately 80% of total relevance with respect to $R\left(\widehat{\Delta}_{\mathrm{mag}(u)}\right)$ (triangle) and $R\left(\widehat{\Lambda}_u\right)$ (square).

termined by five-fold cross-validation, and $\alpha = \hat{\tau}/\sqrt{n}$, where $\hat{\tau} = 1.5\sum_{i=1}^{n}(y_i - \hat{y}_i)^2/n$ with $\hat{y}_i$ denoting the predicted outcome obtained by a three-nearest neighbor algorithm (Cortez et al., 2009; Cherkassky and Ma, 2004). Uncertainties in the parameters are quantified by the semiparametric bootstrap.

The interpretations of the inputs are in Figure 3.3(a), and their relevances are in Figure 3.3(b). From the first figure, increases in alcohol and sulphates are associated with increased preference, while increases in volatile acidity and total sulfur dioxide are associated with decreased preference. For example, an increase in scaled alcohol is associated with an average preference increase of 0.32. The second figure indicates that the most relevant inputs, corresponding to half of the total relevance, are alcohol, sulphates, volatile acidity, and total sulfur dioxide. Changes in these inputs yield, on average, greater changes in wine preference. Cortez et al. (2009) found these same inputs to have the most relevance, albeit in a different order.

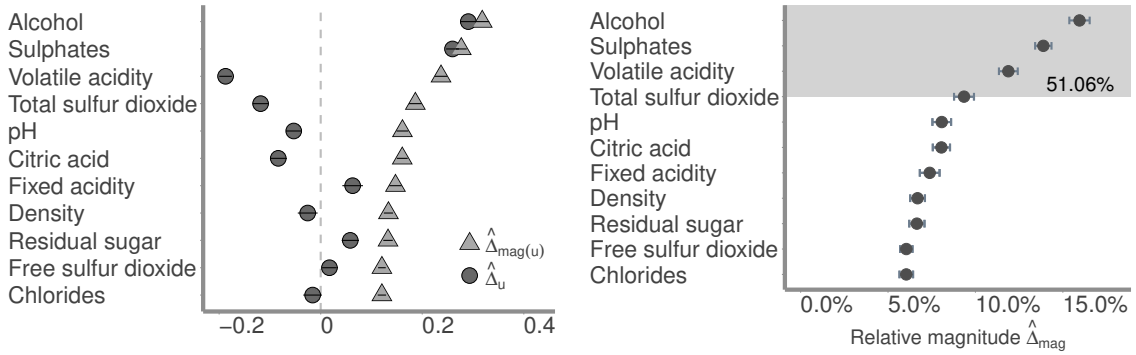


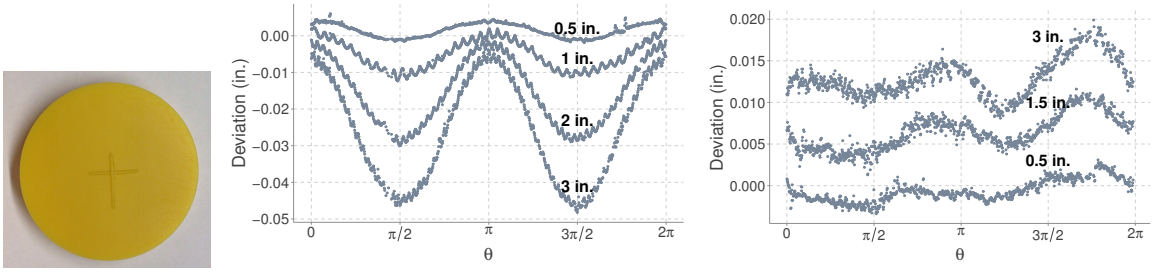
Figure 3.3.: (a) Average predictive comparisons $\widehat{\Delta}_u$ and average magnitude predictive comparisons $\widehat{\Delta}_{\mathrm{mag}(u)}$ for the standardized physicochemical inputs' associations with red wine preference. (b) Screening relevant inputs based on $R\left(\widehat{\Delta}_{\mathrm{mag}(u)}\right)$.

3.4 Screening and interpreting inputs in Bayesian neural network models for shape deviations

3.4.1 Description of additive manufacturing processes and data

Our final case study is on the interpretations of inputs in the Bayesian neural network algorithm devised in Chapter 2 for automated geometric shape deviation modeling across different shapes and processes in an AM system. The additively manufactured products in this study are seven cylinders with negligible heights. A sample product is in Figure 3.4(a). Two different process settings of a single EnvisionTEC ULTRA stereolithography machine, referred to as "process A" and "process B" and described in Table 2.1, were used to manufacture the cylinders. Four cylinders of nominal radii $0.5'', 1'', 2''$, and $3''$ were manufactured under process A, and three cylinders of nominal radii $0.5'', 1.5''$, and $3''$ were manufactured under process B. In-plane shape deviations for the top and bottom boundaries in these products were effectively identical, and so we model the data of the top boundary in-plane deviations. All data were collected in a point cloud format using a single Micro-Vu Vertex system, and the points on a product were identified by Cartesian coordinates that were defined with respect to physical coordinate axes printed directly on the product (Figure 3.4(a)). The in-plane deviations of these products were previously analyzed by Huang et al. (2015b), Luan and Huang (2017), Sabbaghi et al. (2018), and Sabbaghi and Huang (2018) using Bayesian nonlinear regression models.

Our Bayesian neural network approach utilizes functional representations of both in-plane and out-of-plane shape deviations that were formulated by Huang et al. (2015b) and Huang (2016), respectively. Under the former representation, each point i on the boundary of a manufactured product is identified by an angle θ_i under the polar coordinate system of the product on which it resides. The deviation y_i for point i is defined as the difference between the observed radius and the nominal radius of θ_i, where the latter is specified according to the computer-aided design model of the shape (Huang et al., 2015b, p. 432). Figures 3.4(b) and 3.4(c) illustrate the deviations



(a) Manufactured cylinder. (b) In-plane deviations under process A (c) In-plane deviations under process B

Figure 3.4.: Shape deviations for cylinders of different nominal radii (indicated by the labels) manufactured under two distinct processes A and B.

for the seven cylinders across the two AM processes under this representation. Further details can be found in Chapter 2 and Sections 2.3.2 and 2.3.3.

3.4.2 Understanding non-monotonic relationships, two-way interactions, and conditional effects of additive manufacturing inputs from Bayesian neural networks

The Bayesian neural network method described in Chapter 2 possesses a great capability for predicting in-plane and out-of-plane deviations of a wide variety of products, including free-form shapes and products manufactured under distinct processes. However, its complicated neural network structures impede interpretations or insights on the inferred relationships between inputs (e.g., the angle θ and nominal radius r_0 of a point) and shape deviation. We proceed to demonstrate how the estimands in our new predictive comparison methodology yield clear and meaningful interpretations of the AM inputs from these neural network deviation models for cylinders manufactured under the two processes.

Consider the four cylinders under process A. Our inferences on the predictive comparison estimands for these products are summarized in Figure 3.5. There are three important points to observe from these inferences. First, the angle input θ has a harmonic relationship with deviation under process A, and so the standard estimand

Δ_θ is zero. The method of Gelman and Pardoe (2007) then leads to the misleading conclusion that angle is not associated with deviation. In contrast, our estimand $\Delta_{\mathrm{mag}(\theta)}$ enables one to correctly identify angle as a relevant input. Second, we can now identify the two-way interaction between angle and radius as relevant, and interpret it. Two-way interactions were not previously considered by Gelman and Pardoe (2007). Third, our inferences on the conditional average predictive comparison $\Delta_{r_0|\theta}$ in Figure 3.5(b) yields the insight that the association between radius and deviation has a strong dependence on angle. The effect of the nominal radius on deviation is largest in magnitude at the top and bottom poles of cylinders (i.e., $\theta = \pi/2, 3\pi/2$). This corresponds to the fact, inferred from previous non-linear regression analyses, that absolute deviation increases exponentially as a function of nominal radius at the topmost and bottommost points of cylinders manufactured under process A.

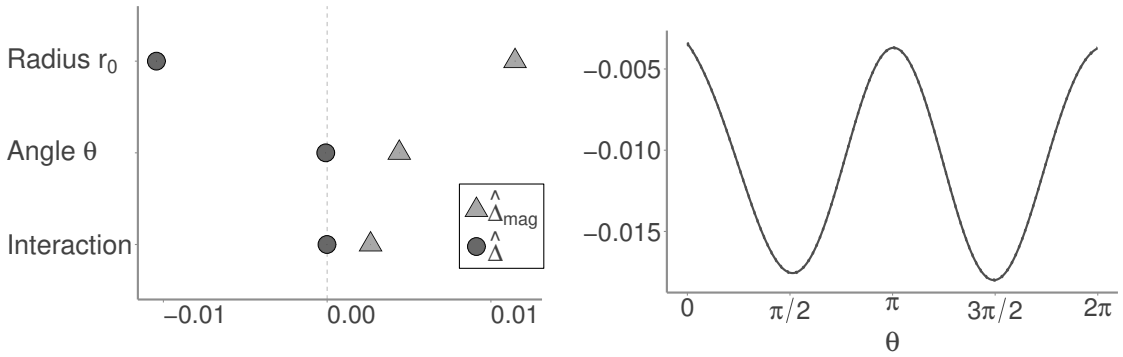
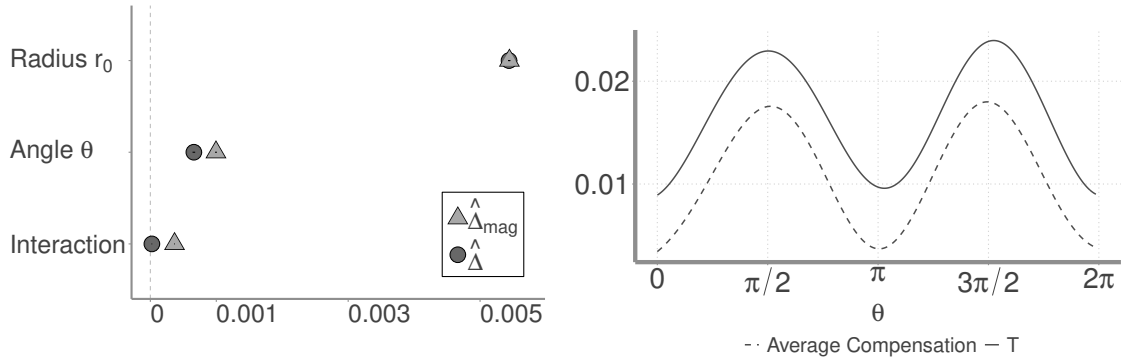


Figure 3.5.: Interpretations of the AM inputs from the neural network model for cylinders under process A. (a) Estimates of the average predictive comparisons $\widehat{\Delta}_u$ (circle) and average magnitude predictive comparisons $\widehat{\Delta}_{\mathrm{mag}(u)}$ (triangle) for the nominal radius and angle inputs, and estimates $\widehat{\Delta}_{r_0\times\theta}, \widehat{\Delta}_{\mathrm{mag}(r_0\times\theta)}$ for their two-way interaction. (b) Estimates for the conditional average predictive comparison of nominal radius given angle, $\widehat{\Delta}_{r_0|\theta}$.

Interpretations of the AM inputs from the neural network model for the three cylinders under process B are similarly obtained and summarized in Figure 3.6. We observe that the average predictive comparisons for nominal radius, angle, and their two-way interaction under process B are inferred to be smaller in absolute value

than those under process A. This can be attributed to the fact that the change in process settings from A to B was essentially an attempted reproduction of the optimum compensation plan of Huang et al. (2015b), which decreased the severity of shape deviations (Sabbaghi and Huang, 2018, p. 2412). Of particular significance are the insights we can obtain from our average conditional predictive comparison for the total equivalent amount of the process setting change in terms of compensation under process A given angle (Sabbaghi and Huang, 2018) (Figure 3.6(b)). The concept of the total equivalent amount of a process setting change in terms of compensation in AM was first formulated by Sabbaghi and Huang (2018), and corresponds to a function that benchmarks the effect of the change using an existing model for the effect of compensation under the previous process. More details on the total equivalent amount, and how it can be inferred and modeled, are in Chapter 2, (Sabbaghi and Huang, 2016), and (Sabbaghi and Huang, 2018). By comparing the estimates of the average conditional predictive comparisons for the total equivalent amount given angle with the average compensation under the process A model derived from (Huang et al., 2015b, p. 434), we can more definitively conclude that cylinders manufactured under B are equivalent to overcompensated cylinders under A. This again corresponds to the previously stated fact, and helps to explain inferences on the AM inputs from the neural network deviation model in light of the complex deviations under process B.



(a) (b)

Figure 3.6.: Interpretations of the AM inputs from the neural network model for cylinders under process B. (a) Estimates of the average predictive comparisons $\widehat{\Delta}_u$ (circle) and average magnitude predictive comparisons $\widehat{\Delta}_{\text{mag}(u)}$ (triangle) for the nominal radius and angle inputs, and estimates $\widehat{\Delta}_{r_0\times\theta}$, $\widehat{\Delta}_{\text{mag}(r_0\times\theta)}$ for their two-way interaction. (b) Estimates of the conditional average predictive comparison for the total equivalent amount of the change in process settings from A to B in terms of compensation under A given angle (solid), and the average compensation derived from (Huang et al., 2015b, p. 434) (dashed).

3.5 Discussion and Extension of Predictive Comparison Methodology

The new predictive comparison methodology described in this chapter helps to address the important objective of obtaining insightful interpretations of the inputs in complex, black box machine learning algorithms and models. We demonstrated the practical significance, and distinct nature, of this method compared to the established approach of Gelman and Pardoe (2007) with both simulation and real-life studies. It is important to recognize that our new predictive comparison methodology is not meant to replace that of Gelman and Pardoe (2007). Instead, it serves as an informative and enlightening supplement for contexts in which standard predictive comparison estimands are insufficient. One such context is shape deviation modeling in AM systems, in which complicated relationships between the AM inputs and deviations generally exist due to the complex physical phenomena and processes involved with AM.

The formulation of the predictive comparisons methodology in this chapter is targeted towards handling inputs on an individual basis. However, in many contexts it is of more interest to assess and interpret the relationships between multiple inputs simultaneously, or between functional forms of the inputs with the outcome. Two general examples of such contexts are provided below.

Example 3.1. Consider the interpretation of a complex image classifier, where an image's pixels are taken as the inputs. In this context, analyzing a group of pixels is generally of more interest than analyzing each individual pixel. This is because the effects of individual pixels may not shed any light on the patterns inferred by the complex classifier. The predictive comparisons methodology as formulated in this chapter is unable to assess the inferred relationships between groups of pixels *simultaneously* with the image's classification.

Example 3.2. Consider an SVM model for income prediction. Suppose the input of interest is an individual's age. In practice, it is useful to understand not only the increase in income that is expected to occur as age increases, but also how the prediction of income is related to different age ranges. Again, the predictive comparison methodology cannot address this objective in its current formulation.

In general, acquiring insights into the relationships between inputs and the outcome is an exploratory task that may require investigations on different functional forms of the inputs. Figure 3.7 illustrates the process for interpreting functional forms of inputs under the predictive comparison methodology. As shown in this figure, in order for one to interpret each new functional form of interest, one must take the additional efforts of first including it as an input, and then refitting the model with the new inputs. As this can be inconvenient in practice, in the next chapter we extend our predictive comparison methodology so as to interpret different functional forms of the inputs without requiring the model to be refitted to the data.

