ID N. 32981

UNIVERSITÀ CAMPUS BIO-MEDICO DI ROMA

DEPARTMENT OF ENGINEERING

Italian National Ph.D. in Artificial Intelligence
Health and Life Sciences
XXXVII Cycle

Resilient Multimodal Learning with Incomplete Information in Biomedical Applications

Supervisors

Paolo Soda

Sara Ramella

Valerio Guarrasi

Candidate

Camillo Maria Caruso

July, 2025

To my Mom

Abstract

The deployment of Artificial Intelligence (AI) in clinical settings is increasingly viewed as a transformative step toward precision medicine and improved patient outcomes. However, the clinical reality of healthcare data is far from ideal. One of the most pervasive challenges remains the issue of missing and incomplete data, a systemic characteristic rather than a sporadic error. Conventional predictive models are often inadequately equipped to handle such irregularities, relying heavily on data preprocessing and imputation strategies that introduce biases and limit generalizability. This dissertation proposes a paradigm shift in the development of machine learning models, advocating for resilient learning systems, architectures inherently designed to operate under partial observability while maintaining high performance and stability. Rather than viewing missingness as a problem to be corrected, the proposed approach treats it as a fundamental property of clinical data that must be embraced and strategically incorporated into the learning process. A key focus is the application of these principles to multimodal learning, where information is drawn from heterogeneous sources, including clinical records, laboratory tests, and radiological images. In such contexts, the absence of entire modalities can significantly hinder model performance if not explicitly addressed. This thesis explores how to design multimodal systems resilient to missing and incomplete data.

To this end, the dissertation first presents NAIM (Not Another Imputation Method), a transformer-based model tailored for tabular data characterized by structural missingness. NAIM leverages a combination of feature-specific embeddings, a masked self-attention mechanism, and a novel regularization strategy that randomly masks inputs during training. This framework enables the model to learn directly from incomplete feature sets, avoiding the biases introduced by imputation. Indeed, unlike traditional methods that depend on complex imputation pipelines, NAIM bypasses the reconstruction step entirely, training directly on what is observed. The model is benchmarked against a wide spectrum of state-of-the-art machine learning and deep learning models across five public classification datasets, as well as on a real-world clinical application involving the prediction of overall survival in cancer patients. Across both experimental settings, NAIM demonstrates strong and consistent per-

formance, particularly under high levels of missingness. These results highlight the models capacity to generalize to realistic clinical scenarios and confirm its practical relevance for resilient AI systems in healthcare.

Building on these insights, the thesis then explores the transition to multimodal modeling by investigating whether combining radiological images (CT scans) and structured clinical data can enhance predictive performance in a real-world setting. A late fusion ensemble approach is employed, where each modality is processed independently and combined at the decision level. This setup reflects a common and practical design in clinical AI systems and ensures a certain degree of modularity. However, by treating modalities independently until the final stage, this approach cannot fully leverage the interdependencies and shared patterns across data types. This observation, supported by the findings of our systematic review on multimodal fusion strategies, underscores the need for more integrated fusion mechanisms capable of modeling cross-modal interactions throughout the learning process.

To overcome this limitation, the dissertation introduces MARIA (Multimodal Attention Resilient to Incomplete datA), a novel transformer-based architecture designed for resilient multimodal data fusion. MARIA integrates modality-specific encoders with a shared attention-based fusion module, using a generalized masking mechanism to handle missing modalities. This intermediate fusion design allows the model to dynamically learn interactions between available inputs while maintaining resilience to missing components. Empirical evaluations across a range of simulated missing data scenarios confirm MARIA's ability to deliver accurate and stable predictions, even under severe data fragmentation.

Altogether, this work outlines a unified methodological framework for designing AI systems that are resilient to data incompleteness in both unimodal and multimodal settings. Rather than relying on preprocessing fixes, resilience is embedded directly into model architectures and training objectives. This design philosophy aligns more closely with the operational constraints of modern clinical environments, where decisions often need to be made based on partial information and data cannot always be recollected or completed. By embracing the imperfection and variability inherent in healthcare data, this dissertation lays the groundwork for the development of practical, adaptable, and resilient AI models, capable of supporting clinicians even in the most uncertain and data-sparse scenarios.

Contents

List of Figures

List of Tables

Chapter 1

Toward Resilient AI in Clinical Practice

Artificial intelligence (AI) is rapidly redefining the landscape of medical research and clinical practice, offering transformative opportunities to enhance the quality, speed, and precision of healthcare delivery. In recent years, the convergence of increasing computational power, an increasing volume of healthcare data, and breakthroughs in deep learning has fueled the development of predictive models capable of extracting complex, non-linear patterns from multimodal biomedical information. These models have demonstrated remarkable performance in a wide array of applications, including disease diagnosis, prognosis, patient stratification, and personalized treatment planning. In high-stakes domains such as oncology, neurology, and cardiology, AI-driven tools have shown the potential to augment clinical decision-making, reduce diagnostic delays, and optimize therapeutic interventions.

This technological progress has been accompanied by a growing interest in deploying AI not only as a research tool but as a core component of modern clinical workflows. Algorithms capable of processing heterogeneous data sources, such as laboratory tests, electronic health records, radiological images, and genomic sequences, are now envisioned as pivotal enablers of precision medicine. The ability to synthesize multimodal data promises a deeper and more holistic understanding of patient health, allowing clinicians to anticipate risks, tailor interventions, and allocate resources more effectively. However, despite this promising scenario, the real-world implementation of AI systems in clinical settings remains filled with substantial challenges. Among the most pervasive and underappreciated obstacles is the incompleteness of clinical data. Unlike curated research datasets, data generated in routine clinical practice are often messy, irregular, and partially observed. Missing values in structured variables, irregularly sampled time-series, incomplete follow-up records, unavail-

able imaging modalities, and inconsistent documentation practices are widespread. These issues stem from a multitude of factors, including diverse institutional protocols, patient heterogeneity, non-standardized data collection, and resource limitations. Additionally, data availability is often hindered by privacy regulations, variations in technology infrastructure, and clinical priorities that do not align with optimal data acquisition. As such, incompleteness is not merely an incidental data quality problem, but rather a fundamental characteristic of clinical information systems.

The presence of missing data poses significant threats to the performance, reliability, and fairness of machine learning models. When not properly addressed, it can lead to biased estimates, reduced statistical power, and impaired generalizability. Traditional methods for handling missingness, such as mean imputation or model-based reconstruction, aim to "complete" the dataset before training. However, these techniques often overlook the underlying structure of the missingness and may introduce artifacts or spurious correlations. Furthermore, they rarely capture the uncertainty associated with imputed values, which can propagate into downstream predictions and decision-making processes. These limitations become particularly pronounced in modern multimodal settings, where different data modalities, such as clinical records, imaging, genomics, and time-series, may be partially or entirely unavailable for subsets of the population.

In light of these challenges, this thesis proposes a paradigm shift: to design AI models that are intrinsically robust to missing data, embedding resilience directly into the model architecture rather than relying on preprocessing or imputation. The goal is to treat missingness not as an error to be corrected, but as a property to be modeled and leveraged. To achieve this, the work presented herein explores a progressive methodological pathway, beginning with the development of models for structured tabular data and advancing toward the design of robust multimodal architectures that can integrate heterogeneous sources of information even in the presence of incomplete observations.

This thesis seeks to bridge the gap between theoretical advances in machine learning and their practical application in clinical contexts by developing systems capable of adapting to the often messy and incomplete nature of real-world healthcare data. Rather than focusing solely on maximizing accuracy under ideal conditions, the models proposed here are designed to remain reliable and effective when faced with the variability, irregularity, and missingness that characterize clinical datasets. Ensuring that these models can function accurately and consistently in such challenging settings is essential for their safe and meaningful use in high-stakes medical decision-making.

