TOWARDS POST-HOC HUMAN-INTERPRETABILITY OF MULTIMODAL NEURAL NETWORKS FOR HEALTHCARE APPLICATIONS

By

Muneeza Azmat

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computational Mathematics, Science and Engineering —Doctor of Philosophy

2023

ABSTRACT

The use of artificial intelligence (AI) in healthcare has rapidly expanded in recent years. Multimodal neural networks (MNNs) that analyze diverse data types like images, lab reports, and genomics data often outperform unimodal approaches in healthcare applications. However, owing to their complex architecture, the decision-making logic of these large AI models is often unknown. This raises serious concerns surrounding model reliability, accountability, patient autonomy, and bias. The black-box nature of these models often makes them unsuitable for high-risk healthcare applications. Research in explainable AI (XAI) is therefore critical for the safe implementation of these models. This dissertation develops a two-phase approach to improve explainability and reliability assessment of MNNs in healthcare: Phase 1 - Explainability via feature importance: we develop a unified framework that quantifies the relative importance of multimodal inputs using post-hoc model-agnostic methods. The estimated importances are validated through importance-known-exactly simulations and agreement between multiple attribution methods. Experiments with multimodal breast tumor and cardiomegaly classifiers demonstrate the technique explains model behavior across diverse data types with high agreement scores and alignment with expert intuition. Phase 2 - Quantifying prediction reliability: we use multimodal input importance to predict the impact of missing inputs on MNN performance. This impact is presented with interpretable performance metrics, including accuracy reduction, providing measures closely tied to model reliability. We also propose an extension of the average model reliability to more fine grain patient-specific reliability estimates using reliability calibration curves. The methods developed in this dissertation offer promising approaches to improve interpretability and quantify reliability of complex MNNs, potentially facilitating their safe adoption in high-risk clinical settings.

We make our world significant by the courage of our questions and the depth of our answers.

- Carl Sagan.

ACKNOWLEDGEMENTS

This has been an exciting, sometimes painful but mostly rewarding journey that would not have been possible without the support of many. I am especially grateful for my dad, Azmat Ullah, who always supported us in the pursuit of our dreams and my mom, Shaheen, who taught us not to take life too seriously. My husband, Aseem, is a constant source of inspiration, his unmatched faith in my abilities lifts me up in my most difficult moments. I also want to thank my incredible siblings - Muneeba, Zarmeen, Osama, and Ayesha, who are my biggest hype-squad. They are always rooting for me with unconditional love and occasional reality checks to keep me grounded. Visits and video-calls with my niece, Amal, have kept me sane during this final stretch. I am incredibly thankful to my amazing loving family.

I also want to sincerely thank my advisor, Prof. Alessio, for being an incredible mentor in research and in life. Whether it be understanding image reconstruction or learning how to refill windshield fluid, every interaction has been a meaningful learning experience. His integrity, empathy, and compassion have profoundly shaped who I am, and who I aspire to be, as a researcher and as a human being.

I want to thank my committee members Prof. Arjun Krishnan, Prof. Selin Aviyente, and Prof. Yuying Xie for their invaluable feedback, and support. I am thankful for the privilege of learning from so many wise mentors.

I also want to thank the incredible people I've met at MSU and East Lansing. A shout-out to the 'Bridge gang' for providing sanity during COVID, Uswa for putting up with my antics and for introducing me to so many cool things around the area, and Wenjie for being the best pep-talk-giver and my partner-in-cry. Also big thank you to Jonathan for enduring my phone-call rants, to Henry for being my brunch and Wharton buddy (the finer things club), and to all the MIDI lab folks for their consistent warmth and support.

Lastly, I want to acknowledge and honor my late grandfather, Hakeem Qudrat Ullah Khan. He was a lifelong learner and a visionary, whose passion for knowledge paved the way for my father, and consequently for me, to embark on this journey of lifelong learning.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

BACKGROUND

1.1 The Black Box Problem in Artificial Intelligence

Artificial intelligence (AI) systems based on machine learning and deep neural networks (DNNs) have become increasingly popular in recent years, revolutionizing a wide range of sectors with their ability to learn from vast data. These computational models, inspired by the human brain, are designed to automatically learn and improve from experience. They have been utilized in various fields like finance, marketing, self-driving cars, voice recognition systems, and healthcare.

The use of AI in healthcare has rapidly expanded in recent years, though its origins trace back decades. Early AI systems were developed in the 1960s to replicate aspects of medical reasoning and decision-making [5]. Expert systems that encoded rules to perform diagnostic tasks emerged in the 1970s and 1980s [6]. Machine learning then enabled statistical pattern recognition for tasks like imaging analysis in the 1990s [7]. With modern advanced deep learning, AI now matches or exceeds clinicians on select diagnostic tasks, as seen across various sub-specialties like dermatology, ophthalmology, and radiology [8, 9, 10]. Beyond diagnosis, AI has applications across healthcare, including automated patient monitoring, personalized treatment recommendations, robotic surgery, and drug discovery [11].

Despite their impressive capabilities and growing prevalence, deep neural networks pose significant challenges, particularly when it comes to understanding their decision-making logic. These models are often referred to as "black boxes" because, while they can make highly accurate predictions, their internal workings that lead to these predictions are not easily interpretable or transparent. The intricacy of neural networks with millions of parameters makes them essentially opaque, even to experts [12]. The black-box problem arises from the complex, nonlinear nature of these models. For example, a neural network might consist of hundreds of layers, each containing numerous neurons with different weights and biases. The interactions between these layers and neurons create a highly intricate web of computations that is not human interpretable. This complexity, while contributing to the model's predictive power, makes it challenging to explain why the model made

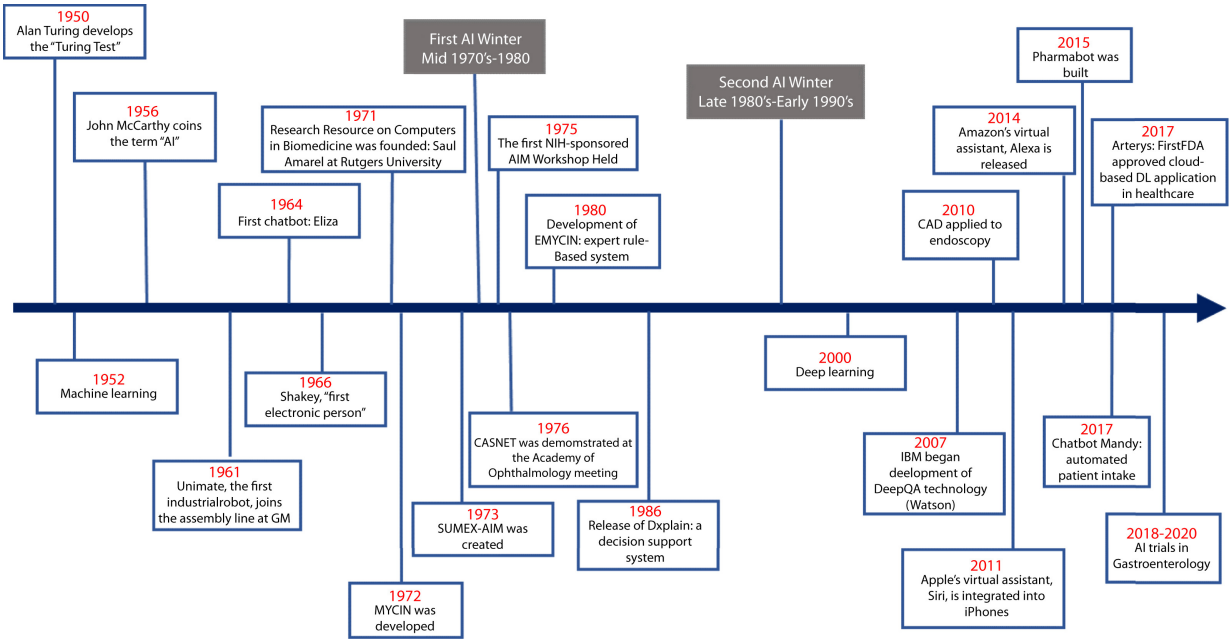


Figure 1.1: Timeline of the evolution of artificial intelligence systems with a focus on applications in healthcare. Key: AI - Artificial Intelligence; DL - Deep Learning; FDA - U.S. Food and Drug Administration; CAD - Computer-Aided Diagnosis. Reprinted from [1] with permission from Elsevier.

a particular decision.

The opacity of these models poses significant challenges for trust, ethics, and accountability. If the reasoning behind a model's outputs and predictions cannot be understood, it becomes difficult to verify that the model is making decisions based on fair, unbiased logic rather than using problematic shortcuts or proxies. This poses a significant hurdle in the broader adoption of these AI systems in healthcare, where decisions can have life-altering consequences. The use of black box models can lead to erroneous predictions with severe consequences.

In their 2019 study, Obermeyer *et al.* showed that an AI algorithm used to guide healthcare decisions was less likely to recommend additional medical care for black patients than for white patients with the same health conditions [13]. Further investigation revealed that the algorithm relied on healthcare costs as a proxy for health needs. Because less money is spent on healthcare for black patients in the U.S., the algorithm incorrectly inferred that they were healthier than equally sick white patients. A different study revealed that the classifiers used for computer-aided diagnosis trained on medical imaging datasets performed poorly for underrepresented genders [14]. This

problem was consistent across various network architectures and datasets, underscoring the need for data diversity and model fairness.

Röösli *et al.* also noted performance disparities in the MIMIC benchmarking model. The model demonstrated less accurate predictions for Black patients and those on Medicaid as compared to individuals with private insurance. Furthermore, the model tended to underestimate risk for Medicare patients and those with a higher number of comorbidities, suggesting possible inequities [15]. Another study demonstrated that an image recognition based AI system designed for cancer treatment recommendations was unable to replicate its performance across various healthcare settings. This study underscores the risks and challenges associated with adapting such systems across diverse environments. It cautions against using these models as black boxes without understanding their limitations and recognizing the importance of region-specific data and inclusive training [16].

The risk of AI systems making errors in healthcare, such as recommending the wrong medication or misdiagnosing a condition, is highlighted in [17]. The authors note that reactions to errors made by AI may differ from those made by humans, and the widespread use of an erroneous AI system could potentially lead to injuries in a large number of patients, unlike errors made by individual healthcare providers.

Black box models in healthcare also present a threat to individual autonomy by obstructing meaningful patient involvement in decision-making processes. In order for patients to exercise their agency, they need to understand the processes and potential outcomes of AI recommendations [18]. The opacity of these models impedes shared decision-making, a crucial aspect of ethical healthcare. Without grasping the underlying logic, patients cannot ensure that algorithms are aligned with their values. In extreme cases, this could even compromise a patient's confidence to refuse treatment. Hence relying on inscrutable AI compromises patients' ability to understand the forces influencing them.

Moreover, healthcare decisions often need to be justified to patients, their families, and in some cases, to legal entities. If a decision made based on a black box model leads to an adverse outcome, it could result in legal issues. The inability of these models to explain their decision making logic,

makes it challenging to assign liability [19, 20]. Holzinger and others propose that explainability could be the solution to this problem [21]. If healthcare professionals are provided with human interpretable explanations of the models logic, then the model becomes similar to other diagnostic tools already in use.

Safety challenges of machine learning systems in healthcare, particularly their black box nature, are also discussed in [22]. The authors argue that while certain aspects of AI systems, like design decisions and training activities, can be examined and explained, the precise workings of an algorithm often remain inscrutable. They suggest that safety governance of AI in healthcare will require frameworks that can explain broader sociotechnical processes, not just the underlying mechanics of the algorithms. However, they note that the inherent inscrutability of some machine learning approaches may make them unsuitable for safety-critical applications.

1.2 The Growing Drive for Explainable AI

In light of the aforementioned risks, there is a growing global demand for explainability in AI to enable oversight and accountability as these transformative technologies become increasingly integrated across healthcare, government, industry, and other domains. Efforts to promote explainable AI have gained momentum across sectors in recent years.

In 2019, the Organisation for Economic Co-operation and Development (OECD) AI Principles endorsed by over 40 countries emphasized the need for trustworthy AI systems. These principles highlight transparency and explainability as key enablers of trust in AI systems to support their widespread diffusion and adoption [23]. The National Institute of Standards and Technology (NIST) in the United States is at the forefront of developing standards, metrics, benchmarks, and tools to address explainable AI as a core component of trustworthy AI. NIST held a virtual workshop on Explainable AI in 2021, bringing together stakeholders from industry, academia, and government to discuss technical needs, challenges, and collaborative opportunities related to explainable AI [24]. The National AI Initiative Act signed into law in January 2021 underscores explainability as an important research priority, calling for coordination across the Federal government to accelerate advances in AI [25]. The Defense Advanced Research Projects Agency's (DARPA) Explainable

AI (XAI) program aims to create a suite of machine learning techniques that yield more explainable models without sacrificing learning performance. A core goal is enabling human users to understand, trust, and manage AI partners through model explainability [26].

On the regulatory front, there have been various initiatives to codify requirements for explainable AI. In the United States, the proposed federal Algorithmic Accountability Act would require companies to conduct assessments of high-risk automated systems for biases and discrimination potentials. It also mandates that companies take corrective actions based on the assessments. The capacity to explain algorithmic decisions in a meaningful way to affected individuals would be important for compliance [27]. The Federal Trade Commission (FTC) has noted that explainability is critical for evaluating important AI properties like fairness, as opaque models preclude assessing underlying biases [28]. In the United Kingdom, guidelines from the Information Commissioner's Office (ICO) state that organizations must be able to explain the decisions, predictions, or recommendations produced by AI systems to affected individuals upon request in non-technical language [29]. The European Union's General Data Protection Regulation (GDPR) has frequently been referenced as establishing a broad "right to explanation" for citizens subject to algorithmic decisions [30]. The EU is working to impose transparency requirements on high-risk AI systems under proposed regulations like the Artificial Intelligence Act [31].

For AI in healthcare, the Food and Drug Administration (FDA) has proposed guiding principles for good machine learning practice in medical device development [32]. The FDA principles stress that interpretability of outputs is critical for the usability and safety of machine learning-based devices and techniques, particularly those involving collaboration between humans and AI algorithms. Interpretability can be promoted through using visualizations to explain the model's predictions, generating natural language rationales, or other strategies to make the model more understandable for users.

Consequently, a primary focus of ongoing research in AI is to develop methods that can enhance the interpretability of these black box models. The goal is to demystify the black box, making the decision-making process of these powerful tools more comprehensible and accountable, thereby

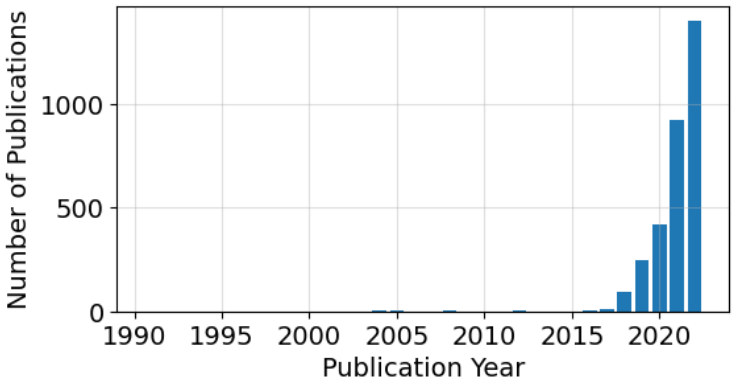


Figure 1.2: Plot of the exponential growth in published research related to 'Explainable AI' over the past decade. This surge in research activity is partially fueled by various international and national incentives and regulatory frameworks. The data for this analysis was sourced from the Web of Science.

addressing the ethical, legal, trust, and safety issues that currently limit their broader use.

1.3 Lexicon of Explainable AI

As evident by the global push for AI regulation, the ideas of interpretability and explainability have gained significant attention. However, the nomenclature and vocabulary used in this area of research can often be confusing due to the interchangeable use of terms. In this section, we give a brief overview to provide clarity on the key terms and concepts used in the scientific literature on interpretable and explainable AI.

Interpretable AI refers to models whose predictions can be readily understood by humans [33]. The term *Intrinsic or Inherent Interpretability* is often associated with simpler models, such as linear regression or decision trees, where the relationship between input and output is known and can be easily understood. These models have clear and explicit rules that relate input features to output predictions.

On the other hand, *Explainable AI* (XAI) is a broader concept that not only includes interpretability but also the techniques and methods that can provide clear, understandable explanations for the decisions made by black box AI models. *Post-hoc Explanations* are methods used to explain model outputs after the model has been trained, without modifying the model itself. *Local Explanations* focus on explaining individual predictions, allowing users to understand why a particular

instance was classified or predicted in a certain way. On the other hand, *global explanations* aim to provide a comprehensive understanding of the overall functioning and logic of the entire model.

Another term that often appears in the literature is *transparency*. In the context of AI, transparency refers to the openness and clarity of an AI system's operations. A transparent AI system is one where all aspects, including the data used for training, the learning algorithm, and the decision-making process, are open and accessible. Transparent models are also referred to as *white-box models* or *glass-box models*. These models provide visibility into their decision-making process, allowing humans to comprehend the factors that influence their predictions.

Lastly, *fairness* and *trust* are key concepts in AI research that are closely tied to interpretability and explainability. Fairness refers to the ability of an AI system to make decisions without bias or discrimination whereas trust in AI refers to the confidence users have in an AI system's reliability and integrity.

1.4 Illuminating the Black Box: Strategies in Explainable AI

In the early days of AI between 1980s-90s, symbolic or rule-based systems dominated, offering transparency and interpretability in their outputs [34]. As machine learning technologies advanced in the 2000s, popular approaches focused on intrinsic interpretability. Strategies developed in this era included sparse linear models, rule-based, and tree-based models. The rise of deep learning in the 2010s, however, introduced high performing opaque models, necessitating the emergence of "Explainable AI" (XAI) to interpret these black-box systems [35]. To illustrate these concepts, Figure 1.3 presents a currently accepted relationship that model performance is often inversely related to interpretability.

The domain of explainable AI encompasses a variety of methodologies, each differing based on specific functional attributes. To organize these diverse concepts, scholars have proposed various taxonomies. Figure 1.4 illustrates a recognized categorization scheme for XAI methods [3].

Popular XAI approaches consist of post-hoc interpretation which can vary from simple visualization methods like partial dependence plots [36] to feature importance or feature attribution methods that quantify the contribution of input features to the model's output. Gradient-based

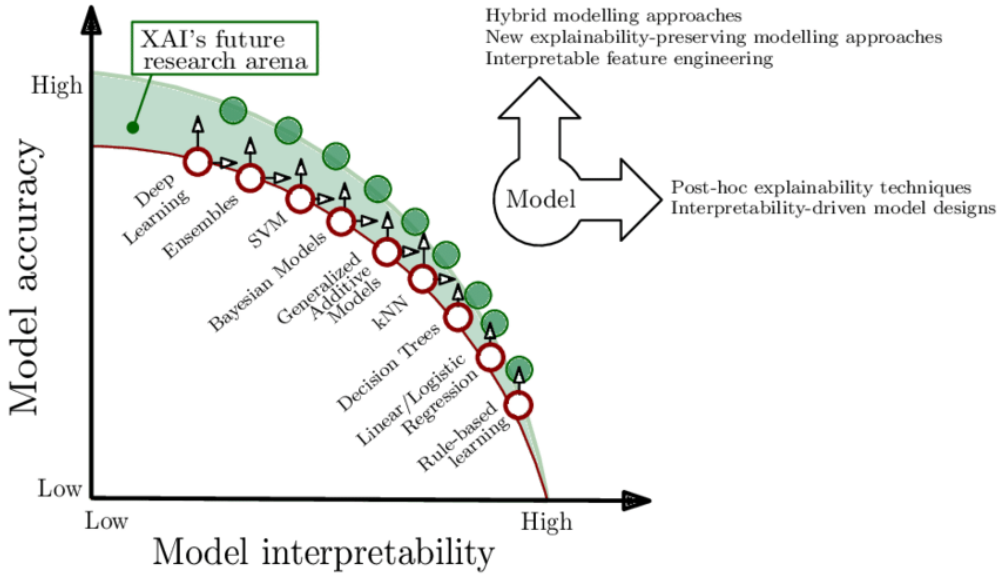


Figure 1.3: The trade-off between model interpretability and performance is illustrated, showing that highly interpretable models like linear regression tend to have lower performance while high-performing opaque models like deep neural networks have low interpretability. Explainable AI methods and tools have promise to increase the interpretability of high-performing opaque models without significantly sacrificing their performance. Reprinted from [2] with permission from Elsevier.

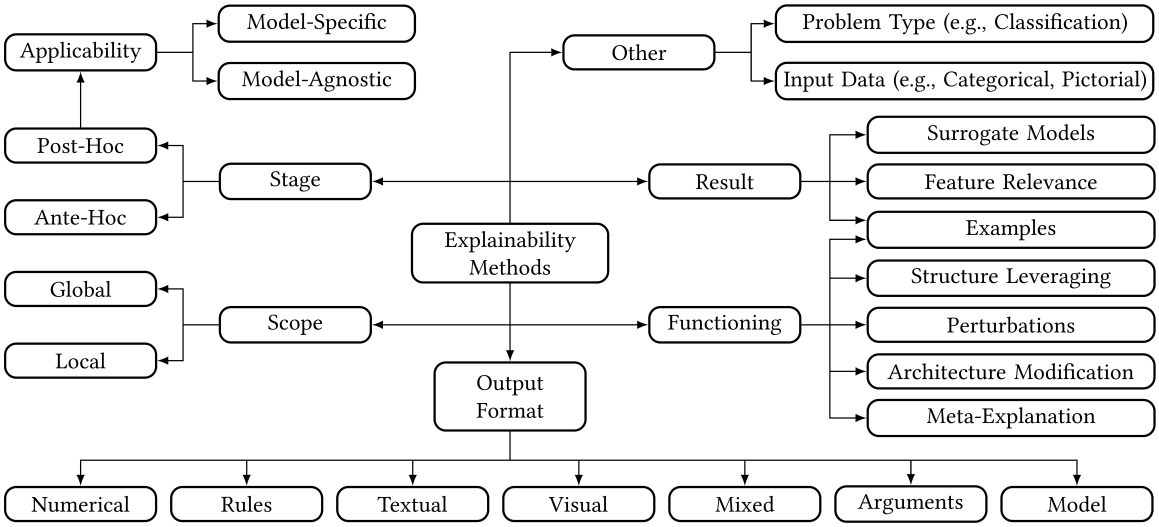


Figure 1.4: Taxonomy of XAI methods combining the conceptual, functioning, and result approaches. The conceptual dimensions of stage, scope, and applicability form the upper levels. The functioning and result of methods are added as dimensions on the lower level. Additional dimensions like output format can be incorporated. Categories are not assumed to be mutually exclusive. Used under CC 4.0 from [3].

methods are also used to identify important input regions. Saliency maps using gradients like Grad-CAM are typically used for interpreting image based convolutional neural networks [37].

Approaches like occlusion analysis and counterfactual explanations manipulate the inputs and the model itself to facilitate a deeper understanding of their functioning. Occlusion analysis methodically masks parts of the input to determine their importance [38]. On the other hand, counterfactual explanations offer insights into model predictions by identifying the minimal change to the input data that would lead to a change in the outcome [39, 40].

Decomposition approaches explain models in terms of their individual components. Methods like testing with concept activation vectors (TCAV) identify concepts that are highly relevant to a prediction [41]. Concept activation vectors link internal neural representations to human-interpretable concepts.

Example-based techniques explain predictions by retrieving similar instances from training data [42]. These include prototype-based methods like influence functions [43] and activation minimization [44]. Influence functions identify the training samples that contribute the most towards predicting a given test sample. Activation minimization identifies examples that strongly activate the function of interest. Other prototype-based methods can learn characteristic prototypes for different classes and are able to identify to the closest training input with those prototypes as explanations at inference time [45, 46].

Model induction involves generating an entirely new model that approximates the behavior of a black-box model. This new model is typically a simpler, interpretable model such as a decision tree, rule set, or linear model. Model induction creates this new model by learning directly from the inputs and outputs of the black-box model. Thus, the new model serves as an approximation or surrogate of the black-box model, allowing researchers to study its decision-making process more readily than they could with the original black-box model.

Similarly, knowledge distillation can also be used as an explainability method. It involves transferring knowledge from a larger, more complex teacher network to a smaller, less intricate student network [47, 48]. The primary objective is to manage the size and complexity of large-

scale neural networks but it can also be used to impart 'distilled' knowledge to a more interpretable model.

Other approaches focus solely on the development of inherently interpretable rule based and symbolic models [49]. Symbolic reasoning emphasizes explicit representation of knowledge using symbolic rules or logical formulas. Recent development of neuro-symbolic AI offer the best of both worlds: the performance of deep learning models and the explainability of symbolic AI. Neuro-symbolic models use symbolic AI for high-level reasoning and neural networks for low-level perception tasks [50].

In medicine analytical XAI approaches, such as predefined kinetic and linear models, feature extraction via correlations and clustering, and sensitivity analysis through perturbations, are also gaining popularity. These models can reveal patterns in genetics and neuroimaging data. Signal inversion methods are less explored despite their potential to probe neural mechanisms [51]. Verbal rule-based systems have also shown promise for interpretable medical predictions like pneumonia risk models [52].

Despite the recent advances in XAI, the black-box problem in deep learning remains unsolved. One of the reasons behind this is that model understanding is subjective and, therefore, difficult to formalize [53]. Furthermore, the insights needed from a model to make it transparent are domain-specific [54]. Consequently, it is challenging to design universal methods for model transparency. Emerging directions include building standardized benchmarks and rigorously evaluating explanations [55]. Improving human-centered design and explanations for non-experts are also crucial areas [56]. Developing theories and formal grounding for interpretability remains an open challenge [57]. This thesis is contributing to XAI research through the presentation of methods offering explanations for black-box models that input multiple modalities and output medical decisions.

CHAPTER 2

METHODS FOR POST-HOC FEATURE IMPORTANCE

As mentioned earlier, predictions alone are not enough for medical applications. The model must also provide some insight into the prediction generation process. In particular, it is often necessary to understand the model in order to perform debugging, bias detection, and failure analysis. Furthermore, insights into the model help the user assess if, when, and how much to trust model predictions. This is a crucial requirement for deploying these models in the real world.

Feature importance or feature attribution is a widely used and well-studied explainability technique [58, 59, 60]. The term *feature* in explainable AI research refers to a measurable property of the model input. Understanding the contribution of an input feature towards a particular decision builds trust with users and can lead to novel scientific discoveries.

Based on their interaction with the predictive models, feature importance methods are classified into filter, wrapper and embedded methods [61]. Filter methods use input data only and are mostly applied as a preprocessing step before training the predictive model. Examples include similarity-based methods, correlation criteria, mutual information, clustering, principal component analysis, and linear discriminant analysis [62]. Wrapper methods such as permutation methods, local model approximations [63], and some gradient-based methods [64] are model agnostic but use model predictions for ranking features. Embedded methods require intricate manipulation of the model. In some direct-objective-optimization-based methods feature ranks are learned in addition to the model parameters [65]. Some methods propagate the feature relevance layer-wise through the DNN [66], while others use special network structures like bijection-layers [67] or self-attention layers [68] to rank features. Table 2.1 provides a comprehensive overview of existing feature attribution methods.