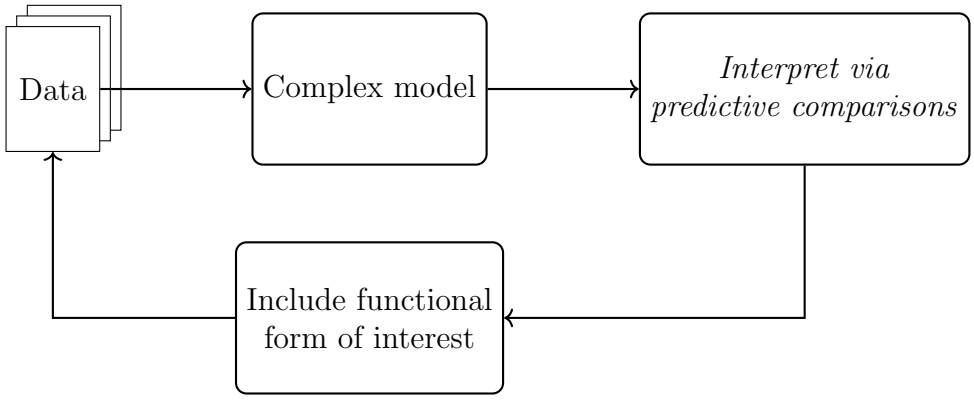


Figure 3.7.: Process flow for interpreting inputs in complex models by means of our predictive comparison methodology. To understand the effects of relevant functional forms of inputs that were not previously considered, one must explicitly include them as inputs and obtain a new model.

4. GENERALIZED PREDICTIVE COMPARISONS FOR INTERPRETING COMPLEX MODELS

4.1 Notation and assumptions

In Chapter 3, we described a predictive comparisons methodology for interpreting and screening inputs on a one-input-at-a-time basis. Here, we extend that methodology to screen and interpret functional inputs simultaneously. A novel aspect of this extension that distinguishes it from the previous methodology is the introduction of new *individual* predictive comparisons for specified observations that can further help address model interpretability for data with complex structures.

Let $p(y \mid x, \beta)$ be a probability density function for a model on data $\{(x_i, y_i)\}_{i=1}^n$, where $y_i \in \mathbb{R}$ is the outcome and $x_i \in \mathbb{R}^K$ is the vector of inputs for observation i, and $\beta \in \mathbb{R}^L$ is the vector of unknown parameters. We partition input x into three different vectors. The first level of the partition is as $x = (u, v)$, where $u \in \mathbb{R}^d$ contains those inputs whose relationships with the outcome are of primary interest and $v \in \mathbb{R}^{K-d}$ contains the remaining entries of x ($1 \leq d \leq K$). The second level of the partition is for the case of $d < K-1$, in which case v is partitioned as $v = (z, v_{-z})$ with $z \in \mathbb{R}^b$ ($b \leq K - d$). Vector z contains those inputs whose effects of interest are the "secondary" effects of two-factor interactions of u and z and the conditional main effects of u given z. We further assume that the inferential uncertainty for β is described by a distribution $p(\beta)$, from which samples $\beta^{(s)}, s = 1, \ldots, S$, can be drawn. Throughout this chapter, $u^{(1)}$ and $u^{(2)}$ will denote two possible vectors of inputs u.

4.2 Generalized predictive comparisons methodology for globally interpreting and screening of inputs

4.2.1 Interpretable estimands

A crucial step in interpreting a complex model is to define *predictive estimands* from the model that are understandable to humans. Ribeiro et al. (2016) focused on interpretable functional forms of the data, such as indicators of the presence or absence of certain inputs. To illustrate their idea, consider a text classifier model. To locally explain an individual prediction, e.g., a sentence, they would assess the effect of the inclusion or exclusion of a certain word on a sentence's predicted class. Gelman and Pardoe (2007) proposed estimands for global interpretability of a single input that effectively capture the expected change in the outcome, as predicted by the model, for an unit increase in the given input. This type of estimand is easily understandable to humans, and was also considered in the estimands in Chapter 3 for screening inputs and understanding their conditional and two-way associations with the outcome.

For the case of multiple inputs and their different functional forms, defining predictive estimands that are readily understandable requires care. Attempting to directly apply the framework of Gelman and Pardoe (2007) or the methodology in Chapter 3 would not suffice because they cannot readily yield interpretable or clear estimands in terms of an increase in *vector* of inputs. In order to derive interpretable estimands for a vector of inputs $u \in \mathbb{R}^d$, we define a direction for the increase by means of an interpretable mapper in Definition 4.1.

Definition 4.1. An interpretable mapper $\text{im} : \mathbb{R}^d \to \mathbb{R}$ is a well-defined function on the domain of the inputs u of primary interest that maps them to the real line, and provides understandable and unambiguous interpretations of u.

Example 4.1. Consider a text classifier model for sentiment analysis in reviews. The outcome in this case is either "positive" or "negative". The inputs are constructed

by means of a bag-of-words representation of a review, i.e., a simplified text representation that describes the occurrences of words within each review. To illustrate, the inputs corresponding to the observed reviews *"Excellent breakfast. Very comfortable beds."* and *"Very bad service. Bad location."* would be represented in Table 4.1 as the frequencies of different words utilized across all reviews. For example, one such input u_1 is the frequency with which *"excellent"* is used in a particular review, and one particular interpretable mapper is $\text{im}(u) = \mathbb{I}\{u_1 > 0\}$, which indicates the presence of u_1.

Table 4.1.: Example of inputs in a bag-of-words text classifier.

	Words Across All Reviews							
Review	*excellent*	*breakfast*	*very*	*comfortable*	*beds*	*bad*	*service*	*location*
1	1	1	1	1	0	0	0	0
2	0	0	1	0	0	2	1	1

Other examples of interpretable mappers that can be applied are the mean and norm functions. The ultimate choice of an interpretable mapper is context dependent, and should be chosen with respect to the utility of the interpretations it can provide. It is important to note that the inclusion of an interpretable mapper allows a deeper exploration of a model in terms of different functional forms of inputs, even when a single input is considered. This fact is further discussed in our case studies.

The remainder of this chapter is organized as follows. In Chapter 4.2.2 we propose generalized predictive comparison estimands for the globally interpretation and screening of inputs. Estimands for capturing the dependencies of the generalized predictive comparisons of certain inputs u upon another inputs z are described in Chapter 4.2.3. We finally propose individual generalized predictive comparisons for interpreting and screening the inputs for a certain observation.

4.2.2 Generalized predictive comparison estimands for interpretable mappers

For a specific transition from $u^{(1)}$ to $u^{(2)}$, the generalized predictive comparison of u on y given v, β, and $\text{im} : \mathbb{R}^d \to \mathbb{R}$ is

$$\delta_{im}\left(u^{(1)} \to u^{(2)}\right) = \left\{\mathbb{E}\left(y \mid u^{(2)}, v, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, v, \beta\right)\right\} \Big/ \left\{\text{im}\left(u^{(2)}\right) - \text{im}\left(u^{(1)}\right)\right\}.$$

This predictive comparison is utilized in the following definitions of generalized average predictive comparison estimands for interpretable mappers.

Definition 4.2. The generalized average predictive comparison (GEAR) estimand for $u \in \mathbb{R}^d$ under $\text{im} : \mathbb{R}^d \to \mathbb{R}$ is

$$\Delta_{\text{im}(u)} = \frac{\displaystyle\int_{(\text{IM})^+} \left\{\mathbb{E}\left(y \mid u^{(2)}, v, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, v, \beta\right)\right\} p(q)\mathrm{d}q}{\displaystyle\int_{(\text{IM})^+} \left\{\text{im}\left(u^{(2)}\right) - \text{im}\left(u^{(1)}\right)\right\} p(q)\mathrm{d}q},$$

where $(\text{IM})^+ = \left\{u^{(1)}, u^{(2)} \in \mathbb{R}^d,\ v \in \mathbb{R}^{K-d},\ \beta \in \mathbb{R}^L : \text{im}\left(u^{(1)}\right) < \text{im}\left(u^{(2)}\right)\right\}$.

Definition 4.3. The generalized average magnitude predictive comparison (GAME) estimand for $u \in \mathbb{R}^d$ under $\text{im} : \mathbb{R}^d \to \mathbb{R}$ is

$$\Delta_{\text{mag}(\text{im}(u))} = \left\{ \frac{\displaystyle\int \left\{\mathbb{E}\left(y \mid u^{(2)}, v, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, v, \beta\right)\right\}^2 p(q)\mathrm{d}q}{\displaystyle\int \left\{\text{im}\left(u^{(2)}\right) - \text{im}\left(u^{(1)}\right)\right\}^2 p(q)\mathrm{d}q} \right\}^{1/2}.$$

The GEAR estimand in Definition 4.2 is interpreted as the expected change in the outcome y as $\text{im}(u)$ increases by an unit, while the GAME estimand in Definition 4.3 can be viewed as the magnitude of the expected change in the outcome as $\text{im}(u)$ changes by one unit. Note that the estimand of Gelman and Pardoe (2007) and those described in Chapter 3.2.1 are, respectively, special cases of the new estimands in

Definitions 4.2 and 4.3 when $d = 1$ and the interpretable mapper is defined as the identity function $\text{im}_{\text{id}}(u) = u$ for all u.

We construct estimators for these estimands as $\widehat{\Delta}_{\text{im}(u)} = S^{-1} \sum_{s=1}^{S} \widehat{\Delta}^{s}_{\text{im}(u)}$, and $\widehat{\Delta}_{\text{mag}(\text{im}(u))} = \left\{ S^{-1} \sum_{s=1}^{S} \widehat{\Delta^2}^{s}_{\text{mag}(\text{im}(u))} \right\}^{1/2}$, where

$$\widehat{\Delta}^{s}_{\text{im}(u)} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ \mathbb{E}\left(y \mid u_j, v_i, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta^{(s)}\right) \right\} w_{ij} \ \text{sign}(\text{im}(u_j) - \text{im}(u_i))}{\sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ \text{im}(u_j) - \text{im}(u_i) \right\} w_{ij} \ \text{sign}(\text{im}(u_j) - \text{im}(u_i))},$$

$$\widehat{\Delta^2}^{s}_{\text{mag}(\text{im}(u))} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ \mathbb{E}\left(y \mid u_j, v_i, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta^{(s)}\right) \right\}^2 w_{ij}}{\sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ \text{im}(u_j) - \text{im}(u_i) \right\}^2 w_{ij}},$$

and weights $w_{ij} = \left\{1 + M(v_i, v_j)\right\}^{-1}$ for distance metric $M : \mathbb{R}^{K-d} \times \mathbb{R}^{K-d} \to \mathbb{R}$ approximate $p(u_j \mid v)$. As in Chapter 3, the w_{ij} reflect the likelihood of a transition from u_i to u_j when $v = v_i$. These estimators are Fisher consistent under the extended stable metric-input distribution assumption in Condition 4.1.

Condition 4.1. Let $M : \mathbb{R}^{K-d} \times \mathbb{R}^{K-d} \to \mathbb{R}_{\geq 0}$ be a metric on the v inputs. Then M provides a stable metric approximation for the marginal probability density function $p(v)$ and the conditional probability density function $p\left(v \mid u\right)$ if, for any $v^{(1)} \in \mathbb{R}^{K-1}$ and $u^{(2)} \in \mathbb{R}^{d}$,

$$\frac{p\left(v^{(1)}\right)}{\int_{\mathbb{R}^{K-d}} \left\{1 + M(v^{(1)}, v)\right\}^{-1} (v) p(v) dv} = \frac{p\left(v^{(1)} \mid u^{(2)}\right)}{\int_{\mathbb{R}^{K-d}} \left\{1 + M(v^{(1)}, v)\right\}^{-1} p\left(v \mid u^{(2)}\right) dv}, \text{ and}$$

$$\int_{\mathbb{R}^{K-d}} \left\{ \frac{p\left(\tilde{v}\right)}{\int_{\mathbb{R}^{K-d}} \left\{1 + M(\tilde{v}, v)\right\}^{-1} p(v) dv} \right\} p\left(\tilde{v}\right) d\tilde{v} < \infty.$$

Theorem 4.1. If the metric M used in $\widehat{\Delta}_{\text{im}(u)}$ and $\widehat{\Delta}_{\text{mag}(\text{im}(u))}$ satisfies the extended stable metric-input distribution assumption, then $\widehat{\Delta}_{\text{im}(u)}$ and $\widehat{\Delta}_{\text{mag}(\text{im}(u))}$ are Fisher consistent for $\Delta_{\text{im}(u)}$ and $\Delta_{\text{mag}(\text{im}(u))}$, respectively.

As before, and similar to the consideration of Gelman and Pardoe (2007), the uncertainties in these estimators are associated with inferences on the unknown model parameters β. Specifically, the parameters are treated as random and the inputs are considered fixed. From sampling theory methods and the Taylor expansion, the standard errors of estimators are thus derived as

$$\text{SE}\left(\widehat{\Delta}_{\text{im}(u)}\right) = (S-1)^{-1/2}\left\{\sum_{s=1}^{S}\left(\widehat{\Delta}^{s}_{\text{im}(u)} - \widehat{\Delta}_{\text{im}(u)}\right)^2\right\}^{1/2},$$

$$\text{SE}\left(\widehat{\Delta}_{\text{mag}(\text{im}(u))}\right) = \frac{\left\{(S-1)^{-1}\sum_{s=1}^{S}\left\{\widehat{\Delta^2}^{s}_{\text{mag}(\text{im}(u))} - \widehat{\Delta^2}_{\text{mag}(\text{im}(u))}\right\}^2\right\}^{1/2}}{2\widehat{\Delta}_{\text{mag}(\text{im}(u))}}.$$

4.2.3 Relational generalized predictive comparisons

We gain more insights into the joint relationships between inputs and the outcome via relational generalized predictive comparisons. One particular focus is on understanding the predictive comparison of $u \in \mathbb{R}^d$ under certain regions or levels of $z \in \mathbb{R}^b$. In this case, we construct $\text{im}(u \mid z) : \mathbb{R}^{d+b} \to \mathbb{R}$ to restrict the domain of z. The conditional average predictive comparison in Chapter 3 is a special case of $\Delta_{\text{im}(u|z)}$ when $u, z \in \mathbb{R}$ and the partition $\boldsymbol{\mathcal{A}}$ represents all levels of z.

Example 4.2. Consider a classifier for identifying good or bad credit risks among loan applicants. Let the input of interest be the amount a client has in their saving account, $u =$ "savings". One might be interested in understanding how the size of the savings account affects the bank's classification across different age groups. Let $\boldsymbol{\mathcal{A}} = \{\mathcal{A}_1, \cdots, \mathcal{A}_{10}\} \in \mathbb{R}^b$ be a partition of input $z =$ "age". One could then define $\text{im}(u \mid z = a) = u\mathbb{I}\{z \in a\}$, for $a \in \{\mathcal{A}_1, \cdots, \mathcal{A}_{10}\}$. Thus $\Delta_{\text{im}(u|z)}$ can be interpreted as the conditional generalized average predictive comparison (COGEAR)

of u in different regions of the z space. Another possible mapper that can incorporate previously specified interpretable mappers for u is $\text{im}(u \mid z) = \text{im}(u)\mathbb{I}\{z \in a\}$.

In addition to conditional relationships, one might be interested in understanding two-way interactions between u and z under certain interpretable mappers. Motivated by the general definition of two-way interactions given by Hinkelmann and Kempthorne (2007, p. 420), Dasgupta et al. (2015, p. 731), and Cheng (2016, p. 72), we proceed to extend the average two-way interaction predictive comparison previously specified in Chapter 3.