A central theme of this thesis is the use of masked self-attention mechanisms and modality-aware fusion strategies. These techniques enable models to condition their predictions on

the available data while ignoring the missing components, thus avoiding the pitfalls of naive imputation. Additionally, several training procedures are introduced to simulate missingness and encourage the model to learn representations that generalize well to incomplete scenarios. These include regularization methods that artificially mask inputs during training, as well as learning objectives that reward the model for performance on partially observed data. Such strategies ensure that the model is not only evaluated but also trained under conditions that mirror real-world incompleteness.

By jointly considering architecture, training strategy, and evaluation framework, this thesis aims to provide a comprehensive and principled approach to robust learning under partial observability. This effort culminates in a series of models and experimental studies that demonstrate how AI systems can be made more resilient to one of the most persistent barriers in medical data science: missingness.

The structure of the thesis reflects this methodological progression and is composed of a collection of research articles, each tackling a key aspect of the robust modeling pipeline:

Chapter 2 introduces NAIM (Not Another Imputation Method), a transformer-based model designed to handle incomplete tabular data. NAIM incorporates masked self-attention and a data corruption regularization scheme to promote robustness during both training and inference. The model is benchmarked against state-of-the-art approaches on multiple datasets, demonstrating competitive performance under varying levels and patterns of missingness.

Chapter 3 shifts focus to a real-world clinical application: the prediction of overall survival (OS) in cancer patients using routinely collected clinical data. This chapter investigates the challenges of irregular temporal sampling, partial histories, and the interpretability of model predictions. It provides insights into how models can be made clinically useful while dealing with highly imperfect and noisy data.

Chapter 4 marks the transition into multimodal modeling, combining clinical features and radiological images (CT scans) to improve survival prediction in non-small cell lung cancer (NSCLC). This study evaluates several fusion strategies and illustrates the potential benefits of multimodal systems.

Chapter 5 introduces MARIA (Multimodal Attention Resilient to Incomplete datA), a novel multimodal transformer that generalizes the masked attention concept to multiple input types. By integrating intermediate fusion and modality-specific masking, MARIA enables predictive modeling even when certain modalities are completely absent. Extensive experiments under various missing-modality scenarios confirm the model's ability to maintain strong performance and stability, establishing it as a benchmark for robust multimodal learning in healthcare.

Finally, the concluding chapter synthesizes the key insights gathered across all contributions, discussing their implications for the development of clinically useful AI systems. It also outlines open challenges and future directions, emphasizing the need for continued progress toward resilient, multimodal, and real-world-ready machine learning models in healthcare.

Chapter 2

NAIM: Resilient Tabular Modeling under Structural Missingness

2.1 Introduction

Tabular data, characterized by data structured in tables with rows and columns, i.e., samples and features, respectively, poses unique challenges in training and testing artificial intelligence (AI) models because they differ from other data structures, such as text and speech that have a sequential nature or the images, which are characterized by spatial coherence. These challenges arise because, unlike in other domains, tabular data often consists of a *heterogeneous mix of categorical and numerical features*. Additionally, each feature can have a *different distribution and scale*, necessitating careful preprocessing steps such as normalization or embedding. Furthermore, tabular datasets frequently contain *missing values*, requiring models that can robustly handle such inconsistencies. To date, machine learning (ML) ruled over deep learning (DL) in many fields involving tabular data, offering methods capable of meeting each of these challenges [1, 2].

More specifically, the missing data issue poses a significant challenge in the realm of tabular data and common reasons for this event are: *i*) human error, where inaccuracies during data entry or collection lead to gaps in information; *ii*) non-response, common in surveys, where participants might skip questions for reasons like privacy or lack of interest; *iii*) data corruption, caused by technical failures or errors; *iv*) attrition, particularly in longitudinal studies, caused by participants drop out; *v*) systematic loss, where data is missing under specific conditions. Each of these factors could play a significant role in the challenge associated with handling missing values in tabular data: this absence of information may affect either training, testing, or both sets at the same time, whereas most of the state-

of-the-art approaches need a complete dataset to function properly. This brings the need for approaches that ignore the missing values and use all the available data to perform the task at hand. The state-of-the-art faced the problem using two main approaches [1, 2]: on the one hand, some methods fill in the missing entries, usually during the preprocessing phase before feeding the data to the model, by imputing the values using a predefined rule or an algorithm [1]; on the other hand, other approaches intrinsically handle missing data during inference time.

Recently, we have witnessed the successes of the transformer architecture in different domains. Since the transformer's core mechanism of self-attention has been adapted to computer vision [3] and speech recognition [4], some studies have started adapting this architecture to tabular data [5, 6, 7]; although, none has proposed specific solutions to handle missing values. This is why in this manuscript we present *"Not Another Imputation Method"* (NAIM), a novel transformer-based model for tabular data, specifically designed to face the missing values problem. The main contributions of this work are:

- The development of a transformer model that integrates feature-specific embeddings and a novel masked self-attention mechanism able to learn only from the available information without any need for imputation of missing values.
- The proposal of a novel regularization technique, which randomly masks each sample at every epoch. This method, presenting a different sample version at each epoch, enables the model to generalize from incomplete data, ensuring more resilient and accurate predictive performance.
- A wide experimental assessment, showing that NAIM achieves better performance than state-of-the-art models in 5 publicly available classification tasks. State-of-the-art approaches consist of 6 ML and 5 DL models, paired with 3 different imputation techniques, when necessary. Results indicate a noteworthy advancement of transformer and not transformer-based DL approaches over ML in addressing the domain of tabular data, highlighting its efficacy and potential in this area.

The manuscript is organized as follows: Section 2.2 presents the state-of-the-art of data imputation techniques and those models able to handle missing values (Section 2.2.1) and the main adaptations of transformers to the tabular data domain (Section 2.2.2). Section 2.3 introduces the proposed model, whereas Sections 2.4 and 2.5 discuss the experimental configuration and the results, respectively. Section 2.6 provides concluding remarks.

2.2 State-of-the-art

Despite the advancements in handling missing values in tabular data, the state-of-the-art shows the following limitations. Current methods primarily focus on imputing missing entries or adapting models to work with incomplete data. However, these strategies often fail to fully leverage the inherent patterns in the data or require extensive preprocessing, potentially leading to information loss or biased predictions. Moreover, while the transformer architecture has shown promising results in various domains, its application to tabular data with missing values has not been realized, indicating a gap in effectively addressing this challenge within the deep learning domain.

This section first presents the most established strategies to address missing values in tabular data (Section 2.2.1) and then we explore existing transformer-based methods for tabular data (Section 2.2.2). These methodologies are then employed as benchmarks in the experiments detailed in Sections 2.4 and 2.5.

2.2.1 SOTA techniques for missing values

The simplest way to handle missing values is the *Complete Case Analysis* [8] that completely excludes samples and/or features with missing values; despite its simplicity, its use can cause a considerable loss of information for the model, which cannot be tolerated to make AI models resilient to be used in practice. Consequently, several approaches have been developed to retain as much information as possible.