While filter, wrapper, and embedded-based feature importance methods each have their advantages and shortcomings, wrapper methods are often preferred due to their model-agnostic nature. Unlike other methods, they only examine model input and output, making them suitable for a wide variety of model architectures. In this context, we focus on exploring several popular and

Table 2.1: Overview of feature importance and attribution methods in explainable AI. The categorization of methods can vary with implementation. The data types listed are the most common ones for each method, some of these methods can be used with other data types with modifications.

Method	**Data**	**Category**	**References**
Permutation Importance	Tabular, Time Series	Filter	Breiman, L. (2001) [69]
Partial Dependence Plots (PDP)	Tabular, Time Series	Filter	Friedman, J. H. (2001) [36]
Saliency Maps	Image	Embedded	Simonyan et al. (2013) [70]
Occlusion	Text, Image	Wrapper	Zeiler, Fergus (2014) [38]
Layer-wise Relevance Propagation (LRP)	Tabular, Text, Image	Embedded	Bach et al. (2015) [66]
Guided Backpropagation	Image	Embedded	Springenberg et al. (2015) [71]
Input Gradients	Tabular, Text, Image	Wrapper	Hechtlinger (2016) [64]
LIME (Local Interpretable Model-agnostic Explanations)	Tabular, Text, Image	Wrapper	Ribeiro et al. (2016) [63]
Grad-CAM	Image	Embedded	Selvaraju et al. (2016) [37]
Quantitative Input Influence (QII)	Tabular, Text, Image	Wrapper	Datta et al. (2016) [72]
SHAP (SHapley Additive exPlanations)	Tabular, Text, Image	Wrapper	Lundberg, Lee (2017) [60]
Integrated Gradients	Tabular, Text, Image	Embedded	Sundararajan et al. (2017) [73]
SmoothGrad	Tabular, Text, Image	Wrapper	Smilkov et al. (2017) [74]
DeepLIFT	Tabular, Text, Image	Wrapper	Shrikumar et al. (2017) [58]
Influence Functions	Tabular	Wrapper	Koh, Liang (2017) [75]
Extremal Perturbations	Text, Image	Wrapper	Fong, Vedaldi (2017) [76]
Contextual Decomposition	Text	Wrapper	Murdoch, Szlam (2017) [77]
Contrastive Explanations Method (CEM)	Tabular, Image	Wrapper	Dhurandhar et al. (2018) [78]
Anchors	Tabular	Filter	Ribeiro et al. (2018) [79]
Model Agnostic suPervised Local Explanations (MAPLE)	Tabular	Wrapper	Plumb et al. (2018) [80]

effective wrapper-based methods. These methods leverage slightly different definitions of feature importance. To effectively describe these methods, we first establish a comprehensive framework. Consider a binary classification problem:

$$F\colon X \in \mathbb{R}^d \longrightarrow y \in \mathbb{R}, \tag{2.1}$$

where F represents the classifier, X is an input sample with d features and y is the predicted probability for $class_1$ [1], where $X = \begin{bmatrix} x_1 & \cdots & x_d \end{bmatrix}^T$, and for all $i = 1, \ldots, d$, $x_i \in \mathbb{R}$ is a feature.

Note that no assumptions are made about the structure of classifier F. All methods discussed below are model-agnostic and F represents an arbitrary binary classifier. These methods can also be adopted for multi-class classification with minor modifications.

2.1 Permutation Feature Importance

The primary goal in classification problems is maximizing the performance of the classifier, which is typically quantified by some score. Therefore, a natural way to estimate feature importance is to study the effect a feature has on a classifier score. One such method is the permutation-based feature importance [69]. It defines the importance of the kth feature as the average decrease in classifier score as the kth feature is permuted m times.

For tabular data, let $D_X = \begin{bmatrix} X_1 & \cdots & X_n \end{bmatrix}^T \in \mathbb{R}^{n \times d}$ be the data matrix with n, d dimensional samples. Rows of D_X represent a new sample and columns represent a feature. Let $\text{Score}(D_X)$ be the average classification performance score of the classifier F on data D_X. The importance of kth feature is given by

$$\text{Permutation Importance} \qquad \text{PERM imp}(k) = \text{Score}(D_X) - \frac{1}{M}\sum_{i=1}^{M} \text{Score}(D_{X(k,i)}), \tag{2.2}$$

where $\text{Score}(D_{X(k,i)})$ is the performance score on the ith permutation of D_X. In each permutation, values in the kth column of D_X are randomly shuffled. The permutation algorithm is detailed in Algorithm 2.1.

[1] Probability of $class_2 = 1 -$ Probability of $class_1$

Algorithm 2.1 Permutation Importance

Input: Trained model (F), Training set (D_X), Number of permutations (M).
Calculate the reference performance Score(D_X) on original data.
for each input feature k **do**
  for $i \leftarrow 1$ to M **do**
    Randomly shuffle values of the kth feature in D_X.
    Calculate model performance Score($D_{X(k,i)}$) on shuffled data.
  end for
  Calculate the importance as average drop in score for permutation in kth feature (2.2).
end for
Scale importances using l_1 normalization.
return Normalized permutation importance.

2.2 Gradient Feature Importance

Studying the impact of a change in input feature on the predicted output can provide insights about feature importance. Hechtlinger [64] uses the gradient to quantify such an impact. The absolute value of the gradient indicates magnitude change of the predicted output for an infinitesimal change in the input feature. The gradient of F with respect to input X is

$$\nabla F(x) = \left[\frac{\partial F(x)}{\partial x_1} \cdots \frac{\partial F(x)}{\partial x_d} \right]^T. \tag{2.3}$$

This method restricts the choice of model F to differentiable classifiers only. The differentiability of deep neural networks depends on the choice of activation function. Common activation functions like sigmoid, Relu, and Tanh are differentiable almost everywhere [2].

We use a central difference approach to numerically approximate the gradient of F at X and define

$$\frac{\partial F(X)}{\partial x_k} = \frac{F(X^{(k+)}) - F(X^{(k-)})}{2\delta x}, \tag{2.4}$$

where

$$X^{(k+)} = X + \delta x \cdot e_k, \quad X^{(k-)} = X - \delta x \cdot e_k, \tag{2.5}$$

$\delta x \in \mathbb{R}$ is the step size, and for all $k = 1, \ldots, d$, $e_k \in \mathbb{R}^d$ is the standard basis vector. The terms $F(X^{(k+)})$ and $F(X^{(k-)})$ are obtained from two forward passes of the model.

[2] differentiable everywhere except on a set of measure zero

The importance of the kth feature is then defined as the absolute value of the partial derivative with respect to x_k. The sample feature importance for a single test sample using the gradient vector is given by

$$\text{GRAD imp}_S(X_i, k) = \left| \frac{\partial F(X)}{\partial x_k} \right|_{X_i}. \tag{2.6}$$

To get the global feature importance, we average all sample feature importances over a test set $\{X_i\}_i^n$. In particular, we define

$$\text{GRAD imp}_G(k) = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{\partial F(X)}{\partial x_k} \right|_{X_i}. \tag{2.7}$$

The input gradient importance algorithm is detailed in Algorithm 2.2.

Algorithm 2.2 Input Gradient Feature Importance

1: **Input:** Trained model (F), Training set D_X.
2: **for** each training sample X in D_X **do**
3: **for** each input feature k **do**
4: Compute the gradient of $F(X)$ w.r.t x_k (2.4).
5: **end for**
6: Take absolute value of gradient vector to get feature importance.
7: Scale importances using l_1 normalization.
8: **end for**
9: Average over all samples to get global gradient importance (2.7).
10: **return** Local and global input gradient feature importance.

2.3 Locally Interpretable Model Agnostic Explanations (LIME)

Another way to get feature importance is to use locally interpretable surrogate models. Linear models are a good choice for surrogate models due to their interpretability and low complexity [63]. For the classifier in (2.1) we can build locally linear surrogate models G. For all $X_i \in \{X_i\}_{i=1}^n$, we can construct

$$G_i(X) : X \in \mathcal{N}(X_i) \longrightarrow y \in \mathbb{R}, \tag{2.8}$$

where $\mathcal{N}(X_i)$ is a set containing samples in the neighborhood of X_i. To populate $\mathcal{N}(X_i)$, we sample from a Gaussian distribution centered at X_i. The surrogate model has the form

$$G_i(X) = W_i^T X, \tag{2.9}$$

where $W_i = \begin{bmatrix} w_{i,1} \cdots w_{i,d} \end{bmatrix}^T$. To ensure that G_i approximates the actual model F centered at X_i, we learn the weights W_i by minimising the weighted least squares loss function

$$\mathcal{L}(W_i) = \sum_{X \in \mathcal{N}(X_i)} e^{-\|X - X_i\|^2} \|F(X) - W_i^T X\|^2. \tag{2.10}$$

For the ith test sample, the importance of kth feature is the absolute value of its corresponding weight defined as

$$\text{LIME imp}_{\text{S}}(X_i, k) = |w_{i,k}|. \tag{2.11}$$

In order to estimate global feature importance using local surrogate models we propose an extension to the LIME. The global feature importance of the kth feature is defined as the average of the local importances estimated by each surrogate model given by

$$\text{LIME imp}_{\text{G}}(k) = \frac{1}{n} \sum_{i=1}^{n} |w_{i,k}|, \tag{2.12}$$

where $w_{i,k}$ is the weight corresponding to the kth feature for the ith surrogate model. The LIME algorithm is detailed in Algorithm 2.3.

Algorithm 2.3 LIME

Input: Trained model (F), Input sample (X), size of neighborhood set (M).
for each training sample X **do**
  Generate M random samples around X to get the neighborhood set $\mathcal{N}(X)$.
  for each sample X' in $\mathcal{N}(X)$ **do**
    Generate model prediction $F(X')$.
    Compute the distance $\|X - X'\|^2$.
  end for
  Fit a linear model G (2.9) using the predictions and weights from the previous step (2.10).
  Take absolute values of coefficients of G to get feature importance (2.11).
  Scale importances using l_1 normalization.
end for
Average over all samples to get global LIME importance (2.12).
return Local and global LIME feature importance.

2.4 SHapley Additive exPlanations (SHAP)

The Shapley value, a concept from cooperative game theory, forms the basis of SHAP, where the classification task is the 'game' and the input features are the 'players'. The Shapley value of

a feature is the average marginal contribution of that feature across all possible combinations of features. The Shapley value for a feature k can be calculated using the following equation:

$$\phi_k(v) = \sum_{C \subseteq \{1,\cdots,d\} \setminus \{k\}} \frac{|C|!(|d| - |C| - 1)!}{|d|!} [v(C \cup \{k\}) - v(C)], \tag{2.13}$$

where d is the total number of features, C is a subset of features with $|C|$ features. $v(C \cup \{k\})$ is the value function for the (coalition) subset C including feature k, $v(C)$ is the prediction for features present in set C marginalized over features that are not included in set C and is given by

$$v_x(C) = \int F(X) d\mathbb{P}_{x \notin C} - E_X(F(X)). \tag{2.14}$$

The exact computation of Shapley values can be computationally expensive as it involves summing over all possible subsets of features and computing the value function. This becomes infeasible for a large number of features. To overcome this, a Monte Carlo estimation of Shapley values can be used [81], which involves randomly sampling permutations of the features. The Monte Carlo estimation of the Shapley value for a feature k can be calculated by using

$$\phi_k(X) = \frac{1}{M} \sum_{z \in Z'} F(h_x(z_{+k})) - F(h_x(z_{-k})), \tag{2.15}$$

where $M = |Z'|$ is the number of sampled combinations. Z' represents a subset of all possible combinations of features, the prime symbol is used to denote that these are not the actual feature values, but a binary vector indicating the presence or absence of a feature in a particular set.

Each element z of Z' is an instance with a subset of its features missing, z_{-k} is constructed from z by setting the kth indicator off ($z_k = 0$) and z_{+k} is constructed from z by setting the kth indicator on ($z_k = 1$).

The mapping function $h_x(z)$ is used to create a synthetic instance by replacing the values in z with the corresponding feature values from the original instance X, defined as

$$h_x(z) = \begin{cases} X_i & \text{if } z_i = 1 \text{ for } i = 1, \cdots, d, \\ 0 \text{ or the mean of feature } i & \text{if } z_i = 0 \text{ for } i = 1, \cdots, d. \end{cases} \tag{2.16}$$

As outlined in algorithm 2.4, the resulting instance $h_x(z)$ is then fed into the machine learning model F to get the prediction. The difference between the feature-present predictions $F(h_x(z_{+k}))$ and the feature-absent predictions $F(h_x(z_{-k}))$ is then used to compute the SHAP value for the kth feature.

Algorithm 2.4 SHAP

Input: Model (F), Training set D_X, Instance X, Number of Monte Carlo samples M
for each input feature k **do**
  Sample M random combinations from all possible combinations of features.
  for z in Z' **do**
    Generate synthetic instance $h_x(z)$ by replacing missing values with expected values from D_X (2.16).
    Original value for kth feature $h_x(z_{+k})$.
    Masked value for kth feature $h_x(z_{-k})$.
    Compute marginal contribution of kth feature in this coalition $F(h_x(z_{+k})) - F(h_x(z_{-k}))$
  end for
  Average over set Z' to get SHAP importance (2.15).
end for
Scale importances using l_1 normalization.
return Normalized SHAP importance.

CHAPTER 3

QUANTIFYING THE BENEFIT OF USING PATIENT-SPECIFIC BLOOD FLOW FOR ASSESSMENT OF CORONARY ARTERY DISEASE RISK

The work in this chapter contributed towards the following:

- M. Azmat, K. Branch, and A. Alessio. '*Virtual Clinical Trial to Evaluate the Benefit of Patient-Specific Blood Flow in CT Assessment of Functional Significance of Coronary Artery Stenosis*'. Presented at BMES Annual Meeting 2020.

- M. Azmat, E. Tu, K. Branch, and A. Alessio. '*Machine Learned Versus Analytical Models for Estimation of Fractional Flow Reserve from CT-Derived Information*'. Presented at SPIE Medical Imaging Conference, 2021.

3.1 Introduction

Coronary artery disease (CAD) is a highly prevalent health condition that poses a significant burden on global health, leading to considerable morbidity and mortality. Traditionally, stress tests and nuclear imaging techniques, such as exercise stress tests, pharmacological stress tests, myocardial perfusion imaging, and single-photon emission computed tomography (SPECT), have been commonly used to assess ischemia, which is directly related to CAD risk. This chapter proposes models to detect functionally significant ischemia, suggesting high-risk CAD; For simplicity, we will use the term "CAD risk" to denote high-risk ischemic, flow-limiting, disease.

However, fractional flow reserve (FFR) is increasingly being used to assess the risk of CAD. FFR is defined as the ratio of the pressure before and after a stenosis as measured during coronary catheterization [82]. This invasive procedure provides a direct and reliable measure of the hemodynamic impact of a coronary stenosis, and it has been shown to avoid unnecessary interventions and improve patient outcomes when used to guide revascularization decisions [83].

Traditionally, FFR is measured invasively during coronary catheterization, which, while accurate, carries inherent risks, costs and recovery time. Therefore, there has been a growing interest in developing non-invasive methods for estimating FFR. One promising approach in this regard is

the use of cardiac computed tomography (CT) imaging combined with data-driven or analytical models.

FFR estimation models use CT angiography to derive patient-specific stenosis geometry, a key determinant of the pressure drop across the stenosis and, consequently, the FFR value [84]. Typically, the FFR estimation methods then either apply computational fluid dynamics (CFD) simulations or machine-learned models to estimate FFR for the patient-specific stenosis geometry [85]. However, most of these models rely on estimated normal values for flow variables, which may not accurately reflect the patient-specific hemodynamic conditions. This limitation could potentially affect the accuracy of FFR estimation and, consequently, the assessment of CAD risk.

We hypothesize that incorporating patient-specific values for hemodynamic parameters, specifically patient-specific blood flow, could improve the estimation accuracy of non-invasive FFR models. patient-specific blood flow can be particularly valuable in patients with microvascular dysfunction, a condition that can affect the blood flow in the coronary arteries and is not captured by traditional FFR measurements [86]. To test this hypothesis, we perform a comparative study where we compare the estimation accuracy of a variety of FFR estimation models against known and true FFR values. We construct models with varying levels of patient-specific information, both with and without patient-specific blood flow information.

However, obtaining patient-specific blood flow information requires additional imaging studies, specifically CT perfusion imaging. CT perfusion imaging provides detailed information about the blood flow in the coronary arteries, which can be used to estimate the patient-specific flow rate. Since the additional imaging carries radiation and monetary costs, it is crucial to quantify the added benefit of using patient-specific blood flow relative to other inputs. This quantification can help justify the use of additional imaging studies and guide the development of cost-effective strategies for non-invasive FFR estimation. To quantify the relative importance of using patient-specific blood flow, we use a machine learning based FFR estimator and perform feature importance analysis using a variety of feature importance estimation methods.

3.2 Methods

To assess the benefit of using patient-specific blood flow for assessment of CAD risk, we set up a virtual clinical trial. We simulated a section of the left anterior descending (LAD) artery with stenosis for 60 reference patients with varying rates of arterial blood flow at stress and different stenosis geometries. We then constructed three different FFR estimation models relying on different levels of patient-specific information: 1) FFR_G was the most primitive model which relies on just the geometric data, 2) FFR_N predicted FFR using normal values for flow parameters in Navier-Stokes equations, and 3) FFR_P used patient-specific values for flow parameters in Navier-Stokes equations to estimate FFR. The ground truth values of fractional flow reserve FFR_{GT} were calculated using high fidelity computational fluid dynamics simulations. For the analytical FFR

Table 3.1: Description of the FFR estimation models used in the comparative study.

Symbol	Name	Description
FFR_{GT}	Ground truth	3D computational fluid dynamics simulation.
FFR_G	Geometric only	Model relying only on the geometry of the stenosis.
FFR_N	Normal flow	Flow-based model using a constant normal blood flow rate.
FFR_P	Patient-specific flow	Flow-based model using patient-specific blood flow rates.

models, we approximated the stenosis geometry with a blunt-plug in a constant diameter artery as shown in Figure 3.1. Table 3.2 lists the parameters that define the patient stenosed LAD model.

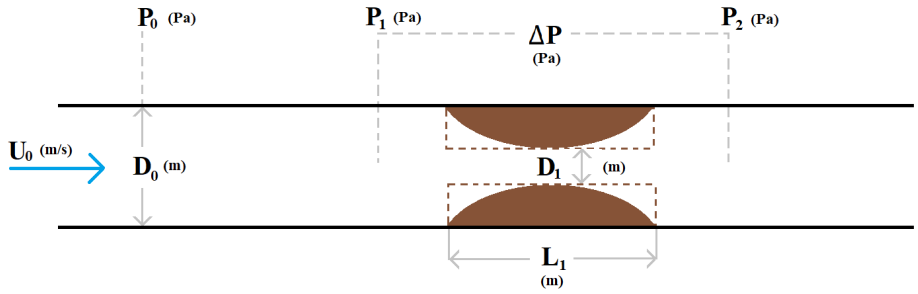


Figure 3.1: Model of a single blunt-plug stenosis in the artery.

3.2.1 Geometric Model (FFR_G)

The geometry-only model relied solely on the stenosis geometry to get a rough estimate of the FFR across the stenosis. FFR_G was calculated as the ratio of maximum stenosed artery diameter

Table 3.2: Geometric and flow parameters used in analytical FFR estimation models.

Symbol	Description	Units
U_0	Average velocity in unobstructed artery	m/s
P_0	Upstream flow pressure in unobstructed artery	Pa
D_0	Diameter of unobstructed artery	m
L_0	Length of upstream unobstructed artery	m
P_1	Flow pressure proximal to the stenosis	Pa
D_1	Minimum stenosis diameter	m
L_1	Length of stenosis	m
P_2	Flow pressure distal to the stenosis	Pa

to unobstructed artery diameter given by

$$FFR_G \triangleq \frac{D_1}{D_0}. \tag{3.1}$$

3.2.2 Flow-Based Model (FFR_N, FFR_P)

The flow-based FFR estimation models were modeled using simplified Navier-Stokes equation for an incompressible Newtonian blood flow in a rigid artery of constant circular cross-section as illustrated in Figure 3.1. The flow was assumed to be fully developed and steady. The pressure drop ΔP across a single blunt-plug stenosis calculated as the sum of viscous and expansion losses in the flow [87] is given by

$$\frac{\Delta P}{\rho U_0^2} = \frac{K_v}{Re_0} + \frac{K_t}{2}\left(\frac{A_0}{A_1} - 1\right), \tag{3.2}$$

$$Re_0 = \frac{\rho U_0 D_0}{\mu}, \tag{3.3}$$

where ρ is the density of blood, μ is the dynamic viscosity of blood, K_v and K_t are viscous and expansion loss coefficients, respectively, Re_0 is the unobstructed Reynolds number, and A_0 and A_1 are the unobstructed and stenosed cross-sectional areas, respectively. The loss coefficients were computed using empirical relationships

$$K_v = 32\frac{0.83L_1 + 1.64D_1}{D_0}\left(\frac{A_0}{A_1}\right)^2, \tag{3.4}$$

$$K_t = 1.52, \tag{3.5}$$

given in [88]. Finally, the flow-informed FFR was defined as the ratio of distal pressure to proximal pressure

$$FFR_{N/P} \triangleq \frac{P_2}{P_1} = \frac{\Delta P + P_1}{P_1}, \tag{3.6}$$

$$FFR_{N/P} = \frac{1}{P_1}\left[\left(\frac{K_v}{Re_0} + \frac{K_t}{2}\left(\frac{A_0}{A_1} - 1\right)\right)\rho U_0^2 + P_1\right]. \tag{3.7}$$

In the case where ΔP was computed using patient-specific blood flow rate (U_{0P}), equation (3.7) returned FFR_P, and in the case where ΔP was computed using a constant normal blood flow rate (U_{0N}), equation (3.7) returned FFR_N. P_1 was calculated from a reference-normal-aortic pressure P_0 by using the Hagen-Poiseuille equation for fully developed flow

$$P_1 = P_0 - \frac{32\mu L_0 U_0}{D_0}. \tag{3.8}$$

3.2.3 Computational Fluid Dynamics Model (FFR_{GT})

To generate reference ground truth values of FFR, we relied on high-fidelity computational fluid dynamics simulations of Newtonian blood flow in a non-rigid artery. The simulation was conducted using Fluent software, version 20.1.0. in 3D space with a steady-state time setup.

3.2.3.1 Geometry

We restricted our analysis to stenosed sections of the LAD artery. For the sake of simplicity, we fixed the geometric parameters of the LAD artery ($D_0 = 4.6mm$) and varied the geometric parameters L_1, D_1 of the stenosis across different patients.

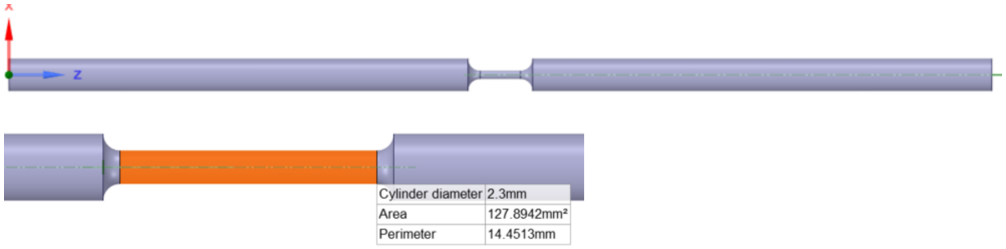


Figure 3.2: Illustration of the three-dimensional blunt-plug arterial stenosis modeled using Ansys.

3.2.3.2 Materials

The blood flow through the artery was assumed to be laminar, and structural and thermal considerations were not taken into account in this study. Material properties used for blood and arterial wall were taken from [89] and are detailed in Table 3.3.

Table 3.3: Material properties for blood and arterial walls used in the CFD simulation.

Property	**Value**		**Units**
	Blood	**Artery wall**	
Density	1050	1075	kg/m^3
Specific heat	3490	3490	K/kgK
Thermal conductivity	0.549	0.476	W/mK
Viscosity	0.0028	-	kg/ms
Molecular weight	28.966	-	$kg/kmol$

3.2.3.3 Solver Settings

The computational fluid dynamics simulation was conducted using Fluent version 20.1.0 to analyze blood flow behavior within a solid environment. The process involved solving the governing equations for fluid flow, and the absolute velocity formulation was activated in the numerical segment. The pressure-velocity coupling utilized the SIMPLE algorithm, and a V-Cycle solver was implemented for the pressure variable. Discretization involved a second-order scheme for pressure and a second-order upwind scheme for momentum equations.

The simulation assumed laminar flow and did not take into account heat transfer, solidification, melting, species transport, pollutants, and structural effects. Relaxation factors were applied to various simulation variables.

3.2.3.4 Boundary Conditions

Boundary conditions were assigned to different zones within the computational domain. The arterial wall was assigned a no-slip condition with the velocity and shear stresses set to zero at the wall. For inflow we specified the inlet velocity profile, based on a fully developed flow assumption. The inlet velocity at position (x, y) on the cross-sectional inlet plane was defined as

$$U(x, y) = U_{maxP}\left(1 - 4\frac{x^2 + y^2}{D_0^2}\right), \tag{3.9}$$

where U_{maxP} is the patient-specific maximum value of velocity in the cross section at the mid-line, and is calculated from patient-specific flow rate inputs as detailed in Table 3.4. The outflow boundary was a pressure outlet with a target mass flow rate. The target output mass flow rate was set equal to inlet mass flow rate given in Table 3.4.

Table 3.4: Range of simulated patient-specific values of blood flow through LAD artery.

	Aortal blood flow	**LAD blood flow**	**Mass flow**		
	Q_{Aorta}	$Q_{LAD} \approx Q_{Aorta}/3$	LAD	U_{0P}	U_{maxP}
	mL/min	m^3/s	kg/s	m/s	m/s
At rest	120.0	6.67e-07	7.07e-04	0.040	0.080
At stress 1	240.0	1.33e-06	1.41e-03	0.080	0.161
At stress 2	360.0	2.00e-06	2.12e-03	0.120	0.241
At stress 3	480.0	2.67e-06	2.83e-03	0.161	0.321

3.2.4 Comparative Analysis

We used the CFD solutions of blood flow in stenosed LAD arteries for 60 patients to obtain the reference FFR_{GT}. For assessment of CAD risk the patients were classified into high versus low risk of CAD by thersholding FFR_{GT} using

$$\text{CAD risk} = \begin{cases} \text{High risk}, & FFR < 0.8, \\ \text{Low risk}, & FFR \geq 0.8. \end{cases} \tag{3.10}$$

To conduct a comparative analysis of different analytical FFR estimation models, we generated a simulated population of 10,000 patients based on the original group of 60 patients by incorporating measurement noise into the patient-specific flow and geometry parameters. This was done to introduce realistic variations. The sampling was performed with prevalence weighting, ensuring that the resulting population reflected a realistic patient population with a prevalence of high-risk CAD set at 63%.

Subsequently, the three models, namely FFR_G, FFR_P, and FFR_N, estimated the FFR values for each sample. Next, these FFR values were used to predict risk of CAD by classifying the samples into high and low risk of CAD using (3.10).