Definition 4.4. The generalized average two-way interaction predictive comparison estimand of $(u, z) \in \mathbb{R}^{d+b}$ under $\text{im}_1 : \mathbb{R}^d \to \mathbb{R}$ for u and $\text{im}_2 : \mathbb{R}^b \to \mathbb{R}$ for z is

$$\Delta_{\text{im}_1(u)\times\text{im}_2(z)} = \frac{\displaystyle\int_{(\text{IM})^+_{1,2}} D\left(\left(u^{(1)}, u^{(2)}\right) \times \left(z^{(1)}, z^{(2)}\right)\right) p^*(q)\,\mathrm{d}q}{\displaystyle\int_{(\text{IM})^+_{1,2}} \left\{\text{im}_1\left(u^{(2)}\right) - \text{im}_1\left(u^{(1)}\right)\right\}\left\{\text{im}_2\left(z^{(2)}\right) - \text{im}_2\left(z^{(1)}\right)\right\} p^*(q)\,\mathrm{d}q},$$

where $D\left(\left(u^{(1)}, u^{(2)}\right) \times \left(z^{(1)}, z^{(2)}\right)\right)$ is defined as in Definition 3.4,

$$p^*(q) = p\left(u^{(1)}, u^{(2)}, z^{(1)}, z^{(2)} \mid v_{-z}\right) p\left(v_{-z}\right) p(\beta), \text{ and}$$
$$(\text{IM})^+_{1,2} = \left\{u^{(1)}, u^{(2)} \in \mathbb{R}^d, z^{(1)}, z^{(2)} \in \mathbb{R}^b,\ v_{-z} \in \mathbb{R}^{K-(d+b)},\ \beta \in \mathbb{R}^L : \text{im}_1(u^{(1)}) < \text{im}_1(u^{(2)}),\right.$$
$$\left.\text{im}_2(z^{(1)}) < \text{im}_2(z^{(2)})\right\}.$$

Our estimator for the generalized average two-way interaction predictive comparison estimand is $\widehat{\Delta}_{\text{im}_1(u)\times\text{im}_2(z)} = S^{-1}\displaystyle\sum_{s=1}^{S} \widehat{\Delta}^s_{\text{im}_1(u)\times\text{im}_2(z)}$, where

$$\widehat{\Delta}^s_{\text{im}_1(u)\times\text{im}_2(z)} = \frac{\displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} d((u_i, u_j) \times (z_i, z_j))\, w_{ij}\, \text{sign}\left\{\kappa_{ij}(\text{im}(u), \text{im}(z))\right\}}{\displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} \kappa_{ij}(\text{im}(u), \text{im}(z))\, w_{ij}\, \text{sign}\left\{\kappa_{ij}(\text{im}(u), \text{im}(z))\right\}},$$

and $\kappa_{ij}(\text{im}(u), \text{im}(z)) = (\text{im}_1(u_j) - \text{im}_1(u_i))\,(\text{im}_2(z_j) - \text{im}_2(z_i))$. As before, the standard error of this estimator is derived as $\text{SE}\left(\widehat{\Delta}_{\text{im}_1(u)\times\text{im}_2(z)}\right) = (S-1)^{-1/2}$ $\left\{\sum_{s=1}^{S}\left(\widehat{\Delta}^{s}_{\text{im}_1(u)\times\text{im}_2(z)} - \widehat{\Delta}_{\text{im}_1(u)\times\text{im}_2(z)}\right)^2\right\}^{1/2}$.

Theorem 4.2. If the metric M in $\widehat{\Delta}_{\text{im}_1(u)\times\text{im}_2(z)}$ satisfies the extended stable metric-input distribution assumption, then the estimator is Fisher consistent for its corresponding estimand.

Example 4.3. Consider the non-linear data generating mechanism between the outcome y and inputs $x_1 \in [0,1]$ and $x_2 \in \{0,1\}$ in Figure 4.1, with

$$y = \begin{cases} x_1^3 & \text{if } x_2 = 1, \\ \exp(-2.2\,x_1) & \text{if } x_2 = 0. \end{cases}$$

In linear models, interactions between inputs are typically formulated in terms of their product. Let $\text{im}_p(x_1, x_2) = x_1\,x_2$ and $\text{im}(x_1 \mid x_2 = a) = x_1\,\mathbb{I}\{x_2 = a\}$ for $a \in \{0,1\}$ be two interpretable mappers that correspond to the product and conditional effects, respectively, of these inputs. Table 4.2 contains the GEAR estimands for the inputs under the previously specified interpretable mappers and the generalized average two-way interaction predictive comparison $\Delta_{x_1\times x_2}$. These values were obtained by means of a grid approximation to the integration. We note that, although $\Delta_{x_1\times x_2}$ and $\Delta_{\text{im}_p(x_1,x_2)}$ yield equivalent results under a linear data generating mechanism, they will not be equivalent under non-linear mechanisms as in this case.

As $\Delta_{x_1\times x_2} > 0$, the average predicted changes in y as x_1 increases by a unit are larger when $x_2 = 1$. Note that this reference to a larger conditional effect incorporates the sign of the actual outcome, and not just its absolute magnitude. This fact is also evident in Figure 4.1, as larger values of x_1 are associated with larger values of y when $x_2 = 1$ but the opposite occurs when $x_2 = 0$. The GEAR estimand $\Delta_{\text{im}_p(x_1,x_2)}$ represents the average predicted increase in y when x_2 changes from 0 to 1. Note that when $x_1 \in [0, 0.63]$, the outcome y is larger for $x_2 = 0$ then $x_2 = 1$. This distinction

between the estimands $\Delta_{\text{im}_p(x_1,x_2)}$ and $\Delta_{x_1 \times x_2}$ is important to recognize when one wishes to interpret the relationships inferred by machine learning algorithms and models on non-linear relationships between an outcome and inputs.

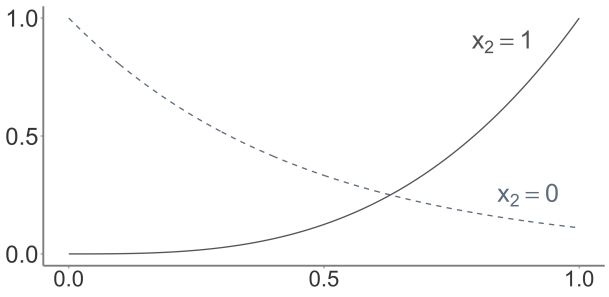


Figure 4.1.: Non-linear outcome in Ex. 4.3.

Table 4.2.: Estimands in Ex. 4.3.

Estimand	*Value*
$\Delta_{\text{im}_p(x_1,x_2)}$	-6.1×10^{-3}
$\Delta_{x_1 \times x_2}$	1.72
$\Delta_{\text{im}(x_1 \mid x_2=1)}$	0.90
$\Delta_{\text{im}(x_1 \mid x_2=0)}$	-1.21

4.2.4 Individual generalized predictive comparisons

In some cases one may be interested in understanding the relationships between the inputs and outcome for a specific observation. One example of such a situation is in image classification. In this case, it may be more interpretable to evaluate what drives the classification of a certain image than attempting to understand the model for the entire data set of images. This idea is further discussed in Chapter 4.3.2.

Definition 4.5. The individual generalized average predictive comparison (iGEAR) and individual generalized average magnitude predictive comparison (iGAME) estimands for observation i with inputs $x_i = (u_i, v_i) \in \mathbb{R}^K$, in which $u_i \in \mathbb{R}^d$ represents the input of interest, under $\text{im} : \mathbb{R}^d \to \mathbb{R}$ are, respectively,

$$\Delta^{(i)}_{\text{im}(u)} = \frac{\int_{\text{IM}^{(i)+}} \left\{ \mathbb{E}\left(y \mid u^{(2)}, v_i, \beta\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta\right) \right\} p\left(u^{(2)} \mid v_i\right) p(\beta) \mathrm{d}u^{(2)} \mathrm{d}\beta}{\int_{\text{IM}^{(i)+}} \left\{ \text{im}\left(u^{(2)}\right) - \text{im}\left(u_i\right) \right\} p\left(u^{(2)} \mid v_i\right) p(\beta) \mathrm{d}u^{(2)} \mathrm{d}\beta},$$

$$\Delta^{(i)}_{\text{mag}(\text{im}(u))} = \left\{ \frac{\int \left\{ \mathbb{E}\left(y \mid u^{(2)}, v_i, \beta\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta\right) \right\}^2 p\left(u^{(2)} \mid v_i\right) p(\beta) \mathrm{d}u^{(2)} \mathrm{d}\beta}{\int \left\{ \text{im}\left(u^{(2)}\right) - \text{im}\left(u_i\right) \right\}^2 p\left(u^{(2)} \mid v_i\right) p(\beta) \mathrm{d}u^{(2)} \mathrm{d}\beta} \right\}^{1/2},$$

and $\text{IM}^{(i)+} = \left\{u^{(2)} \in \mathbb{R}^d,\ \beta \in \mathbb{R}^L : \text{im}\left(u_i\right) < \text{im}\left(u^{(2)}\right)\right\}$.

Our individual predictive comparison estimand differs from current local interpretability methods in three ways. First, it neither depends on a perturbation distribution, nor does it focus on small regions of the domain around an observation i. Second, it does not rely on the application of a more interpretable model compared to the fitted machine learning algorithm/model to understand the effects of inputs on the outcome for individual i. Finally, as we shall proceed to demonstrate, our methodology quantifies the uncertainty of the inferences.

Our estimators for these new estimands are $\widehat{\Delta}^{(i)}_{\text{im}(u)} = S^{-1}\sum_{s=1}^{S}\widehat{\Delta}^{(i),s}_{\text{im}(u)}$, and $\widehat{\Delta}^{(i)}_{\text{mag}(\text{im}(u))} = \left\{S^{-1}\sum_{s=1}^{S}\widehat{\Delta^2}^{(i),s}_{\text{mag}(\text{im}(u))}\right\}^{1/2}$, where

$$\widehat{\Delta}^{(i),s}_{\text{im}(u)} = \frac{\sum_{j=1}^{n}\left\{\mathbb{E}\left(y \mid u_j, v_i, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta^{(s)}\right)\right\} w_{ij}\,\text{sign}(\text{im}(u_j) - \text{im}(u_i))}{\sum_{j=1}^{n}\left\{\text{im}(u_j) - \text{im}(u_i)\right\} w_{ij}\,\text{sign}(\text{im}(u_j) - \text{im}(u_i))},$$

$$\text{and } \widehat{\Delta^2}^{(i),s}_{\text{mag}(\text{im}(u))} = \frac{\sum_{j=1}^{n}\left\{\mathbb{E}\left(y \mid u_j, v_i, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta^{(s)}\right)\right\}^2 w_{ij}}{\sum_{j=1}^{n}\left\{\text{im}(u_j) - \text{im}(u_i)\right\}^2 w_{ij}}.$$

In contrast to the previous results on Fisher consistency, the assessment of Fisher consistency for the estimands in Definition 4.5 will be focused on the population of individuals with $v = v_i$.

Theorem 4.3. If the metric M in $\widehat{\Delta}^{(i)}_{\text{im}(u)}$, and $\widehat{\Delta}^{(i)}_{\text{mag}(\text{im}(u))}$ satisfies the extended stable metric-input distribution assumption, then these estimators are Fisher consistent for their corresponding estimands.

We derive the standard errors of these estimators as

$$\mathrm{SE}\left(\widehat{\Delta}^{(i)}_{\mathrm{im}(u)}\right) = (S-1)^{-1/2}\left\{\sum_{s=1}^{S}\left(\widehat{\Delta}^{(i),s}_{\mathrm{im}(u)} - \widehat{\Delta}^{(i)}_{\mathrm{im}(u)}\right)^2\right\}^{1/2},$$

$$\mathrm{SE}\left(\widehat{\Delta}^{(i)}_{\mathrm{mag}(\mathrm{im}(u))}\right) = \left(2\widehat{\Delta}^{(i)}_{\mathrm{mag}(\mathrm{im}(u))}\right)^{-1}\left\{(S-1)^{-1}\sum_{s=1}^{S}\left\{\widehat{\Delta^2}^{(i),s}_{\mathrm{mag}(\mathrm{im}(u))} - \widehat{\Delta^2}^{(i)}_{\mathrm{mag}(\mathrm{im}(u))}\right\}^2\right\}^{1/2}.$$

Example 4.4. Consider the text classifier model for sentiment analysis in reviews discussed in Example 4.1. Let the input of interest u_3 be the frequency with which *"very"* is used, and the interpretable mapper be the indicator $\mathrm{im}(u_3) = \mathbb{I}\{u_3 > 0\}$. In practice, the word *"very"* can be associated to either a strong positive or a strong negative feeling. Due to its two possible connotations, understanding the association of *"very"* with individual reviews may be more interpretable and provide more interesting insights to the patterns inferred by the text classifier than attempting to analyze its association over all reviews.

4.3 Illustrative Studies

4.3.1 Simulation study on Bayesian neural networks

The first illustration of our methodology involves a simulation study based on examples from Linkletter et al. (2006), Williams et al. (2006), and Surjanovic and Bingham (2013). In this simulation study we consider two processes that involve the same five inputs, and each generate 500 observations. The outcomes y_1 and y_2 for the two processes are generated according to

$$y_1 = f_1(x_1) + 0.3f_2(x_2) + \epsilon_1,$$
$$y_2 = f_3(x_3, x_4) + 0x_5 + \epsilon_2,$$

respectively, where $\epsilon_1 \sim \mathrm{N}(0, 0.05^2)$, $\epsilon_2 \sim \mathrm{N}(0, 0.01^2)$, $x_1, x_2 \sim \mathrm{Unif}(0,1)$, $x_3, x_4, x_5 \sim \mathrm{Unif}(-\pi, \pi)$, $f_1(x) = \sin(x)$, $f_2(x) = \sin(5\,x)$, and $f_3(x_3, x_4) = f_4(x_3)f_5(x_4)$ with

$f_4(x_3) = (x_3 + 1)$ and $f_5(x_4) = \cos(\pi x_4)$. All of the inputs and error terms are mutually independent. We model the observed outcomes for the two processes using Bayesian neural networks with one hidden layer, and three and five hidden neurons respectively. Inputs x_1 and x_2 are used to model y_1, and inputs x_3, x_4, x_5 are used to model y_2. These models provide a good fit to the observed data. The formal model specifications are

$$y_{1,i} = \beta_0 + \sum_{m=1}^{3} \beta_m \tanh\left(\alpha_{0,m} + \sum_{k=1}^{2} \alpha_{k,m} x_{i,k}\right) + \epsilon_{1,i},$$
$$y_{2,i} = \tilde{\beta}_0 + \sum_{m=1}^{5} \tilde{\beta}_m \tanh\left(\tilde{\alpha}_{0,m} + \sum_{k=3}^{5} \tilde{\alpha}_{k,m} x_{i,k}\right) + \epsilon_{2,i},$$

where $\epsilon_{k,i} \sim N(0, \sigma_k^2)$ for $k \in \{1, 2\}$, and the prior densities are

$$p(\boldsymbol{\alpha}, \boldsymbol{\beta}, \log \sigma_1) \propto \exp\left(\frac{-\boldsymbol{\alpha}^\mathsf{T}\boldsymbol{\alpha}}{20}\right) \exp\left(\frac{-\boldsymbol{\beta}^\mathsf{T}\boldsymbol{\beta}}{20}\right),$$
$$p\left(\tilde{\boldsymbol{\alpha}}, \tilde{\boldsymbol{\beta}}, \log \sigma_2\right) \propto \exp\left(\frac{-\tilde{\boldsymbol{\alpha}}^\mathsf{T}\tilde{\boldsymbol{\alpha}}}{20}\right) \exp\left(\frac{-\tilde{\boldsymbol{\beta}}^\mathsf{T}\tilde{\boldsymbol{\beta}}}{20}\right).$$

For comparison purposes, the data were also modeled using the linear models

$$y_{1,i} = \gamma_{1,0} + \gamma_{1,1} f_1(x_{1,i}) + \gamma_{1,2} f_1(x_{2,i}) + \epsilon_{1,i},$$
$$y_{2,i} = \gamma_{2,0} + \gamma_{2,1} f_3(x_{3,i}, x_{4,i}) + \gamma_{2,2} x_{5,i} + \epsilon_{2,i}.$$

For each individual input, and for the pair of inputs (x_3, x_4), we estimated the GEAR for the interpretable mappers that correspond to the functions underlying the data generating mechanisms for the two processes. Two examples are $\text{im}(x_1) = f_1(x_1)$ and $\text{im}(x_3, x_4) = f_1(x_1)$. In addition, we also estimated the relevant generalized average two-way interaction predictive comparisons. Tables 4.3(a) and 4.3(b) summarize the results obtained over 1000 simulations. Our GEAR estimators provide comparable results to the linear model estimators. For the first process, they recover the true

effects of $f_1(x_1)$ and $f_2(x_2)$ on y_1. Simimlarly, the true effects of $f_3(x_3, x_4)$ and x_5 on y_2 are recovered for the second process.