ML approaches

State-of-the-art techniques for managing missing values mainly focus either on imputing missing values [1, 8, 9] or on applying a specific strategy for missing entries [1, 9]. In [1], where the authors benchmark different approaches to handle missing values, the imputation techniques are distinguished in *Constant Imputation* and *Conditional Imputation*. The first, despite its simplicity, can effectively support models by replacing missing values with a central tendency measure, e.g., mean imputation; the latter comprises techniques recognized for leveraging patterns within the data to predict missing values accurately, e.g., Multiple Imputation by Chained Equations (MICE) [10] and K-Nearest Neighbors imputation (KNN) [11]. Furthermore, the authors in [1] also introduce the *Missing Incorporated in Attributes* (MIA) strategy, specifically advantageous for tree-based models, which incorporate missingness directly into the model, thus preserving all available data. By presenting some of the main possible approaches to handle missing values, this analysis underlines the multitude of avail-

able options and the need to identify the most suitable one for the specific task at hand. Furthermore, the prevalence of imputation approaches over models that can handle missing values, which are tree-based only, highlights the need for approaches that can ignore missing values rather than impute them. Indeed, identifying the most suitable imputation method for a specific downstream task in advance is a considerable challenge. Ideally, it would be better to rely solely on the data at hand, avoiding the introduction of potentially misleading information. This approach, unlike imputation strategies that rely on the training set, would ensure a robust analysis, ignoring missing values entirely and making the most of the available data.

DL approaches

DL approaches have introduced new perspectives and possibilities compared to traditional ML methods, particularly in handling complex and structured data scenarios. A representative example of this advancement in the missing data handling problem is GRAPE [12], a graph neural network (GNN)-based architecture specifically designed for tabular data analysis able to simultaneously perform feature imputation and label prediction. GRAPE, remapping the dataset onto a bipartite graph composed of sample nodes and feature nodes, intrinsically handles missing values by simply omitting edges between nodes where data is unavailable. This setting lets GRAPE formulate feature imputation as an edge-level prediction problem and classification as a node-level prediction task.

Nevertheless, in recent years, transformer models, originally developed for natural language processing (NLP) [13], have become predominant in various application domains, including computer vision [3] and speech recognition [4]. Although some extensions have been proposed to analyze tabular data, many transformer-based approaches still struggle to effectively manage missing data.

2.2.2 Transformer for tabular data

Currently, there are 3 main transformer-based approaches, each developing a different aspect of the transformer architecture to adapt it to tabular data: TabNet [5], TabTransformer [6], and FTTransformer [7].

TabNet [5] leverages the self-attention mechanism to selectively focus on specific features for decision-making, which is crucial in interpreting high-dimensional data. Learning a sparse masking matrix and mimicking the boosting technique by adding a layer at each step, TabNet can perform a dynamic feature selection. Its design facilitates interpretability, allowing it to offer insights into feature importance and decision pathways, a critical requirement in fields

like finance and healthcare where understanding model decisions is as important as their accuracy.

TabTransformer [6] addresses the embedding of categorical features within tabular data. It applies a transformer-based self-attention mechanism to create contextual embeddings for categorical features, similar to how transformers process words in NLP. By doing so, it captures complex inter-feature relationships and dependencies, enhancing the model's predictive performance. Given $x \in \mathbb{R}^{n \times m}$ as a mixture of categorical x^{cat} and numerical x^{num} features, each categorical feature x_i^{cat}, where i denotes the i-th feature, is encoded twice: on the one hand, a feature-specific lookup table E_i^{pos} encodes the value with respect only to the possible categorical values of that feature; on the second hand all categorical values are encoded using a shared lookup table E that embeds all the categorical features' values. The resulting embedding vector for the i-th categorical feature is denoted as e_i^{cat}:

$$e_i^{cat} = [E_i^{pos}(x_i^{cat}), E(x_i^{cat})]. \tag{2.1}$$

Instead, the numerical features x^{num} simply undergo a normalization step.

FTTransformer [7] further explores the potential of transformer models in extracting patterns and interactions within tabular data, which are often overlooked by traditional machine learning methods. Specifically, this model addresses the embedding of numerical features applying two different approaches for the numerical and categorical features. The embedding e_i^{num} for a given numerical feature x_i^{num} is computed as follows:

$$e_i^{num} = b_i + x_i^{num} \cdot W_i^{num}, \quad x_i^{num} \in \mathbb{R}, \quad e_i^{num} \in \mathbb{R}^{d_e} \tag{2.2}$$

where the d_e-dimensional embedding vector is computed as the sum of the i-th feature bias, $b_i \in \mathbb{R}^{d_e}$, and the multiplication of the feature value with the feature-specific vector $W_i^{num} \in \mathbb{R}^{d_e}$. Instead, the embedding e_i^{cat} for a given categorical feature x_i^{cat} is implemented as the lookup table E_i^{cat}, which associates a d_e-dimensional trainable vector to the k_i entries of the specific categorical feature:

$$e_i^{cat} = b_i + E_i^{cat}(x_i^{cat}), \quad x_i^{cat} \in \mathbb{N}_{k_i}, \quad e_i^{cat} \in \mathbb{R}^{d_e}. \tag{2.3}$$

2.2.3 Current limitations

As illustrated in the previous sections, current state-of-the-art methods still have several limitations. Among traditional ML approaches, none of the methods truly ignore missing values; the MIA strategy, which does not impute missing data, attempts to incorporate missingness

into the learning process rather than bypassing it completely. Regarding DL, GRAPE is the only architecture that can explicitly ignore missing features, but its dependence on GNN structures limits its integration and scalability. In contrast, transformer-based models, not having dedicated embeddings and a masking mechanism, cannot handle missing data. This shortcoming highlights the need for transformer-based solutions that can robustly ignore missing features while maintaining architectural flexibility and compatibility with broader modeling frameworks.

2.3 Methods

This section first presents a formal definition of the problem; then it examines the proposed NAIM architecture (Section 2.3.2), presenting the innovation we made to the embedding of the features and the novel masked self-attention mechanism to dynamically handle missing values; finally, in section 2.3.3, we introduce a novel regularization strategy that aims to enable the model to learn how to handle missing data even if they are not present in the training set, and to extract the optimal representation for each feature by preventing any co-adaptation between them.

2.3.1 Problem definition

Let $X \in \mathbb{R}^{n \times m}$ represent a data matrix with n samples and m features, where the element x_i^{num} corresponds to a numerical feature of the i-th sample, while x_i^{cat} represents a categorical feature of the same sample. In scenarios involving missing data, some of these feature values are unavailable. This is captured, for each sample separately, by a binary mask matrix $M \in \{-\infty, 0\}^{m \times m}$, where the entry $M_{ij} = 0$ indicates that the i-th feature is observed and can attend to the j-th feature, while $M_{ij} = -\infty$ prevents attention from the i-th feature to the j-th feature when the latter is missing. Typically, datasets are also associated with labels for a specific downstream task, which we denote as $Y \in \mathbb{R}^n$. Our focus is on predicting labels for test instances.

2.3.2 NAIM architecture

The transformer architecture [13], originally developed for text translation, includes two main blocks: the first, known as the *encoder*, extracts a hidden representation from the data, whilst the second, named the *decoder*, reconstructs the hidden representation in the context of the specific domain. For instance, in the NLP domain, the decoder generates the next token or performs complex interactions given an input sequence. Nevertheless, the

decoder may not be required in scenarios where decision-making is the primary objective. Given that most of the tasks related to tabular data are classification or regression, the transformer-based models mostly make use of the *encoder* part only with a fully-connected (*FC*) module for the final prediction [5, 6, 7]. However, to perform other tasks, e.g., features imputation or self-supervised pretraining, the *decoder* part could also be used to reconstruct the feature representation in another domain [5]. Indeed, when developing NAIM to get a model that handles missing features and learns from the available information without any need for imputation to perform a classification task, we opted for the encoder-only architecture followed by a *FC* module (Figure 2.1), which receives as input the normalized and concatenated version of the *encoder* output (grey block).