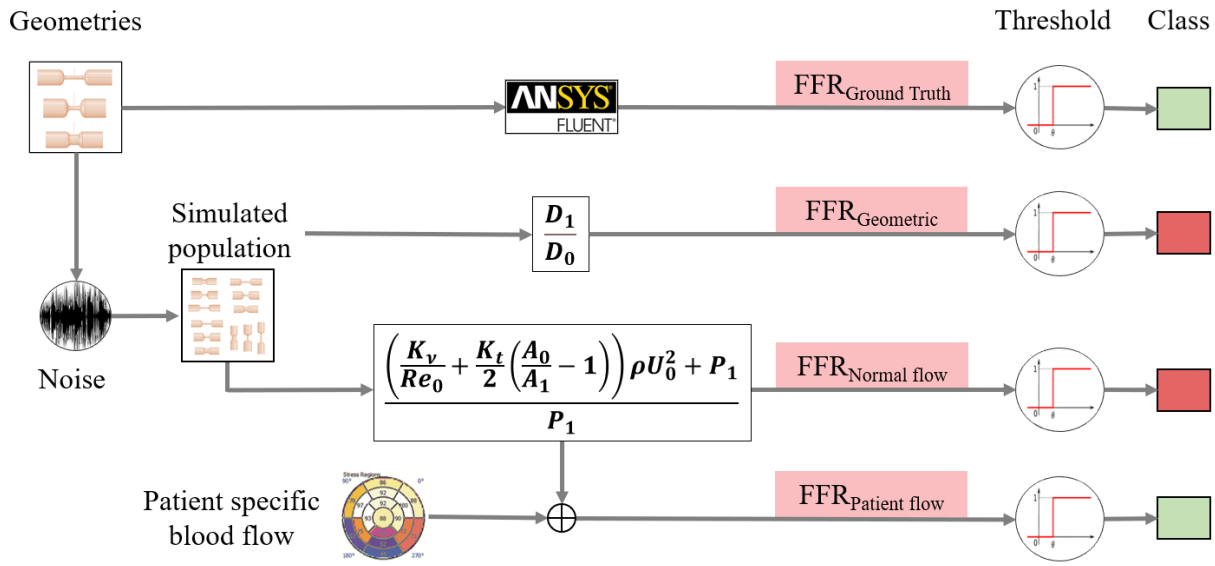


Figure 3.3: Schematic of the virtual clinical trial to quantify the added benefit of using patient-specific blood flow rate for CAD assessment.

By comparing the classification performance of these models, we aim to demonstrate the added benefit of incorporating patient-specific blood flow information in assessing CAD risk. Figure 3.3 shows the schematic of this comparative study.

3.3 Results

The computational fluid dynamics and the analytical model were used to simulate blood flow and estimate FFR across the stenosis in all simulated patients. Figure 3.4 shows the velocity profiles of the blood flow in the stenosed coronary artery after the convergence of the CFD solution. To estimate FFR, average static pressure along different axial cross-sections was plotted, which resulted in a static pressure profile along the center line of the stenosed artery, as depicted in Figure 3.5. This figure shows a significant drop in pressure, indicative of the presence of stenosis.

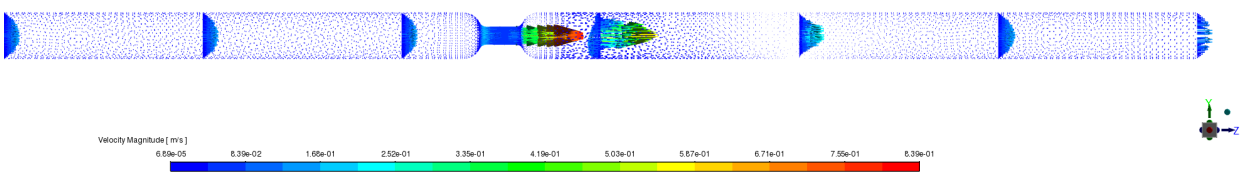


Figure 3.4: Velocity profiles of the blood flow along different cross sections in an unobstructed artery.

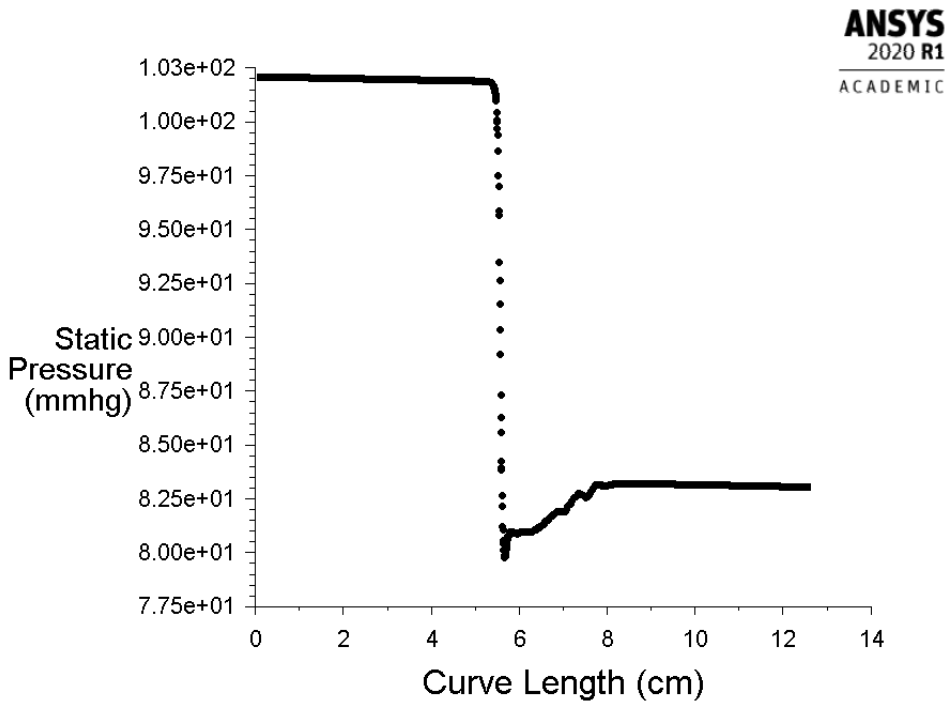


Figure 3.5: Mean static pressure values along the arterial centerline, as determined by the converged CFD solution. The term curve length refers to distance along the length of the artery.

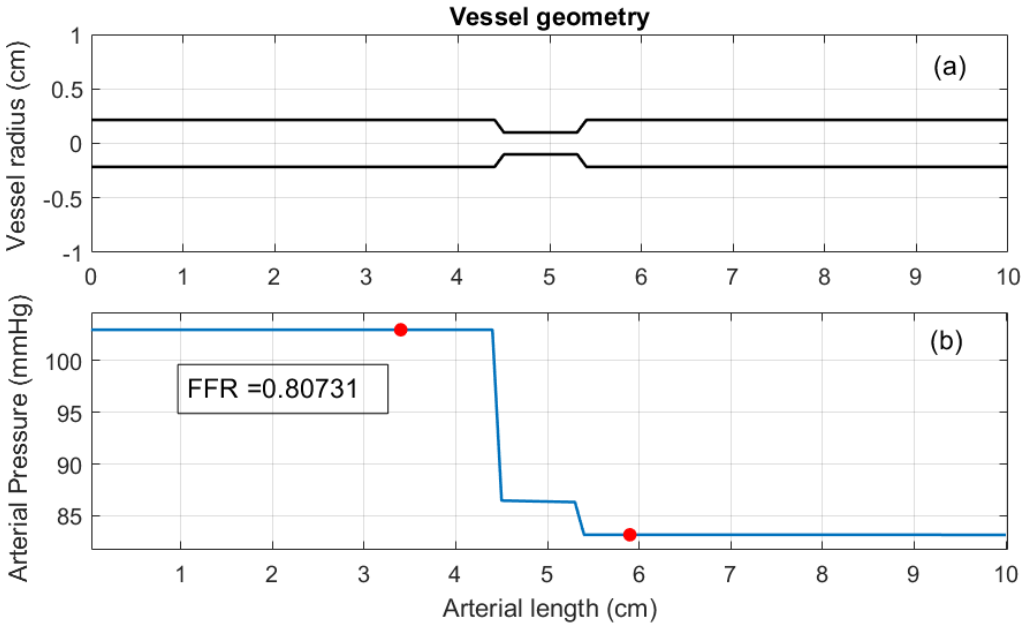


Figure 3.6: Flow solution of the analytical model FFR_P implemented in Matlab; (a) shows the stenosed geometry which is the a 2D projection of the geometry used for the CFD analysis in Figure 3.5; (b) shows the static pressure along the artery, where the red dots represent the probes at which the proximal and distal pressures were measured for calculating FFR.

Figure 3.6 displays the pressure solution along the stenosis artery, which was obtained using simplified Navier-Stokes equations FFR_P. For identical geometric and flow inputs, static pressure estimates from FFR_P align closely with the CFD results as seen in Figures 3.5 and 3.6.

The estimated FFR values from all models were used to categorize patients into high and low risk of CAD. Next, the performance was compared with the ground truth values. Figure 3.7 showcases the results of this comparative study. We conducted the experiments at varying levels of measurement noise, plotting the receiver operating characteristic (ROC) curves for patient classification into high and low CAD risk categories using the three proposed models at each noise level.

Of all three models, the patient-specific analytical model demonstrates the best performance, while the normal flow rate FFR model holds a slight advantage over the pure geometric model, which lacks any information on flow dynamics. The marked difference in the area-under-the-curve (AUC) for the patient-specific model underscores the importance of using patient-specific flow data when determining the risk of CAD.

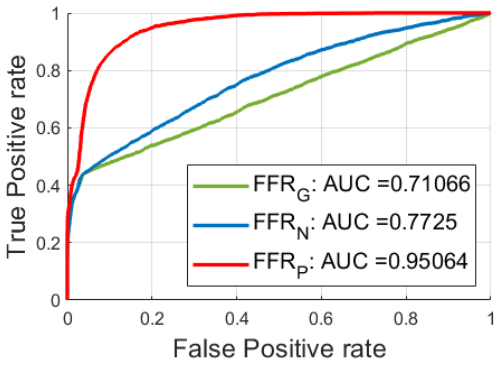
Figure 3.7 plots the ROC curves for the FFR estimates across varying levels of noise. The patient-specific flow-informed FFR_P has the best classification performance, followed by normal flow-informed FFR_N and then geometric FFR_G.

3.4 Limitations and Improvements

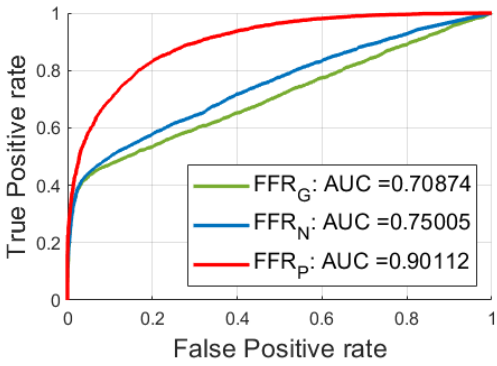
We were able to demonstrate that adding patient-specific blood flow information improves classification accuracy. However, this analysis was limited in its ability to quantify the importance of patient-specific blood flow in comparison to other anatomical features. With the aforementioned feature ranking tools at our disposal we estimated the importance of each feature separately.

In order to quantify the relative contribution of patient-specific blood flow and other geometric inputs towards assessment of CAD risk, we performed a feature importance analysis. We constructed and trained a multilayered perceptron (MLP) based binary classifier that classified patients into high versus low risk of CAD.

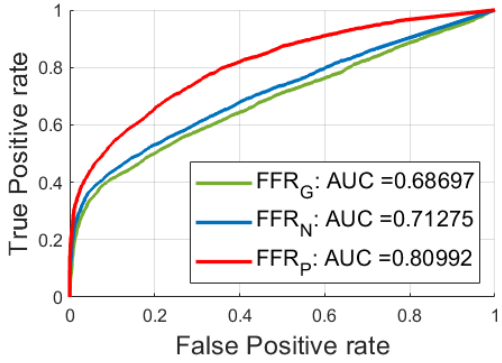
The classifier was trained on ground truth CAD risk labels and the following features from the



(a) 5% noise



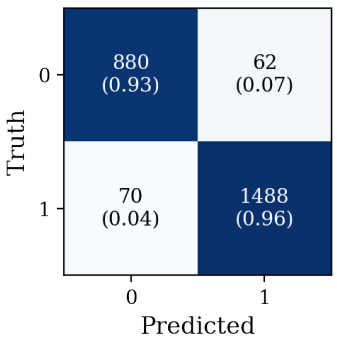
(b) 10 % noise



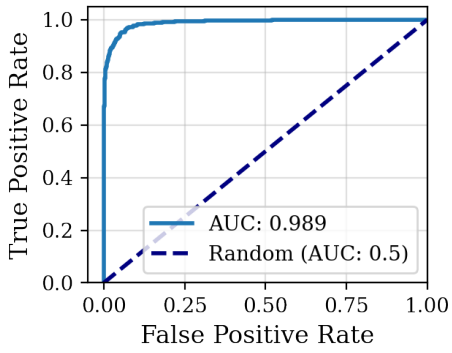
(c) 20 % noise

Figure 3.7: ROC curves for classification into high risk versus low risk for CAD, for varying levels of noise.

simulated population: U_{0P}, D_1, D_0, L_1, and $D_{stn} \triangleq \frac{D_1}{D_0}$. Note that the constant normal flow rate U_{0N} was not used because it gets scaled to zero during data pre-processing. The trained model had an average test AUC of 0.98 with 0.003 standard deviation.



(a) Confusion matrix



(b) Receiver operating characteristic curve

Figure 3.8: Performance summary of the trained CAD risk classifier on test set.

We performed feature importance analysis using permutation, input gradient, LIME, and SHAP to quantify the relative importance of the flow and geometric variables. We also computed the average importance generated from the four estimation methods as an additional measure of importance.

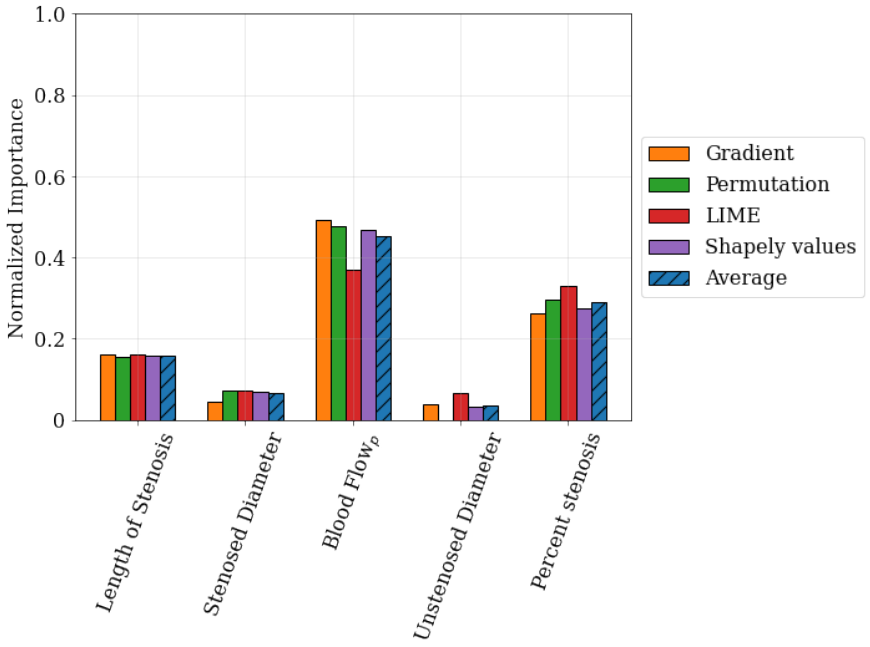


Figure 3.9: Relative importance of patient-specific features for classification of CAD risk.

Table 3.5: Average feature importance across all methods.

Feature	**Symbol**	**Average Importance**
Length of stenosis	L_1	0.159
Stenosed diameter	D_1	0.065
Patient-specific blood flow	U_{0P}	0.451
Upstream unobstructed diameter	D_0	0.034
Percent stenosis	$\%stn$	0.291

To validate the estimated importances, we used 'validation by agreement': measuring the concordance among the importances generated by a variety of importance estimation methods. The level of agreement between these methods was quantified using cosine similarity and root-mean-square Error (RMSE). Figure 3.9 shows the relative importance of the patient-specific features returned by different methods.

Table 3.6: Pairwise cosine similarity and RMSE between normalized input importance estimated by permutation (PERM), input gradient (GRAD), LIME, SHAP, and average of the four methods (AVG) for the CAD risk classifier.

Cosine Similarity

	GRAD	PERM	LIME	SHAP	AVG
GRAD	1.00	0.99	0.97	1.00	1.00
PERM	0.99	1.00	0.98	1.00	1.00
LIME	0.97	0.98	1.00	0.98	0.99
SHAP	1.00	1.00	0.98	1.00	1.00
AVG	1.00	1.00	0.99	1.00	1.00

RMSE

	GRAD	PERM	LIME	SHAP	AVG
GRAD	0.00	0.03	0.07	0.02	0.02
PERM	0.03	0.00	0.06	0.02	0.02
LIME	0.07	0.06	0.00	0.05	0.04
SHAP	0.02	0.02	0.05	0.00	0.01
AVG	0.02	0.02	0.04	0.01	0.00

The results in Table 3.6 demonstrate a high level of agreement among the feature importances generated by the different methods, thereby affirming the validity of our findings. Using the feature ranking methods we were able to successfully quantify the relative importance of patient-specific features for classification of CAD and illustrate the impact of using various patient-specific anatomical and flow features.

3.5 Summary

This chapter presents a medically relevant classification problem, CAD detection, and also a virtual clinical trial to evaluate different machine learned classification models. This problem also provides a test platform for evaluating the four conventional feature importance estimation approaches advanced in the following chapters. Results demonstrate successful identification of CAD risk with different models and, moreover, successful ranking of input feature importance that aligns with expert intuition.

CHAPTER 4

UNIFIED FRAMEWORK FOR MULTIMODAL FEATURE IMPORTANCE

The work in this chapter contributed towards the following:

- M. Azmat, A. Alessio. '*Feature Importance Estimation Using Gradient Based Method for Multimodal Fused Neural Networks*'. Presented at IEEE NSS-MIC-RTSD, 2022.
- M. Azmat, A. Alessio. '*Adaptable Feature Importance Estimation Framework for Fusion-based Multimodal Deep Neural Networks*'. Presented at SNMMI Annual Meeting, 2023.

4.1 Introduction

Multimodal neural networks (MNNs) are machine learning models that analyze data from multiple modalities, such as images, text, and audio. By combining multiple sources of information, multimodal models can often achieve higher performance than models that rely on a single modality [90, 91, 92]. In recent years, deep learning-based multimodal NNs have shown great potential as decision support systems by generating predictions that offer a comprehensive view, enhancing decision-making accuracy [93, 94, 95, 96].

Popular architectures of multimodal models are based on additive approaches. That is, input or learned features from different modalities are aggregated to make a decision such as in ensemble-based models [97], joint training models [98], and fusion-type models [99].

Fusion-type models are the most popular choice for multimodal architecture. Depending on where the modalities are fused, these models are classified as early-fusion, late-fusion, or joint-fusion models. Early fusion directly concatenates raw input features before passing them to a single neural network [100, 101]. This approach is simple but lacks explicit modeling of interactions between modalities [102]. Late fusion uses separate models for each modality and combines their outputs. Voting, stacking, and mixture of experts are common late fusion techniques. However, these models often do not effectively leverage cross-modal relationships during training [103]. Joint

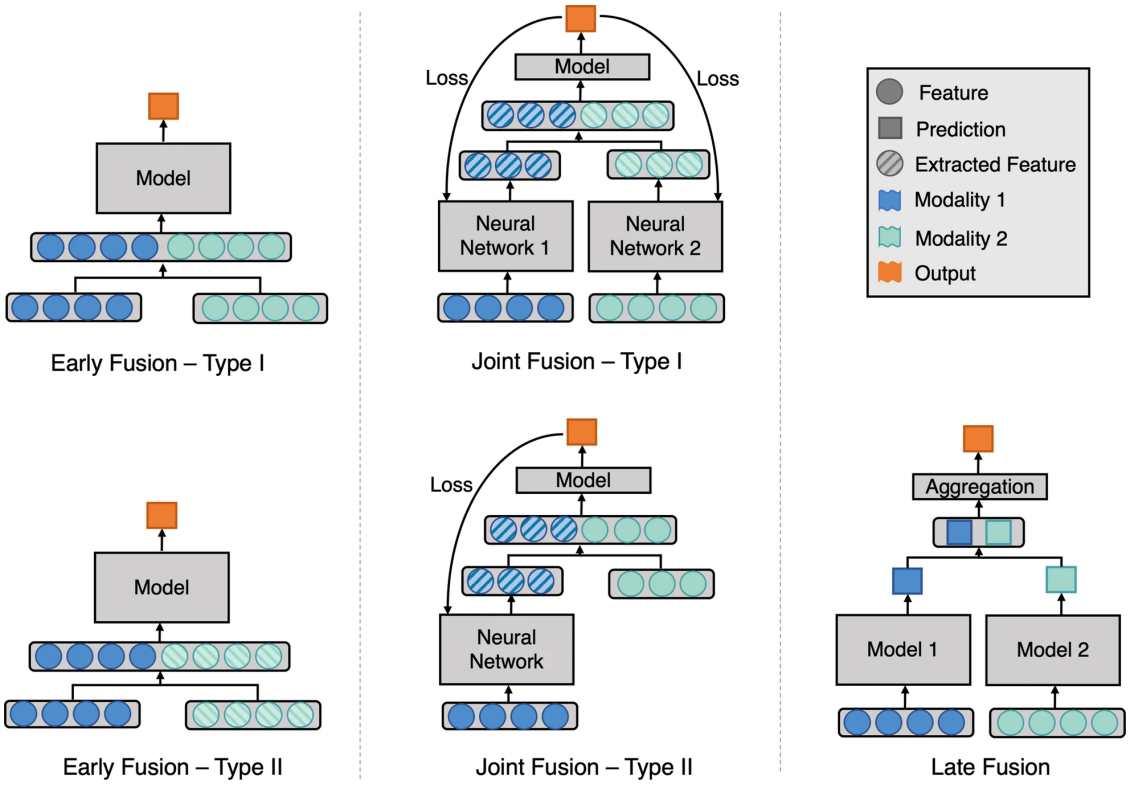


Figure 4.1: Model architecture for various multimodal fusion strategies. The left diagram illustrates early fusion, where original or extracted features are merged at the input level. The middle diagram represents hybrid or joint fusion, where original or extracted features are combined at the input level and the model is trained end-to-end. The right diagram shows late fusion, where predictions are consolidated at the decision level. Used under CC 4.0 from [4].

or hybrid fusion models incorporate both early and late fusion. For example, separate encoders can extract features from each modality, followed by fusion layers and joint training [104].

In the healthcare domain, the ability of MNNs to integrate different types of data such as electronic health records, medical images, and genetic information provides substantial advantages. MNNs have demonstrated their efficacy in medical image analysis, where images supply visual context and text contributes descriptive insights. The fusion of data from MRI, PET scans, and electronic health records can enhance brain tumor diagnosis and prognosis predictions [4]. Similarly, MNNs that incorporate fundus images, OCT scans, and clinical data have proven useful in facilitating diabetic retinopathy screening [105]. In patient monitoring scenarios, data streams from bedside equipment, wearable devices, and medical records can be synthesized to predict adverse events or disease trajectories [106].

Despite the promising advancements of MNNs, their complex architectures present unique challenges for explainability. While techniques like attention mechanisms and feature attribution can help explain predictions generated by unimodal neural networks, they often fall short with architectures that integrate cross-modal interactions. The complexity introduced by heterogeneous input types, feature blending, and high-dimensional latent spaces in multimodal networks can further obfuscate the decision-making process.

The development of explainability methods capable of addressing the specific challenges of multimodal learning continues to be an active research area. Recent studies have investigated both intrinsic and post-hoc explanation techniques suited to these models. Intrinsic methods aim to construct intrinsically interpretable model architectures. For instance, using attention scores as a proxy for feature importance in late fusion models for detecting hate speech from multimodal textual, cultural, and social data [107]. Post-hoc techniques based on occlusion, gradients, and perturbations have been explored for multimodal visual question answering (VQA) models. Authors of [108] propose perceptual score: a perturbation based metric that can be used to probe multimodal models and understand their reliance on different input data types for VQA data.

Multimodal explainability methods for medical applications are limited and often rely on fixed, pre-trained, modality-specific models, acting as feature extractors [109]. Alternately, they employ naive early fusion schemes [94, 110]. These approaches fail to fully leverage the capabilities of multimodal learning as they are heavily reliant on fixed pre-trained encoders, rather than learning cross-modal representations from scratch. As a result, the potential for explainability is limited, leaving interactions between modalities largely unexplored. Other multimodal implementations use medical images from different clinical modalities such as T1-weighted and T2-weighted MRIs. However, given the homogeneous nature of the input data structure, these approaches are not truly multimodal [111].

In summary, the development of explainable AI for multimodal learning in medical applications continues to pose a significant challenge. Existing methods are often restricted to single modalities or lack a comprehensive evaluation. There is a need for new techniques that can explain interactions

Table 4.1: Overview of Nomenclature in Multimodal Neural Networks.

Term	Definition	Example
Modality	Refers to the different types of data sources used in a multimodal network.	Images, Tabular data, Text.
Input	Refers to raw data that characterizes different attributes of the patient. Inputs can originate from various sources and have different data types.	MRI image, Age, Blood pressure.
Features	Represent the measurable properties or characteristics of the input data that are relevant to the task at hand. These can be obtained from inputs using models or preprocessing techniques.	Radiomic features, One-hot encoded categorical features.
Deep Features	Denote the high-level learned representations obtained after passing the feature through multiple layers of a neural network.	Convolutional features.
Fusion Features	Specifically refers to deep features in the fusion layer where features from more than one modality are being fused.	Fusion of clinical data and MRI: Concatenated deep feature vector.

between modalities. Adapting these solutions to meet clinical needs and applications is a crucial direction for the future of multimodal explainable AI in healthcare.

Our proposed solution is a framework for explaining the functionality of multimodal neural networks. Our framework adapts concepts from unimodal feature importance and modifies them for a multimodal model. We employ a hybrid fusion architecture where the model is trained end-to-end, enabling us to learn task-relevant features and cross-modal relationships. Our approach uses truly multimodal and heterogeneous input data, such as images, tabular data, and categorical features. The details of our proposed framework are described in the following section.

4.2 Proposed Framework

Our framework relies on a hybrid fusion architecture for multimodal learning as shown in Figure 4.2. Each input uses a modality-specific module that can be treated as a feature extractor. Deep features from the feature extractors are concatenated in the fusion layer. The fusion layer is used as an input to a fully connected network that generates model predictions. All weights are learned through end-to-end training. Table 4.1 gives definitions and examples of nomenclature used in this thesis and literature in general.