Table 4.3.: Summaries of generalized average predictive comparisons over 1000 simulated datasets. (a) Mean estimates with corresponding standard errors and coverage calculated based on 95% confidence intervals for $\widehat{\beta}$ and 95% central posterior intervals for GEAR. (b) Mean estimates with corresponding standard errors for the two-way generalized average predictive comparisons.

(a)

Input under IM	*Outcome*	*Estimates (SE)*		*Coverage (%)*	
		$\widehat{\gamma}$	$\widehat{\Delta}_{\text{im}(u)}$	$\widehat{\gamma}$	$\widehat{\Delta}_{\text{im}(u)}$
$f_1(x_1)$	y_1	1.000 (0.009)	1.000 (0.011)	95.9	95.1
$f_2(x_2)$		0.300 (0.003)	0.299 (0.004)	95.5	94.4
$f_3(x_3, x_4)$	y_2	1.000 (0.005)	0.997 (0.007)	95.1	94.5
x_5		-0.0001 (0.008)	-0.0007 (0.009)	94.3	93.9

(b)

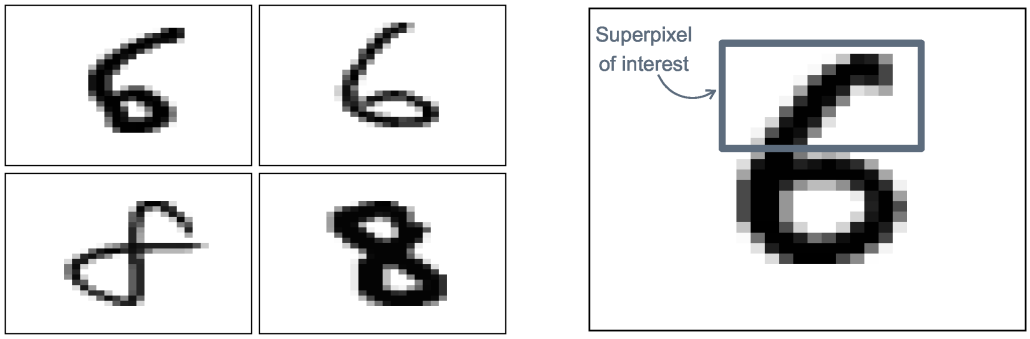
Two-way interaction	*Outcome*	$\widehat{\Delta}_{\text{im}_1(u)\times\text{im}_2(z)}$ *(SE)*
$x_1 \times x_2$	y_1	-0.0035 (0.032)
$f_1(x_1) \times f_2(x_2)$		0.0012 (0.014)
$x_3 \times x_4$	y_2	-0.0463 (0.015)
$x_3 \times x_5$		-0.0001 (0.007)
$x_4 \times x_5$		0.030 (0.004)
$f_4(x_3) \times f_5(x_4)$		0.997 (0.012)
$f_3(x_3, x_4) \times x_5$		0.001(0.0117)

4.3.2 Understanding a BART classifier for handwritten digits

Image processing and classification constitute significant applications of machine learning models (Egmont-Petersen et al., 2002; Rawat and Wang, 2017). We illustrate the scope of our methodology for these applications, and the insights it can yield into complex models for image classification, through a study of the BART classifier for the MNIST handwritten digit image data. The training data is composed of $10,806$

handwritten "6" and "8"s, and the test data contains an additional $1,932$ images of these digits. Examples of the data are in Figure 4.2(a). The inputs in this context are the intensities of the individual pixels in a 28×28 digit image, with the values for each being between 0 (white) and 1 (black). The model outcome is the probability of an image being classified as a "6". The fitted BART model exhibited high predictive performance, with accuracy values of 99.3% and 98.71% for the training and test data, respectively.

An effective approach to understand an image classifier is to consider a particular image (Montavon et al., 2018). Accordingly, our individual predictive comparisons can be useful for interpreting the previously fitted BART image classifier. The specific image that we consider for this purpose is presented in Figure 4.2(b). We performed two sets of analyses for this image. The interest of the first set of analyses is on understanding the effects of the group of pixels u (referred to as the "super-pixel") delimited by the grey rectangle in Figure 4.2(b), and the interest of the second is on all of the pixels u_{upper} located in the upper half of the image. We note that u corresponds to the image's most discriminating feature between classes "6" and "8". For these two analyses, we let v and v_{upper} denote the images that do not contain u and u_{upper}, respectively.



(a) (b)

Figure 4.2.: (a) Examples of MNIST handwritten digit image data used to train BART model. (b) Image used to assess BART model. Pixels inside grey rectangle represent the input vector u for the first analysis.

For handwritten "8"s with similar v, we expect a higher quantity of colored pixels throughout the super-pixel, while for "6"s we expect high intensities only in the super-pixel's diagonal. To assess whether such patterns are effectively learned by our model, we consider two interpretable mappers: the norm of the entire super-pixel, and the norm of only the diagonal entries in the super-pixel. Also, as we are focused on this particular image, we examine these effects of the super-pixel while keeping the rest of the image (which is contained in v) fixed. Table 4.4 summarizes the results of these two generalized average predictive comparisons for Figure 4.2(b). We observe that as the norm of intensity in the entire super-pixel increases, the probability of classifying it as a "6" decreases. Furthermore, the probability of classifying it as "6" increases as the norm of the intensity in the diagonal entries increases.

We now consider the second analysis on u_{upper}, which is treated as the super-pixel of interest. As symmetry can distinguish the two numbers, we adopt as our interpretable mapper a measure of symmetry between the upper and lower halves for this specific image. Symmetry here is measured as a percentage and defined as (Marola, 1989)

$$\text{symmetry}(u_{\text{upper}} \mid v_{\text{upper}}) = 100 \left(\frac{\sum_{k=1}^{392} \left(u_{\text{upper}}^{(k)}\right) \left(v_{\text{upper}}^{(k)}\right)}{\sum_{k=1}^{392} \left(u_{\text{upper}}^{(k)}\right)^2} \right),$$

where $u_{\text{upper}}^{(k)}$ denotes the pixel's intensity, and $u_{\text{upper}}^{(k)}$ and $v_{\text{upper}}^{(k)}$ are associated with pixels in symmetric positions in the digit's upper and lower halves, respectively. The results for these mappers are also summarized in Table 4.4. We observe that as symmetry between the upper and lower halves increases, the overall probability of classifying the image as a "6" decreases.

In a similar manner, we considered the effects of symmetry between upper and lower halves over all digits. We obtained the GEAR estimate of -0.007 with standard deviation 0.0004, indicating that as symmetry between the upper and lower halves

increases, the probability of classifying a handwritten digit as a "6" decreases. Hence, our methodology enables us to understand and confirm in a straightforward manner that the BART model is learning patterns similar to those used by humans when distinguishing between the two digits of "6" and "8".

Table 4.4.: Interpretations of the handwritten digit classifier. Understanding the effects of a super-pixel of interest as its interpretable mapper increases on the probability of classifying the individual digit in Figure 4.2(b) as a "6".

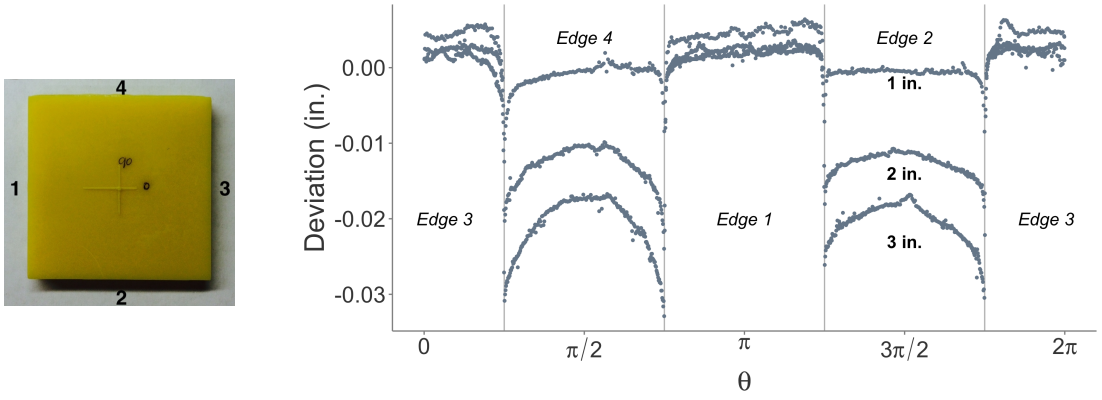
Input	*Interpretable mapper*	*iGEAR Estimates (SE)*
Super-pixel u	$\text{im}(u) = \|\|u\|\|_2$	-0.124 (0.009)
Super-pixel u	$\text{im}(u) = \|\|\text{diag}(u)\|\|_2$	0.152 (0.017)
Super-pixel u_{upper}	$\text{im}(u) = \text{symmetry}(u)$	-0.0194 (0.001)

4.4 Screening and interpreting functional inputs in Bayesian NN models for shape deviations

Our final case study is on the interpretations of functional forms of inputs in the Bayesian NNs devised in Chapter 2. As discussed in the previous chapters, automatically modeling geometric shape deviations in AM systems across shapes and process, and obtaining insights on the inferred relationships between the inputs and shape deviations, are crucial to advance the potential of AM. The first issue was addressed in Chapter 2 via our Bayesian NN methodology, and the later issue was only partially addressed in Chapter 3. Specifically, the formulation of our previous predictive comparison methodology in that chapter does not directly yield insights on the inferred relationships between functional forms of inputs and shape deviations. We now proceed to demonstrate how the estimands in our generalized predictive comparison methodology address this limitation of the previous method. The specific case study under consideration involves the interpretation of functional forms of inputs for additively manufactured cubes with small heights, one of which is illustrated in Figure 4.3(a). Note that, although the Bayesian ELM model for squares was fitted

using data from both cylinders and cubes, here we consider interpretations specific for cubes based on their Bayesian ELM model.

As previously described, each point i on the boundary of a cube is identified by an angle θ_i, a nominal radius function $r_i(\theta_i \mid r_0)$ (where r_0 represents the known nominal radius of its circumcircle), and an indicator for the edge $\text{edge}(\theta_i) \in \{1, \cdots, 4\}$ of the cube on which θ_i resides. Edges 1 and 3 refer to the vertical edges of the cubes, and edges 2 and 4 refer to the horizontal edges (Figure 4.3(a)). Also, the outcome y_i of in-plane shape deviation is defined as the difference between the observed radius and the nominal radius function of θ_i. Figure 4.3(b) contains the deviation profiles for three additive manufactured cubes whose circumcircles are of nominal radii $1''$, $2''$, and $3''$. Most of the deviations are negative, which indicate that the manufactured products exhibit shrinkage in comparison to the computer-aided models. Further details can be found in Chapters 2 and 2.3.4.



(a) (b)

Figure 4.3.: (a) A sample additively manufactured cube with small height. (b) Observed in-plane shape deviations for different cubes manufactured under the same stereolithography process.

We observe from Figure 4.3(b) that edge orientation has an effect on the observed deviations. Also, different edges, even within the same orientation, exhibit distinct deviation profiles, which indicates the importance of considering the different edges in the deviation model. In this context, understanding the effects of orientation on de-

viation instead of the effects of edge itself may be more interpretable and more useful in order to gain further insights into the manufacturing process. As such, although the Bayesian NN models in Chapter 2 use edge as an input, it is of more interest to interpret the inferred effects of edge orientation on shape deviation. Our generalized predictive comparison methodology requires no further efforts to perform such interpretations, in terms of fitting a new model that explicitly includes edge orientation as an input as would be required for the previous predictive comparisons methodology in Chapter 3. Moreover, the generalized predictive comparison methodology enables one to "mine" the model in terms of exploring several relationships inferred by the model through different functional forms of the inputs.

It is important to note that the categorical input edge must be taken into account to properly compute the weights for the predictive comparison methodology. For a quantitative input u, we partition v as $v = (\text{edge}, z)$, where z is a remaining quantitative input. We then define the weights as $w_{ij} = \mathbb{I}\left\{\text{edge}_i = \text{edge}_j\right\} \left\{1 + M\left(z_i, z_j\right)\right\}^{-1}$. For the nominal radius input $r(\cdot)$, we considered the identity function as the interpretable mapper, and for the edge input the indicator function for whether it has a horizontal orientation was used as the interpretable mapper, i.e., $\text{im}_E(\text{edge}) = \mathbb{I}\{\text{edge} \in \{2, 4\}\}$. Specifying an interpretable mapper for the θ input required extra care. This is because in the third edge, the angles are defined in the domain $(0, \pi/4) \cup (7\pi/4, 2\pi)$ (Figure 4.3(b)). To avoid gaps between the angles for this edge, we utilized as our interpretable mapper

$$\text{im}(\theta_i) = \begin{cases} \theta_i - \mathbb{I}(\theta_i > 0)2\pi & \text{if edge}_i = 3, \\ \theta_i & \text{if edge}_i \neq 3. \end{cases}$$

Figure 4.4(b) summarizes the results from our generalized predictive comparison methodology for functional forms of the inputs and their interactions. We observe that as we change from a vertical to a horizontal orientation, deviation decreases on average by $0.01''$. Alternatively, horizontal edges are shrinking $0.01''$ more than

vertical edges on average. The interaction between horizontal edge and nominal radius indicates a larger effect of nominal radius in edges 2 and 4. Results for the generalized predictive comparisons of angle and nominal radius function under different edges are in Figure 4.4(c). These results indicate some interactions between nominal radius function and angle for all edges excluding the first. In addition, they suggest that an increase in radius yields, on average, higher deviations in absolute value in horizontal edges compared to vertical edges.

Generalized conditional predictive comparisons for radius under different angles are summarized in Figure 4.4(d). We note that when the deviation data for these products were collected, different angles were measured across the products in the sense that an angle θ_i observed in one product was not necessarily measured for the other products. As such, we separated the domain of the angles into 50 equally spaces subsets $\boldsymbol{\vartheta} = \{\vartheta_1, \cdots, \vartheta_{50}\}$, and calculated $\widehat{\Delta}_{\text{im}(r_0|\theta)}$, where $\text{im}(r_0 \mid \theta \in \vartheta_t) = r_0\, \mathbb{I}\{\theta \in \vartheta_t\}$ for $t \in \{1, \cdots, 50\}$. A thinner partition on the angle's domain could be considered, however it would introduce artifacts of lesser smooth curves, which are not of interest here.

Besides exploring the relationships of functional forms of inputs with shape deviations for previously manufactured cubes, the insights that our methodology yields into the AM process can enable us to learn how to improve the dimensional accuracy of future shapes to be manufactured by the process. For example, suppose we wish to manufacture a rectangular prism that has never been manufactured. The results that we obtained from our methodology in this study could then help guide us with respect to the orientation that should be adopted for the first manufacture of the rectangular prisms.