Our approach exploits the *Feature Embedding* (yellow block) and *Masked Multi-Head Attention* (red block) steps reported in the figure. More specifically, our idea stems from observing that the use of the padding index in the lookup table paired with the masked self-attention mechanism, could be an interesting and unexplored way to handle missing entries for both the categorical and numerical types of features in tabular data.

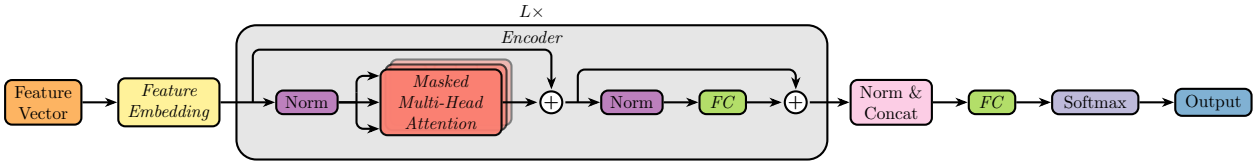

Figure 2.1: The architecture of NAIM, composed of the *Feature Embedding*, the *Encoder* equipped with the *Masked Multi-Head Attention* mechanism, and the final classification head [13].

Focusing on the *Feature Embedding*, which is a fundamental step to find a richer representation of the input data useful in the downstream task, we observed that the use of the padding index is reported in literature [6, 7], but none are able to handle missing values for both categorical and numerical features. Indeed, with the embedding lookup table E_i paired with the masked self-attention, we can completely exclude the missing values from the gradient calculations defining the $k_i + 1^{th}$ entry, named *padding index* or ‹pad›, which is assigned to a not-trainable vector of zeros. Using the notation introduced in section 2.2.2, the embedding of categorical features e_i^{cat} is given by:

$$e_i^{cat} = b_i + E_i^{cat}(x_i^{cat}), \quad x_i^{cat} \in \mathbb{N}_{k_i}, \quad e_i^{cat} \in \mathbb{R}^{d_e} \tag{2.4}$$

whereas we defined an embedding lookup table E_i^{num} for each numerical feature with 2 possible entries, named *present* and *missing*. Using the padding index of the lookup table, these entries are associated with a trainable and a not-trainable d_e-dimensional vector, re-

spectively. Then, we scaled the numerical embedding vector e_i^{num} by multiplying it by the feature value x_i^{num} when present:

$$e_i^{num} = b_i + x_i^{num} \cdot E_i^{num}(x_i^{num}), \quad x_i^{num} \in \mathbb{R}, \quad e_i^{num} \in \mathbb{R}^{d_e}. \tag{2.5}$$

This setup, further described with an example reported in Figure 2.2, ensures that both types of features can be handled when presenting missing values without impacting the learning process. The figure provides an example of a feature vector with 2 categorical (x^{cat}) and 2 numerical (x^{num}) features. For each of these 2 types of features, there are one missing and one non-missing value, which are then embedded using the respective look-up tables in the *Feature Embedding* block. In particular, we can notice that both the missing features are encoded using the entry relative to the missing value of the look-up tables, represented with the ? symbol. After obtaining the embedding vectors of the features, these are concatenated in the embedded representation e, which will be fed to the encoder later in the model architecture.

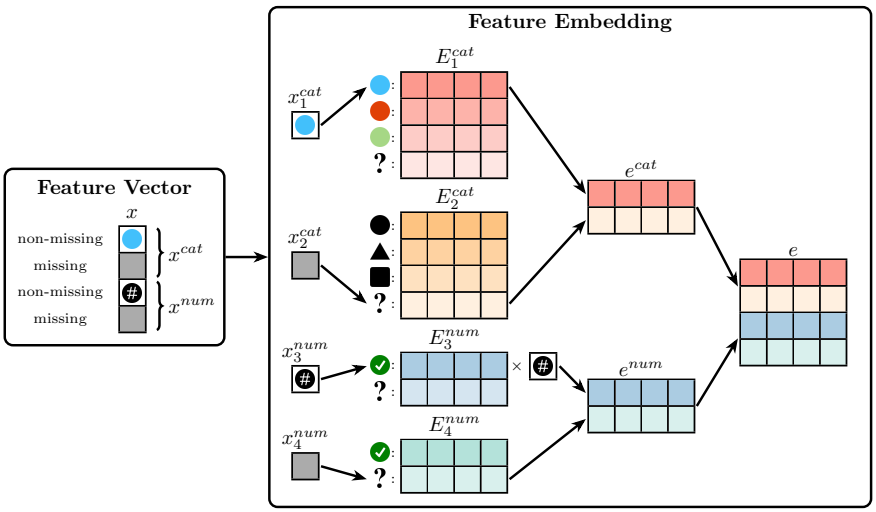

Figure 2.2: The proposed *Feature Embedding* process for tabular data. In the example, the feature vector x has 4 features: 2 categorical (x^{cat}) and 2 numerical (x^{num}). The colors (●, ●, ●) and the shapes (●, ▲, ■) are examples of possible values for the first categorical feature x_1^{cat} and for the second one x_2^{cat}, respectively, whilst ✓ stands for a non-missing numerical feature and # for its value. Finally, ? indicates the padding index related to missing features for both types of features. In the *Feature Embedding* block, we can see how the embedding e of the feature vector x is composed of the concatenation of embedded representations of the categorical and numerical features, denoted as e^{cat} and e^{num}, respectively. These representations are composed by the concatenation of the vectors associated with each feature value, selected using the feature-specific lookup tables E_i^{cat} and E_i^{num}.

Focusing on the masked self-attention mechanism to adapt it effectively for tabular data

and to completely mask out the contribution of the missing features, we first argued that it is necessary to modify it. Indeed, masked self-attention, a variant of the standard attention mechanism, plays a key role: it allows the model to focus only on certain parts of the input sequence while ignoring others. Traditionally, after the *Feature Embedding* step, the transformer calculates the query, key, and value matrices, denoted as Q, K, and V respectively, through linear transformations of e, i.e., mapping the embedding into a smaller space d_h based on the chosen number of heads h:

$$\begin{cases} Q = e \cdot W^Q, & d_h = d_e/h \\ K = e \cdot W^K, & W^Q, W^K, W^V \in \mathbb{R}^{d_e \times d_h} \\ V = e \cdot W^V, & Q, K, V \in \mathbb{R}^{n \times d_h} \end{cases} \tag{2.6}$$

where W^Q, W^K and W^V are weights matrices learned during training. Then, in the standard masked self-attention,

$$Attention(Q, K, V) = \underbrace{softmax\left(\frac{QK^T}{\sqrt{d_h}} + M\right)}_{A \in \mathbb{R}^{n \times n}} V \tag{2.7}$$

the attention matrix A uses the masking matrix $M \in \mathbb{R}^{n \times n}$ that sums $-\infty$ to the weights that should be ignored, zeroing their influence after the *softmax* operation. In the NLP domain, this technique is mainly employed to achieve 2 different purposes: on the one hand, the use of a mask M such as

$$M_{ij} = \begin{cases} 0 & \text{if } i \leq j \\ -\infty & \text{if } i > j \end{cases}, \quad M = \begin{bmatrix} 0 & \text{-}\infty & \dots & \text{-}\infty & \text{-}\infty \\ 0 & 0 & \dots & \text{-}\infty & \text{-}\infty \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & \text{-}\infty \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix} \tag{2.8}$$

allows the model to ignore future positions in the sequence, avoiding any leakage of information when training the model with the same sentence at different stages of completion; on the other hand, it could be used a mask M that cancels out the contributions of the last

tokens

$$M_{ij} = \begin{cases} 0 & \text{if } x_j \neq \text{‹pad›} \\ -\infty & \text{if } x_j = \text{‹pad›} \end{cases}, \quad M = \begin{bmatrix} 0 & 0 & ... & 0 & \text{-}\infty \\ 0 & 0 & ... & 0 & \text{-}\infty \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & ... & 0 & \text{-}\infty \\ 0 & 0 & ... & 0 & \text{-}\infty \end{bmatrix} \tag{2.9}$$

allowing the model to analyze sequences shorter than the dimension n used for training.