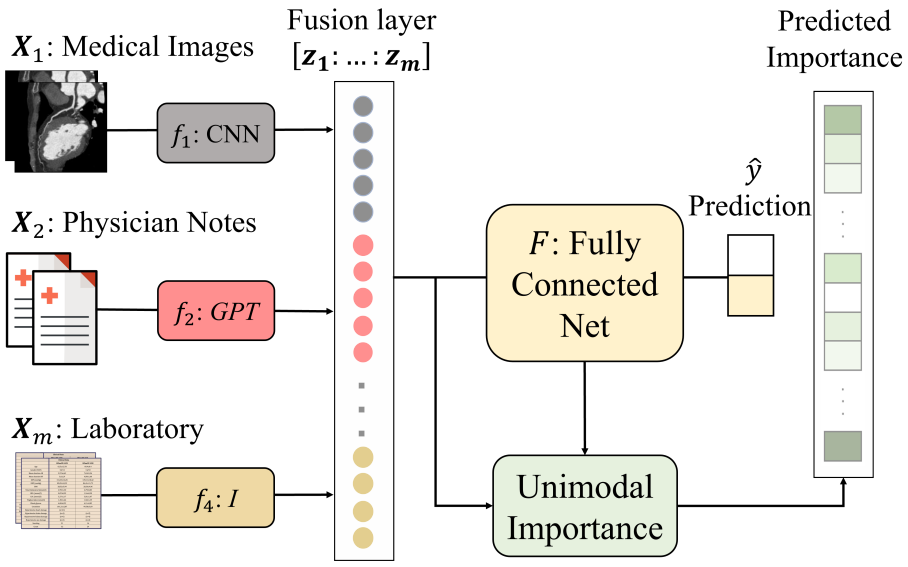


Figure 4.2: Proposed method for multimodal feature importance. A hybrid fusion architecture supporting multimodal inputs is trained in an end-to-end manner. Features in the fusion layer are used to estimate feature importance of the upstream inputs. The post-fusion architecture typically consists of fully connected layers. The feature importance module can be replaced with any post-hoc attribution method.

Since the post-fusion neural network block in Figure 4.2 takes in homogeneous fusion features from the shared representation space and returns model predictions, we can treat it as a unimodal model. As a result, we can modify the unimodal feature importance methods, described in chapter 2, to quantify multimodal input importance. Unlike prior use, we apply these methods at the fusion layer to estimate the importance of fusion features, then aggregate all of the fusion feature importances from a contributing input to estimate the importance of each input.

We will now formalize the proposed framework. All methods discussed are model-agnostic and can be adopted for multi-class classification with minor modifications. Consider a binary multimodal classification problem:

$$\Theta : \mathbf{X} \longrightarrow y \in \mathbb{R}, \tag{4.1}$$

where F represents the classifier, $\mathbf{X} = \begin{bmatrix} X_1 & \ldots & X_m \end{bmatrix}^T$ is the multimodal input consisting of m sub inputs X_i which can have different dimensions $\mathbb{R}^{d_i}$ depending on their modality.

Using the hybrid fusion architecture in Figure 4.2, each input is passed through a modality-

specific feature extractor f_i, to generate features Z_i corresponding to the input.

$$Z_i = f_i(X_i). \tag{4.2}$$

These features are then concatenated in the fusion layer to generate fusion features $\mathbf{Z}$

$$\mathbf{Z} \triangleq \begin{bmatrix} Z_1 & \cdots & Z_m \end{bmatrix}^T, \tag{4.3}$$

and passed as inputs to the classifier F to generate predictions y

$$y = F(\mathbf{Z}), \tag{4.4}$$

$$\Theta(\mathbf{X}) = F\left(\begin{bmatrix} f_1(X_1) & \cdots & f_m(X_m) \end{bmatrix}^T\right). \tag{4.5}$$

4.2.1 Multimodal Permutation Importance

Let Score$(\cdot)$ denote a function that returns the average classification performance score of the classifier F on a set of inputs. Let $\mathcal{S}_Z = \{\mathbf{Z} : \forall \mathbf{X} \in \mathcal{S}_N\}$ be the set of fusion features generated from $\mathcal{S}_N$ a multimodal input set with N samples. The permutation importance of jth fusion feature is given by

$$\text{PERM imp}(j) = \text{Score}(\mathcal{S}_Z) - \frac{1}{N_p}\sum_{i=1}^{N_p} \text{Score}\left(\mathcal{S}_{Z(j,i)}\right), \tag{4.6}$$

where $\mathcal{S}_{Z(j,i)}$ is the ith permutation of $\mathcal{S}_{Z(j,i)}$ in the jth fusion feature, and N_p is a hyperparameter that controls the number of permutations. In each permutation, values of the jth feature are randomly shuffled across the set. Importance of the kth multimodal input is computed by aggregating (averaging) the PERM importances of features from input k

$$\text{MM-PERM imp}(k) = \sum_{j \text{ from input } k} \left|\text{PERM imp}(j)\right|. \tag{4.7}$$

4.2.2 Multimodal Input Gradient Importance

The importance of input k can be approximated by aggregating the gradient-based importances of the fusion features from input k, and averaged over the set $\mathcal{S}_Z$ computed as

$$\text{MM-GRAD imp}(k) = \frac{1}{N}\sum_{\mathbf{Z}\in\mathcal{S}_Z} \left\| \frac{\partial F(\mathbf{Z})}{\partial Z_k} \right\|_1, \tag{4.8}$$

where Z_k is the vector of deep features from input k. Note that the l_1 norm sums up the absolute values of gradients with respect to all fusion features coming from input k.

4.2.3 Multimodal LIME

Let G be a surrogate model for the classifier F in the local neighborhood of the sample $\mathbf{Z}$ given by

$$G(\mathbf{V}) = \mathbf{W}^T\mathbf{V}, \tag{4.9}$$

where $\mathbf{V}$ is sampled from a multivariate Gaussian centered at $\mathbf{Z}$. To ensure that G approximates the actual model F locally, we learn the weights W by minimising the weighted least squares error

$$\mathcal{L}(W) = \sum_{\mathbf{V}} e^{-\|\mathbf{V}-\mathbf{Z}\|^2}\|F(\mathbf{V}) - \mathbf{W}^T\mathbf{V}\|^2, \tag{4.10}$$

where $\mathbf{W} = \begin{bmatrix} W_1 & \cdots & W_m \end{bmatrix}^T$ is the vector of learned coefficients, W_k represents the sub vector containing LIME coefficients for fusion features Z_k extracted from input X_k. Importance of the kth input is computed by aggregating values of coefficients corresponding to features from input k given by

$$\text{MM-LIME imp}(k) \triangleq \frac{1}{N}\sum_{\mathbf{Z}\in\mathcal{S}_Z} ||W_k||_1. \tag{4.11}$$

4.2.4 Multimodal SHAP

Shapley values use the net contribution of a feature across all combinations of feature interactions to quantify its importance. The marginal contribution of a feature for a particular combination is calculated as the difference in model predictions when the feature is included or excluded. To avoid exploring all possible combinations, a Monte Carlo approximation can be used to sample N_s combinations. The shapely value based importance of the jth fusion feature is given by

$$\phi_j(\mathbf{Z}) = \frac{1}{N_s}\sum_{i=1}^{N_s} F\left(mask_i(\mathbf{Z})^{+j}\right) - F\left(mask_i(\mathbf{Z})^{+j}\right), \tag{4.12}$$

where $mask_i(.)^{+j}$ generates a masked instance by replacing random feature values (generated by ith Monte Carlo sampling) in $\mathbf{Z}$ with expected values from the background dataset $\mathcal{S}_Z$ while keeping the original value for the jth. Whereas $mask_i(.)^{-j}$ replaces the actual value for j with the masked

value from the background data set. The SHAP importance of kth multimodal input is estimated by aggregating the shapely values of fusion features extracted from the kth input.

$$\text{MM-SHAP imp}(k) = \frac{1}{N} \sum_{\mathbf{Z} \in \mathcal{S}_Z} \sum_{j \text{ from input } k} \left| \phi_j(\mathbf{Z}) \right|. \tag{4.13}$$

4.2.5 Multimodal Average Importance

We propose an additional approach, MM-AVG imp, that takes the average of the importances returned by input gradient, permutation, LIME, and SHAP methods. The key motivation is that each technique approximates model behavior and defines importance slightly differently. In real-world cases without ground truth feature importance, it may be preferable to use a more comprehensive metric that combines multiple notions of importance, rather than relying solely on one definition.

4.3 Simulation Platform

For real-world problems, true values for feature importance are often unknown. As a solution, we used synthetic classification tasks with predefined decision functions, which allowed us to generate ground truth for feature importances and thereby validating our multimodal input importance methodology. We explored a variety of synthetic classification problems using controlled decision

Table 4.2: Synthetic decision functions and the corresponding normalized ground truth input importance.

id	**Decision function**	**Ground truth feature importance**			
		$\mathbf{X}_1$	$\mathbf{X}_2$	$\mathbf{X}_3$	$\mathbf{X}_4$
1	$\|\|Z_1\|\|_1$	1	0	0	0
2	$\|\|Z_2\|\|_1$	0	1	0	0
3	$\|\|Z_3\|\|_1$	0	0	1	0
4	$\|\|Z_4\|\|_1$	0	0	0	1
5	$\sum_{i=1,2} \|\|Z_i\|\|_1$	0.5	0.5	0	0
6	$\sum_{i=3,4} \|\|Z_i\|\|_1$	0	0	0.5	0.5
7	$\sum_{i=1}^{4} \|\|Z_i\|\|_1$	0.25	0.25	0.25	0.25
8	$e^{Z_{7,2}} ln\left(Z_{1,1} + Z_{2,1}\right)^2$	$4\left\|\frac{e^{Z_{2,7}}}{(Z_{1,1}+Z_{1,2})}\right\|$	$\left\|e^{Z_{2,7}} ln\left(Z_{1,1} + Z_{1,2}\right)\right\|$	0	0

Where, $Z_{i,k}$ is the ith fusion feature from the kth input used during ground truth feature importance generation.

functions to generate ground truth labels and feature importances for our multimodal data. The decision functions, detailed in Table 4.2, combined encoded multimodal inputs and noise to generate ground truth labels and ground truth input importances.

The multimodal inputs were encoded using pre-trained, modality-specific feature extractors tailored to each input data type. It's important to note that these pre-trained encoders were utilized solely for ground truth generation in our analysis. To ensure fairness and prevent leakage, we used a different architecture in our multimodal classifier.

With the analytical form of the decision functions known, ground truth input importances were given by absolute value of gradient of the function with respect to encoded inputs, which were then summed and normalized to get input level importances.

This simulation environment allowed us to precisely define the importance of different inputs and thoroughly test the approach across diverse multimodal use cases. Given that true feature importances are not known for real-world datasets, the use of real data with synthetic decision function allowed us to quantitatively validate the proposed methodology for estimating multimodal feature importance against known true values.

4.3.1 Multimodal Data

We simulated a variety of synthetic multimodal data sets each containing four inputs for a binary classification task, where $\mathbf{X}_1, \mathbf{X}_2$ were images and $\mathbf{X}_3, \mathbf{X}_4$ were tabular. For image inputs, we used 28×28 pixel abdominal CT scan images from OrganA and OrganC medMNIST dataset [112]. Whereas tabular inputs were sampled from a multivariate Gaussian distribution, with one set of inputs drawn from a cross-correlated distribution to model dependent features. Ground truth class labels were generated by thresholding the decision function for a class-balance classification problem. In total, 10 different decision functions simulated 10 different train/test data sets containing 10000 samples each.

4.3.2 Model Architecture

We used the same model architecture for all classification problems and retrained the model from scratch each time. We constructed a CNN-based encoder to process image inputs and a

standard scaler for tabular data, as summarized in Table 4.3. The image encoder was composed of CNN layers with 6, 16, and 64 filters of size 3×3 and each layer was followed with max pooling using a 2×2 window. The convolutional layers were then followed by a fully connected layer that flattened the convolution outputs to a vector of encoded features. The post-fusion architecture was comprised of three fully connected layers that processed the encoded images and standard-scaled tabular inputs, providing the class probabilities.

Table 4.3: Description of inputs and encoders used for training the multimodal classifiers for synthetic data.

	Input	**Modality**	**Encoder** (f)	**Encoded Dimension**
$\mathbf{X}_1$	Abdominal CT	Image	CNN	1×10
$\mathbf{X}_2$	Abdominal CT	Image	CNN	1×10
$\mathbf{X}_3$	Independent features	Tabular	Standard Scaler	1×10
$\mathbf{X}_4$	Correlated features	Tabular	Standard Scaler	1×10

To guarantee the models' reliance on the input was not influenced by the scale of fusion features, we implemented batch normalization within the fusion layer. An important implementation detail to note is that the normalization must occur before concatenating the features, so that the gradients are evaluated on normalized fusion features. This ensures that scale does not impact the importance estimates returned by the gradient method.

The models used an Adam optimizer with a fixed learning rate 1×10^{-3} for approximately 20 epochs or when the validation accuracy went over 95%. The weights were initialized using default PyTorch initialization. Training was done on a CPU with an average epoch time of approximately 70 seconds.

4.4 Results

We used classification accuracy on the test set as a metric to assess if the learned model is a good approximation of the ground truth decision function. This is critical because performance of feature importance methods is limited by the performance of the learned predictive model. All trained models achieve $\geq$ 92% classification accuracy on independent test data. The side by side comparison of input importances generated by the different methods in Figure 4.3 shows that the

importance estimates are consistent across different estimation methods and our approach is able to identify the top contributing inputs in most cases.

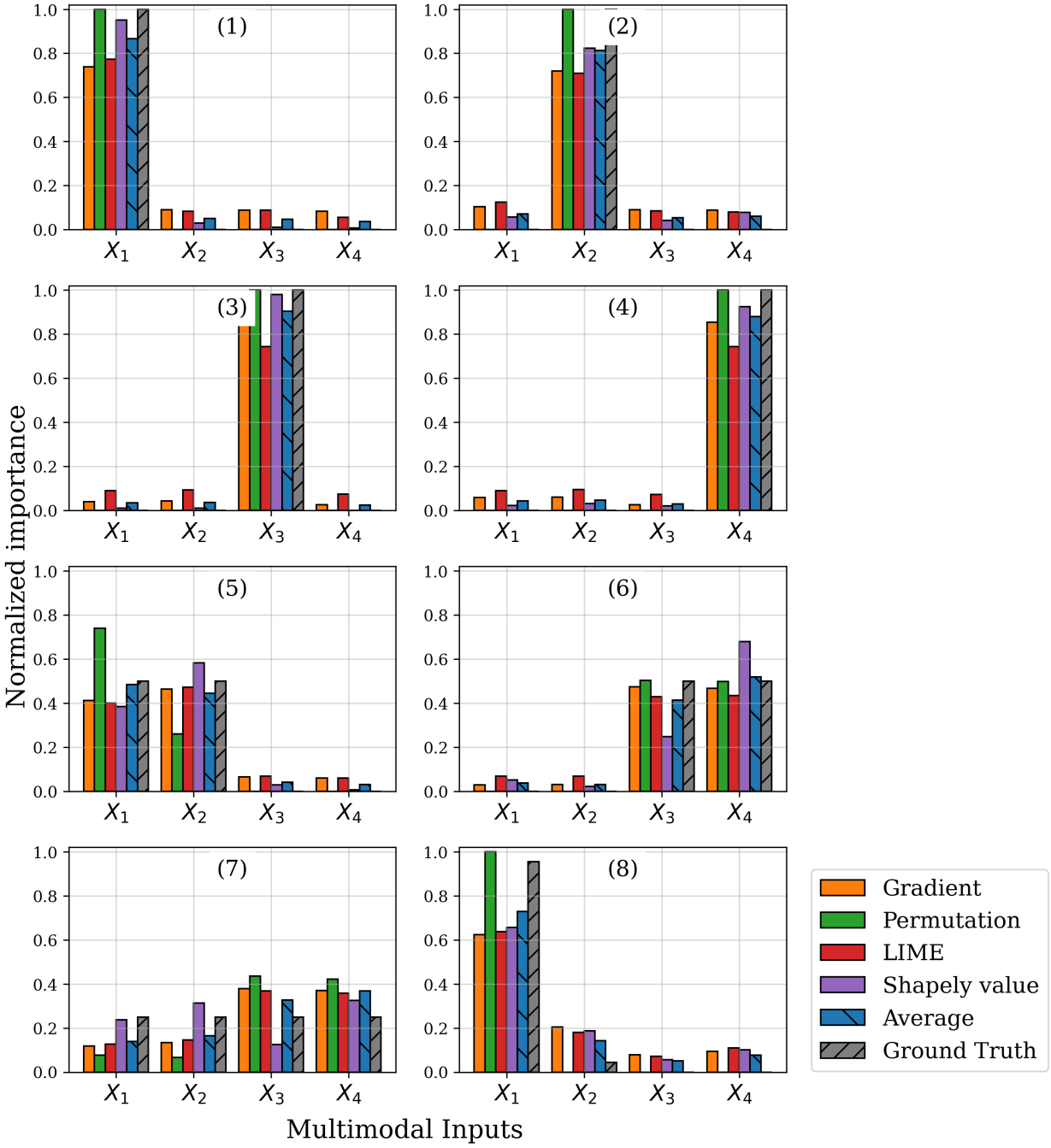


Figure 4.3: Plots of normalized ground truth versus average feature importance returned by four estimation methods plus an average value across the methods. Each subplot represents a different test case corresponding to decision functions given in Table. 4.2. The predicted feature important values closely estimate known ground truth and display a consistent ranking of features.

Table 4.4 shows the percent relative error of predicted and ground truth importances estimated by the different methods averaged over the different decision functions. The percent relative error for all inputs averaged over all decision functions lies within 9% of the ground truth importance. These experiments repeated for multiple decision functions and data establish the validity of our approach for estimating importance of multimodal inputs. In Chapter 5 we demonstrate performance in real data sets where we don't have access to ground truth importances.

Table 4.4: Percent relative error and RMSE in feature importance estimates for the proposed methods compared to synthetic ground truth from synthetic decisions functions employing four multimodal inputs.

Feature Importance Method	**Percent relative error**					**RMSE**
	$\mathbf{X_1}$	$\mathbf{X_2}$	$\mathbf{X_3}$	$\mathbf{X_4}$	Mean	
Gradient	8.67	7.95	6.50	6.76	7.47	0.122
Permutation	4.02	4.38	1.87	1.74	3.00	0.088
LIME	10.1	8.94	8.18	8.00	8.83	0.135
Shapely value	5.07	5.29	5.46	5.10	5.23	0.100
Average	5.84	5.71	4.87	4.98	5.35	0.084

Average method averages the importances from aforementioned methods into a single unified importance metric.

Table 4.5: Pairwise cosine similarity and RMSE between normalized input importance estimated by proposed methods for the synthetic classification problems.

Cosine Similarity

	GRAD	PERM	LIME	SHAP	AVG
GRAD	1.00	0.97	0.99	0.97	0.99
PERM	0.97	1.00	0.96	0.94	0.98
LIME	0.99	0.96	1.00	0.97	0.99
SHAP	0.97	0.94	0.97	1.00	0.98
AVG	0.99	0.98	0.99	0.98	1.00

RMSE

	GRAD	PERM	LIME	SHAP	AVG
GRAD	0.00	0.13	0.04	0.10	0.04
PERM	0.13	0.00	0.15	0.15	0.10
LIME	0.04	0.15	0.00	0.11	0.06
SHAP	0.10	0.15	0.11	0.00	0.08
AVG	0.04	0.10	0.06	0.08	0.00

The input importance methods used in our analysis rely on slightly different definitions of importance, therefore, we want to validate that the importance values returned by these methods are consistent and agree well with each other. To quantify this agreement, we calculated the cosine similarity and root mean squared error (RMSE) between the normalized importances from each method. The cosine similarity results in Table 4.5 are computed across all decision functions, with pairwise similarities between all methods. The values show strong agreement in the normalized importances returned by the different techniques across a variety of decision functions.

Additionally, Table 4.5 reports the RMSE of normalized importances between pairs of methods, with the values representing RMSE across all decision functions. The RMSE values indicate consistentcy across our methods. Despite the different theoretical formulations of the methods, they produce aligned rankings and similar normalized importances, as evidenced by the high similarity and low error across methods.

4.5 Summary

This chapter proposes methods to apply feature importance estimation at the deep fusion layer and then aggregate those values to estimate input importance for multimodal neural networks. Through controlled simulation experiments, we demonstrate effective and consistent input importance estimation using four different importance estimation techniques. These methods provide information on the relative importance of different inputs in a multimodal model decision.

CHAPTER 5

ESTIMATING MODEL RELIABILITY WITH MISSING DATA THROUGH MULTIMODAL IMPORTANCE

M. Azmat, H. Fessler, G. Holste, A.Alessio, '*Predicting Impact of Missing Modalities on Classification Performance in Multimodal Models: A Unified Framework for Multimodal Input Importance*', Submitted to IEEE Journal of Biomedical and Health Informatics.

5.1 Introduction

In recent years, deep learning-based MNNs have shown great potential as decision support systems in healthcare [93, 94, 95, 96]. However, despite the substantial potential of MNNs for medical and clinical applications, the use of multiple modalities also introduces challenges. One leading challenge is how to handle and interpret the impact of missing or incomplete data. In medical settings, it is common to encounter scenarios where some modalities are missing or incomplete, either from incomplete medical records, excessive costs, or potential risks to the patient. Previous studies have demonstrated that missing data can result in varying levels of performance reduction for multimodal models [113, 114], but there is a lack of research on methods for predicting the impact of missing modalities in multimodal learning. In this work we develop a method to predict the performance degradation of multimodal deep NNs in cases of one or more missing input modalities. Since model performance reflects its reliability, predicting how performance declines can provide indirect insights about model reliability when inputs are missing. Predicting performance degradation resulting from missing data could have an invaluable role in increasing the interpretablity of MNNs. Additionally, it has the potential to inform decisions about which modalities and tests are critical for specific patients. This could ultimately lead to a reduction in the number of unnecessary imaging and lab procedures, potentially having a significant impact on healthcare costs and patient safety.

To demonstrate significance of this approach, consider a MNN trained and deployed to use 5 multimodal inputs (labeled A-E) to perform classification. If this model encounters a patient with only 4 of the inputs (A-D), it would be beneficial to know *a priori* the potential value of obtaining

input (E). If the performance gain of adding input (E) is insignificant, the costs and risks of getting that input can be avoided. Conversely, if the performance gain is significant, the collection of input (E) can be prioritized for the patient. In short, this approach has the potential to enable clinical decision makers to better interpret the performance of machine learned models and ultimately make better informed decisions about any additional information needed for specific patients.

Furthermore, knowing the performance of the model in the presence of noisy or missing data is also crucial for the change control plan portion of the FDA proposed regulatory framework for Artificial Intelligence/Machine Learning (AI/ML)-based Software as a Medical Device (SaMD) [32]. The change control plan is intended to ensure that changes to the software do not negatively impact its safety, efficacy, or performance [115]. A priori knowledge of the model performance in the presence of missing data allows for a more accurate assessment of the potential risks associated with changes to the software and enables more accurate post-market surveillance, essential for identifying any problems or issues that may arise after the software has been released. This can help to ensure that the software remains safe and effective for use in clinical settings.

This work is based on the hypothesis that the *performance degradation of MNNs due to a missing input is correlated with the importance of the missing input*. To predict the effects of multimodal mean imputation on classification performance, we propose a two-step method reliant on 1) multimodal feature importance estimation and 2) performance degradation estimation for multimodal missing data.

Step 1, multimodal input importance: We use methods developed in Chapter 4 to estimate importance of multimodal inputs. In the absence of ground truth importance values for real data, we employ a suite of distinct feature importance methods and establish a consensus across these approaches.

Step 2, performance degradation estimation for multimodal missing data: During inference with a MNN, missing inputs are generally treated with imputation methods to fill in gaps in input data. There are several popular methods of input imputation, including mean and median imputation [116], K-nearest neighbors imputation [117], multiple imputation, data augmentation [118], and

machine learning based approaches [119]. These methods can be applied to medical datasets where information comes in various forms such as medical images, patient records, and genetic data. The choice of imputation method depends on the specific characteristics of the dataset and the goals of the machine learning model. While there has been substantial research in developing variations of data imputation methods to deal with missing data, there has been limited research in trying to predict the impact of data imputation on model performance. In this work, we implement and evaluate a method that linearly relates performance degradation to missing input importance.

To summarize, we propose an approach to enhance the interpretability of multimodal models. The modality-level importance estimation step provides insight into the model's dependence on input data. The second step leverages the importance metrics generated in the first step, utilizing them to shed light on the model's performance limitations and behavior in the absence of certain inputs. Our primary contributions lie in offering a deeper understanding of the model's decision-making process, and quantifying its reliability under unique circumstances.

5.2 Methods

To predict classification performance in the case of missing or imputed data, we propose a linear model that relates input importance to model performance. This assumption is based on the hypothesis that input importance is, and should be, proportional to model performance. The predicted classification performance, $\text{Score}_{\mathbf{X}_k}$, when input $\mathbf{X}_k$ is imputed (missing) for the set of missing inputs $\mathcal{K}$ is

$$\text{Score}_{\mathbf{X}_k} = \text{Score}_{\Phi} - \sum_{k \in \mathcal{K}} \overline{\text{imp}}(\mathbf{X}_k)(\text{Score}_{\Phi} - \text{Score}_{\mathbf{X}_1:\mathbf{X}_m}). \tag{5.1}$$

In this relationship, $\overline{\text{imp}}(\mathbf{X}_k)$ is the normalized aggregated importance of the kth input estimated from one of the four methods discussed in Chapter 4. The Score_{Φ} is the reference score without input imputation and $\text{Score}_{\mathbf{X}_1:\mathbf{X}_m}$ is the baseline score when all m modalities are imputed. It should be stressed that the $\overline{\text{imp}}(\mathbf{X}_k)$ terms are normalized to sum to one for a given classifier model. The relationship in (5.1) is intuitive with the difference in the later score terms providing the range of performance degradation that is tempered by the relative importance of the missing input. In the

extreme when all modalities are missing, the classification performance reverts to $\text{Score}_{\mathbf{X}_1:\mathbf{X}_m}$.

A key advantage of this approach is its ability to predict model's performance degradation with little added cost. For a classification problem with m inputs and N samples, using the linear approximation (5.1)) we can get the estimates for model performance with $O(m)$ complexity. Alternately, if we calculate the predictions by performing imputation on data the complexity would be

$$\underbrace{O(N)}_{\text{Imputation}} + O(N \underbrace{(ml_1 + l_2l_3 + ...)}_{\text{Forward pass}}), \tag{5.2}$$

where l_i is the dimension of the weights in ith layer. For machine learning models $N >> m$. This added benefit becomes more significant for large data with more inputs.

5.3 Data Collection and Pre-processing

We performed our analysis for two real world problems: 1) multimodal breast tumor classification problem, and 2) multimodal cardiomegaly classification problem.

5.3.1 Breast Tumor Data

For the breast tumor classification study, we used fully anonymized data from an IRB-approved study of 5,248 women who had 10,185 breast cancer examinations between July 2005 and November 2015 [96]. Each patient received a dynamic contrast-enhanced magnetic resonance imaging (DCE-MRI) exam, and a subset of patients received a mammogram (76.5%) or underwent breast tissue biopsy (26.8%). We considered each breast as a separate case and cropped the DCE-MRI images to store them as single breast images.

We included MRI images without artifacts that had been scored using the Breast Imaging-Reporting and Data System (BI-RADS) and had a known 12-month post-MRI cancer status. Breasts labeled 'Malignant' had been diagnosed with breast cancer, confirmed by pathology, either at the time of examination or within 12 months after MRI. All other breasts were labeled as 'Benign'. We also included additional features, such as patient age, clinical indication for MRI, and background parenchymal enhancement from MRI.