4.5 Discussions

The current trade-off between model complexity and interpretability hinders the effective use of machine learning algorithms and models in practice. The generalized

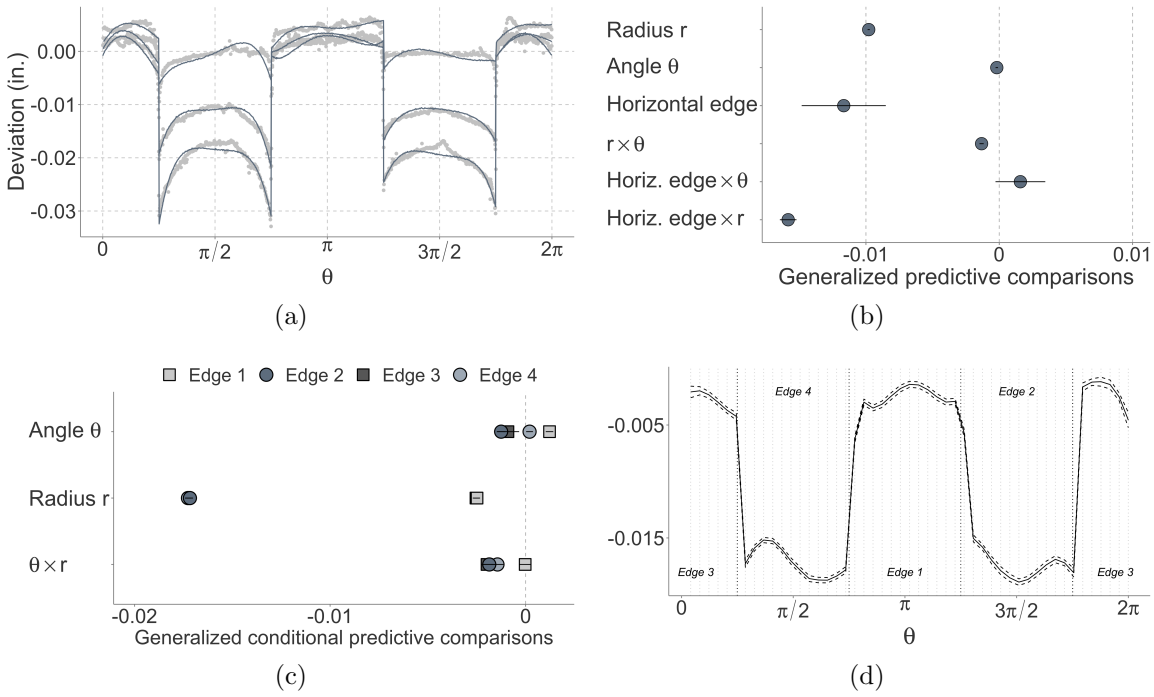


Figure 4.4.: Interpretations for Bayesian NN in-plane deviation model of manufactured cubes. (a) Mean prediction based on model (solid lines), and observed deviations (grey dots). (b) Generalized predictive comparisons for the nominal radius function r, angle θ, and edge inputs, and all of their two-way interactions, under our specified interpretable mappers. Dots represent mean values and bars indicate one standard deviation. (c) Generalized conditional predictive comparisons for nominal radius function r, angle θ, and their interaction, under our specified interpretable mappers, for each edge on the cube. Vertical edges are represented by squares and horizontal edges by circles. Dots represent mean values and bars indicate one standard deviation. (d) Generalized conditional predictive comparison for nominal radius function r (black lines) across different ranges of angle θ (grey vertical dotted lines). The solid black line represents the mean values, and the dashed black lines indicate one standard deviation.

predictive comparison methodology in this chapter can address this issue by enabling one to assess and interpret the relationships between multiple inputs simultaneously, and between functional forms of the inputs with the outcome. We demonstrated the practical relevance of our method with simulations and real-life case studies that utilize BART and NN models. Of particular significance is the illustration of our method for shape deviation models in AM systems. This study illuminated complicated re-

lationships between different functional forms of the AM inputs and deviations that arise due to the complex physical phenomena and processes involved with AM.

5. CONCLUDING REMARKS

Additive manufacturing systems have great potential to fundamentally transform the manner in which people interact with manufacturing. By reducing fabrication complexity and liberating product design processes for online communities of users, AM systems can inaugurate an exciting new era of cybermanufacturing with positive effects that far exceed those of current manufacturing systems. However, deviation modeling and control of the vast variety of shapes manufactured under distinct processes, while satisfying significant time and resource constraints, is a significant issue that must be addressed to realize the potential of AM systems. Furthermore, although machine learning can be used to produce highly accurate predictive models for AM systems, its fundamental limitation is the incomplete set of tools available to scientists and engineers that can yield interpretations of them to better understand AM systems. Obtaining insightful interpretations of black box machine learning algorithms and models is of critical importance for their effective application in real-world AM systems.

We effectively addressed the first challenge in AM systems by developing an automated Bayesian ELM model building method. Our method sequentially leverages prior deviation models and data via four simple steps to automate model specifications of new shapes and processes. The use of Bayesian statistics in our method is important because it provides a formal and straightforward inferential framework for this sequential leveraging of prior information. The power and scope of our method were illustrated by several case studies on in-plane and out-of-plane deviations of different shapes under distinct AM processes. As was demonstrated in these case studies, our method produced effective deviation models in a simple and efficient manner without requiring the use of specialized domain knowledge on specific AM processes. The corresponding significant implication is that our method can abstract from particular

shapes and processes to underpin cross-cutting deviation model building in AM systems more broadly. In this respect, our method can enable smarter deviation control for AM systems, and thereby help to advance their future growth and adoption for a large community of AM users. Novel statistical innovations in our method include principled and connectable NN structures that facilitate model transfer, and a method to tune the random selection of the hidden neuron coefficients for improved predictive performance. These two contributions can be applied beyond our immediate AM context to reduce the amount of ad-hoc tuning and model fitting typically performed for NNs.

Our new predictive comparison and generalized predictive comparison methodologies help to address the second challenge in AM systems. Our new estimands in these methodologies effectively enable one to "mine" a model, in the sense of (a) interpreting the inferred associations between inputs and/or functional forms of inputs with the outcome, (b) uncovering the relevance of multiple inputs, and (c) interpreting the inferred conditional and two-way associations of the inputs with the outcome. These methodologies have a broader scope of application beyond the Bayesian neural network, BART, and SVM algorithms that were considered in our case studies. Indeed, it is applicable to any machine learning algorithm or model that yields predictions of outcomes and enables the construction of a distribution capturing inferential uncertainty for unknown parameters.

New avenues of future research are illuminated by our methodologies. One exciting research problem is automated deviation modeling for a new type of shape without the use of any manufactured products of that shape, but only using products from similar classes of shapes. This is also known as prescriptive deviation modeling Luan and Huang (2017), and is of fundamental importance to the practical operation of AM systems. An additional problem is the creation of diagnostic methods to identify possible violations of the model assumptions. To illustrate, the transfer of a geometric shape deviation model across different shapes currently relies on an additivity assumption, which if not met could introduce deficiencies in the model's

predictions. A diagnostic methodology combined with our method's inherent feedback loop mechanism, and the extension of our automated method to prescriptive modeling would further help to realize the potential of AM systems. Another direction that is related to the automated deviation modeling method is improving different aspects of the ELMs involved in the method. Although ELMs demonstrate great promise by reducing an entire class of NNs to a simple linear regression, they can be potentially improved by developing an automated approach to select the number of hidden neurons in a Bayesian framework, methods to tune the random assignment for the inner parameters to improve out-of-sample prediction. The latter was discussed in Chapter 2.2 and Appendix A.1. For the predictive comparison methodologies, one new research problem is the use of inferences on generalized predictive comparison estimands to identify deficiencies or limitations that exist with a particular machine learning algorithm or model fitted to a dataset. This advance could then enable one to understand how to modify the algorithm or model so as to eliminate the deficiencies and obtain a better fit. These new research problems and directions can play critical roles for the operation of new advanced manufacturing systems with complicated inputs. They can also yield new developments and methods that will be of broader scientific relevance for the effective application of machine learning methods in the physical and engineering sciences.

REFERENCES

REFERENCES

Berger, J. O. (1985). *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag.

Bourell, D., Leu, M., and Rosen, D. (2009). Roadmap for additive manufacturing: Identifying the future of freeform processing. Technical report, National Science Foundation and the Office of Naval Research.

Buckholtz, B., Ragai, I., and Wang, L. (2015). Cloud manufacturing: Current trends and future implementations. *Journal of Manufacturing Science and Engineering*, 137(4):040902.

Campbell, T., Williams, C., Ivanova, O., and Garrett, B. (2011). *Could 3D Printing Change the World? Technologies, Potential, and Implications of Additive Manufacturing*. Washington, DC: Atlantic Council.

Casella, G. and Berger, R. L. (2002). *Statistical inference*, volume 2. Duxbury Pacific Grove, CA.

Cheng, C.-S. (2016). *Theory of factorial design: Single-and multi-stratum experiments*. Chapman and Hall/CRC.

Cherkassky, V. and Ma, Y. (2004). Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks*, 17(1):113–126.

Chipman, H. A., George, E. I., McCulloch, R. E., et al. (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298.

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553.

Cortez, P. and Silva, A. M. G. (2008). Using data mining to predict secondary school student performance. In *Proceedings of Fifth FUture BUsiness TEChnology Conference (FUBUTEC 2008)*, pages 5–12. EUROSIS.

Cox, D. R. and Hinkley, D. V. (1974). *Theoretical Statistics*. Chapman and Hall/CRC, 1 edition.

Dasgupta, T., Pillai, N. S., and Rubin, D. B. (2015). Causal inference from 2k factorial designs by using potential outcomes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(4):727–753.

Deng, L., Yu, D., et al. (2014). Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387.

Doshi-Velez, F. and Kim, B. (2017). Towards A Rigorous Science of Interpretable Machine Learning. *arXiv preprint arXiv:1702.08608*.

Duane, S., Kennedy, A., Pendleton, B. J., and Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *Annals of Statistics*, 7(1):1–26.

Efron, B. and Tibshirani, R. J. (1994). *An Introduction to the Bootstrap*. Chapman and Hall/CRC, 1 edition.

Egmont-Petersen, M., de Ridder, D., and Handels, H. (2002). Image processing with neural networksa review. *Pattern recognition*, 35(10):2279–2301.

Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 222:309–368.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, pages 1189–1232.

Gelman, A. and Pardoe, I. (2007). Average predictive comparisons for models with nonlinearity, interactions, and variance components. *Sociological Methodology*, 37(1):23–51.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741.

Gibson, I., Rosen, D., and Stucker, B. (2009). *Additive Manufacturing Technologies: Rapid Prototyping to Direct Digital Manufacturing*. Springer Verlag.

Goldstein, A., Kapelner, A., Bleich, J., and Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

GTAI (2017). *Industrie 4.0: Smart manufacturing for the future*. Germany Trade & Invest.

Hinkelmann, K. and Kempthorne, O. (2007). *Design and analysis of experiments, volume 1: introduction to experimental design*, volume 592. John Wiley & Sons.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366.

Huang, G., Huang, G.-B., Song, S., and You, K. (2015a). Trends in extreme learning machines: A review. *Neural Networks*, 61:32–48.

Huang, G.-B. and Chen, L. (2007). Convex incremental extreme learning machine. *Neurocomputing*, 70(16):3056–3062.

Huang, G.-B. and Chen, L. (2008). Enhanced random search based incremental extreme learning machine. *Neurocomputing*, 71(16):3460–3468.

Huang, G.-B., Chen, L., and Siew, C. K. (2006). Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Trans. Neural Networks*, 17(4):879–892.

Huang, G.-B., Wang, D., and Lan, Y. (2011). Extreme learning machines: a survey. *International Journal of Machine Learning and Cybernetics*, 2(2):107–122.

Huang, G.-B., Zhu, Q.-Y., and Siew, C.-K. (2004). Extreme learning machine: a new learning scheme of feedforward neural networks. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, volume 2, pages 985–990. IEEE.

Huang, Q. (2016). An analytical foundation for optimal compensation of three-dimensional shape deviations in additive manufacturing. *Journal of Manufacturing Science and Engineering*, 138(6):061010.

Huang, Q., Nouri, H., Xu, K., Chen, Y., Sosina, S., and Dasgupta, T. (2014). Statistical predictive modeling and compensation of geometric deviations of 3D printed products. *ASME Transactions, Journal of Manufacturing Science and Engineering*, 136(6):061008 (10 pages).

Huang, Q., Zhang, J., Sabbaghi, A., and Dasgupta, T. (2015b). Optimal offline compensation of shape shrinkage for 3D printing processes. *IIE Transactions on Quality and Reliability*, 47(5):431–441.

Huang, S., Liu, P., Mokasdar, A., and Hou, L. (2013). Additive manufacturing and its societal impact: a literature review. *The International Journal of Advanced Manufacturing Technology*, 67(5):1191–1203.

Jin, Y., Qin, S., and Huang, Q. (2016). Offline predictive control of out-of-plane geometric errors for additive manufacturing. *ASME Transactions on Manufacturing Science and Engineering*, 138(12):121005 (7 pages).

Libbrecht, M. W. and Noble, W. S. (2015). Machine learning applications in genetics and genomics. *Nature Reviews Genetics*, 16(6):321.

Linkletter, C., Bingham, D., Hengartner, N., Higdon, D., and Ye, K. Q. (2006). Variable selection for gaussian process models in computer experiments. *Technometrics*, 48(4):478–490.

Lipovetsky, S. and Conklin, M. (2001). Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, 17(4):319–330.

Lipton, Z. C. (2016). The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*.

Luan, H. and Huang, Q. (2017). Prescriptive modeling and compensation of in-plane geometric deviations for 3D printed freeform products. *IEEE Transactions on Automation Science and Engineering*, 14(1):73–82.

Lundberg, S. and Lee, S.-I. (2016). An unexpected unity among methods for interpreting model predictions. *arXiv preprint arXiv:1611.07478*.

Mahalanobis, P. C. (1927). Analysis of race mixture in Bengal. *Journal of the Asiatic Society of Bengal*, 23:301–333.

Mahalanobis, P. C. (1936). On the generalized distance in statistics. National Institute of Science of India.

Marola, G. (1989). On the detection of the axes of symmetry of symmetric and almost symmetric planar images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(1):104–108.

McCulloch, R., Sparapani, R., Gramacy, R., Spanbauer, C., and Pratola, M. (2018). *BART: Bayesian Additive Regression Trees*. R package version 1.8.

McDonnell, M. D., Tissera, M. D., Vladusich, T., van Schaik, A., and Tapson, J. (2015). Fast, simple and accurate handwritten digit classification by training shallow neural network classifiers with the "extreme learning machine" algorithm. *PLOS ONE*, 10(8):e0134254.

Meng, X.-L. (1994). Posterior predictive p-values. *Annals of Statistics*, 22(3):1142–1160.

Meng, X.-L. (2010). Machine learning with human intelligence: Principled corner cutting (pc 2). In *Plenary Invited Talk, Annual Conference on Neural Information Processing Systems (NIPS)*, volume 163.

Montavon, G., Samek, W., and Müller, K.-R. (2018). Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15.

Oakley, J. E. and O'Hagan, A. (2004). Probabilistic sensitivity analysis of complex models: a Bayesian approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(3):751–769.

Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Prieto, A., Prieto, B., Ortigosa, E. M., Ros, E., Pelayo, F., Ortega, J., and Rojas, I. (2016). Neural networks: An overview of early research, current frameworks and new challenges. *Neurocomputing*, 214:242–268.

Rawat, W. and Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). Why should I trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2018). Anchors: High-precision model-agnostic explanations.

Rubin, D. (1984). Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics*, 12(4):1151–1172.

Sabbaghi, A. and Huang, Q. (2016). Predictive model building across different process conditions and shapes in 3D printing. Dallas, Texas. 2016 IEEE International Conference on Automation Science and Engineering (CASE 2016).

Sabbaghi, A. and Huang, Q. (2018). Model transfer across additive manufacturing processes via mean effect equivalence of lurking variables. *Annals of Applied Statistics*, 12(4):2409–2429.

Sabbaghi, A., Huang, Q., and Dasgupta, T. (2018). Bayesian model building from small samples of disparate data for capturing in-plane deviation in additive manufacturing. *Technometrics*, 60(4):532–544.

Schmutzler, C., Boeker, C., and Zaeh, M. F. (2016a). Investigation of deviations caused by powder compaction during 3D printing. *Procedia CIRP*, 57:698–703.

Schmutzler, C., Zimmermann, A., and Zaeh, M. F. (2016b). Compensating warpage of 3D printed parts using free-form deformation. *Procedia CIRP*, 41:1017–1022.