Considering the structure of these 2 possible masks, it is straightforward that only the latter could be adapted to mask missing features, since the former would mask large portions of the samples, even related to features that are non-missing. On this ground, the use of the Eq. 2.7 paired with the mask M in Eq. 2.9 used to mask the columns related to the missing values, might seem a viable solution, but, as shown in Figure 2.3 with colors and reported in Appendix A with an example, it does not cancel out the undesired value. Indeed, in Figure 2.3 we observe both the application of traditional masked attention and our new proposal: starting from left to right we have the matrices Q, K and V generated from the example feature vector shown in Figure 2.2, where the second and fourth features were missing. Immediately afterward, we report the attention matrix resulting from the calculation of the QK^T product, which is useful for understanding the next steps and especially how the contributions of the different features mix up together. In the next step, we show that the traditional masked attention mechanism, by masking the second and fourth columns, eliminates some of the contributions of the missing features, but leaves others on the second and fourth rows. For this reason, we here propose a new masked self-attention designed to completely mask out the contributions of the missing values using twice the same mask M from equation 2.9:

$$\mathit{Attention}(Q, K, V) = \mathit{ReLU}\left(\mathit{softmax}\left(\frac{QK^T}{\sqrt{d_h}} + M\right) + M^T\right)V \tag{2.10}$$

In this way, the attention matrix's rows and columns related to missing values are now assigned zero attention, ignoring their contribution.

2.3.3 Regularization technique

Given the capacity of NAIM to effectively handle missing features, we introduce a novel regularization technique designed to enhance model robustness by simulating missing data scenarios. This approach, inspired by the Cutout technique [14] randomly masks elements

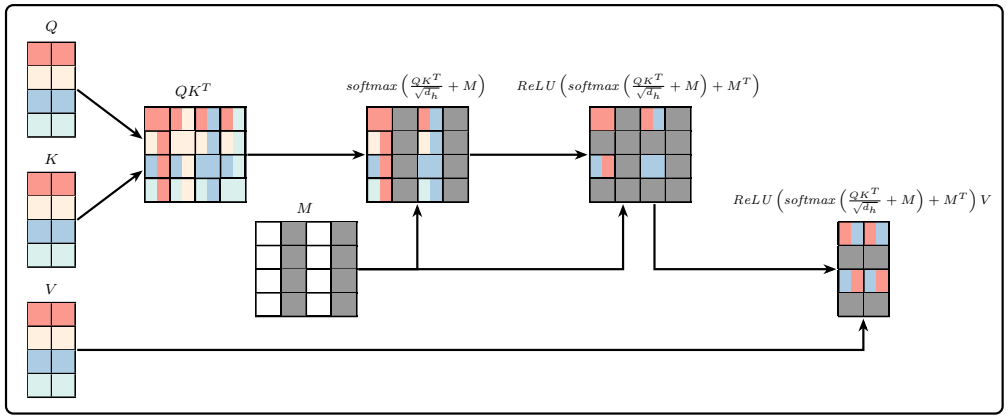

Figure 2.3: The proposed masked self-attention mechanism, designed to effectively ignore the impact of missing entries within the attention matrix. In the example, the QK^T matrix, obtained by the multiplication of the Q and K representations, is reported as an example of how the contributions of the different features, identified by different colors, mix up together. Next, the classic masked self-attention mechanism is applied, and some of the contributions of the missing features (indicated with □ and □) remain. Then, the proposed attention mechanism ensures that the influence of these missing values is completely masked out, before the multiplication by the representation V of the sample.

within feature vectors (Figure 2.4), taking into account real-world data's inherent variability and incompleteness. By integrating this regularization technique, we aim to improve the model's ability to generalize from incomplete patterns, thus enhancing predictive performance on datasets with varying degrees of missing information.

Given the feature vector $x \in \mathbb{R}^n$, where n represents the dimensionality of the vector, let $v \leq n$ be the number of non-missing elements. The vector may be fully populated with $v = n$ or contain missing values with $v < n$. The process is governed by a binary decision variable $B \sim \text{Bernoulli}(0.5)$, which determines whether the masking will be applied to the instance x. If a sample is selected for masking, a random count of c elements to mask is chosen uniformly from the set $\{1, 2, \ldots, v-1\}$, ensuring that at least one element remains unmasked. Finally, c non-missing elements within x are randomly chosen and set their values to *missing*, resulting in the augmented vector.

The masking operation simulates scenarios of incomplete data, a common challenge in practical applications. This methodological approach is systematically applied during the training phase, ensuring that each instance x underwent the described random masking process with a probability of 50%: this introduces variability in the input data, encouraging the model to learn more robust features that are not overly dependent on the presence of specific elements within the feature vector.

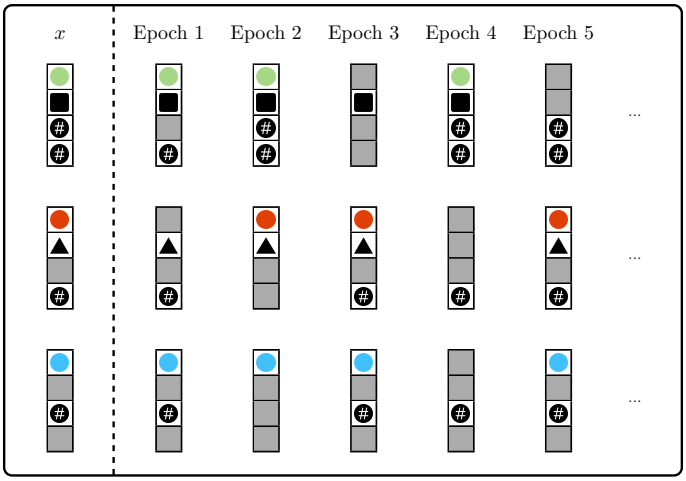

Figure 2.4: The proposed regularization strategy performed at every epoch before feeding the sample to the model. The colors (●, ●, ●) and the shapes (▲, ■) are examples of possible values for the categorical features, whilst # stands for a numerical feature. In the example are reported 3 feature vectors and their masked versions created in 5 different epochs. It should be noted that, when some features are originally missing, only the non-missing entries can be masked.

2.4 Experimental Configuration

In this section, we first report the data used in the experiments, the preprocessing applied and the metric used in the evaluation (Section 2.4.1), then we list the combinations of models and imputers used as competitors (Section 2.4.2). We provide some implementation details about the parameters used during training in Appendix B.