To overcome potential biases from artifacts or signal changes caused by biopsy, we transformed

the DCE-MRI images to 2D maximum intensity projection (MIP) images, which retain only high contrast enhancement information and effectively remove any artifacts. The MIP images were resized to 224×224 pixels, and pixel intensities in the top 0.5% were removed. The remaining intensity values were normalized, and basic information from the images and tabular clinical features were added to the dataset.

Based on the unimodal feature importance results from an earlier study [96], we selected a subset of four most significant tabular inputs. Incorporating additional non-imaging features beyond this subset did not yield any notable improvements in the model performance. Consequently, we chose (1) Age: Patient's age at the time of the MRI study, (2) Max intensity: maximum pixel intensity in the MIP image, (3) Breast Density: Mammographic breast density via BI-RADS assessment (Fibroglandular, Dense, or Extremely Dense), and (4) MRI Indication: Clinical indication for MRI study (Screening, Diagnostic, or Known Cancer).

Finally, we created a balanced subset of the dataset consisting of 6,842 breast images and their associated non-image features. The non-image features were normalized, and the dataset was randomly split into three balanced sets: 4,180 cases for training (61.1%), 650 cases for validation (9.5%), and 2,012 cases for testing (29.4%). To avoid possible data leakage, samples from the sample patient were not shared across these sets.

5.3.2 MIMIC Data for Cardiomegaly Classification

For the cardiomegaly classification problem, we leveraged the open source, multimodal MIMIC-IV and MIMIC-CXR datasets [120, 121, 122] available through credentialed access. The MIMIC-CXR dataset contains over 377,110 chest radiographs, each affiliated with radiology reports, corresponding to 227,835 radiographic studies involving around 65,000 unique patients. MIMIC-CXR contains pre-generated ground truth labels for around 12 diseases derived from the radiology reports using the CheXpert labeler [123]. We include data for studies with definite labels of cardiomegaly present or absent.

We only used the Posterior-Anterior views of the radiographs that were preprocessed using a center crop on the longer dimension of the image, generating a square image. This was followed by a

resizing to 224×224 pixels and normalization to range between $[-1024, 1024]$. This pre-processing was chosen to generate images that are compatible for use with the pre-trained TorchXrayVision models [124].

Additionally, tabular demographic information matching the radiographs was extracted from MIMIC-IV. The demographic inputs include age, gender (identified as male or female), type of insurance (grouped as Medicaid, Medicare, or others), marital status (noted as divorced, married, single, or widowed), and ethnicity (categorized into American Indian/Alaska Native, Asian, Black/African American, Hispanic/Latino, White, and other demographics).

The final multimodal dataset for cardiomegaly classification contained 13,786 images of 8,940 unique patients, each matched with corresponding demographic data. This dataset was then partitioned into training, testing, and validation sets using patient identifiers to prevent data leakage. Each set had approximately 65% prevalence of cardiomegaly.

5.4 Model Setup and Training

5.4.1 Multimodal Breast Tumor Classifier

Figure 5.1 provides an overview of the architecture for the multimodal breast cancer classification model. For the image encoder, ResNet50 was adapted to accommodate a single-channel input. The classification head was removed, an average pooling followed by a fully connected layer was added after layer 4 bottleneck 2 in the PyTorch implementation of ResNet50, effectively encoding the 2048 convolution features from the image to 10 deep features for fusion. As discussed in [96], the choice of encoded dimension is arbitrary and has no significant impact on model performance. For the tabular inputs we use standard scaling and one-hot-encoding. Table 5.1 shows an overview of the input modalities and their corresponding encoding schemes.

5.4.2 Multimodal Cardiomegaly Classifier

For the cardiomegaly classifier, we modified the pre-trained Densenet121-res224-chex (DenseNet 224×224 model trained on CheXpert dataset) to generate deep features for the image modality. The 1024-dimensional vector from the final dense-layer was passed through a fully connected layer that

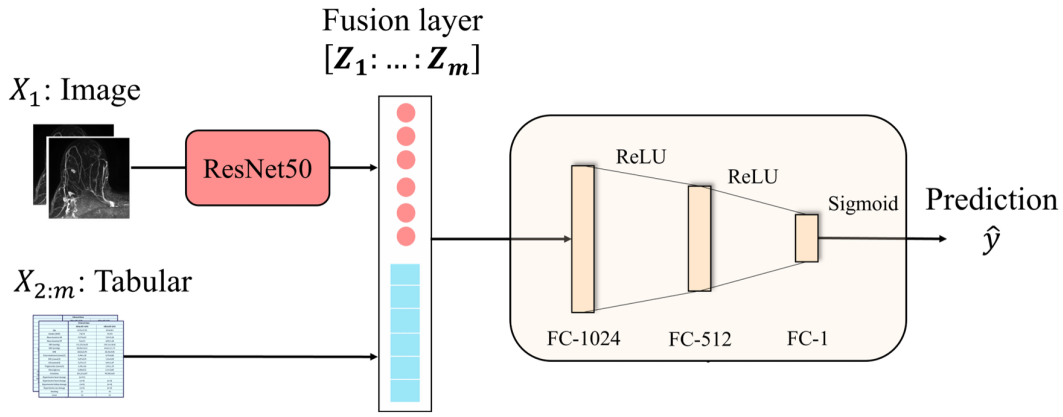


Figure 5.1: Architecture of the hybrid fusion model used for classifying breast MRI's using multimodal data. Resnet50 is used to extract fusion features from images while tabular inputs are pre-processed using standard scalar and one hot encoding. All weights are learnable and the model is trained end-to-end.

Table 5.1: Description of Inputs used for training the multimodal breast tumor classifier and the corresponding encoding schemes.

	Input	**Modality**	**Encoder** (f)	**Encoded Dimension**
$\mathbf{X}_1$	MRI	Image	ResNet50	1×10
$\mathbf{X}_2$	Age	Tabular	Standard Scaler	1×1
$\mathbf{X}_3$	Max Intensity	Tabular	Standard Scaler	1×1
$\mathbf{X}_4$	Breast Density	Tabular	One-hot-encoder	1×3
$\mathbf{X}_5$	MRI Indication	Tabular	One-hot-encoder	1×3

returned a 10-dimensional encoded image vector. The choice of encoding dimension was arbitrary and did not impact model performance significantly. Weights of the pre-trained DenseNet were frozen, whereas weights for last fully connected layer were learned during training of the multimodal classifier. Table 5.2 lists model inputs and their corresponding encoding methodologies.

Table 5.2: Description of Inputs used for training the multimodal cardiomegaly classifier and the corresponding encoding schemes.

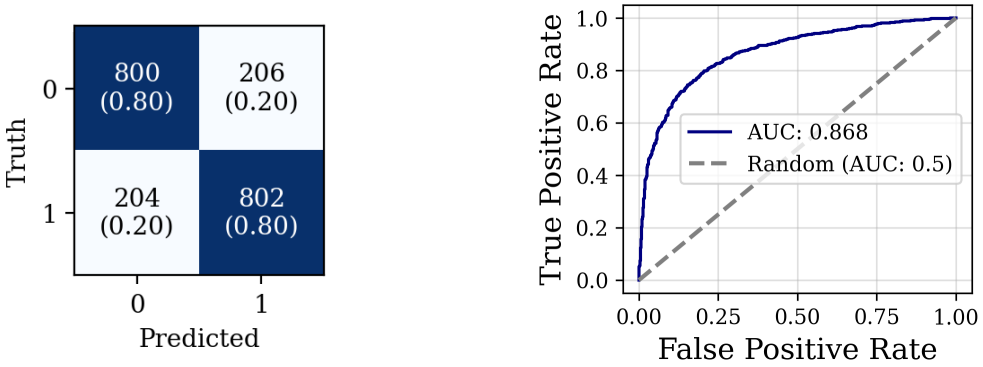
	Input	**Modality**	**Encoder** (f)	**Encoded Dimension**
$\mathbf{X}_1$	Radiograph	Image	DenseNet	1×10
$\mathbf{X}_2$	Age	Tabular	Standard Scaler	1×1
$\mathbf{X}_3$	Gender	Tabular	One-hot-encoder	1×2
$\mathbf{X}_4$	Insurance	Tabular	One-hot-encoder	1×3
$\mathbf{X}_5$	Marital Status	Tabular	One-hot-encoder	1×4
$\mathbf{X}_6$	Ethnicity	Tabular	One-hot-encoder	1×6

The breast tumor and cardiomegaly models were trained using the Adam optimizer [125], with a fixed learning rate of 1×10^{-4} and without any learning rate scheduling. The models were trained for a total of approximately 200 epochs or until the validation area under the curve (AUC) failed to demonstrate any improvement in the last 20 epochs, whichever occurred first. The PyTorch [126] default settings were utilized for model weight initialization except for where pre-trained weights were used. The training process was executed on a single NVIDIA Tesla V100S GPU, and the average epoch time was approximately 17 seconds for the breast tumor classifier and approximately 40 seconds for the cardiomegaly classifier. Weights from the epoch with the highest validation AUC were selected and used in further analysis of the trained model.

5.5 Results

5.5.1 Breast Tumor Classification

The trained breast tumor classification model has an AUC of 0.868 on a hold-out test dataset. Model sensitivity and specificity on the test dataset are 0.795 and 0.796 respectively, using the optimal threshold of 0.52. This performance is summarized in Figure 5.2. Figure 5.3 displays correctly and incorrectly classified samples from the test set and provides examples of a true positive, true negative, false positive, and false negative instance.

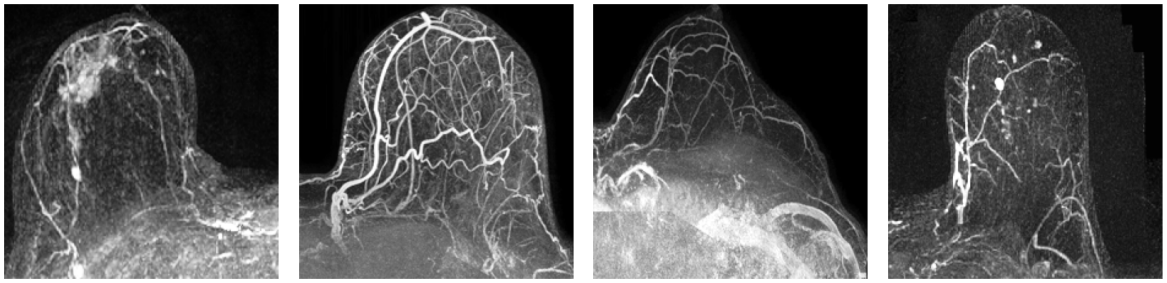


(a) Confusion matrix

(b) Receiver operating characteristic curve

Figure 5.2: Performance of the the trained breast tumor classifier on test set.

Figure 5.4 presents the input importance estimates. In the absence of ground truth feature importance, we validated our results by assessing the agreement between the different importance



(a) True positive (b) True Negative (c) False Positive (d) False Negative

Figure 5.3: Examples of different classification outcomes of the trained breast tumor classifier on the test set.

methods. Table 5.3 displays the cosine similarity and RMSE between methods. The average (AVG) calculates the mean importance across all these methods and renormalizes the features to sum to one. The results demonstrate a high level of agreement, with cosine similarity ≥ 0.97 and RMSE ≤ 0.11, among the importance values generated by the different estimation methods.

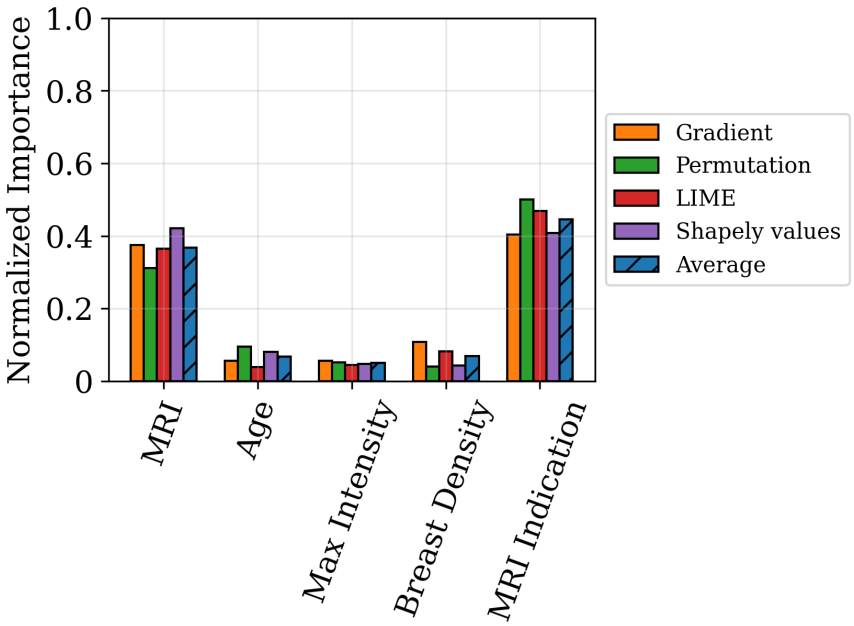


Figure 5.4: Comparison of normalized feature importance results and associated feature ranks using gradient, permutation, LIME, and shapely values based methods for the multimodal breast tumor classifier. AVG reports the mean importance across the four methods.

Another approach to validate these importance estimates is via expert opinion. For breast tumor classification, the importance estimation aligns well with expert intuition, as majority of the tumor-related information is contained in the MRI image. Furthermore, the MRI indication, which

Table 5.3: Pairwise cosine similarity and RMSE between normalized input importance estimated by proposed methods for the multimodal breast tumor classifier.

Cosine Similarity

	GRAD	PERM	LIME	SHAP	AVG
GRAD	1.00	0.97	0.99	0.99	1.00
PERM	0.97	1.00	0.99	0.97	0.99
LIME	0.99	0.99	1.00	0.99	1.00
SHAP	0.99	0.97	0.99	1.00	0.99
AVG	1.00	0.99	1.00	0.99	1.00

RMSE

	GRAD	PERM	LIME	SHAP	AVG
GRAD	0.00	0.06	0.03	0.07	0.02
PERM	0.06	0.00	0.04	0.11	0.05
LIME	0.03	0.04	0.00	0.10	0.03
SHAP	0.07	0.11	0.10	0.00	0.07
AVG	0.02	0.05	0.03	0.07	0.00

includes a 'known cancer' category, can provide significant insights to aid in the classification decision.

Table 5.4: Predicted performance of multimodal breast tumor classifier in the case of a single missing input.

Imputed	**AVG**	**True Accuracy**		**Predicted Accuracy**		
Input	**importance**	**Mean**	**STD**	**Mean**	**STD**	**RMSE**
MRI	0.368	0.708	0.010	0.685	0.007	0.025
Age	0.068	0.790	0.009	0.772	0.008	0.018
Max Intensity	0.050	0.793	0.009	0.778	0.008	0.016
Breast Density	0.069	0.792	0.009	0.772	0.008	0.020
Indication	0.446	0.724	0.010	0.662	0.007	0.063

Equation (5.1) is used to predict the performance using AVG importance of imputed inputs.

Once we had estimates for input importance of our model, we then used them to predict the model's performance for missing inputs. For non-categorical data, missing inputs were replaced with their mean value, while for categorical data, the most frequent category from the training set was used. We used accuracy as the Score(.) function in (5.1) during evaluation. For missing inputs, the imputation was done at the fusion layer. Non-categorical inputs were imputed with the mean value, whereas categorical inputs were replaced with the most frequent category value. To assess

the effectiveness of our approach, we measured the predicted accuracy using (5.1) and compared it with the computed accuracy value after imputation, referred to as true accuracy. We used 200 bootstrap realizations of the test set to obtain the mean and standard deviation of the prediction and true accuracy.

Table 5.4 presents the mean predicted and mean experimental accuracies for cases of test data with one missing input using the AVG estimated importances in (5.1). For the case of a single missing input, our proposed linear relationship is able to predict the model performance within less than 3% for most features and within 7% when Indication was missing.

The analysis was then applied to cases where more than one input was absent. We designed experiments that encompassed all possible permutations of present and absent inputs. For each experiment, we predicted the missing input model performance using our proposed linear relationship in (5.1), and compared the predicted performance with the actual performance on a test set, where the corresponding inputs were replaced with their mean values. Figure 5.5 illustrates these comparisons, contrasting the predictive and experimental missing input model performance for four distinct importance estimation methods. Each datum point on the plot represents an experiment with a unique combination of present and absent inputs.

Figure 5.6 shows the proportionality between AVG importance of missing inputs and degradation in model performance, supporting our hypothesis. We show results from our proposed linear relationship (5.1) and the best linear unbiased estimator (BLUE) [127] of the true drop in model accuracy. Our proposed linear relationship predicts that for missing inputs with cumulative importance of 0.1 normalized units (n.u.) the model's accuracy decreases from its reference value by 2.89%. This is similar to the BLUE prediction of a 3.22% drop in model accuracy. In approximately 70% of the experiments, the prediction of performance loss due to missing input falls within a 5% error margin as detailed in Table 5.5. For all experiments the predicted drop in model performance is highly correlated (ρ=0.92) with the missing input importance.

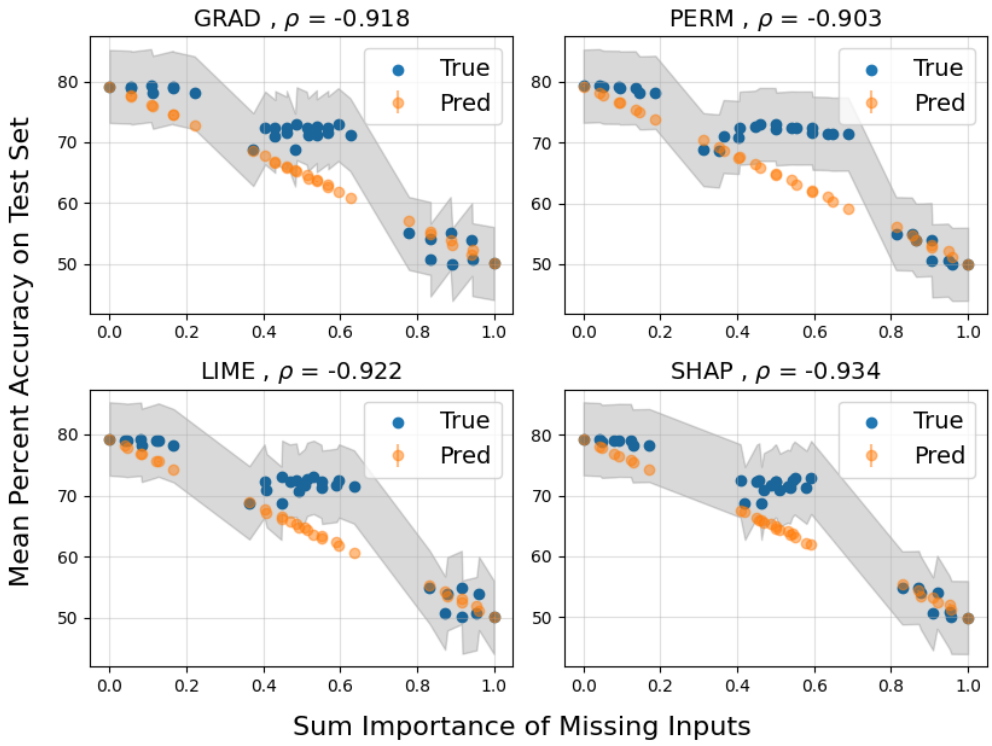


Figure 5.5: Comparison of predicted and true breast tumor classification performance reduction as a function of missing input importance using gradient (GRAD), permutation (PERM), LIME, and shapely values (SHAP). The Pearson correlation coefficient, ρ, is between the model test performance and aggregated importance of missing inputs.

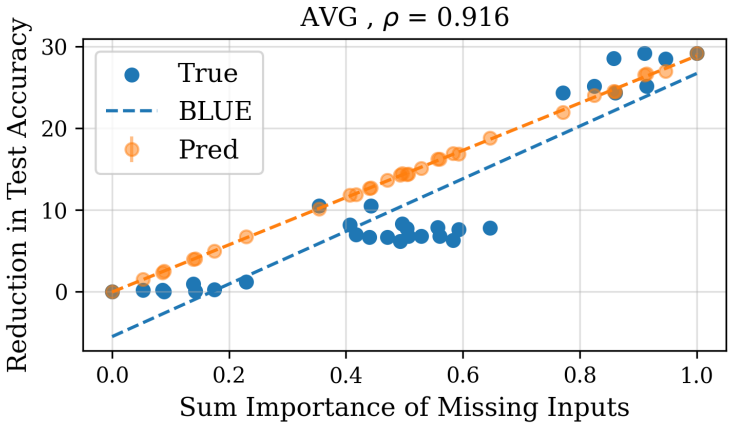


Figure 5.6: Breast tumor classification performance reduction as a function of missing input importance. This presents predictions, "Pred", using AVG method. BLUE represents the best linear fit of the true drop in model test accuracy. The Pearson correlation coefficient, ρ, between the drop in model test performance and the sum importance of missing inputs.

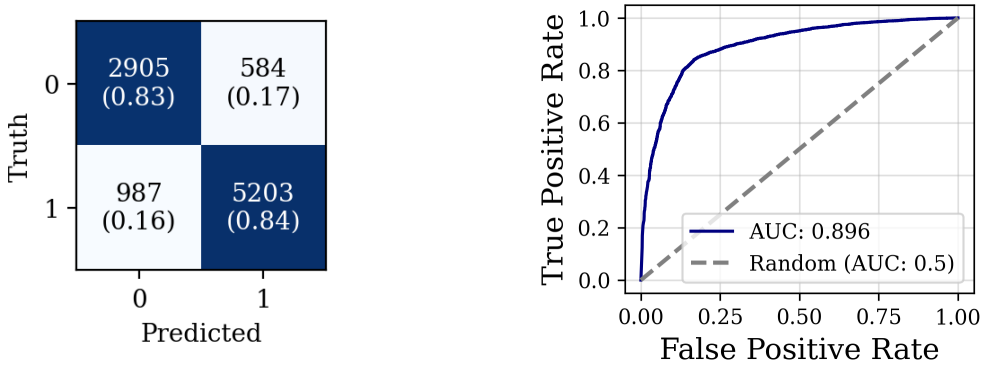
Table 5.5: Predicted and true performance of multimodal breast tumor classifier in the case of multiple missing input. Each row corresponds to a different experiment with a unique subset of missing inputs. The predicted accuracy is obtained using (5.1), and the experimental accuracy is the computed accuracy on an imputed test set.

Imputed input					Aggregated	True Accuracy		Predicted Accuracy		
Img.	**Age**	**MIP**	**B.Den.**	**Ind.**	**Importance**	**Mean**	**STD**	**Mean**	**STD**	**RMSE**
					0.000	0.792	0.009	0.792	0.009	0.004
				✗	0.446	0.725	0.010	0.666	0.006	0.060
			✗		0.069	0.792	0.009	0.772	0.008	0.020
			✗	✗	0.514	0.724	0.010	0.645	0.008	0.080
		✗			0.050	0.790	0.009	0.778	0.008	0.014
		✗		✗	0.496	0.725	0.010	0.647	0.007	0.078
		✗	✗		0.119	0.791	0.009	0.758	0.008	0.034
		✗	✗	✗	0.565	0.725	0.010	0.627	0.007	0.098
	✗				0.068	0.789	0.009	0.772	0.009	0.018
	✗			✗	0.513	0.716	0.010	0.646	0.008	0.070
	✗		✗		0.136	0.790	0.009	0.753	0.008	0.037
	✗		✗	✗	0.582	0.717	0.010	0.626	0.007	0.092
	✗	✗			0.118	0.781	0.009	0.758	0.008	0.025
	✗	✗		✗	0.564	0.715	0.010	0.628	0.007	0.087
	✗	✗	✗		0.186	0.782	0.010	0.737	0.008	0.045
	✗	✗	✗	✗	0.632	0.716	0.011	0.609	0.008	0.107
✗					0.368	0.689	0.010	0.689	0.007	0.011
✗				✗	0.814	0.549	0.011	0.559	0.009	0.015
✗			✗		0.436	0.689	0.011	0.669	0.007	0.023
✗			✗	✗	0.882	0.549	0.011	0.540	0.010	0.015
✗		✗			0.418	0.708	0.010	0.669	0.007	0.040
✗		✗		✗	0.864	0.540	0.011	0.539	0.010	0.009
✗		✗	✗		0.487	0.709	0.010	0.650	0.007	0.060
✗		✗	✗	✗	0.932	0.540	0.011	0.520	0.011	0.022
✗	✗				0.435	0.723	0.013	0.667	0.007	0.057
✗	✗			✗	0.881	0.506	0.010	0.540	0.009	0.034
✗	✗		✗		0.504	0.724	0.012	0.648	0.007	0.077
✗	✗		✗	✗	0.950	0.506	0.010	0.520	0.010	0.015
✗	✗	✗			0.486	0.728	0.009	0.650	0.007	0.079
✗	✗	✗		✗	0.931	0.501	0.011	0.521	0.010	0.020
✗	✗	✗	✗		0.554	0.731	0.010	0.631	0.007	0.100
✗	✗	✗	✗	✗	1.000	0.499	0.011	0.499	0.011	0.000

Key: Img=MRI image, MIP=MIP maximum intensity, B.Den=Breast density, Ind=Indication.

5.5.2 Cardiomegaly Classification

The trained model for cardiomegaly classification model shows good classification performance with an AUC of 0.896, a sensitivity of 0.84, and specificity of 0.83 on a hold-out test dataset, with an optimal threshold of 0.32. Figure 5.7 illustrates model performance on test data and Figure 5.8 shows sample images corresponding to classification outcomes of true positive, true negative, false positive, and false negative for the trained cardiomegaly classifier from the test set.



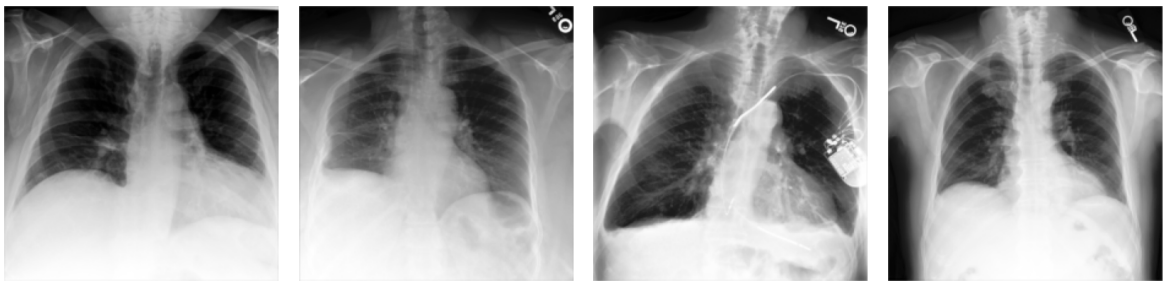
(a) Confusion matrix

(b) Receiver operating characteristic curve

Figure 5.7: Performance of the the trained cardiomegaly classifier on test set.

For the input importance estimates in Figure 5.9, the cosine similarity and RMSE values between the importance estimation methods are shown in Tables 5.6, demonstrating a strong consensus among the importance values estimated by the four distinct methods.