Soria-Olivas, E., Gomez-Sanchis, J., Martin, J. D., Vila-Frances, J., Martinez, M., Magdalena, J. R., and Serrano, A. J. (2011). BELM: Bayesian extreme learning machine. *IEEE Transactions on Neural Networks*, 22(3):505–509.

Surjanovic, S. and Bingham, D. (2013). Virtual library of simulation experiments: Test functions and datasets. Retrieved August 10, 2018, from `http://www.sfu.ca/~ssurjano`.

Tao, X., Zhou, X., He, Y.-L., and Ashfaq, R. A. R. (2016). Impact of variances of random weights and biases on extreme learning machine. *JSW*, 11(5):440–454.

Tong, K., Joshi, S., and Lehtihet, E. (2008). Error compensation for fused deposition modeling (fdm) machine by correcting slice files. *Rapid Prototyping Journal*, 14(1):4–14.

Tong, K., Lehtihet, E., and Joshi, S. (2003). Parametric error modeling and software error compensation for rapid prototyping. *Rapid Prototyping Journal*, 9(5):301–313.

Štrumbelj, E. and Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, 41(3):647–665.

Vapnik, V. (1998). *Statistical Learning Theory*. Wiley, New York, 1 edition.

Wang, H., Katz, R., and Huang, Q. (2005). Multi-operational machining processes modeling for sequential root cause identification and measurement reduction. *Journal of Manufacturing Science and Engineering*, 127(3):512–521.

Wang, W., Cheah, C., Fuh, J., and Lu, L. (1996). Influence of process parameters on stereolithography part shrinkage. *Materials & Design*, 17(4):205–213.

Williams, B., Higdon, D., Gattiker, J., Moore, L., McKay, M., Keller-McNulty, S., et al. (2006). Combining experimental data and computer simulations, with an application to flyer plate experiments. *Bayesian Analysis*, 1(4):765–792.

Wu, C. F. J. and Hamada, M. S. (2009). *Experiments: Planning, Analysis, and Optimization*. Wiley Series in Probability and Statistics. Wiley, 2 edition.

Wu, D., Rosen, D. W., Wang, L., and Schaefer, D. (2015). Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation. *Computer-Aided Design*, 59:1–14.

Zhou, C. and Chen, Y. (2012). Additive manufacturing based on optimized mask video projection for improved accuracy and resolution. *Journal of Manufacturing Processes*, 14(2):107–118.

APPENDICES

A. SUPPLEMENTARY MATERIAL FOR CHAPTER 2

A.1 Preventing saturation of hidden neurons in Bayesian ELM deviation models

Saturation of a hidden neuron refers to the activation function returning a relatively constant value for a range of inputs. An illustration for the hyperbolic tangent activation function is in Fig. A.1. To illustrate saturation in the context of deviation modeling via ELMs, consider modeling in-plane deviations of uncompensated cylinders (shape class 1) manufactured under a fixed process p as in the case studies. Let the inputs for each point i be $\mathbf{z}\left(\theta_i, r_{i,1}^{\text{nom}}\right) = \left(\theta_i, r_{i,1}^{\text{nom}}\right)^{\mathsf{T}}$, and set $h_{i,m} = g\left(\alpha_{m,0}^{(1,p)} + \alpha_{m,1}^{(1,p)}\theta_i + \alpha_{m,2}^{(1,p)} r_{i,1}^{\text{nom}}\right)$ as the neuron in entry (i,m) of $\mathbf{H}_{1,p}$ for random $\alpha_{m,0}^{(1,p)}, \alpha_{m,1}^{(1,p)}, \alpha_{m,2}^{(1,p)}$. Suppose $\alpha_{m,1}^{(1,p)} = 0.9, \alpha_{m,0}^{(1,p)}, \alpha_{m,2}^{(1,p)} \geq 0$, and $\theta_i = 2$. Then $h_{i,m} = g\left(\alpha_{m,0}^{(1,p)} + 1.8 + \alpha_{m,2}^{(1,p)} r_{i,1}^{\text{nom}}\right) \geq 0.94$ as $r_{i,1}^{\text{nom}} \geq 0.5$. Thus $h_{i,m} \approx 1$, and so is effectively an "activated neuron" irrespective of $r_{i,1}^{\text{nom}}$. The broader, practical lesson to be drawn is that if many neurons are saturated, then the relationships between deviation and inputs will not be effectively learned.

To illustrate how our new random assignment mechanism addresses the saturation issue in a principled manner, consider again the previous setting. Suppose $(-2.5, 2.5)$ is taken as the non-saturation region for the activation function. In our mechanism,

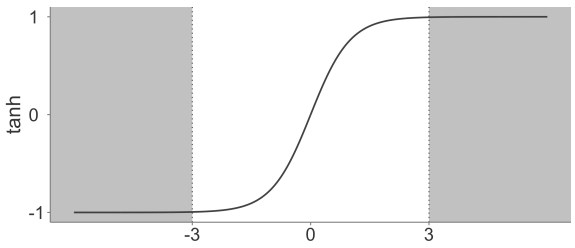


Figure A.1.: Saturation regions (shaded) for the hyperbolic tangent.

we choose a_1, a_2 such that $0 \leq |a_1\theta_i + a_2 r_{i,1}^{\text{nom}}| \leq 2.5$ across most of the θ_i and $r_{i,1}^{\text{nom}}$. Specifically, we allocate a proportion λ of the range $(0, 2.5)$ to input θ that depends on the standard deviations b_θ and b_r, with $\lambda = b_\theta/(b_\theta + b_r)$, and define $a_1 = 5\lambda/(2\theta_{\max})$, $a_2 = (2 - a_1\theta_{\max})/r_{\max}$, where $\theta_{\max}$ and $r_{\max}$ are the largest angle and nominal radius, respectively. Note that although a_1 and a_2 involve the maximum values of the corresponding inputs, our mechanism is not equivalent to rescaling the respective inputs to lie in the range $(-1, 1)$. The $\alpha_{m,0}^{(1,p)}$ are still drawn from Uniform$(-1, 1)$ independently. Also, this new mechanism does not force all hidden neurons to be non-saturated. The general case involving more inputs follows in a similarly straightforward fashion by allocating different portions of $(0, 2.5)$ to them.

Standard tuning approaches to avoid saturation in ELMs (and thereby enhance their predictive performances) involve input normalization and/or rescaling, and the selection of different activation functions (e.g., the rectified linear unit). However, if no changes are made to the random mechanism, these methods typically fail to yield satisfactory models, and are particularly ineffective in the AM context. We accordingly developed this principled random mechanism for our Bayesian ELM methodology so as to reduce the likelihood of randomly selecting a large number of saturated hidden neurons.

A.2 Bayesian ELM model comparisons

Soria-Olivas et al. (2011) proposed a Bayesian framework for ELMs, referred to as BELMs, in which the parameters are optimized iteratively using the ML-II method of Berger (1985). The authors did not modify the usual random mechanism for the hidden neuron parameters. We compare the deviation models obtained from our methodology to those obtained from BELMs for the case studies in Section 2.3 that have at least four products. For each, one of the products is taken as the test data and the others are the training data. For example, in the case study of Section 2.3.2, the deviations for the 2″ cylinder were taken as the test data, and the

other cylinders were used to fit the models. This was done because at least three shapes are needed to learn the relationships between the inputs and deviation. The methodologies are compared graphically and via root mean squared error (RMSE), defined as $\left\{\sum_{i=1}^{n}\left(y_i-\hat{y}_i\right)^2/n\right\}^{1/2}$ where y_i and $\hat{y}_i$ are the observed and predicted deviations, respectively, for point i, and n is the size of the test data. In all of these comparisons, the inputs were not scaled to lie within $(-1,1)$, as that yields worse results.

First, consider in-plane cylinder deviations under stereolithography process A. Deviations from the $0.5'', 1''$, and $3''$ cylinders formed the training data, and those from the $2''$ cylinder were the test data. We set $M_{1,\mathrm{A}} = 40$ for both methods. We conclude from the comparisons in Fig. A.2(a) and Table A.1(a) that our method yields better out-of-sample predictions.

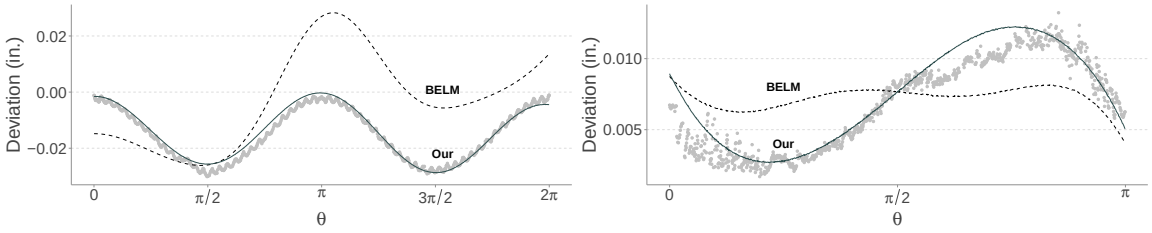
Table A.1.: Comparison of the posterior summaries for RMSE under our methodology and the standard BELM. (a) In-plane deviations of test cylinder under process A. (b) Out-of-plane deviations of test semi-cylinder under process B.

(a)

Model	**RMSE *(SE)***
Our	0.066 (0.003)
BELM	0.605 (0.020)

(b)

Model	**RMSE *(SE)***
Our	0.041 (0.002)
BELM	0.094 (0.003)



(a) (b)

Figure A.2.: Posterior predictive mean trends obtained from our methodology (solid line), and those obtained from the standard BELM method (dashed lines). (a) In-plane deviations (dots) for the test cylinder under process A. (b) Out-of-plane deviations (dots) for the vertical semi-cylinder under B.

Second, consider out-of-plane deviations of vertical semi-cylinders under stereolithography process B. Three vertical semi-cylinders of nominal radii $0.5'', 0.8''$, and $3''$ formed the training data, and the $1.5''$ vertical semi-cylinder formed the test data. We set the number of hidden neurons for both methods to 40. Fig. A.2(b) and Table A.1(b) summarizes the comparison of the predictions from these two methods, and leads again to the conclusion that our method is better.

B. SUPPLEMENTARY MATERIAL FOR CHAPTER 3

B.1 Fisher consistency proofs for predictive comparison estimators

This section provides the Fisher consistency proofs for the average predictive comparison estimators given in Chapter 3 of the dissertation. The sample of inputs $x_1, \ldots, x_n$ are assumed to be independent and identically distributed according to the probability density function $p(x)$, and the sample of parameters $\beta^{(1)}, \ldots, \beta^{(s)}$ are assumed to be independent and identically distributed according to the probability density function $p(\beta)$ that captures the uncertainty associated with β. Following Gelman and Pardoe (2007), $p(x)$ is assumed to be independent of β.

Definition B.1. For a real-valued function t of a sample of independent and identically distributed random variables $x_1, \ldots, x_n$ that is defined as a functional of their empirical distribution function $\hat{F}_n$, i.e., $t \equiv t\left(\hat{F}_n\right)$, the plug-in evaluation of t for the cumulative distribution function F from which the variables are generated is $t(F)$.

Definition B.2. For two real-valued functions t_1 and t_2 of a sample of independent and identically distributed random variables $x_1, \ldots, x_n$ that are defined as functionals of their empirical distribution function $\hat{F}_n$, the plug-in evaluations of $t_1 + t_2$, $t_1 - t_2$, and $t_1 t_2$ for the cumulative distribution function F from which the variables are generated are $t_1(F) + t_2(F)$, $t_1(F) - t_2(F)$, and $t_1(F)t_2(F)$, respectively. Furthermore, if $t_2(F) \neq 0$, then the plug-in evaluation of t_1/t_2 is $t_1(F)/t_2(F)$.

Definition B.3 (Fisher consistency (Fisher (1922); Cox and Hinkley (1974), p. 287))**.** Let θ be an estimand defined from a cumulative distribution function F. A statistic t that is a functional of the empirical distribution function $\hat{F}_n$ for a sample generated independently and identically from F is a Fisher consistent estimator for θ if $t(F) = \theta$.

Following the construction of the estimators in Sections 3.1 and 3.2, let $M : \mathbb{R}^{K-1} \times \mathbb{R}^{K-1} \to \mathbb{R}_{\geq 0}$ denote a metric on $\mathbb{R}^{K-1}$, and for any $v^{(1)} \in \mathbb{R}^{K-1}$ let $f_{v^{(1)}} : \mathbb{R}^{K-1} \to [0, 1]$ be defined as $f_{v^{(1)}}(v) = \left\{1 + M\left(v^{(1)}, v\right)\right\}^{-1}$ for $v \in \mathbb{R}^{K-1}$ (the fixed metric M is excluded in the notation for $f_{v^{(1)}}$ to simplify the exposition, with the understanding that this function does indeed depend on the selected M). The Mahalanobis (1927, 1936) metric is primarily considered in the dissertation, and its formal definition is provided below.

Definition B.4. The Mahalanobis metric for two vectors $v^{(1)}, v^{(2)} \in \mathbb{R}^{K-1}$ is

$$\left(v^{(1)} - v^{(2)}\right)^{\mathsf{T}} S^{-1} \left(v^{(1)} - v^{(2)}\right),$$

where S is a positive definite $(K-1) \times (K-1)$ matrix.

As stated in Section 3.1, Condition 3.1 on the distributions and metric for the v inputs will be considered throughout the study of Fisher consistency for the estimators. The following sequence of lemmas will be used in the Fisher consistency proofs.

Lemma B.1. Suppose metric M provides stable metric approximations as in Condition 3.1 for the marginal and conditional probability density function values $p(v)$ and $p(v \mid u)$ for any $v \in \mathbb{R}^{K-1}$ and $u \in \mathbb{R}$. Let $v^{(1)} \in \mathbb{R}^{K-1}$ such that $p\left(v^{(1)}\right) \neq 0$, $u^{(2)} \in \mathbb{R}$, and define the statistics $t_1 = \sum_{j=1}^{n} \mathbb{I}\left(u_j = u^{(2)}\right) f_{v^{(1)}}(v_j)/n$ and $t_2 = \sum_{j=1}^{n} f_{v^{(1)}}(v_j)/n$, where $\mathbb{I}(\cdot)$ is the indicator function. Then t_1/t_2 is Fisher consistent for the conditional probability density function value $p\left(u^{(2)} \mid v^{(1)}\right)$.

Proof. By inspection, the plug-in evaluations of t_1 and t_2 are $t_1(F) = \int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v)$ $p\left(u^{(2)}, v\right) dv = p\left(u^{(2)}\right) \int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v) p\left(v \mid u^{(2)}\right) dv$ and $t_2(F) = \int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v) p(v) dv$, respectively. From Defintion B.2, the plug-in evaluation of t_1/t_2 is then $p\left(u^{(2)}\right) \int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v) p\left(v \mid u^{(2)}\right) dv / \int_{\mathbb{R}^{K-1}} f_{v^{(1)}}(v) p(v) dv$. Thus, by virtue of the stable metric-input distribution assumption, the plug-in evaluation of t_1/t_2 is equivalent to $p\left(u^{(2)}\right) p\left(v^{(1)} \mid u^{(2)}\right) / p\left(v^{(1)}\right)$, which equals $p\left(u^{(2)} \mid v^{(1)}\right)$. □

This result can be understood by recognizing that, when considering the entire population, the statistic t_1/t_2 in Lemma C.1 is effectively a weighted average of counts of inputs $\left(u^{(2)}, v\right)$ with larger weights assigned to those counts in which v is closer to $v^{(1)}$. Under the stable metric-input distribution assumption, this yields $p\left(u^{(2)} \mid v^{(1)}\right)$.