2.4.1 Data and Evaluation

We evaluate NAIM and competitor models on 5 publicly available datasets (reported in Table 2.1) from the UCI repository [15]. We selected these datasets as they are used to benchmark models for tabular data in [6]:

- **Adult** [16]: it is a dataset focusing on income prediction, which tries to predict people that make over $50k$ a year, from those that earn less than that amount, based on personal attributes such as education, occupation, and hours per week worked;
- **BankMarketing** [17]: its task is to predict whether the client will subscribe to a term deposit or not, and it comprises data from direct marketing campaigns of a Portuguese banking institution;

Dataset	# of samples	# of features	# of categorical features	Class distribution
Adult [16]	48842	14	8	**0**: 37155 **1**: 11687
BankMarketing [17]	41188	20	10	**0**: 36548 **1**: 4640
OnlineShoppers [18]	12330	17	7	**0**: 10422 **1**: 1908
SeismicBumps [19]	2584	18	4	**0**: 2414 **1**: 170
Spambase [20]	4601	57	0	**0**: 2788 **1**: 1813

Table 2.1: Datasets' details and references. Datasets' information consists of the number of samples, the number of features, how many of these features are categorical, and the classes' distribution.

- **OnlineShoppers** [18]: it involves predicting whether a visitor will make a purchase based on session information, like page views and time spent on the site;
- **SeismicBumps** [19]: it aims at predicting seismic hazards in coal mines, i.e., whether any seismic bump with an energy higher than 10^4 J was registered in the next 8 hours, based on seismic activity and energy measurements;
- **Spambase** [20]: it involves identifying whether an email is spam or not based on word frequencies and other email attributes.

We note that this selection of datasets shares the same type of task, i.e., classification, and that all of them are complete, except for Adult, which has only 1% of missing data. The datasets exemplify well-curated data; however, in practical real-world scenarios, this level of completeness may not always be achievable.

For each dataset, we normalize the numerical features in the range $[0, 1]$ and apply one-hot encoding to the categorical features, before feeding them to the models. In the experiments featuring NAIM, HistGradientBoost, TabNet, TabTransformer, FTTransformer and XGBoost, we do not apply the one-hot encoding, since their implementations can handle categorical features. The preprocessing step is calibrated on the training data and then applied to validation and testing sets.

Our experiments focus on artificially generated *Missing Completely At Random* (MCAR) values, the most common evaluation regime used in missing data papers [12]. Indeed, MCAR's lack of systematic bias means that the remaining data can be considered a fair representation of the whole. We aim to test our models under various missing data scenarios by introducing missing values at different percentages (denoted as p) across both training and testing sets. More specifically, we artificially generated missing percentages in the training and test sets separately, equal to 0%, 5%, 10%, 25%, 50% and 75% of the data in the set, for a total of 36 possible variations with repetitions. This allows us to methodically delineate varying scenarios of data completeness and missingness: i) a scenario with a well-curated

data collection for both training and testing sets, i.e., 0% of missing data in training and in testing; *ii*) a well-curated collection for the training set followed by a testing set with missing values, e.g., 0% of missing values in training and 25% in testing; *iii*), a scenario characterized by faulty data collection practices leading to missing values in the training phase, yet a completely curated test set without missing values, e.g., more common cases with 5%, 10%, or 25% of missing data in training and 0% in testing, or even more extreme cases where 50% or 75% of data are missing in the training set, but in the test set they are all not missing; *iv*) scenarios, which could represent domains where complete data collection is not possible, characterized by missing values in both training and testing sets, e.g., situations in which low percentages such as 5%, 10% or 25% of the data are missing in both sets, or even more extreme circumstances in which 50% or 75% of the values are missing. Given a targeted missing data percentage p and the total number of samples j in the considered set, we calculated the total number of values to be masked ($j \cdot n \cdot p$), considering any pre-existing missing values ($j \cdot n \cdot p - \sum_j (n - v_j)$). Following this, a random masking matrix of dimension $j \times n$ is generated to match the set structure. We ensured that at least one value in any fully masked row or column was replaced, avoiding complete data loss in any specific dimension. This approach leads to samples and features exhibiting varied percentages of missing values, all adhering to the MCAR paradigm.

2.4.2 Competitors

We conduct an extensive comparison of our methodology against approaches that incorporate missing data imputation as preprocessing followed by model training, and against those models able to intrinsically handle missing values. In this respect, our analysis exploits 11 unique model competitors, each integrated with the 3 main imputation techniques, i.e., the mean constant imputation, the KNN and the MICE with their default parameters [1]. This is summarized in Table 2.2, where the first column distinguishes between ML and DL approaches, the second lists the base learners, columns $3-5$ report the imputation techniques, and finally last column shows the use of an intrinsic strategy. In this table, each of the 35 competitors is therefore marked by an "$\times$".

We can categorize the competing models based on the subsequent analyses we intend to conduct. Our objective is to benchmark NAIM against leading methodologies in the field of missing tabular data, specifically machine learning models combined with imputation techniques. Listed alphabetically, these models include:

- **AdaBoost** [21]: it is employed as a cascade of classifiers, which is particularly effective for its ability to adaptively focus on hard-to-classify instances, enhancing overall model

Type	Model	Imputers			
		Mean Constant	KNN	MICE	Intrinsic
Machine Learning	AdaBoost [21]	×	×	×	
	Decision Tree [22]	×	×	×	×
	HistGradientBoost [1]	×	×	×	×
	Random Forest [23]	×	×	×	×
	SVM [24]	×	×	×	
	XGBoost [25]	×	×	×	×
Deep Learning	MLP [26]	×	×	×	
	GRAPE [12]				×
	TabNet [5]	×	×	×	
	TabTransformer [6]	×	×	×	
	FTTransformer [7]	×	×	×	

Table 2.2: Combinations of models and imputers used as competitors in the experiments. Each competitor is represented by an "×", given by the combination of a learner and an imputation technique, further to intrinsic strategies.

performance. This characteristic makes it a robust choice for handling diverse datasets, including those with imputed values.

- **Decision Tree** [22]: it is a versatile machine learning algorithm particularly useful for analyzing tabular datasets. It excels in its interpretability, as it visually represents decision-making processes and variable importance, enabling straightforward insights into complex data relationships and patterns.

- **HistGradientBoost** [1]: a variation of a gradient-boosted tree model, recognized for its efficiency in handling large datasets and its ability to improve upon the limitations of traditional gradient boosting by optimizing for speed and memory usage. Its inclusion allows for the assessment of advanced boosting techniques in missing data scenarios.

- **Random Forest** [23]: an ensemble of decision trees, it is noted for its robustness against overfitting. Leveraging a multitude of decision trees enhances the model's accuracy and reliability, making it an exemplary model for evaluating the strengths of ensemble methods in the tabular data domain.

- **Support Vector Machine** (SVM) [24]: it is included for its versatility in dealing with non-linear data separations, offering, as a kernel machine equipped with the RBF kernel, a contrast to tree-based models in handling imputed datasets. The SVMs capacity to project data into higher dimensions where classes are more easily separable makes it a valuable model for comparison.

- **XGBoost** [25]: a sophisticated variation of AdaBoost that employs a gradient descent procedure to minimize the loss when adding weak learners, it stands out for its exceptional performance with tabular data. Its advanced handling of regularization and scalability positions XGBoost as the leading model for comparison in studies involving tabular datasets [2].

This selection of models, each with unique approaches for handling tabular data, provides a comprehensive background for evaluating the performance of our proposed methodology. Furthermore, considering that Decision Tree, HistGradientBoost, Random Forest and XGBoost offer implementations incorporating the MIA strategy, we assessed their effectiveness in managing missing values.