The importance estimation, validated by agreement, also aligns well with our intuition. The majority of the information regarding heart size is present in the radiographs, and while there



(a) True positive (b) True Negative (c) False Positive (d) False Negative

Figure 5.8: Examples of different classification outcomes of the trained cardiomegaly classifier on the test set.

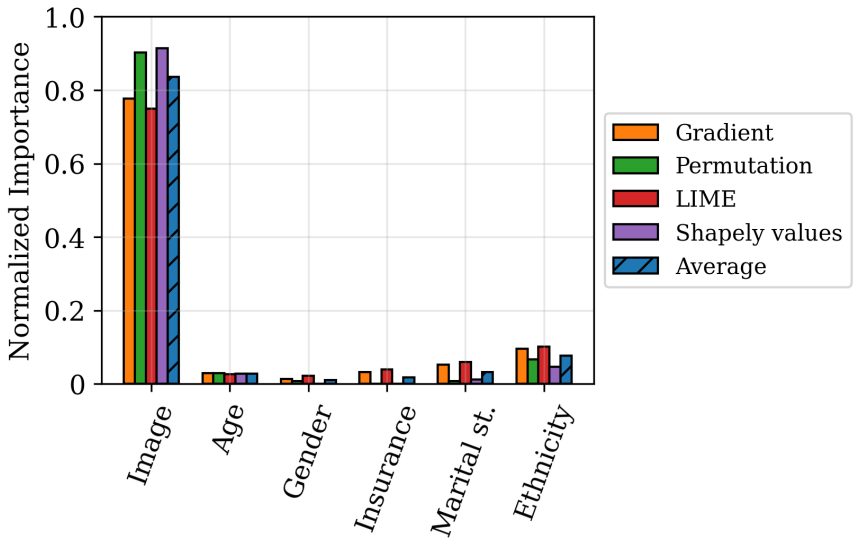


Figure 5.9: Comparison of normalized feature importance results and associated feature ranks using gradient, permutation, LIME, and shapely values based methods for the multimodal cardiomegaly classifier. AVG reports the mean importance across the four methods.

Table 5.6: Pairwise Cosine similarity and RMSE between normalized input importances estimated by different methods for the multimodal cardiomegaly classifier.

Cosine Similarity

	GRAD	PERM	LIME	SHAP	AVG
GRAD	1.00	1.00	1.00	1.00	1.00
PERM	1.00	1.00	0.99	1.00	1.00
LIME	1.00	0.99	1.00	0.99	1.00
SHAP	1.00	1.00	0.99	1.00	1.00
AVG	1.00	1.00	1.00	1.00	1.00

RMSE

	GRAD	PERM	LIME	SHAP	AVG
GRAD	0.00	0.06	0.01	0.06	0.03
PERM	0.06	0.00	0.07	0.01	0.03
LIME	0.01	0.07	0.00	0.08	0.04
SHAP	0.06	0.01	0.08	0.00	0.04
AVG	0.03	0.03	0.04	0.04	0.00

is not a causal relationship, certain ethnicities have been demonstrated to have elevated risks for cardiovascular diseases and hypertension [128, 129], which can be primary contributors to cardiomegaly [130, 131]. Consequently, these findings not only offer us an understanding of the inputs on which the models depend, but also help us calibrate our confidence in the model's prediction based on domain knowledge.

Next, we used the estimated importances to provide an additional layer of interpretability by understanding the model limitations under missing inputs. Table 5.7 illustrates the comparison between the actual and predicted model performance in cases with a single missing input. The predicted missing input model performance lies within 3% of the true value.

Table 5.7: Predicted performance of multimodal cardiomegaly classifier in the case of a single missing input.

Imputed	**AVG**	**True Accuracy**		**Predicted Accuracy**		
Input	**importance**	**Mean**	**STD**	**Mean**	**STD**	**RMSE**
Radiograph	0.836	0.646	0.004	0.671	0.004	0.025
Age	0.028	0.832	0.004	0.827	0.004	0.005
Gender	0.011	0.834	0.004	0.831	0.004	0.003
Insurance	0.018	0.831	0.004	0.829	0.004	0.003
Marital st.	0.033	0.834	0.004	0.827	0.004	0.007
Ethnicity	0.078	0.829	0.004	0.818	0.003	0.011

Equation (5.1) is used to predict the performance using AVG importance of imputed inputs.

For cases with multiple missing inputs, the experiments, each of which is represented by a point on the plots in Figure 5.10, generate model performance predictions that lie within 5% error margin. Comparison of predicted performance drop from (5.1) with predicted performance drop from BLUE of the true drop in model accuracy are illustrated in Figure 5.11. Equation (5.1) predicts that for missing inputs with cumulative importance of 0.1 normalized units (n.u.) the accuracy of cardiomegaly classifier will drop from its reference value by 1.93%. Compared to the BLUE prediction that for missing inputs with cumulative importance of 0.1 n.u. the models accuracy will drop from its reference value by 2.24%. In contrast to the breast tumor classification, the cardiomegaly dataset contains a single dominant input that governs the primary trend in model

performance, resulting in a high correlation between input importance and true missing input model performance. However, upon further examination of the two clusters of experiments in Figure 5.11 (those with and without radiographs), we find that the importance remains highly correlated with model performance even within the clusters. This further demonstrates that performance reduction is correlated with input importance.

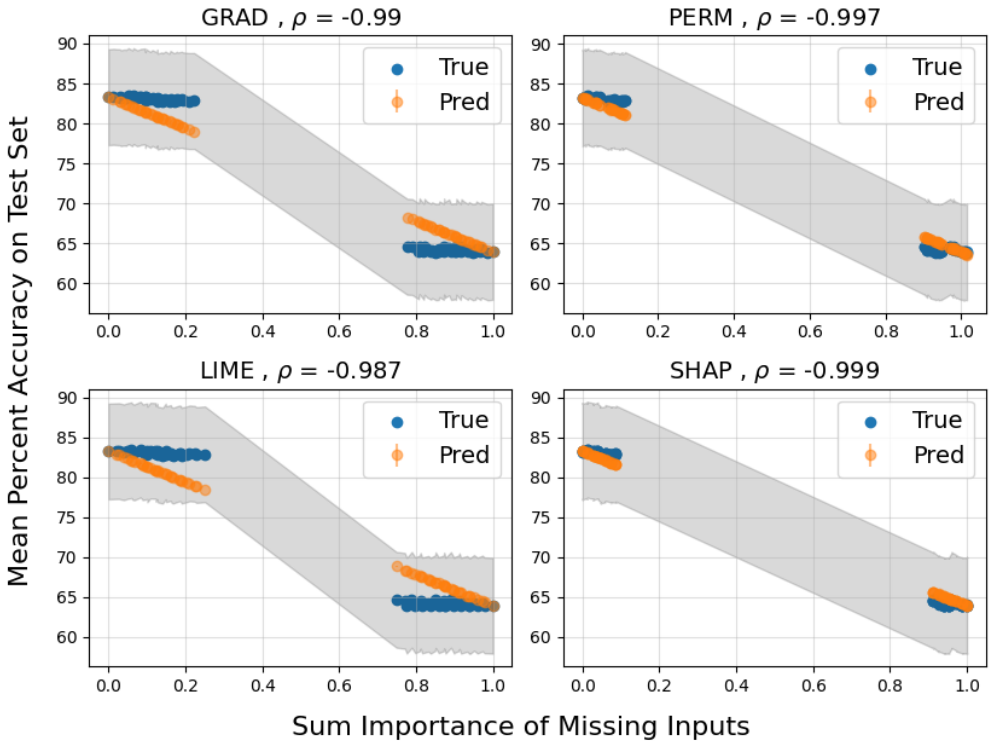


Figure 5.10: Comparison of predicted and true cardiomegaly classification performance reduction as a function of missing input importance in the case of one or more missing inputs using proposed methods. ρ is the Pearson correlation coefficient between the model test performance and aggregated importance of missing inputs.

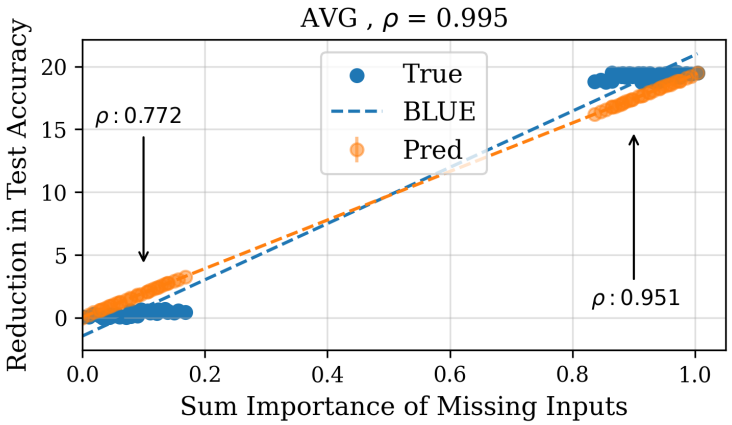


Figure 5.11: Cardiomegaly classification performance reduction as a function of missing input importance. Presents predictions, "Pred", using AVG importances. BLUE represents the best linear fit of the true drop in model test accuracy. ρ is the Pearson correlation coefficient between the drop in model test performance and sum importance of missing inputs.

5.6 Conclusion

In this study, we introduced a unified framework for estimating the importance of multimodal inputs in fusion-based multimodal neural networks. Previous interpretability work involving multimodal data had employed fixed feature extractors to obtain deep features from each modality [94, 132]. A novelty of our approach lay in the fact that the fusion model, including the feature extractors, was trained end-to-end for a specific task. Consequently, the features extracted were those fine-tuned, likely most pertinent, to a particular classification task.

Our unified multimodal input importance framework was agnostic to the type of estimation methods used, allowing us to utilize a range of importance estimation methods. Another strength of our proposed framework was that the importance estimates did not rely on the input data dimension, allowing us to compare, for example, the importance of a 224×224 image input to a 1×3 categorical input.

We addressed the challenge of validating the importance estimates by testing the framework in a controlled environment with synthetic data, custom decision functions, and complete control over the ground truth feature importance values. Our framework was then applied to provide insights into the decision-making logic of two multimodal classifiers trained to classify breast tumors and cardiomegaly from multimodal data. With this real data, we did not have ground truth

Table 5.8: Predicted and true performance of multimodal cardiomegaly classifier in the case of multiple missing input. Each row corresponds to a different experiment with a unique subset of missing inputs. The predicted accuracy is obtained using (5.1), and the experimental accuracy is the computed accuracy on an imputed test set.

Imputed input						Agg.	True Accuracy		Predicted Accuracy		
Img.	**Age**	**Gen.**	**Ins.**	**Mar.**	**Eth.**	**Imp.**	**Mean**	**STD**	**Mean**	**STD**	**RMSE**
						0.000	0.833	0.004	0.833	0.004	0.001
					✗	0.078	0.829	0.004	0.817	0.003	0.011
				✗		0.033	0.834	0.004	0.827	0.004	0.008
				✗	✗	0.111	0.830	0.004	0.811	0.003	0.019
			✗			0.018	0.831	0.004	0.829	0.004	0.003
			✗		✗	0.096	0.828	0.004	0.814	0.004	0.013
			✗	✗		0.051	0.833	0.004	0.823	0.004	0.010
			✗	✗	✗	0.129	0.829	0.003	0.808	0.003	0.021
		✗				0.011	0.833	0.004	0.831	0.004	0.003
		✗			✗	0.089	0.829	0.004	0.816	0.003	0.014
		✗		✗		0.044	0.834	0.004	0.824	0.004	0.010
		✗		✗	✗	0.122	0.830	0.004	0.809	0.003	0.021
		✗	✗			0.029	0.832	0.004	0.827	0.004	0.005
		✗	✗		✗	0.107	0.828	0.004	0.812	0.003	0.016
		✗	✗	✗		0.062	0.833	0.004	0.821	0.003	0.013
	✗	✗	✗	✗	✗	0.140	0.829	0.004	0.806	0.004	0.023
	✗					0.028	0.832	0.004	0.827	0.004	0.005
	✗				✗	0.106	0.829	0.004	0.813	0.003	0.016
	✗			✗		0.061	0.833	0.004	0.821	0.003	0.012
	✗			✗	✗	0.139	0.830	0.004	0.806	0.003	0.024
	✗		✗			0.046	0.832	0.004	0.824	0.004	0.008
	✗		✗		✗	0.124	0.827	0.004	0.809	0.003	0.019
	✗		✗	✗		0.079	0.833	0.003	0.818	0.003	0.015
	✗		✗	✗	✗	0.157	0.828	0.004	0.803	0.003	0.026
	✗	✗				0.039	0.833	0.004	0.825	0.004	0.008
	✗	✗			✗	0.117	0.829	0.004	0.810	0.004	0.019
	✗	✗		✗		0.072	0.834	0.004	0.819	0.004	0.015
	✗	✗		✗	✗	0.150	0.830	0.003	0.804	0.003	0.026
	✗	✗	✗			0.057	0.832	0.004	0.821	0.004	0.010
	✗	✗	✗		✗	0.135	0.827	0.004	0.806	0.003	0.021
	✗	✗	✗	✗		0.090	0.832	0.004	0.815	0.004	0.017
	✗	✗	✗	✗	✗	0.168	0.830	0.004	0.801	0.003	0.029
✗						0.836	0.646	0.004	0.671	0.004	0.025
✗					✗	0.913	0.646	0.004	0.657	0.005	0.011
✗				✗		0.869	0.642	0.005	0.665	0.004	0.023
✗				✗	✗	0.947	0.642	0.004	0.649	0.005	0.007
✗			✗			0.854	0.647	0.004	0.668	0.004	0.021

Table 5.8 (cont'd.).

Imputed input						Agg.	True Accuracy		Predicted Accuracy		
Img.	**Age**	**Gen.**	**Ins.**	**Mar.**	**Eth.**	**Imp.**	**Mean**	**STD**	**Mean**	**STD**	**RMSE**
✗			✗		✗	0.932	0.646	0.004	0.652	0.005	0.007
✗			✗	✗		0.887	0.642	0.004	0.661	0.004	0.019
✗			✗	✗	✗	0.965	0.643	0.005	0.646	0.005	0.004
✗		✗				0.846	0.645	0.004	0.669	0.004	0.024
✗		✗			✗	0.924	0.645	0.004	0.654	0.005	0.009
✗		✗		✗		0.880	0.642	0.004	0.663	0.004	0.021
✗		✗		✗	✗	0.957	0.642	0.004	0.648	0.005	0.006
✗		✗	✗			0.864	0.645	0.004	0.665	0.004	0.020
✗		✗	✗		✗	0.942	0.645	0.004	0.650	0.005	0.005
✗		✗	✗	✗		0.898	0.642	0.004	0.660	0.004	0.018
✗		✗	✗	✗	✗	0.975	0.642	0.004	0.644	0.005	0.003
✗	✗					0.864	0.639	0.005	0.665	0.004	0.026
✗	✗				✗	0.942	0.640	0.005	0.651	0.005	0.011
✗	✗			✗		0.897	0.640	0.005	0.660	0.004	0.020
✗	✗			✗	✗	0.975	0.640	0.005	0.645	0.005	0.005
✗	✗		✗			0.882	0.639	0.005	0.662	0.004	0.023
✗	✗		✗		✗	0.960	0.640	0.005	0.648	0.005	0.008
✗	✗		✗	✗		0.915	0.640	0.005	0.656	0.004	0.016
✗	✗		✗	✗	✗	0.993	0.640	0.005	0.641	0.005	0.001
✗	✗	✗				0.875	0.640	0.005	0.664	0.004	0.024
✗	✗	✗			✗	0.953	0.639	0.005	0.649	0.004	0.009
✗	✗	✗		✗		0.908	0.639	0.005	0.657	0.004	0.018
✗	✗	✗		✗	✗	0.986	0.639	0.005	0.642	0.005	0.003
✗	✗	✗	✗			0.893	0.640	0.004	0.660	0.004	0.021
✗	✗	✗	✗		✗	0.971	0.639	0.005	0.645	0.005	0.006
✗	✗	✗	✗	✗		0.926	0.640	0.005	0.654	0.005	0.014
✗	✗	✗	✗	✗	✗	1.000	0.639	0.005	0.638	0.005	0.001

Table continued. Key : Agg. Imp.=Aggregrated Importance, Img=Chest radiograph, Gen=Gender, Ins=Insurance, Mar=Marital status, Eth=Ethnicity

feature importance knowledge and therefore validated our importance estimates by quantifying the agreement across estimates returned by different methods. Furthermore, the estimated AVG importances aligned well with expert intuition and passed the validation by agreement test.

We further enhanced the model's interpretability by using the estimated importances to predict the model's performance in the special case of missing inputs. Our goal was to provide non-technical users with an understanding of the model's prediction reliability in terms of accuracy.

We hypothesized that the degradation in model performance in the absence of certain inputs was proportional to the importance of those inputs. We designed numerous experiments to test how closely our prediction of the model performance aligned with the true model performance in the absence of inputs. Our results across two different multimodal datasets and two different fusion-based classifiers showed a high correlation between the importance of missing inputs and model performance, supporting our hypothesis. A limitation of our approach was the use of a linear relationship between input importance and missing input model performance, which might not adequately capture the combined importance of inputs. Despite this limitation, we consistently observed a high correlation between input importance and missing input model performance. Future work could explore different non-linear relationships. This study represented a step towards providing an additional layer of understanding of the model's limitations and operational capabilities. It also aided in answering questions related to cost-benefit analysis, such as the value of acquiring additional input data on a patient when the performance degradation might be minimal.

CHAPTER 6

TOWARDS SAMPLE-LEVEL RELIABILITY ESTIMATION

6.1 Introduction

In Chapter 5, we discussed the use of model accuracy as a human-interpretable measure of model reliability. However, this analysis was conducted on set-level, wherein these metrics were estimated across the entire validation dataset, failing to offer individual sample-specific insights. Traditional deep network designs yield sample-specific predictions without accompanying reliability measures. Ideally, we would like to obtain sample-specific reliability estimates to quantify confidence in each individual prediction.

Various approaches have been proposed to obtain sample-wise measures of prediction reliability that align with accuracy. Bayesian neural networks induce probabilistic outputs by placing prior distributions over network weights and propagating this uncertainty through to predictions [133]. Dropout sampling at test time enables uncertainty approximation through Monte Carlo simulation by running predictions on multiple dropout masked versions of the model [134]. Conformal prediction provides a distribution-free framework to derive prediction intervals guaranteed to contain new samples at a specified confidence level based on a calibration set [135].

Existing measures of sample-level reliability often rely on specific architectural modifications in the model or generate variance estimates that are less intuitive and interpretable, especially for non-experts. Therefore, in this analysis, we aim to develop a sample-level reliability metric that is straightforward and understandable. To this end, we use local accuracy as an interpretable and tangible measure of reliability at the sample level. Local accuracy simply conveys the empirical performance of the model in the vicinity of a given sample, providing an accessible reliability quantification.

The methods discussed in this chapter aim to map properties of the input sample to local accuracy through calibration techniques. The key insight is that model performance depends heavily on the data sample and can vary greatly across different regions of the input space. While performance metrics like accuracy, AUC, and cross-entropy provide a set-level average view, they fail to account

for this variability. A model may produce reliable predictions in some areas of the input space while faltering in others. Intuitively, if a model makes accurate predictions in the region around a given sample, then it is likely to be accurate on that individual sample as well.

In this analysis we generate reliability estimates using local accuracy about a sample, that are transparent and meaningful to all users. We analyze model performance across different sample populations segmented based on sample properties. Calibrating these sample-specific attributes could produce granular reliability estimates tied to local accuracy. In summary, while the set-level performance prediction offers useful insights into model reliability, sample-specific analysis is needed. We aim to transition from set-level reliability to sample-level reliability which could increase model transparency, evaluation rigor, and safety for real-world deep learning systems.

6.2 Methods

We defined the reliability of a model's prediction for a single sample as the *local accuracy of the model in a small neighborhood around that sample*. The key intuition being that samples surrounded by other samples on which the model performs accurately are likely to also be classified correctly. To construct neighborhoods, we used samples that were similar to the target sample based on a pre-defined property of the input for example distance in the feature space, cosine similarity, or predicted probability. We then took samples within a radius threshold on that metric to form the neighborhood. By averaging model accuracy on those neighborhood samples, we generated a local reliability estimate for the target sample.

Through this process, we constructed a reliability calibration curve (RCC) that relates the pre-defined sample property to local accuracy offering a nuanced insight into model performance by providing fine-grained reliability estimates for individual samples based on model performance in local neighborhoods. It is important to note that these RCCs can be constructed using any performance metric. However, we specifically chose accuracy due to its intuitive and accessible nature, making it easily interpretable not just for experts, but for laypeople as well.

We explored four main approaches for constructing RCCs, each depending on different properties of the input sample.

1. Mahalanobis distance: we calculated the Mahalanobis distance of each sample from the distribution of the training set in fusion feature space. The Mahalanobis-based RCC maps this distance to local accuracy.

2. Cosine similarity: we computed the cosine similarity between the fusion features for each sample and the training set. The cosine similarity-based RCC maps this similarity metric to local accuracy.

3. UMAP dimensionality reduction: we used the Uniform Manifold Approximation and Projection (UMAP) method, where the encoded features of the multimodal input from the fusion layer were projected down to low dimensional space. The resultant calibration curve maps distance of the sample from the training data in the UMAP-reduced feature space to local accuracy.

4. Prediction probabilities: we used the model's prediction probabilities to construct neighborhoods and build a RCC that maps the prediction probability to local accuracy.

The RCCs were learned using validation data by regressing local accuracy against the chosen sample property across varied neighborhood sizes. Their performance was evaluated using a hold out test set. We evaluated the RCCs on the following criteria:

- Granularity: The curve should account for a wide range of local accuracy values, providing more fine-grained reliability estimates.

- Convergence: As neighborhood size increases, the predicted local accuracy should approach the global accuracy on the full validation set.

- Generalization: We quantified generalization via the RMSE between the predicted local accuracy from the RCC and the true local accuracy on a holdout test set.

6.2.1 Mahalanobis Distance-based Reliability Calibration Curve (M-RCC)

Mahalanobis distance is a multivariate generalization of measuring the number of standard deviations a point is away from the mean of a distribution [136]. It equals zero when a point lies at the distribution mean and grows as the point moves away along the principal component axes.

Mahalanobis distance has been used to successfully identify out-of-distribution or distribution-shifted samples by quantifying distance in the input or feature space [137]. The key intuition is that larger Mahalanobis distances indicate dissimilarity from the training distribution.

Continuing with the notation introduced in Chapter 4, we define the Mahalanobis distance of a test sample $\mathbf{Z}$ from the set of encoded multimodal training inputs $\mathcal{S}_Z$, in the fusion layer, as

$$D_M(\mathbf{Z}, \mathcal{S}_Z) = \sqrt{(\mathbf{Z} - \overline{\mathbf{Z}})^T \Sigma_Z^{-1} (\mathbf{Z} - \overline{\mathbf{Z}})}, \tag{6.1}$$

where $\overline{\mathbf{Z}}$ is the mean of samples in the set $\mathcal{S}_Z$ given by

$$\overline{\mathbf{Z}} = \frac{1}{N} \sum_{\mathbf{Z} \in \mathcal{S}_Z} \mathbf{Z}, \tag{6.2}$$

and Σ_Z is the covariance matrix for samples in the set $\mathcal{S}_Z$ given by

$$\Sigma_Z = \frac{1}{N-1} \sum_{\mathbf{Z} \in \mathcal{S}_Z} (\mathbf{Z} - \overline{\mathbf{Z}})(\mathbf{Z} - \overline{\mathbf{Z}})^T. \tag{6.3}$$

We computed the Mahalanobis distance between the multimodal fusion features of a test sample and the distribution of the training set fusion features providing a sample-specific measure of how well the model's internal representation aligns with the training data. We used these distances to construct neighborhoods around the test sample for estimating local model accuracy.

6.2.2 Cosine Similarity-based Reliability Calibration Curve (C-RCC)

The cosine similarity between a test sample $\mathbf{Z}$ and the mean $\overline{\mathbf{Z}}$ of the encoded training set $\mathcal{S}_Z$, in the fusion layer, is given by:

$$D_C(\mathbf{Z}, \mathcal{S}_Z) = \frac{\mathbf{Z} \cdot \overline{\mathbf{Z}}}{\|\mathbf{Z}\|_2 \|\overline{\mathbf{Z}}\|_2}, \tag{6.4}$$

where the numerator denotes the dot product between test sample and mean of the training set and $\|.\|_2$ denotes the 2-norm of a vector.

We used the cosine similarity between the input sample at inference time and the training data representations in the fusion layer of the neural network. The basic motivation being that test points located in sparse regions of the input space, far from the bulk of training data, will likely yield less reliable predictions [138]. This indicates that a functional relationship exists between the sample similarity and local model performance. The cosine similarity was therefore used to build C-RCC.

6.2.3 UMAP-based Reliability Calibration Curve (U-RCC)

UMAP is a non-linear dimensionality reduction method [139]. Its goal is to find a low-dimensional embedding of the data that best preserves the global topological structure of the high-dimensional input data. We define the sets

$$\begin{aligned} \mathcal{S}_Z: &\quad \text{set of high-dimensional input data points,} \\ \mathcal{S}_z: &\quad \text{set of low-dimensional embeddings.} \end{aligned}$$

UMAP finds a low-dimensional representation $\mathbf{z} \in \mathcal{S}_z$ of the high-dimensional data $\mathbf{Z} \in \mathcal{S}_Z$ that preserves global data structure. This is achieved by first constructing a graph in the high-dimensional space. For each sample, the nearest neighbors are computed and weights are assigned to the graph edges connecting the sample and its neighbors. Weights w_{ij}^{high} between two samples $\mathbf{Z}_i$ and $\mathbf{Z}_j$ are calculated as

$$w_{ij}^{high} = \exp\left(- \frac{d(\mathbf{Z}_i, \mathbf{Z}_j) - \rho_i}{\sigma_i} \right), \tag{6.5}$$

where $d(\mathbf{Z}_i, \mathbf{Z}_j)$ is the distance between points, ρ_i controls the local neighborhood size, and σ_i controls the fuzziness of neighborhoods.

Next, a graph in the low-dimensional space in constructed. For the low-dimensional graph, the weights are computed as

$$w_{ij}^{low} = \frac{1}{1 + a \cdot d(\mathbf{z}_i, \mathbf{z}_j)^{2b}}, \tag{6.6}$$

where, a and b are hyperparameters of UMAP, $d(.,.)$ is a distance function, and $\mathbf{z}_i$ and $\mathbf{z}_j$ are low-dimensional representations of the ith and jth sample respectively. In order to learn the low-dimensional representation, the cross-entropy loss between the high-dimensional and low-

dimensional graphs, given by

$$\mathcal{L} = \sum_{i,j} w_{ij}^{high} \log\left(\frac{w_{ij}^{high}}{w_{ij}^{low}}\right) + (1 - w_{ij}^{high}) \log\left(\frac{1 - w_{ij}^{high}}{1 - w_{ij}^{low}}\right), \tag{6.7}$$

is minimized using stochastic gradient descent. Rather than quantifying distances in the original fusion feature space, we first projected the fusion features into a lower-dimensional space that preserved the local structure of the data. We used UMAP to project the fusion features into a lower-dimensional space and then computed distances between the input sample and training data in this UMAP-reduced space to construct the UMAP-based reliability calibration curve (U-RCC). The UMAP induced distance between a test sample $\mathbf{Z}$ and the mean $\overline{\mathbf{Z}}$ of the encoded training set $\mathcal{S}_Z$, in the fusion layer, is given by

$$D_U(\mathbf{Z}, \mathcal{S}_Z) = \|\mathbf{Z} - \overline{\mathbf{Z}}\|_2, \tag{6.8}$$

where the numerator denotes the dot product between test sample and mean of the training set and $\|.\|_2$ denotes the 2-norm of a vector.