Lemma B.2. Suppose metric M satisfies the stable metric-input distribution assumption. Let $u^{(1)} \in \mathbb{R}, v^{(1)} \in \mathbb{R}^{K-1}, \beta \in \mathbb{R}^L$, and $g_{u^{(1)},v^{(1)},\beta} : \mathbb{R} \to \mathbb{R}$ be defined based on $u^{(1)}, v^{(1)}$, and β. Define the statistics $t_1 = \sum_{j=1}^{n} g_{u^{(1)},v^{(1)},\beta}(u_j) f_{v^{(1)}}(v_j)/n$ and $t_2 = \sum_{j=1}^{n} f_{v^{(1)}}(v_j)/n$. Then t_1/t_2 is Fisher consistent for $\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}(u) p\left(u \mid v^{(1)}\right) du$.

Proof. The statistic t_1/t_2 can be written as the sum of statistics

$$\sum_{j=1}^{n} \left[g_{u^{(1)},v^{(1)},\beta}(u_j) \left\{ \frac{\sum_{k=1}^{n} \mathbb{I}(u_k = u_j) f_{v^{(1)}}(v_k)/n}{\sum_{k=1}^{n} f_{v^{(1)}}(v_k)/n} \right\} \right].$$

By virtue of Lemma C.1, for each u_j the statistic

$$g_{u^{(1)},v^{(1)},\beta}(u_j) \left\{ \frac{\sum_{k=1}^{n} \mathbb{I}(u_k = u_j) f_{v^{(1)}}(v_k)/n}{\sum_{k=1}^{n} f_{v^{(1)}}(v_k)/n} \right\}$$

is Fisher consistent for $g_{u^{(1)},v^{(1)},\beta}(u_j) p\left(u_j \mid v^{(1)}\right)$. By considering Definition B.2 and the sum of these statistics, t_1/t_2 is accordingly Fisher consistent for $\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}(u)$ $p\left(u \mid v^{(1)}\right) du$. □

Lemma B.3. Suppose metric M satisfies the stable metric-input distribution assumption. Let $g_{u^{(1)},v^{(1)},\beta} : \mathbb{R} \to \mathbb{R}$ be defined for any $u^{(1)} \in \mathbb{R}, v^{(1)} \in \mathbb{R}^{K-1}$, and $\beta \in \mathbb{R}^L$. Define the statistic

$$t_3 = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{\sum_{j=1}^{n} g_{u_i,v_i,\beta}(u_j)f_{v_i}(v_j)}{\sum_{j=1}^{n} f_{v_i}(v_j)}\right\}.$$

Then t is Fisher consistent for $\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) p\left(u^{(1)},u^{(2)},v^{(1)}\right) du^{(1)}du^{(2)}dv^{(1)}$, where $u^{(1)}$ and $u^{(2)}$ are independent u input variables.

Proof. By the law of total expectation, for two independent u input variables $u^{(1)}$ and $u^{(2)}$,

$$\begin{aligned}&\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) p\left(u^{(1)},u^{(2)},v^{(1)}\right) du^{(1)}du^{(2)}dv^{(1)}\\ &\qquad = \int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\left\{\int_{\mathbb{R}} g_{u^{(1)},v^{(1)}\beta}\left(u^{(2)}\right) p\left(u^{(2)} \mid v^{(1)}\right) du^{(2)}\right\} p\left(u^{(1)},v^{(1)}\right) du^{(1)}dv^{(1)}.\end{aligned}$$

Now for each input (u_i, v_i), the statistic $\left\{\sum_{j=1}^{n} g_{u_i,v_i,\beta}(u_j)f_{v_i}(v_j)\right\} / \sum_{j=1}^{n} f_{v_i}(v_j)$ is Fisher consistent for $\int_{\mathbb{R}} g_{u_i,v_i,\beta}\left(u^{(2)}\right) p\left(u^{(2)} \mid v_i\right) du^{(2)}$ by virtue of Lemma C.2. It then follows from Definition B.2 and the observations above that t is Fisher consistent for $\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) p\left(u^{(1)},u^{(2)},v^{(1)}\right) du^{(1)}du^{(2)}dv^{(1)}$. □

Lemma B.4. Suppose metric M satisfies the stable metric-input distribution assumption. Let $g_{u^{(1)},v^{(1)},\beta} : \mathbb{R} \to \mathbb{R}$ be defined for any $u^{(1)} \in \mathbb{R}, v^{(1)} \in \mathbb{R}^{K-1}$, and $\beta \in \mathbb{R}^L$. Then

$$t_4 = \frac{1}{Sn}\sum_{s=1}^{S}\sum_{i=1}^{n}\left\{\frac{\sum_{j=1}^{n} g_{u_i,v_i,\beta^{(s)}}(u_j)f_{v_i}(v_j)}{\sum_{j=1}^{n} f_{v_i}(v_j)}\right\}$$

is Fisher consistent for $\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) p\left(u^{(1)},u^{(2)},v^{(1)}\right) p(\beta) du^{(1)} du^{(2)} dv^{(1)} d\beta$, where $u^{(1)}$ and $u^{(2)}$ are independent u input variables.

Proof. By the law of total expectation, for two independent u input variables $u^{(1)}$ and $u^{(2)}$,

$$\begin{aligned}&\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) p\left(u^{(1)},u^{(2)},v^{(1)}\right) p(\beta) du^{(1)} du^{(2)} dv^{(1)} d\beta \\ &\quad = \int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}} \left\{\int_{\mathbb{R}} g_{u^{(1)},v^{(1)}\beta}\left(u^{(2)}\right) p\left(u^{(2)} \mid v^{(1)}\right) du^{(2)}\right\} p\left(u^{(1)},v^{(1)}\right) p(\beta) du^{(1)} dv^{(1)} d\beta.\end{aligned}$$

From Lemma C.3, the statistic

$$\frac{1}{n}\sum_{i=1}^{n}\left\{\frac{\sum_{j=1}^{n} g_{u_i,v_i,\beta^{(s)}}(u_j) f_{v_i}(v_j)}{\sum_{j=1}^{n} f_{v_i}(v_j)}\right\}$$

is Fisher consistent for $\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\left\{\int_{\mathbb{R}} g_{u^{(1)},v^{(1)}\beta^{(s)}}\left(u^{(2)}\right) p\left(u^{(2)} \mid v^{(1)}\right) du^{(2)}\right\} p\left(u^{(1)},v^{(1)}\right) du^{(1)} dv^{(1)}$ for each $\beta^{(s)}$. The final result then follows from Definition B.2 and the assumption that the $\beta^{(s)}$ are independent and identically distributed draws from $p(\beta)$.

□

Lemma B.5. For any $u^{(1)} \in \mathbb{R}, v^{(1)} \in \mathbb{R}^{K-1}$, and $\beta \in \mathbb{R}^L$, let $g_{u^{(1)},v^{(1)},\beta} : \mathbb{R} \to \mathbb{R}$ be defined such that for any $u^{(2)} \in \mathbb{R}$, $g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) = g^*_{v^{(1)},\beta}\left(u^{(2)}\right) - g^*_{v^{(1)},\beta}\left(u^{(1)}\right)$ for some function $g^*_{v^{(1)},\beta} : \mathbb{R} \to \mathbb{R}$ that is defined based on just $v^{(1)}$ and β. Then for two identically distributed u input variables $u^{(1)}$ and $u^{(2)}$,

$$\begin{aligned}&\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) \operatorname{sign}\left(u^{(2)} - u^{(1)}\right) p\left(u^{(1)},u^{(2)},v^{(1)}\right) p(\beta) du^{(1)} du^{(2)} dv^{(1)} d\beta \\ &= 2\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) \mathbb{I}\left(u^{(1)} < u^{(2)}\right) p\left(u^{(1)},u^{(2)},v^{(1)}\right) p(\beta) du^{(1)} du^{(2)} dv^{(1)} d\beta.\end{aligned}$$

Proof. Excluding sets of measure zero, $\text{sign}\left(u^{(2)} - u^{(1)}\right) = 2\mathbb{I}\left(u^{(1)} < u^{(2)}\right) - 1$. Hence the first integral above is equivalent to

$$2\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)\mathbb{I}\left(u^{(1)} < u^{(2)}\right)p\left(u^{(1)},u^{(2)},v^{(1)}\right)p(\beta)du^{(1)}du^{(2)}dv^{(1)}d\beta$$
$$-\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)p\left(u^{(1)},u^{(2)},v^{(1)}\right)p(\beta)du^{(1)}du^{(2)}dv^{(1)}d\beta.$$

As $\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)p\left(u^{(1)},u^{(2)},v^{(1)}\right)p(\beta)du^{(1)}du^{(2)}dv^{(1)}d\beta = 0$ from the given property of $g_{u^{(1)},v^{(1)},\beta}$, the final result follows accordingly. □

The Fisher consistency proofs for Theorems 3.1, 3.2, and 3.3 follow below.

Proof of Theorem 3.1. For any $u^{(1)}, u^{(2)} \in \mathbb{R}$, $v^{(1)} \in \mathbb{R}^{K-1}$, and $\beta \in \mathbb{R}^L$, let $h_{u^{(1)},v^{(1)}}\left(u^{(2)}\right) = u^{(2)} - u^{(1)}$ and $g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) = \mathbb{E}\left(y \mid u^{(2)}, v^{(1)}, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, v^{(1)}, \beta\right)$. Note that $w_{ij} = f_{v_i}(v_j)$, and

$$\widehat{\Delta}_u = \frac{1}{S}\sum_{s=1}^{S}\left[\frac{\displaystyle\sum_{i=1}^{n}\left\{\left(\frac{1}{n}\sum_{j=1}^{n}w_{ij}\right)\left(\frac{\displaystyle\sum_{j=1}^{n} g_{u_i,v_i,\beta^{(s)}}(u_j)w_{ij}\text{sign}(u_j - u_i)/n}{2\displaystyle\sum_{j=1}^{n}w_{ij}/n}\right)\right\}}{\displaystyle\sum_{i=1}^{n}\left\{\left(\frac{1}{n}\sum_{j=1}^{n}w_{ij}\right)\left(\frac{\displaystyle\sum_{j=1}^{n} h_{u_i,v_i}(u_j)w_{ij}\text{sign}(u_j - u_i)/n}{2\displaystyle\sum_{j=1}^{n}w_{ij}/n}\right)\right\}}\right].$$

Then by virtue of Definition B.2, the stable metric-input distribution assumption, and the previous lemmas, it follows that $\widehat{\Delta}_u$ is Fisher consistent for Δ_u. □

Proof of Theorem 3.2. By virtue of Definition B.2, the stable metric-input distribution assumption, Theorem 3.1, and the fact that the numerator and denominator of $\widehat{\Delta}^2_{\text{mag}(u)}$ are the squares of the numerator and denominator of $\widehat{\Delta}_u$, respectively, it follows that $\widehat{\Delta}^2_{\text{mag}(u)}$ is Fisher consistent for $\Delta^2_{\text{mag}(u)}$, and so $\widehat{\Delta}_{\text{mag}(u)}$ is Fisher consistent for $\Delta_{\text{mag}(u)}$. Also, for any $v \in \mathbb{R}^{K-1}$ and $\beta \in \mathbb{R}^L$, a Fisher consistent estimator

of $\overline{E_{u|v}(y \mid u, v, \beta)}$ is $\sum_{j=1}^{n} f_v(v_j)\mathbb{E}\left(y \mid u_j, v, \beta\right) / \sum_{j=1}^{n} f_v(v_j)$. Thus from the previous lemmas, $\widehat{\Lambda}_u^2$ is Fisher consistent for Λ_u^2, and so $\widehat{\Lambda}_u$ is Fisher consistent for Λ_u. □

Proof of Theorem 3.3. The proofs for $\widehat{\Delta}_{u|z}$, $\widehat{\Delta}_{u\times z}$, $\widehat{\Delta}_{\mathrm{mag}(u\times z)}$, and $\widehat{\Lambda}_{(u\times z)}$ follow by the same arguments that were employed in the proofs of Theorem 3.1 and Theorem 3.2. Note that $\widehat{\Delta}_{u|z}$ is $\widehat{\Delta}_u$ for a specific level of input z, and so its proof considers integration over v_{-z}, not v. □

B.2 Standard errors for predictive comparison estimators

This section provides the calculations of the standard errors for the average predictive comparison estimators given in Chapter 3 of the dissertation. These calculations are performed under the same assumptions on the samples of inputs and parameters as invoked in Appendix B.1.

Proposition B.1. For any input u in x and $A_u \in \left\{\widehat{\Lambda}_u, \widehat{\Delta}_{\mathrm{mag}(u)}\right\}$, the standard error for the selected average predictive comparison estimator is

$$\mathrm{SE}\left(A_u\right) = \frac{1}{2A_u}\left[\frac{1}{S-1}\sum_{s=1}^{S}\left\{\left(A_u^{(s)}\right)^2 - A_u^2\right\}^2\right]^{1/2},$$

where

$$\widehat{\Lambda}_u^{(s)} = \sum_{i=1}^{n}\left\{\mathbb{E}\left(y \mid x_i, \beta^{(s)}\right) - \overline{E_{u|v_i}\left(y \mid u, v_i, \beta^{(s)}\right)}\right\}^2,$$

$$\widehat{\Delta}_{\mathrm{mag}(u)}^{(s)} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\left\{\mathbb{E}\left(y \mid u_j, v_i, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, v_i, \beta^{(s)}\right)\right\}^2 w_{ij}}{\left(\sum_{i=1}^{n}\sum_{j=1}^{n}(u_j - u_i)^2 w_{ij}\right)}.$$

Proof. The standard errors are derived according to the Taylor expansion calculation of Gelman and Pardoe (2007, p. 40) in which, for any random variable A, the standard

deviation of $A^{1/2}$ is approximately equal to the standard deviation of A divided by $2\sqrt{A}$. By treating β as random and x as fixed, the standard deviation of A_u^2 is

$$\left[\frac{1}{S-1}\sum_{s=1}^{S}\left\{\left(A_u^{(s)}\right)^2 - A_u^2\right\}^2\right]^{1/2},$$

and the final result follows accordingly. □

Proposition B.2. For any input u in x and $A_u \in \left\{\widehat{\Lambda}_u, \widehat{\Delta}_{\text{mag}(u)}\right\}$, the standard error for the relative predictive comparison is

$$\text{SE}\left\{R(A_u)\right\} = \frac{\text{SE}(A_u)\sum_{k\neq u} A_k}{\left(\sum_k A_k\right)^2},$$

where A_k is either $\widehat{\Delta}_{\text{mag}(k)}$ or $\widehat{\Lambda}_k$ (corresponding to the choice of A_u) for all k in x.

Proof. For $A_u \in \left\{\widehat{\Delta}_{\text{mag}(u)}, \widehat{\Lambda}_u\right\}$, $\text{SE}\left\{g\left(A_u\right)\right\} = SE(A_u)g'\left(A_u\right)$ by the Delta Method (Casella and Berger, 2002, p. 240). From the definition of $R(A_u)$,

$$\frac{\partial R\left(A_u\right)}{\partial A_u} = \left\{\left(\sum_{k=1}^{K} A_k\right) - A_u\right\} \Big/ \left(\sum_{k=1}^{K} A_k\right)^2 = \sum_{k\neq u} A_k \Big/ \left(\sum_{k=1}^{K} A_k\right)^2.$$

The result follows accordingly. □

Proposition B.3. For any inputs u and z in x, and for $V_{(u,z)} \in \left\{\widehat{\Delta}_{u|z}, \widehat{\Delta}_{u\times z}\right\}$, the standard errors for the estimators of the average conditional and two-way interaction predictive comparison estimators are $\text{SE}\left(V_{(u,z)}\right) = \left\{\sum_{s=1}^{S}\left(V_{(u,z)}^{(s)} - V_{(u,z)}\right)^2 \Big/ (S-1)\right\}^{1/2}$, where

$$\widehat{\Delta}_{u|z}^{(s)} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}\left\{\mathbb{E}\left(y \mid u_j, z, v_{-z,i}, \beta^{(s)}\right) - \mathbb{E}\left(y \mid u_i, z, v_{-z,i}, \beta^{(s)}\right)\right\}\text{sign}(u_j - u_i)}{\sum_{i=1}^{n}\sum_{j=1}^{n}(u_j - u_i)w_{ij}\text{sign}(u_j - u_i)},$$

$$\widehat{\Delta}^{(s)}_{u\times z} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} d((u_i,u_j)\times(z_i,z_j))w_{ij}\text{sign}\{(u_j-u_i)(z_j-z_i)\}}{\sum_{i=1}^{n}\sum_{j=1}^{n}(u_j-u_i)(z_j-z_i)w_{ij}\text{sign}\{(u_j-u_i)(z_j-z_i)\}}.$$

Proof. The standard errors are derived in a similar manner as that in (Gelman and Pardoe, 2007, p. 39). The β in $V_{(u,z)} \in \left\{\widehat{\Delta}_{u|z}, \widehat{\Delta}_{u\times z}\right\}$ are treated as random, and x as fixed. Thus, by standard sampling theory methods, the standard errors are $\text{SE}\left(V_{(u,z)}\right) = \left\{\sum_{s=1}^{S}\left(V^{(s)}_{(u,z)} - V_{(u,z)}\right)^2 \Big/ (S-1)\right\}^{1/2}$. □

B.3 Supplementary information for predicting and understanding student performance

Descriptions of the inputs considered in the student performance case study in Section 3.3 are provided in Table B.1. Each student's outcome is their final grade, which is a numeric value ranging from 0 to 20.