Lastly, we aim to showcase the performance of our method in comparison to other DL models specifically designed for tabular data. Our comparative analysis extends to 5 advanced architectures, each selected for its novel approach to handling tabular data, particularly when combined with imputation techniques for managing missing values. These models, discussed in sections 2.2.1 and 2.2.2, bring unique perspectives and methodologies to the challenges of tabular data analysis:

- **Multilayer Perceptron** (MLP) [26]: a foundational DL model, it stands out for its simplicity and versatility. It consists of multiple layers of neurons, each fully connected to those in the next layer, enabling the model to capture complex nonlinear relationships between features. Despite its straightforward structure, MLP's performance hinges on the quality of input data, making it crucial to pair with effective imputation techniques to address missing values.

- **GRAPE** [12]: designed explicitly to tackle missing data within tabular datasets, this model employs a graph-based approach, representing each data instance as a node and modeling feature interactions as edges. Leveraging GNNs, it propagates information effectively across the nodes, allowing the model to learn directly from incomplete data without preliminary imputation. This structure enables GRAPE to capture complex relationships inherent to tabular data, enhancing robustness and predictive performance even in the presence of extensive missingness.

- **TabNet** [5]: leveraging the self-attention mechanism, it dynamically selects which features to focus on for making decisions. Its ability to perform dynamic feature selection mimics the boosting technique, enhancing its interpretability. This model is particularly suited to interpreting high-dimensional data, offering insights into feature importance and decision pathways.

- **TabTransformer** [6]: this model innovates by embedding categorical features within tabular data, applying transformer-based self-attention mechanisms to create contextual embeddings. It captures complex inter-feature relationships and dependencies, significantly improving the predictive performance on tabular datasets.
- **FTTransformer** [7]: it further explores the potential of transformers in extracting patterns and interactions within tabular data, employing distinct embedding strategies for numerical and categorical features.

These DL models, chosen for their innovative handling of tabular data with respect to classical ML approaches, are integrated with 3 main imputation techniques since none of them can handle missing values for both categorical and numerical features.

2.4.3 Evaluation Metrics and Statistical Analysis

For evaluation, each dataset is divided into 5 stratified cross-validation splits, to maintain the original class distribution, and for each fold, 20% of the training samples are selected for validation. We evaluate each experiment averaging out the values of Area Under the ROC curve (AUC) computed in the different cross-validation folds. AUC is a valuable metric to evaluate classification tasks, since it represents the degree to which a model can correctly classify positive and negative instances across all possible thresholds, making it a comprehensive measure of model performance, even in case of imbalanced data [27].

To ensure the statistical validity of our results, we investigated the statistical differences between the predictions made by NAIM and those by the competitors using the Wilcoxon signed-rank test. To this end, we assessed the rate of experiments for each dataset individually, among the 36 variations with repetitions of the 6 percentages of missing values artificially generated, in which NAIM achieved statistically superior and inferior results compared to the competitors, setting $p = 0.05$.

2.5 Results and Discussions

As outlined in the previous section, we compare NAIM with 35 leading competitors in ML and DL for tabular data, and we test the performance with 36 different combinations of percentages of missing values both in the training and testing sets, across the various datasets, resulting in a total of 6480 experiments, 1296 per dataset. Therefore, in analyzing the performance of the different approaches, we considered each level of missing data within the training set separately, delineating performance metrics as the percentages of missing

data in the testing set increased. In Table C.1 we reported for each combination of model and strategy to handle missing values (reported in different rows), the average performance, in terms of AUC and standard error, across the 5 datasets under consideration, obtained at the different percentages of missing values (reported in the different columns). In particular, the first rows denote the percentage of missing values used in training and the specific percentage of missing data in the test set. Similarly, in Appendix C, we also reported the detailed tables per dataset (Tables C.2 - C.6). As we can see in Table C.1, NAIM achieves the best performance, highlighted in bold, in most scenarios (23 out of 36). To be more specific, we conducted the statistical analysis described in section 2.4.3 and reported in D, to compare the predictions made by NAIM in comparison with those of its competitors. This analysis shows that on average the proposed model achieves better performance in 58.7% of the cases, while it only loses in 1.6% of the cases.

To further analyze the results we divide the competitors into groups and plot the performance constructing 6 separate charts, one for each level of missing data within the training set, which report the performance metrics as the percentage of missing data in the test set increases. We use this representation to compare NAIM with different groups of competitors, namely ML and DL models paired with the imputation techniques (Figure 2.5), model implementing some intrinsic strategy (Figure 2.6) and the various imputation techniques regardless of the model used (Figure 2.7). To present our findings, these charts report the average performance values.

As a first analysis, we compare NAIM with ML and DL models coupled with the 3 different imputers under consideration. In Figure 2.5, in panels A and B for the ML and DL models respectively, we reported the average performance across all 5 datasets and the 3 imputation techniques under consideration. These charts show the decreasing trend in model's performance as the percentage of missing values in the testing set increases, a finding that aligns with our expectations. Interestingly, NAIM stands out for its consistently superior average performance across all levels of missing data, maintaining its high ranking even in the optimal scenario where no data is missing. This observation not only underscores the distinct advantage of NAIM over ML and DL models but also highlights the unexplored potential of DL methods in improving both the model's performance when handling missing data. Furthermore, the performance difference is particularly evident when the training set has 0% missing data (first chart on the left in the first row for each panel A and B). This highlights a significant challenge that current methodologies face: the need to learn how to handle missing data during the training phase, to then correctly infer on testing data with missing values. This aspect emphasizes a critical limitation of conventional approaches and the need for innovative strategies that can effectively address data incompleteness. In this