6.2.4 Prediction Probability-based Reliability Calibration Curve (P-RCC)

Most neural network-based classifiers use the softmax function in the final layer to generate probabilities of class labels. Given a vector $\mathbf{V} \in \mathbb{R}^n$ from the last layer of an n-class classifier, the softmax function generates a vector $P \in \mathbb{R}^n$ of probability distribution over a list of model outputs, where for all $i = 1, \cdots, n$ the entries of P are given by

$$p_i = Softmax(V_i) = \frac{e^{V_i}}{\sum_{j=1}^{n} e^{V_j}}, \tag{6.9}$$

where $P = \begin{bmatrix} p_1 & \cdots & p_n \end{bmatrix}^T$, all elements of the resultant vector lie in the range $(0, 1)$, and $\sum_{i=1}^{n} p_i = 1$.

We used the prediction probabilities generated by the model as the sample-specific attribute for generating the P-RCC.

6.2.5 Model Setup and Data

For evaluating our RCCs, we utilized the same data, model setup, and classification problems as described in Chapter 4 and detailed in Tables 4.2 and 4.3. This provides a controlled environment

covering a variety of classification problems with existing trained models that can be readily used to test our new approach.

To construct a RCC for a classifier, first we defined a sample-level property $Prop$ that we wish to calibrate. Where $Prop$ can be Mahalanobis distance-based (D_M) described in (6.1), cosine similarity-based (COS) described in (6.4), UMAP-based (D_U) described in (6.8), or prediction probability-based (P) described in (6.9).

Using a validation set $\mathcal{V}_X$, we stratified or binned the samples based on their $Prop$ values. Within each bin, we computed the local accuracy (Acc_l) and the average $Prop$ value. We also computed the weights w associated with each bin using the density of samples in the bin. The ordered pairs ($Prop, Acc_l$) of average property value and local accuracy in a bin were generated for all bins. The ordered pair data along with the weights were then used to fit a calibration curve h. This process was repeated for N_p bootstrap iterations of the validation set over a variety of bin sizes N_{bin} generating $N_p \times N_{bin}$ calibration curves. RCC was then generated as the mean curve using the calibration curves ($h_{i,j}$) where i represents ith bootstrap iteration and j represents jth bin size. RCC mapping can be represented as

$$RCC : \mathbf{X} \rightarrow \text{Accuracy}\left(\mathcal{N}_{Prop}(\mathbf{X})\right), \tag{6.10}$$

where $\mathcal{N}_{prop}$ is a neighborhood in $Prop$ about $Prop(\mathbf{X})$. The bootstrapping also provides confidence intervals around the generated RCC. An overview of the RCC generation approach is given in Algorithm 6.1.

6.3 Results

As described earlier, we used three main metrics for evaluating the generated RCCs: granularity, convergence, and generalizability. All the RCCs satisfy the convergence property because for a single bin (all neighboring samples that share similar sample property), local accuracy is equal to the set accuracy. Therefore, we focus the following discussion on assessing the generalization and granularity of the RCCs. Figure 6.1 illustrates the calibration curves generated for the classification problems highlighted in Table 6.1. The red dots, representing local neighborhoods, are ordered

Algorithm 6.1 RCC

Input: Validation set $\mathcal{V}_X$, sample property $Prop$, number of bootstrap iterations (N_p), range of neighborhood sizes (N_{bins}).
for $i \leftarrow 1$ to N_p **do**
  Take a random subset $\mathcal{V}_X^i$ of the validation set.
  for $j \leftarrow 1$ to N_{bins} **do**
    Generate histogram of $Prop$ values with j bins for $\mathcal{V}_X^i$.
    Calculate local accuracy (Acc_l) for samples in each bin.
    for $k \leftarrow 1$ to j **do**
      Generate ordered pairs of average $Prop$ value and local accuracy in the kth bin $(Prop, Acc_l)_k$
      Calculate weights (w_k) corresponding to each ordered pair as the density of samples in the bin.
    end for
    Use the ordered pairs $(Prop, Acc_l)_k$ weighted by w_k to fit a curve $h_{i,j}$ for ith bootstrap with j bins.
  end for
end for
Use the curves $h_{i,j}$ to compute the mean RCC and the 95% confidence interval.
return RCC

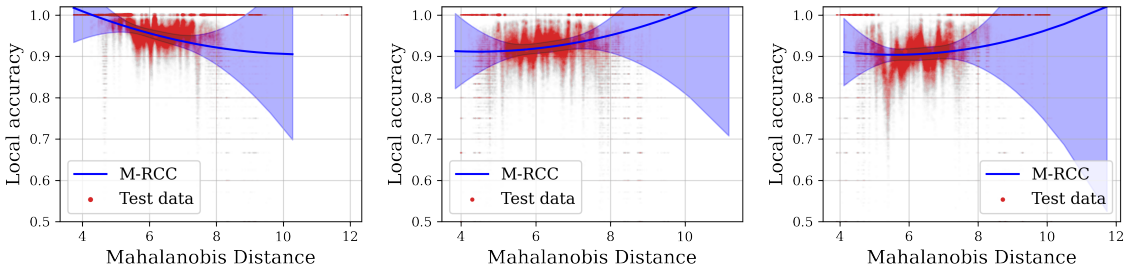


Figure 6.1: Generated M-RCCs for problems 1,7, and 8 (L-R) in Table 6.1. The mean calibration curve, depicted by the blue line, is constructed using validation data, the blue shaded region represents a 95% confidence interval. The red dots represent pairs of average Mahalanobis distance and local accuracy of test data, calculated over local neighborhoods, repeated for multiple bootstrap iterations. The test data is weighted based on the density of samples in the neighborhood.

Table 6.1: Validation results of the Mahalanobis distance based reliability calibration curve (M-RCC).

id	Classification problem	Average test accuracy	Granularity		Generalization RMSE
			min local acc.	max local acc.	
1	$\|\|Z_1\|\|_1$	0.95	0.90 ± 0.117	1.00 ± 0.043	0.0379
7	$\sum_{i=1}^{4} \|\|Z_i\|\|_1$	0.91	0.91 ± 0.018	1.00 ± 0.167	0.0496
8	$e^{Z_{2,7}} \ln\left(Z_{1,1} + Z_{1,2}\right)^2$	0.90	0.90 ± 0.008	1.00 ± 0.253	0.0532

The minimum and maximum local accuracies are reported ± standard deviation.

pairs of property value and local accuracy derived from bootstrapping the test set. These pairs are weighted according to the sample density in each neighborhood. The solid blue line depicts the RCC generated from the validation set using Algorithm 6.1, and the shaded area corresponds to a 95% confidence interval.

Generalization of the RCC is evaluated by calculating the RMSE between the ordered pairs derived from the test data and the mean calibration curve. Table 6.1 shows results for the Mahalanobis-based RCC. While the curve generalizes reasonably to the holdout test set, its granularity is limited. This means it does not reveal a wide range of local accuracy values and cannot provide fine-grained reliability measures. This suggests the Mahalanobis distance of a sample from the training set is not highly representative of the model's local performance trends. While Mahalanobis distance has been successfully used to detect distribution shifts [137], it does not reveal local performance trends well.

Similarly, the cosine similarity-based curves fail to capture informative local trends, with most curves centered around the set-level accuracy in Figure 6.2. These similarity-based approaches are often better suited for detecting out-of-distribution samples.

The UMAP-based RCCs demonstrate good generalization in Table 6.3 but, like other distance-based methods, suffer from lack of granularity. The UMAP-projected fusion features in Figure 6.3 show that UMAP preserves the discriminative power of the fusion features, as the class-labeled plots remain separated. Since we construct U-RCC using the sample distance from the center of training distribution, we see the local accuracy increase with greater distances.

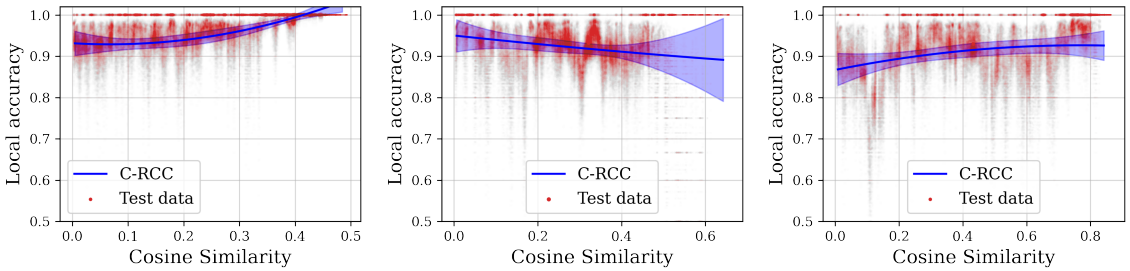


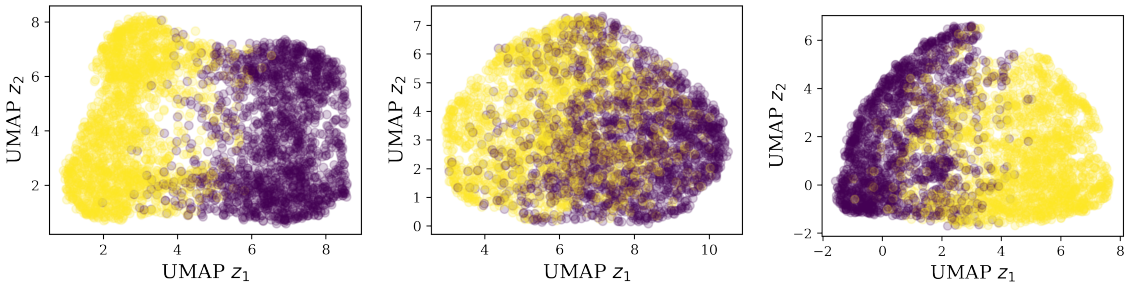
Figure 6.2: Generated C-RCCs for problems 1,7, and 8 (L-R) in Table 6.2 The mean calibration curve, depicted by the blue line, is constructed using validation data, the blue shaded region represents a 95% confidence interval. The red dots represent pairs of average cosine similarity and local accuracy of test data, calculated over local neighborhoods, repeated for multiple bootstrap iterations. The test data is weighted based on the density of samples in the neighborhood.

Table 6.2: Validation results of the cosine similarity based calibration curve (C-RCC).

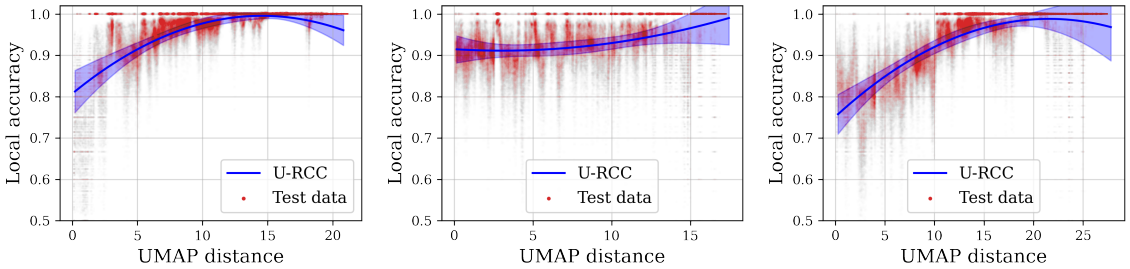
id	**Classification problem**	**Average test accuracy**	**Granularity**		**Generalization RMSE**
			min local acc.	**max local acc.**	
1	$\|\|Z_1\|\|_1$	0.95	0.93 ± 0.007	1.00 ± 0.013	0.0347
7	$\sum_{i=1}^{4} \|\|Z_i\|\|_1$	0.91	0.89 ± 0.051	0.95 ± 0.020	0.0542
8	$e^{Z_{2,7}} \ln\left(Z_{1,1} + Z_{1,2}\right)^2$	0.90	0.87 ± 0.020	0.93 ± 0.013	0.0637

Figure 6.4 shows the generated RCCs using the model predicted probability for each sample. Our model outputs class probabilities from the softmax layer. The results demonstrate that in addition to reasonable generalization, the P-RCC based on softmax probabilities exhibits substantially more granularity compared to the previously discussed distance and similarity-based methods. This can be attributed to the fact that those approaches measured distances and similarities in the fusion layer, while the probability-based PRCC leverages the outputs from the final layer of the model. Therefore, it takes advantage of the full discriminative power of the end-to-end architecture trained specifically for this task.

Since P-RCC significantly outperforms the previous metrics, we present full results for all explored cases in Table 6.4. The probability-based RCCs is a valuable tool that can enable fine-grained reliability quantification and interpretability compared to distance/similarity-based approaches in the fusion layer.



(a) Visualization of UMAP-reduced fusion features for problems 1,7, and 8 (L-R) in Table 6.3. Samples from all four input modalities are reduced to two discriminative features z_1 and z_2, colored by class label. The UMAP preserves global and local structure, maintaining class separation. The distance-based calibration curve below uses the UMAP distance of a test sample from the training distribution center. As distance increases towards the extremes, class separation improves and local accuracy increases.

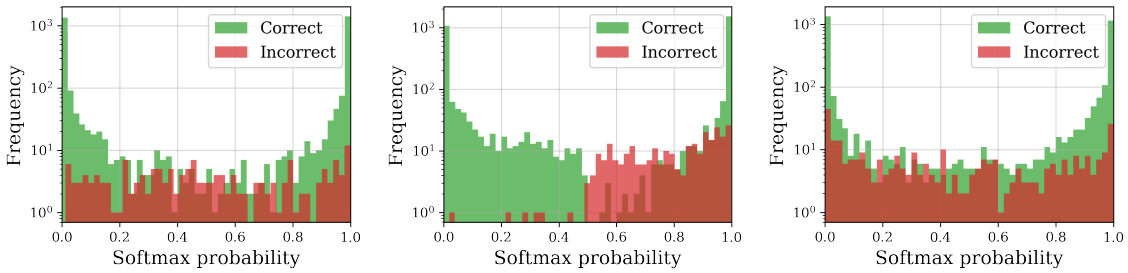


(b) Generated U-RCCs for problems 1,7, and 8 (L-R) in Table 6.3. The mean calibration curve, depicted by the blue line, is constructed using validation data, the blue shaded region represents a 95% confidence interval. The red dots represent pairs of average UMAP distance and local accuracy of test data, calculated over local neighborhoods, repeated for multiple bootstrap iterations. The test data is weighted based on the density of samples in the neighborhood.

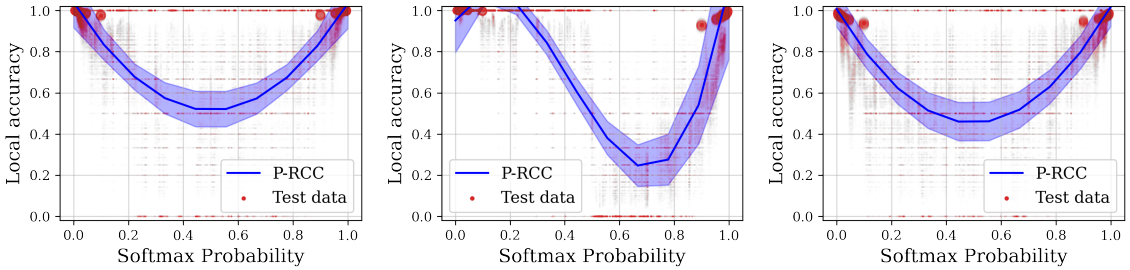
Figure 6.3: RCCs based on the euclidean distance of a sample from mean of training data in UMAP-projected space.

Table 6.3: Validation results of the UMAP-based reliability calibration curve (U-RCC).

id	**Classification problem**	**Average test accuracy**	**Granularity**		**Generalization RMSE**
			min local acc.	**max local acc.**	
1	$\|\|Z_1\|\|_1$	0.95	0.81 ± 0.026	1.00 ± 0.003	0.0461
7	$\sum_{i=1}^{4} \|\|Z_i\|\|_1$	0.91	0.91 ± 0.008	0.99 ± 0.033	0.0496
8	$e^{Z_{2,7}} \ln\left(Z_{1,1} + Z_{1,2}\right)^2$	0.90	0.76 ± 0.024	0.99 ± 0.013	0.0539



(a) Histogram of softmax probabilities generated by models for problems 1,7, and 8 (L-R) in Table 6.4. The bins of the histogram represent local neighborhoods in the validation set.



(b) Generated P-RCCs for for problems 1,7, and 8 (L-R) in Table 6.4. The mean calibration curve, depicted by the blue line, is constructed using validation data, the blue shaded region represents a 95% confidence interval. The red dots represent pairs of average softmax probability and local accuracy of test data, calculated over local neighborhoods, repeated for multiple bootstrap iterations. The test data is weighted based on the density of samples in the neighborhood.

Figure 6.4: RCCs based on model prediction probability.

Table 6.4: Validation results of the prediction probability based calibration curve (P-RCC).

id	**Classification problem**	**Average test accuracy**	**Granularity**		**Generalization**
			min local acc.	**max local acc.**	**RMSE**
1	$\|\|Z_1\|\|_1$	0.95	0.52 ± 0.044	1.00 ± 0.062	0.0882
2	$\|\|Z_2\|\|_1$	0.93	0.59 ± 0.034	1.00 ± 0.052	0.0844
3	$\|\|Z_3\|\|_1$	0.99	0.68 ± 0.065	1.00 ± 0.048	0.0667
4	$\|\|Z_4\|\|_1$	0.99	0.74 ± 0.066	1.00 ± 0.042	0.0670
5	$\sum_{i=1,2} \|\|Z_i\|\|_1$	0.92	0.53 ± 0.034	1.00 ± 0.057	0.1040
6	$\sum_{i=3,4} \|\|Z_i\|\|_1$	0.99	0.68 ± 0.074	1.00 ± 0.056	0.0669
7	$\sum_{i=1}^{4} \|\|Z_i\|\|_1$	0.91	0.25 ± 0.052	1.00 ± 0.177	0.1163
8	$e^{Z_{2,7}} \ln\left(Z_{1,1} + Z_{1,2}\right)^2$	0.90	0.46 ± 0.047	1.00 ± 0.049	0.0992

6.4 RCCs in the Case of Missing Inputs

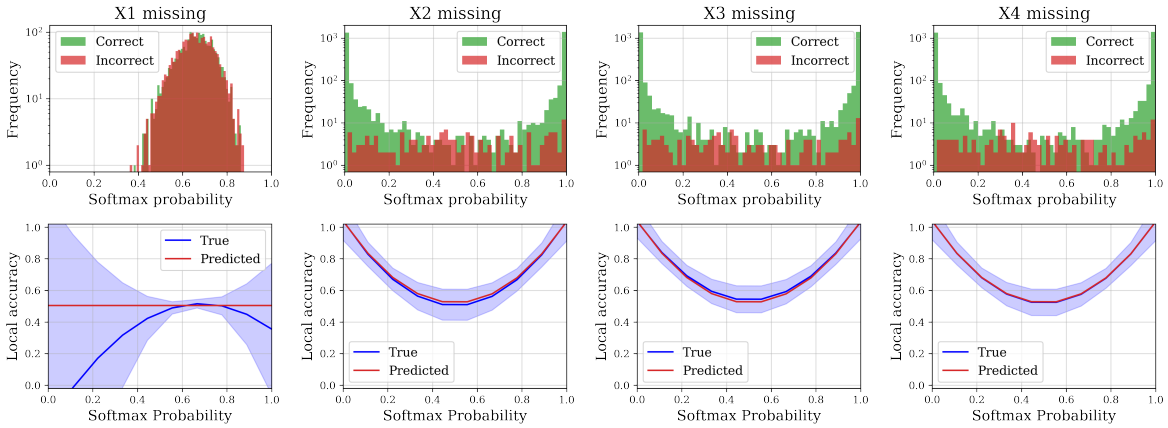
After constructing and evaluating the reliability curves, we used them to extend the analysis to missing data scenarios. Specifically, we proposed a modified version of (5.1) to generate sample-level reliability estimates in the case of missing inputs:

$$\text{Local Score}_{\mathbf{X}_k} = \text{Local Score}_{\Phi} - \sum_{k \in \mathcal{K}} \overline{\text{imp}}(\mathbf{X}_k)(\text{Local Score}_{\Phi} - \text{Score}_{\mathbf{X}_1:\mathbf{X}_m}). \tag{6.11}$$

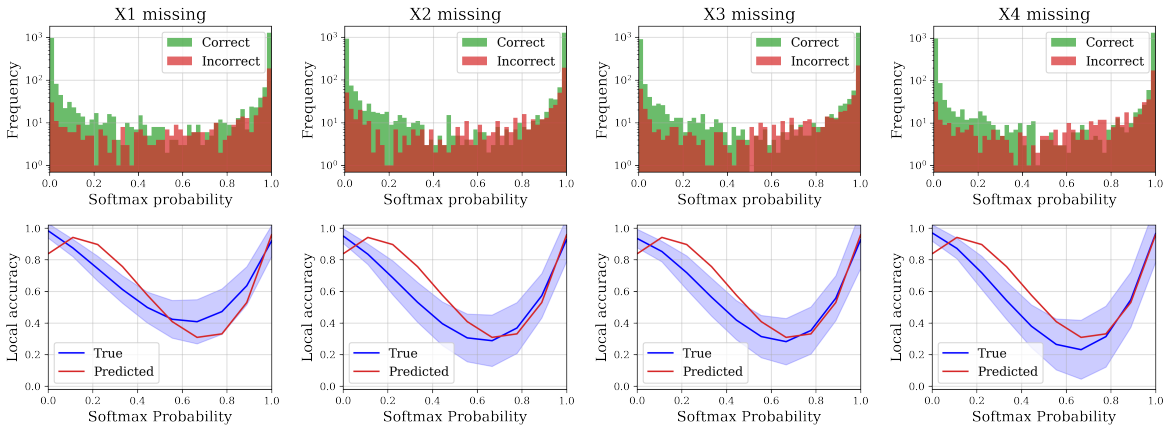
Where the reference local score without missing data came from the RCC. In this formulation, the local accuracy with missing input X_1 is proportional to the importance of X_1 scaled by the reference local accuracy. This allowed us to estimate the impact of missing data on a sample at inference time using the multimodal input importance and RCC.

Using the same framework as before, we estimated P-RCCs for missing input cases across the classifiers trained on problems in Table 6.4. To estimate the P-RCC with missing inputs, we used (6.11), relating local accuracy to the importance of the missing input. We used this to predict how the calibration curve would change for different missing inputs. We compared the estimated P-RCC to the true P-RCC generated on modified validation data where the missing input was mean-imputed. We perform this analysis for each classifier, removing one input at a time and imputing it with the mean value.

Figure 6.5 shows results for a subset of classification problems, clearly demonstrating that the model prediction probability distribution and the calibration curve behavior change drastically when an important input is missing, while remaining relatively unchanged for less important inputs. Our estimated P-RCC aligns well with the true P-RCC. This supports our initial intuition that the drop in local model performance, caused by missing inputs, is proportional to the importance of the missing input. However, since we use a simple linear relationship, and testing is on controlled classification tasks, further ablation experiments are needed to robustly estimate sample-level reliability under missing data.



(a) Average feature importances from left to right: $X_1 = 1.00, X_2 = 0.00, X_3 = 0.00, X_4 = 0.00$. Top row: softmax probability distributions on the validation set for each missing input case. Bottom row: True P-RCC generated on data with missing input (blue line) compared to predicted PRCC estimated from (6.11) (red line). When the most important input X_1 is missing, the softmax probability distribution shifts lower and the true P-RCC drops steeply, aligned with the prediction. With less important inputs $X_2, \cdots, X_4$ missing, the probability and PRCC remain relatively unchanged, also matched by the estimate.



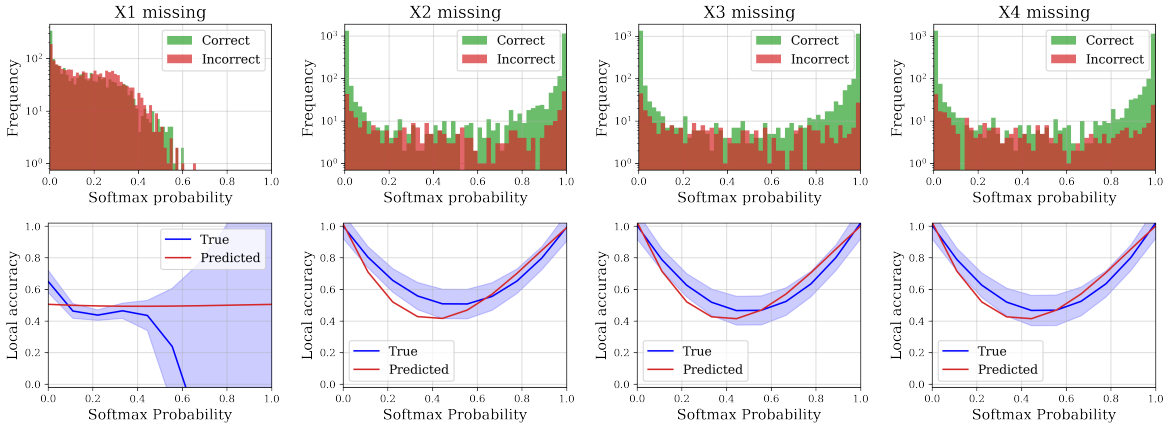
(b) Average feature importances from left to right: $X_1 = 0.25, X_2 = 0.25, X_3 = 0.25, X_4 = 0.25$. Top row: softmax probability distributions on the validation set for each missing input case. Bottom row: True P-RCC generated on data with missing input (blue line) compared to predicted PRCC estimated from (6.11) (red line). Since all inputs are equally important, shifts in the softmax probability distribution and P-RCC are consistent across missing input cases.

Figure 6.5: Comparison of estimated and true P-RCCs for holdout test data in the case of missing multimodal inputs. (a) shows results for problem 1, (b) shows results for problem 7, and (c) shows results for problem 8 in Table 6.4.

6.5 Conclusion and Future Work

This work explored RCCs for multimodal neural networks to provide sample-level reliability quantification. While promising, more work needs to be done to address granularity limitations for the distance-based methods. Representative sample properties need to be explored to construct bet-

Figure 6.5 (cont'd.).



(c) Average feature importances from left to right: $X_1 = 0.97, X_2 = 0.02, X_3 = 0.00, X_4 = 0.00$. Top row: softmax probability distributions on the validation set for each missing input case. Bottom row: True P-RCC generated on data with missing input (blue line) compared to predicted PRCC estimated from (6.11) (red line). shifts in the softmax probability distribution and P-RCC are aligned with estimates and are proportional to the importance of missing inputs.

ter calibration curves. For example, this current work relies on one dimensional sample properties; future work could look for RCC relationships between model performance and multi-dimensional representations of a sample. As the current distance and similarity metrics lack sufficient granularity, potential future work may achieve better localization by looking at class-wise distances.