Table B.1.: Inputs in the student performance case study of Section 3.3 (Cortez and Silva, 2008).

Input	Description
school	Student's school (binary: Gabriel Pereira or Mousinho da Silveira)
sex	student's sex (binary: female or male)
age	Student's age (numeric: from 15 to 22)
address	Student's home address type (binary: urban or rural)
famsize	Family size (binary: less or equal to 3 or greater than 3)
Pstatus	Parent's cohabitation status (binary: living together or apart)
Medu	Mother's education (numeric: none, primary education (4th grade), 5th to 9th grade, secondary education or higher education)
Fedu	Father's education (numeric: none, primary education (4th grade), 5th to 9th grade, secondary education or higher education)
Mjob	Mother's job (nominal: teacher, health care related, civil services (e.g. administrative or police), at home or other)
Fjob	Father's job (nominal: teacher, health care related, civil services (e.g. administrative or police), at home or other)
reason	Reason to choose this school (nominal: close to home, school reputation, course preference or other)

continued on next page

Table B.1.: *continued*

Input	Description
guardian	Student's guardian (nominal: mother, father or other)
traveltime	Home to school travel time (numeric: <15 min., 15 to 30 min., 30 min. to 1 hour, or >1 hour)
studytime	Weekly study time (numeric: <2 hours, 2 to 5 hours, 5 to 10 hours, or >10 hours)
failures	Number of past class failures (numeric: n if $1 \leq n < 3$, else 4)
schoolsup	Extra educational support (binary: yes or no)
famsup	Family educational support (binary: yes or no)
paid	Extra paid classes (binary: yes or no)
activities	Extra-curricular activities (binary: yes or no)
nursery	Attended nursery school (binary: yes or no)
higher	Wants to take higher education (binary: yes or no)
internet	Internet access at home (binary: yes or no)
romantic	With a romantic relationship (binary: yes or no)
famrel	Quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
freetime	Free time after school (numeric: from 1 - very low to 5 - very high)
goout	Going out with friends (numeric: from 1 - very low to 5 - very high)
Dalc	Workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
Walc	Weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
health	Current health status (numeric: from 1 - very bad to 5 - very good)
absences	Number of school absences (numeric: from 0 to 93)

C. SUPPLEMENTARY MATERIAL FOR CHAPTER 4

C.1 Fisher consistency proofs for generalized predictive comparisons estimators

This section contains the Fisher consistency proofs for the generalized predictive comparison estimators given in Chapter 4. We assume that sample of inputs $x_1, \ldots, x_n$ and the sample of parameters $\beta^{(1)}, \ldots, \beta^{(s)}$ are independent and identically distributed according to the probability density functions $p(x)$ and $p(\beta)$, respectively. Following Gelman and Pardoe (2007) $p(x)$ is assumed to be independent of β.

The proofs for Fisher consistency of the generalized predictive comparison estimators are extensions of the proofs provided in Appendix B.1. Let $M : \mathbb{R}^{K-d} \times \mathbb{R}^{K-d} \to \mathbb{R}_{\geq 0}$ denote a metric on $\mathbb{R}^{K-d}$, and for any $v^{(1)} \in \mathbb{R}^{K-d}$. The Mahalanobis (1927, 1936) metric is primarily considered in the dissertation, and its formal definition B.4 is extended for $v^{(1)}, v^{(2)} \in \mathbb{R}^{K-d}$ below.

Definition C.1. The Mahalanobis metric for two vectors $v^{(1)}, v^{(2)} \in \mathbb{R}^{K-d}$ is

$$M\left(v^{(1)}, v^{(2)}\right) = \left(v^{(1)} - v^{(2)}\right)^{\mathsf{T}} S^{-1} \left(v^{(1)} - v^{(2)}\right),$$

where S is a positive definite $(K-d) \times (K-d)$ matrix.

The following sequence of lemmas will be used in the Fisher consistency proofs. All lemmas are extensions of the lemmas in Appendix B.1 for $u \in \mathbb{R}^d$. Thus, all proofs follow similar arguments that were employed in Appendix B.1.

Lemma C.1. Suppose metric M provides stable metric approximations as in Condition 3.1 for the marginal and conditional probability density function values $p(v)$ and $p(v \mid u)$ for any $v \in \mathbb{R}^{K-d}$ and $u \in \mathbb{R}^d$. Let $v^{(1)} \in \mathbb{R}^{K-d}$ such that $p\left(v^{(1)}\right) \neq 0$, $u^{(2)} \in \mathbb{R}^d$, and define the statistics $t_1 = \sum_{j=1}^{n} \mathbb{I}\left(u_j = u^{(2)}\right) \left\{1 + M\left(v^{(1)}, v_j\right)\right\}^{-1} / n$ and

$t_2 = \sum_{j=1}^{n} \left\{1 + M\left(v^{(1)}, v_j\right)\right\}^{-1} / n$, where $\mathbb{I}(\cdot)$ is the indicator function. Then t_1/t_2 is Fisher consistent for the conditional probability density function value $p\left(u^{(2)} \mid v^{(1)}\right)$.

Lemma C.2. Suppose metric M satisfies the stable metric-input distribution assumption. Let $u^{(1)} \in \mathbb{R}^d, v^{(1)} \in \mathbb{R}^{K-d}, \beta \in \mathbb{R}^L$, and $g_{u^{(1)},v^{(1)},\beta} : \mathbb{R}^d \to \mathbb{R}$ be well-defined based on $u^{(1)}, v^{(1)}$, and β. Define the statistics $t_1 = \sum_{j=1}^{n} g_{u^{(1)},v^{(1)},\beta}(u_j) \left\{1 + M\left(v^{(1)}, v_j\right)\right\}^{-1} / n$ and $t_2 = \sum_{j=1}^{n} \left\{1 + M\left(v^{(1)}, v_j\right)\right\}^{-1} / n$. Then t_1/t_2 is Fisher consistent for $\int_{\mathbb{R}^d} g_{u^{(1)},v^{(1)},\beta}(u) p\left(u \mid v^{(1)}\right) du$.

Lemma C.3. Suppose metric M satisfies the extended stable metric-input distribution assumption. Let $g_{u^{(1)},v^{(1)},\beta} : \mathbb{R}^d \to \mathbb{R}$ be defined for any $u^{(1)} \in \mathbb{R}d, v^{(1)} \in \mathbb{R}^{K-d}$, and $\beta \in \mathbb{R}^L$. Define the statistic

$$t_3 = \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{\sum_{j=1}^{n} g_{u_i,v_i,\beta}(u_j) \left\{1 + M\left(v^{(1)}, v_j\right)\right\}^{-1}}{\sum_{j=1}^{n} \left\{1 + M\left(v^{(1)}, v_j\right)\right\}^{-1}} \right\}.$$

Then t_3 is Fisher consistent for $\int_{\mathbb{R}^{K-d}} \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) p\left(u^{(1)}, u^{(2)}, v^{(1)}\right) du^{(1)} du^{(2)} dv^{(1)}$, where $u^{(1)}$ and $u^{(2)}$ are independent u input variables.

To facilitate exposure of Lemmas C.4 and C.5, consider $q = \left(u^{(1)}, u^{(2)}, v^{(1)}, \beta\right)$, and $p(q) = p\left(u^{(1)}, u^{(2)}, v^{(1)}\right) p(\beta)$.

Lemma C.4. Suppose metric M satisfies the extended stable metric-input distribution assumption. Let $g_{u^{(1)},v^{(1)},\beta} : \mathbb{R}^d \to \mathbb{R}$ be well-defined for any $u^{(1)} \in \mathbb{R}^d, v^{(1)} \in \mathbb{R}^{K-d}$, and $\beta \in \mathbb{R}^L$. Then

$$t_4 = \frac{1}{Sn} \sum_{s=1}^{S} \sum_{i=1}^{n} \left\{ \frac{\sum_{j=1}^{n} g_{u_i,v_i,\beta^{(s)}}(u_j) \left\{1 + M\left(v_i, v_j\right)\right\}^{-1}}{\sum_{j=1}^{n} \left\{1 + M\left(v_i, v_j\right)\right\}^{-1}} \right\}$$

is Fisher consistent for $\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-1}}\int_{\mathbb{R}}\int_{\mathbb{R}} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)p(q)dq$, where $u^{(1)}$ and $u^{(2)}$ are independent u input variables.

Lemma C.5. For any $u^{(1)} \in \mathbb{R}^d, v^{(1)} \in \mathbb{R}^{K-d}$, and $\beta \in \mathbb{R}^L$, let $g_{u^{(1)},v^{(1)},\beta} : \mathbb{R}^d \to \mathbb{R}$ be well-defined such that for any $u^{(2)} \in \mathbb{R}^d$, $g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) = g^*_{v^{(1)},\beta}\left(u^{(2)}\right) - g^*_{v^{(1)},\beta}\left(u^{(1)}\right)$ for some function $g^*_{v^{(1)},\beta} : \mathbb{R}^d \to \mathbb{R}$ that is well-defined based on just $v^{(1)}$ and β. In addition, let $f : \mathbb{R}^d \to \mathbb{R}$ be a well-defined Interpretable Mapper for u. Then for two identically distributed u input variables $u^{(1)}$ and $u^{(2)}$,

$$\begin{aligned}&\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-d}}\int_{\mathbb{R}^d}\int_{\mathbb{R}^d} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)\operatorname{sign}\left(f\left(u^{(2)}\right) - f\left(u^{(1)}\right)\right)p(q)dq\\ &= 2\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-d}}\int_{\mathbb{R}^d}\int_{\mathbb{R}^d} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)\mathbb{I}\left\{f\left(u^{(1)}\right) < f\left(u^{(2)}\right)\right\}p(q)dq.\end{aligned}$$

Proof. Excluding sets of measure zero, $\operatorname{sign}\left(f\left(u^{(2)}\right) - f\left(u^{(1)}\right)\right) = 2\,\mathbb{I}\left\{f\left(u^{(1)}\right) < f\left(u^{(2)}\right)\right\} - 1$. Thus the first integral above is equivalent to

$$\begin{aligned}&2\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-d}}\int_{\mathbb{R}^d}\int_{\mathbb{R}^d} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)\mathbb{I}\left\{f\left(u^{(1)}\right) < f\left(u^{(2)}\right)\right\}p(q)dq\\ &- \int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-d}}\int_{\mathbb{R}^d}\int_{\mathbb{R}^d} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)p(q)dq.\end{aligned}$$

As $\int_{\mathbb{R}^L}\int_{\mathbb{R}^{K-d}}\int_{\mathbb{R}^d}\int_{\mathbb{R}^d} g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right)p(q)dq = 0$ from the given property of $g_{u^{(1)},v^{(1)},\beta}$, the final result follows accordingly. □

The Fisher consistency proofs for Theorems 4.1, 4.2, and 4.3 follow below.

Proof of Theorem 4.1. For any $u^{(1)}, u^{(2)} \in \mathbb{R}^d$, $v^{(1)} \in \mathbb{R}^{K-d}$, $\beta \in \mathbb{R}^L$, and interpretable mapper $f : \mathbb{R}^d \to \mathbb{R}$, let $h_{u^{(1)},v^{(1)}}\left(u^{(2)}\right) = f\left(u^{(2)}\right) - f\left(u^{(1)}\right)$ and $g_{u^{(1)},v^{(1)},\beta}\left(u^{(2)}\right) = \mathbb{E}\left(y \mid u^{(2)}, v^{(1)}, \beta\right) - \mathbb{E}\left(y \mid u^{(1)}, v^{(1)}, \beta\right)$. Note that $w_{ij} = \{1 + M(v_i, v_j)\}^{-1}$, and

$$\widehat{\Delta}_{f(u)} = \frac{1}{S}\sum_{s=1}^{S}\left[\frac{\displaystyle\sum_{i=1}^{n}\left\{\left(\frac{1}{n}\sum_{j=1}^{n}w_{ij}\right)\left(\frac{\displaystyle\sum_{j=1}^{n}g_{u_i,v_i,\beta^{(s)}}(u_j)w_{ij}\text{sign}(f(u_j)-f(u_i))/n}{2\displaystyle\sum_{j=1}^{n}w_{ij}/n}\right)\right\}}{\displaystyle\sum_{i=1}^{n}\left\{\left(\frac{1}{n}\sum_{j=1}^{n}w_{ij}\right)\left(\frac{\displaystyle\sum_{j=1}^{n}h_{u_i,v_i}(u_j)w_{ij}\text{sign}(f(u_j)-f(u_i))/n}{2\displaystyle\sum_{j=1}^{n}w_{ij}/n}\right)\right\}}\right].$$

Then by virtue of Definition B.2, the extended stable metric-input distribution assumption, and the previous lemmas, it follows that $\widehat{\Delta}_{f(u)}$ is Fisher consistent for $\Delta_{f(u)}$.

By virtue of Definition B.2, the extended stable metric-input distribution assumption, the result above, and the fact that the numerator and denominator of $\widehat{\Delta}^2_{\text{mag}(f(u))}$ are the squares of the numerator and denominator of $\widehat{\Delta}_{f(u)}$, respectively, it follows that $\widehat{\Delta}^2_{\text{mag}(f(u))}$ is Fisher consistent for $\Delta^2_{\text{mag}(f(u))}$, and so $\widehat{\Delta}_{\text{mag}(f(u))}$ is Fisher consistent for $\Delta_{\text{mag}(f(u))}$. □

Proof of Theorem 4.2. By virtue of Definition B.2, the extended stable metric-input distribution assumption, Theorem 4.1, previous lemmas, and the fact that for any $v \in \mathbb{R}^{K-d}$ and $\beta \in \mathbb{R}^L$, a Fisher consistent estimator of $\overline{E_{u|v}(y \mid u, v, \beta)}$ is $\sum_{j=1}^{n}\{1 + M(v, v_j)\}^{-1}\mathbb{E}(y \mid u_j, v, \beta) / \sum_{j=1}^{n}\{1 + M(v, v_j)\}^{-1}$, it follows that $\widehat{\Delta}_{f(u)\times g(z)}$ is Fisher consistent for $\Delta_{f(u)\times g(z)}$. □

Proof of Theorem 4.3. The proofs for $\widehat{\Delta}_{i,f(u)}$ and $\widehat{\Delta}_{i,\text{mag}(f(u))}$ follow by the same arguments that were employed in the proofs of Theorem 4.1 and Theorem 4.2. □

VITA

VITA

Raquel de Souza Borges Ferreira was born in Belo Horizonte, Minas Gerais, Brazil. She received a bachelor's degree in Statistics from Universidade Federal de Juiz de Fora in 2012, and a master's degree in Statistics from Universidade Federal de Minas Gerais in 2014. She continued her graduate studies in Statistics at Purdue University under Dr. Arman Sabbaghi. Her research interests are varied.