Additionally, the simplicity of the importance-based RCC estimation provided reasonable but preliminary reliability estimates for missing inputs. More extensive validation across diverse problems is needed to fully develop a robust methodology for missing data scenarios.

In conclusion, this preliminary work highlighted several opportunities to refine the RCC approach, including improving localization, validating on more complex examples, identifying optimal model layers for distance based methods, and applying to real classification problems. Addressing these limitations is an important next step in improving sample-level reliability quantification in multimodal models. The methods developed here provide a foundation to build on, through improvements in granularity, generalization, and missing data handling.

CHAPTER 7

CONCLUSION

This dissertation explored methods to improve the human-interpretability of multimodal deep learning models for healthcare applications. We proposed a unified framework for estimating input feature importance in multimodal classifiers. Validation on synthetic data with ground truth importances showed our approach could accurately recover true feature importance. Analysis of real multimodal tumor classification and cardiomegaly detection models provided intuitive explanations of the black-box models. Our framework was agnostic to the underlying importance estimation technique, providing flexibility. By comparing importance across multimodal inputs, we gained insight into how different data types like images, text, and lab tests contributed to predictions.

To further enhance interpretability, we used the estimated importances to predict how model performance degraded with missing inputs. Across two clinical tasks, we showed input importance was strongly correlated with a drop in accuracy when that input was removed. This will enable understanding of model limitations and cost-benefit analysis for acquiring additional patient data. While our results demonstrated a strong correlation between performance degradation and input importance, future work could explore other functional relationships between the two. Additionally, our analysis has centered on binary classification problems using accuracy as the evaluation metric. An important next step would be extending the techniques to handle multi-class tasks, addressing extreme class imbalance, and leveraging metrics such as balanced accuracy to improve robustness. With these kinds of expansions, the foundations established in this dissertation can continue to mature.

We also constructed reliability calibration curves to quantify model reliability at the sample level. Initial results demonstrated the promise of this approach for per-sample reliability estimation. At the same time, they revealed opportunities to enhance the granularity and expand validation across diverse real-world tasks. Our missing data importance model provided reasonable preliminary reliability estimates, establishing a foundation to build upon through further refinement and extensive real-world testing.

In conclusion, this dissertation makes significant strides in advancing the interpretability of multimodal deep learning for healthcare applications through input importance estimations and novel reliability calibration techniques. While work remains in improving localization, expanding validation, and clinical translation, the critical foundations have been laid. The opportunities uncovered to refine the methodology highlight the fruitful research directions ahead. Overall, this dissertation establishes a robust framework and springboard for increasing interpretability of powerful multimodal AI systems poised to transform medicine.

BIBLIOGRAPHY

[1] V. Kaul, S. Enslin, and S. A. Gross, "History of artificial intelligence in medicine," *Gastrointestinal endoscopy*, vol. 92, no. 4, pp. 807–812, 2020.

[2] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, *et al.*, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information fusion*, vol. 58, pp. 82–115, 2020.

[3] T. Speith, "A review of taxonomies of explainable artificial intelligence (xai) methods," in *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 2239–2250, 2022.

[4] S.-C. Huang, A. Pareek, S. Seyyedi, I. Banerjee, and M. P. Lungren, "Fusion of medical imaging and electronic health records using deep learning: a systematic review and implementation guidelines," *NPJ digital medicine*, vol. 3, no. 1, p. 136, 2020.

[5] S. M. Wraith, J. S. Aikins, W. J. Clancey, L. M. Fagan, W. J. van Melle, B. G. Buchanan, R. Davis, A. C. Scott, E. H. Shortliffe, S. G. Axline, *et al.*, "Computerized consultation system for selection of antimicrobial therapy," *American Journal of Hospital Pharmacy*, vol. 33, no. 12, pp. 1304–1308, 1976.

[6] P. Szolovits, R. S. Patil, and W. B. Schwartz, "Artificial intelligence in medical diagnosis," *Annals of internal medicine*, vol. 108, no. 1, pp. 80–87, 1988.

[7] S.-C. B. Lo, H.-P. Chan, J.-S. Lin, H. Li, M. T. Freedman, and S. K. Mun, "Artificial convolution neural network for medical image pattern recognition," *Neural networks*, vol. 8, no. 7-8, pp. 1201–1214, 1995.

[8] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[9] J. De Fauw, J. R. Ledsam, B. Romera-Paredes, S. Nikolov, N. Tomasev, S. Blackwell, H. Askham, X. Glorot, B. O'Donoghue, D. Visentin, *et al.*, "Clinically applicable deep learning for diagnosis and referral in retinal disease," *Nature medicine*, vol. 24, no. 9, pp. 1342–1350, 2018.

[10] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, *et al.*, "Chexnet radiologist-level pneumonia detection on chest x-rays with deep learning," *arXiv preprint arXiv:1711.05225*, 2017.

[11] K.-H. Yu, A. L. Beam, and I. S. Kohane, "Artificial intelligence in healthcare," *Nature biomedical engineering*, vol. 2, no. 10, pp. 719–731, 2018.

[12] J. Burrell, "How the machine 'thinks': Understanding opacity in machine learning algorithms," *Big data & society*, vol. 3, no. 1, p. 2053951715622512, 2016.

[13] Z. Obermeyer, B. Powers, C. Vogeli, and S. Mullainathan, "Dissecting racial bias in an algorithm used to manage the health of populations," *Science*, vol. 366, no. 6464, pp. 447–453, 2019.

[14] A. J. Larrazabal, N. Nieto, V. Peterson, D. H. Milone, and E. Ferrante, "Gender imbalance in medical imaging datasets produces biased classifiers for computer-aided diagnosis," *Proceedings of the National Academy of Sciences*, vol. 117, no. 23, pp. 12592–12594, 2020.

[15] E. Röösli, S. Bozkurt, and T. Hernandez-Boussard, "Peeking into a black box, the fairness and generalizability of a mimic-iii benchmarking model," *Scientific Data*, vol. 9, no. 1, p. 24, 2022.

[16] C. Liu, X. Liu, F. Wu, M. Xie, Y. Feng, and C. Hu, "Using artificial intelligence (watson for oncology) for treatment recommendations amongst chinese patients with lung cancer: feasibility study," *Journal of medical Internet research*, vol. 20, no. 9, p. e11087, 2018.

[17] B. Institution, "Risks and remedies for artificial intelligence in health care," 2021. Accessed: 2023-06-14.

[18] D. S. Watson, J. Krutzinna, I. N. Bruce, C. E. Griffiths, I. B. McInnes, M. R. Barnes, and L. Floridi, "Clinical applications of machine learning algorithms: beyond the black box," *Bmj*, vol. 364, 2019.

[19] E. Vayena, A. Blasimme, and I. G. Cohen, "Machine learning in medicine: addressing ethical challenges," *PLoS medicine*, vol. 15, no. 11, p. e1002689, 2018.

[20] L. Hoffman, E. Benedetto, H. Huang, E. Grossman, D. Kaluma, Z. Mann, and J. Torous, "Augmenting mental health in primary care: a 1-year study of deploying smartphone apps in a multi-site primary care/behavioral health integration program," *Frontiers in psychiatry*, p. 94, 2019.

[21] A. Holzinger, B. Haibe-Kains, and I. Jurisica, "Why imaging data alone is not enough: Ai-based integration of imaging, omics, and clinical data," *European Journal of Nuclear Medicine and Molecular Imaging*, vol. 46, pp. 2722–2730, 2019.

[22] C. Macrae, "Governing the safety of artificial intelligence in healthcare," *BMJ quality & safety*, vol. 28, no. 6, pp. 495–498, 2019.

[23] OECD, "Recommendation of the council on artificial intelligence." https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0449, 2019. Accessed: 2023-07-18.

[24] N. I. of Standards and T. (NIST), "Nist explainable ai workshop summary." https://www.nist.gov/publications/nist-explainable-ai-workshop-summary, 2020. Accessed: 2023-07-18.

[25] U. Congress, "H.r.6216 - national artificial intelligence initiative act of 2020." https://www.congress.gov/bill/116th-congress/house-bill/6216, 2020. Accessed: 2023-07-18.

[26] D. Gunning and D. Aha, "Darpa's explainable artificial intelligence (xai) program," *AI magazine*, vol. 40, no. 2, pp. 44–58, 2019.

[27] U. Congress, "H.r.2231 - algorithmic accountability act of 2019." https://www.congress.gov/bill/116th-congress/house-bill/2231, 2020. Accessed: 2023-07-18.

[28] F. T. Commission, "Big data: A tool for inclusion or exclusion? understanding the issues." https://www.ftc.gov/system/files/documents/reports/big-data-tool-inclusion-or-exclusion-understanding-issues/160106big-data-rpt.pdf, 2016. Accessed: 2023-07-18.

[29] I. C. Office, "Explaining decisions made with artificial intelligence." https://ico.org.uk/media/for-organisations/guide-to-data-protection/key-dp-themes/explaining-decisions-made-with-artificial-intelligence-1-0.pdf, 2020. Accessed: 2023-07-18.

[30] E. Union, "Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation)." https://eur-lex.europa.eu/eli/reg/2016/679/oj, 2016. Accessed: 2023-07-18.

[31] E. Union, "Proposal for a regulation of the european parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts." https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0206, 2021. Accessed: 2023-07-18.

[32] Food, D. Administration, *et al.*, "Proposed regulatory framework for modifications to artificial intelligence/machine learning (ai/ml)-based software as a medical device (samd)," 2019.

[33] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM computing surveys (CSUR)*, vol. 51, no. 5, pp. 1–42, 2018.

[34] E. Shortliffe, *Computer-based medical consultations: MYCIN*, vol. 2. Elsevier, 2012.

[35] C. Molnar, *Interpretable machine learning*. Lulu. com, 2020.

[36] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[37] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra, "Grad-cam: Why did you say that?," *arXiv preprint arXiv:1611.07450*, 2016.

[38] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pp. 818–833, Springer, 2014.

[39] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," in *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pp. 607–617, 2020.

[40] Y. Goyal, Z. Wu, J. Ernst, D. Batra, D. Parikh, and S. Lee, "Counterfactual visual explanations," 2019.

[41] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, *et al.*, "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)," in *International conference on machine learning*, pp. 2668–2677, PMLR, 2018.

[42] R. Caruana, H. Kangarloo, J. D. Dionisio, U. Sinha, and D. Johnson, "Case-based explanation of non-case-based learning methods.," in *Proceedings of the AMIA Symposium*, p. 212, American Medical Informatics Association, 1999.

[43] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," 2020.

[44] A. Nguyen, J. Yosinski, and J. Clune, "Understanding neural networks via feature visualization: A survey," 2019.

[45] C. Chen, O. Li, D. Tao, A. Barnett, C. Rudin, and J. K. Su, "This looks like that: deep learning for interpretable image recognition," *Advances in neural information processing systems*, vol. 32, 2019.

[46] E. A. Barnes, R. J. Barnes, Z. K. Martin, and J. K. Rader, "This looks like that there: Interpretable neural networks for image tasks when location matters," *Artificial Intelligence for the Earth Systems*, vol. 1, no. 3, p. e220001, 2022.

[47] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[48] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, pp. 1789–1819, 2021.

[49] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature machine intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[50] A. d. Garcez and L. C. Lamb, "Neurosymbolic ai: The 3 rd wave," *Artificial Intelligence Review*, pp. 1–20, 2023.

[51] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5188–5196, 2015.

[52] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission," in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1721–1730, 2015.

[53] S. Rüping *et al.*, "Learning interpretable models," 2006.

[54] A. A. Freitas, "Comprehensible classification models: A position paper," *SIGKDD Explor. Newsl.*, vol. 15, p. 1–10, mar 2014.

[55] A. Jacovi, A. Marasović, T. Miller, and Y. Goldberg, "Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in ai," in *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pp. 624–635, 2021.

[56] D. Wang, Q. Yang, A. Abdul, and B. Y. Lim, "Designing theory-driven user-centric explainable ai," in *Proceedings of the 2019 CHI conference on human factors in computing systems*, pp. 1–15, 2019.

[57] Z. C. Lipton, "The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery.," *Queue*, vol. 16, no. 3, pp. 31–57, 2018.

[58] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," 2019.

[59] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *CoRR*, vol. abs/1312.6034, 2014.

[60] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[61] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, p. 1157–1182, Mar. 2003.

[62] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," vol. 50, pp. 1–45, Jan. 2018.

[63] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you? explaining the predictions of any classifier," ACM, Aug. 2016.

[64] Y. Hechtlinger, "Interpretation of prediction models using the input gradient," 2016.

[65] M. Wojtas and K. Chen, "Feature importance ranking for deep learning," *CoRR*, vol. abs/2010.08973, 2020.

[66] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PLOS ONE*, vol. 10, p. e0130140, July 2015.

[67] Y. Li, C.-Y. Chen, and W. W. Wasserman, "Deep feature selection: Theory and application to identify enhancers and promoters," vol. 23, pp. 322–336, May 2016.

[68] B. Skrlj, S. Dzeroski, N. Lavrac, and M. Petkovic, "Feature importance estimation with self-attention networks," in *Proceedings of the 24th European Conference on Artificial Intelligence*, 2020.

[69] L. Breiman *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[70] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.

[71] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:1412.6806*, 2014.

[72] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," in *2016 IEEE symposium on security and privacy (SP)*, pp. 598–617, IEEE, 2016.

[73] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International conference on machine learning*, pp. 3319–3328, PMLR, 2017.

[74] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "Smoothgrad: removing noise by adding noise," *arXiv preprint arXiv:1706.03825*, 2017.

[75] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *International conference on machine learning*, pp. 1885–1894, PMLR, 2017.

[76] R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2950–2958, 2019.

[77] W. J. Murdoch and A. Szlam, "Automatic rule extraction from long short term memory networks," *arXiv preprint arXiv:1702.02540*, 2017.

[78] A. Dhurandhar, P.-Y. Chen, R. Luss, C.-C. Tu, P. Ting, K. Shanmugam, and P. Das, "Explanations based on the missing: Towards contrastive explanations with pertinent negatives," *Advances in neural information processing systems*, vol. 31, 2018.

[79] M. T. Ribeiro, S. Singh, and C. Guestrin, "Anchors: High-precision model-agnostic explanations," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[80] G. Plumb, D. Molitor, and A. S. Talwalkar, "Model agnostic supervised local explanations," *Advances in neural information processing systems*, vol. 31, 2018.

[81] E. Štrumbelj and I. Kononenko, "Explaining prediction models and individual predictions with feature contributions," *Knowledge and information systems*, vol. 41, no. 3, pp. 647–665, 2014.

[82] N. H. Pijls, B. de Bruyne, K. Peels, P. H. van der Voort, H. J. Bonnier, J. Bartunek, and J. J. Koolen, "Measurement of fractional flow reserve to assess the functional severity of coronary-artery stenoses," *New England Journal of Medicine*, vol. 334, no. 26, pp. 1703–1708, 1996.

[83] P. A. Tonino, B. De Bruyne, N. H. Pijls, U. Siebert, F. Ikeno, M. vant Veer, V. Klauss, G. Manoharan, T. Engstrøm, K. G. Oldroyd, *et al.*, "Fractional flow reserve versus angiography for guiding percutaneous coronary intervention," *New England Journal of Medicine*, vol. 360, no. 3, pp. 213–224, 2009.

[84] P. D. Morris, D. Ryan, A. C. Morton, R. Lycett, P. V. Lawford, D. R. Hose, and J. P. Gunn, "Virtual fractional flow reserve from coronary angiography: modeling the significance of coronary lesions: results from the virtu-1 (virtual fractional flow reserve from coronary angiography) study," *JACC: Cardiovascular Interventions*, vol. 6, no. 2, pp. 149–157, 2013.

[85] A. Coenen, Y.-H. Kim, M. Kruk, C. Tesche, J. De Geer, A. Kurata, M. L. Lubbers, J. Daemen, L. Itu, S. Rapaka, *et al.*, "Diagnostic accuracy of a machine-learning approach to coronary computed tomographic angiography-based fractional flow reserve: result from the machine consortium," *Circulation: Cardiovascular Imaging*, vol. 11, no. 6, p. e007217, 2018.

[86] V. R. Taqueti, L. J. Shaw, N. R. Cook, V. L. Murthy, N. R. Shah, C. R. Foster, J. Hainer, R. Blankstein, S. Dorbala, and M. F. Di Carli, "Excess cardiovascular risk in women relative to men referred for coronary angiography is associated with severely impaired coronary flow reserve, not obstructive disease," *Circulation*, vol. 135, no. 6, pp. 566–577, 2017.

[87] D. F. Young and F. Y. Tsai, "Flow characteristics in models of arterial stenoses—i. steady flow," *Journal of biomechanics*, vol. 6, no. 4, pp. 395–410, 1973.

[88] B. Seeley and D. Young, "Effect of geometry on pressure losses across models of arterial stenoses," *Journal of Biomechanics*, vol. 9, pp. 439–448, Jan. 1976.

[89] F. A. Duck, *Physical properties of tissues: a comprehensive reference book*. Academic press, 2013.

[90] Z. Guo, X. Li, H. Huang, N. Guo, and Q. Li, "Deep learning-based image segmentation on multimodal medical imaging," *IEEE Transactions on Radiation and Plasma Medical Sciences*, vol. 3, no. 2, pp. 162–169, 2019.

[91] T. Xu, H. Zhang, X. Huang, S. Zhang, and D. N. Metaxas, "Multimodal deep learning for cervical dysplasia diagnosis," in *International conference on medical image computing and computer-assisted intervention*, pp. 115–123, Springer, 2016.

[92] A. Akselrod-Ballin, M. Chorev, Y. Shoshan, A. Spiro, A. Hazan, R. Melamed, E. Barkan, E. Herzel, S. Naor, E. Karavani, *et al.*, "Predicting breast cancer by applying deep learning to linked health records and mammograms," *Radiology*, vol. 292, no. 2, pp. 331–342, 2019.

[93] X. Ma and F. Jia, "Brain tumor classification with multimodal mr and pathology images," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 5th International Workshop, BrainLes 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 17, 2019, Revised Selected Papers, Part II 5*, pp. 343–352, Springer, 2020.

[94] L. R. Soenksen, Y. Ma, C. Zeng, L. Boussioux, K. Villalobos Carballo, L. Na, H. M. Wiberg, M. L. Li, I. Fuentes, and D. Bertsimas, "Integrated multimodal artificial intelligence framework for healthcare applications," *NPJ Digital Medicine*, vol. 5, no. 1, p. 149, 2022.

[95] R. Yan, F. Zhang, X. Rao, Z. Lv, J. Li, L. Zhang, S. Liang, Y. Li, F. Ren, C. Zheng, *et al.*, "Richer fusion network for breast cancer classification based on multimodal data," *BMC Medical Informatics and Decision Making*, vol. 21, no. 1, pp. 1–15, 2021.

[96] G. Holste, S. C. Partridge, H. Rahbar, D. Biswas, C. I. Lee, and A. M. Alessio, "End-to-end learning of fused image and non-image features for improved breast cancer classification from mri," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3294–3303, 2021.

[97] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*, pp. 1–15, Springer Berlin Heidelberg, 2000.

[98] N. Neverova, C. Wolf, G. Taylor, and F. Nebout, "Moddrop: Adaptive multi-modal gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, p. 1692–1706, aug 2016.

[99] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," 2011.

[100] C. G. M. Snoek, M. Worring, and A. W. M. Smeulders, "Early versus late fusion in semantic video analysis," in *ACM International Conference on Multimedia*, pp. 399—-402, 2005.

[101] H. Gunes and M. Piccardi, "Affect recognition from face and body: early fusion vs. late fusion," *2005 IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, pp. 3437–3443 Vol. 4, 2005.

[102] D. Ramachandram and G. W. Taylor, "Deep multimodal learning: A survey on recent advances and trends," *IEEE signal processing magazine*, vol. 34, no. 6, pp. 96–108, 2017.

[103] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 2, pp. 423–443, 2018.

[104] P. Atrey, M. Hossain, A. El Saddik, and M. Kankanhalli, "Multimodal fusion for multimedia analysis: A survey," *Multimedia Syst.*, vol. 16, pp. 345–379, 11 2010.

[105] Y. Li, M. El Habib Daho, P.-H. Conze, H. Al Hajj, S. Bonnin, H. Ren, N. Manivannan, S. Magazzeni, R. Tadayoni, B. Cochener, *et al.*, "Multimodal information fusion for glaucoma and diabetic retinopathy classification," in *International Workshop on Ophthalmic Medical Image Analysis*, pp. 53–62, Springer, 2022.

[106] S. Niu, Q. Yin, Y. Song, Y. Guo, and X. Yang, "Label dependent attention model for disease risk prediction using multimodal electronic health records," in *2021 IEEE International Conference on Data Mining (ICDM)*, pp. 449–458, IEEE, 2021.

[107] P. Vijayaraghavan, H. Larochelle, and D. Roy, "Interpretable multi-modal hate speech detection," *arXiv preprint arXiv:2103.01616*, 2021.

[108] I. Gat, I. Schwartz, and A. Schwing, "Perceptual score: What data modalities does your model perceive?," *Advances in Neural Information Processing Systems*, vol. 34, pp. 21630–21643, 2021.

[109] H. Suresh, N. Hunt, A. Johnson, L. A. Celi, P. Szolovits, and M. Ghassemi, "Clinical intervention prediction and understanding using deep networks," *arXiv preprint arXiv:1705.08498*, 2017.

[110] S. El-Sappagh, J. M. Alonso, S. R. Islam, A. M. Sultan, and K. S. Kwak, "A multilayer multimodal detection and prediction model based on explainable artificial intelligence for alzheimer's disease," *Scientific reports*, vol. 11, no. 1, p. 2660, 2021.

[111] W. Jin, X. Li, and G. Hamarneh, "Evaluating explainable ai on a multi-modal medical imaging task: Can existing algorithms fulfill clinical requirements?," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 11945–11953, 2022.

[112] J. Yang, R. Shi, D. Wei, Z. Liu, L. Zhao, B. Ke, H. Pfister, and B. Ni, "Medmnist v2: A large-scale lightweight benchmark for 2d and 3d biomedical image classification," *arXiv preprint arXiv:2110.14795*, 2021.

[113] N. Z. Abidin, A. R. Ismail, and N. A. Emran, "Performance analysis of machine learning algorithms for missing value imputation," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 6, 2018.

[114] E. T. Capariño, A. M. Sison, and R. P. Medina, "Application of the modified imputation method to missing data to increase classification performance," in *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pp. 134–139, IEEE, 2019.

[115] K. N. Vokinger, S. Feuerriegel, and A. S. Kesselheim, "Continual learning in medical devices: Fda's action plan and beyond," *The Lancet Digital Health*, vol. 3, no. 6, pp. e337–e338, 2021.

[116] A. R. T. Donders, G. J. Van Der Heijden, T. Stijnen, and K. G. Moons, "A gentle introduction to imputation of missing values," *Journal of clinical epidemiology*, vol. 59, no. 10, pp. 1087–1091, 2006.

[117] P. Jonsson and C. Wohlin, "An evaluation of k-nearest neighbour imputation using likert data," in *10th International Symposium on Software Metrics, 2004. Proceedings.*, pp. 108–118, IEEE, 2004.

[118] D. B. Rubin, "The calculation of posterior distributions by data augmentation: Comment: A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when fractions of missing information are modest: The sir algorithm," *Journal of the American Statistical Association*, vol. 82, no. 398, pp. 543–546, 1987.

[119] L. Tran, X. Liu, J. Zhou, and R. Jin, "Missing modalities imputation via cascaded residual autoencoder," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1405–1414, 2017.

[120] A. Johnson, M. Lungren, Y. Peng, Z. Lu, R. Mark, S. Berkowitz, and S. Horng, "Mimic-cxr-jpg-chest radiographs with structured labels," *PhysioNet*, 2019.

[121] A. E. Johnson, L. Bulgarelli, L. Shen, A. Gayles, A. Shammout, S. Horng, T. J. Pollard, B. Moody, B. Gow, L.-w. H. Lehman, *et al.*, "Mimic-iv, a freely accessible electronic health record dataset," *Scientific data*, vol. 10, no. 1, p. 1, 2023.

[122] A. Johnson, L. Bulgarelli, T. Pollard, L. A. Celi, R. Mark, and S. Horng IV, "Mimic-iv-ed," *PhysioNet*, 2021.

[123] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya, *et al.*, "Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, pp. 590–597, 2019.

[124] J. P. Cohen, J. D. Viviano, P. Bertin, P. Morrison, P. Torabian, M. Guarrera, M. P. Lungren, A. Chaudhari, R. Brooks, M. Hashir, *et al.*, "Torchxrayvision: A library of chest x-ray datasets and models," in *International Conference on Medical Imaging with Deep Learning*, pp. 231–249, PMLR, 2022.

[125] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR (Poster)*, 2015.

[126] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[127] C. R. Henderson, "Best linear unbiased estimation and prediction under a selection model," *Biometrics*, pp. 423–447, 1975.

[128] N. Chaturvedi, "Ethnic differences in cardiovascular disease," *Heart*, vol. 89, no. 6, pp. 681–686, 2003.

[129] A. K. Kurian and K. M. Cardarelli, "Racial and ethnic differences in cardiovascular disease risk factors," *Ethnicity & disease*, vol. 17, no. 1, pp. 143–152, 2007.

[130] "Cardiomegaly." https://www.ncbi.nlm.nih.gov/books/NBK542296/. Accessed: 2023-06-12.

[131] "Enlarged heart - symptoms and causes." https://www.mayoclinic.org/diseases-conditions/enlarged-heart/symptoms-causes/syc-20355436. Accessed: 2023-06-12.

[132] Q. McNamara, A. De La Vega, and T. Yarkoni, "Developing a comprehensive framework for multimodal feature extraction," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1567–1574, 2017.

[133] Y. Kwon, J.-H. Won, B. J. Kim, and M. C. Paik, "Uncertainty quantification using bayesian neural networks in classification: Application to biomedical image segmentation," *Computational Statistics & Data Analysis*, vol. 142, p. 106816, 2020.

[134] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Proceedings of The 33rd International Conference on Machine Learning* (M. F. Balcan and K. Q. Weinberger, eds.), vol. 48 of *Proceedings of Machine Learning Research*, (New York, New York, USA), pp. 1050–1059, PMLR, 20–22 Jun 2016.

[135] G. Shafer and V. Vovk, "A tutorial on conformal prediction.," *Journal of Machine Learning Research*, vol. 9, no. 3, 2008.

[136] P. C. Mahalanobis, "On the generalized distance in statistics," *Proceedings of the National Institute of Sciences (Calcutta)*, vol. 2, pp. 49–55, 1936.

[137] J. Ren, S. Fort, J. Liu, A. G. Roy, S. Padhy, and B. Lakshminarayanan, "A simple fix to mahalanobis distance for improving near-ood detection," *arXiv preprint arXiv:2106.09022*, 2021.

[138] K. Lee, H. Lee, K. Lee, and J. Shin, "Training confidence-calibrated classifiers for detecting out-of-distribution samples," *arXiv preprint arXiv:1711.09325*, 2017.

[139] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

APPENDIX

Open Source Code

Code developed for Chapter 3 is available: https://github.com/MA/FFR.

Code developed for Chapter 4 is available: https://github.com/MA/SYN.

Code developed for Chapter 5 is available: https://github.com/MA/MII.