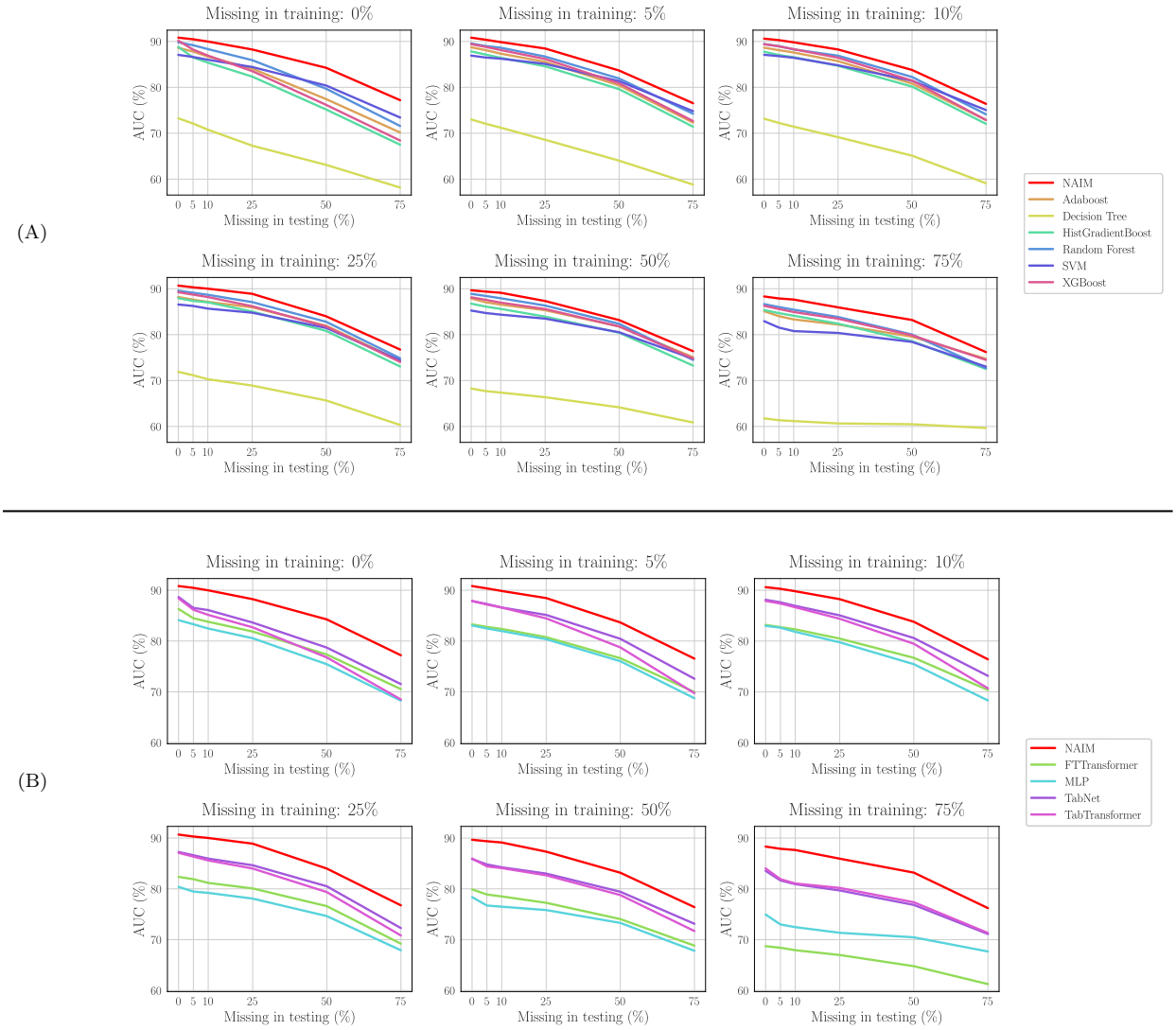

Figure 2.5: Comparison between the test performance of NAIM and the competitor models averaged over the 5 dataset and the 3 imputation strategies: panel A reports the comparison between NAIM and the ML models, whereas panel B compares NAIM with the DL models.

respect, we deem that the better performance of NAIM is due to our regularization technique that enables it to learn how to handle missing values even if all the training data are present.

Then, as a second analysis, we compare NAIM against the models capable of intrinsically handling missing values. In Figure 2.6, which reports the performance of these approaches, the average performances are computed across the 5 datasets. Turning our focus on the top left chart, which plots the performance in the 0% missing values in training scenario, we notice that the difference in performance as the percentage of missing values in the test set increases is higher compared to the other 5 charts where the training is performed on

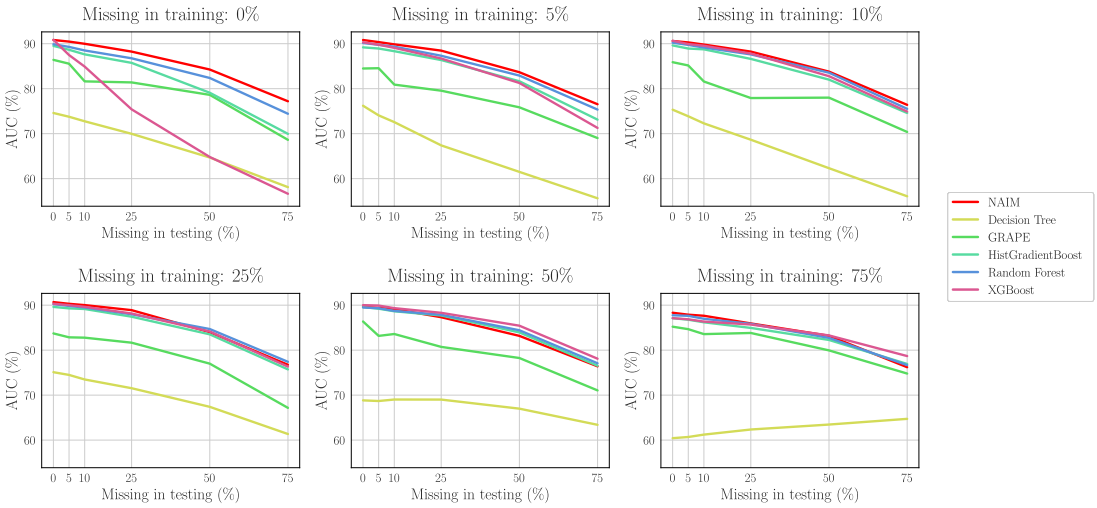

Figure 2.6: Comparison between the test performance of NAIM and the competitors implementing an intrinsic strategy averaged over the 5 dataset.

data containing missing values. This performance drop confirms that neither the imputation techniques nor the intrinsic strategies are effective if no missing values have been seen during training, whereas the stable performances of NAIM validate our novel regularization technique, which allows it to adapt to the presence of missing data in all training scenarios. These charts also show the competitive performance obtained by XGBoost and Random Forest, each delivering comparable performance across a spectrum of scenarios characterized by missing data in the training set, including the optimal scenario of 0% missing in both sets. It is worth noting that in the scenario with 50% missing data in training (second row, second chart from the left), they are even able to outperform NAIM regardless of the percentage of missing values in the test set. Moreover, it is noteworthy that HistGradientBoost achieves performance similar to those of XGBoost and Random Forest, although slightly inferior. On the opposite, Decision Tree and GRAPE, despite being theoretically capable of handling missing data, consistently fail to achieve comparable performance to other models in both missing data scenarios during training and testing.

Subsequently, as a third analysis, to further assess NAIM's ability to handle missing data, we compare it against imputation strategies. In contrast to the previous analyses, here the average performance for each of the imputation methods, reported in Figure 2.7, are computed by averaging the results across the 5 datasets and the 10 models available. The figure illustrates how, in scenarios involving complete training data (first row, first chart from the left), models trained with data imputed using the KNN imputer exhibit slightly

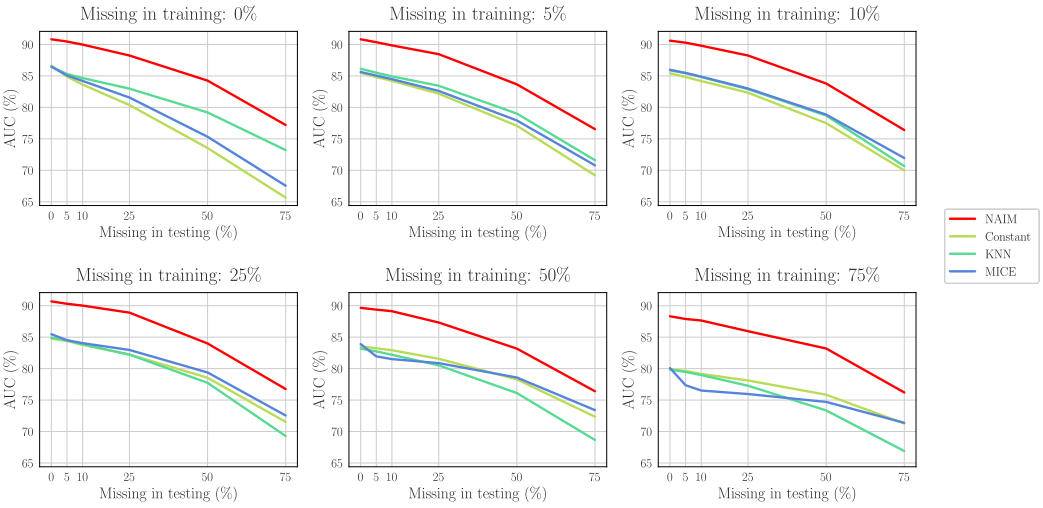

Figure 2.7: Comparison between the test performance of NAIM and the competitor imputers averaged over the 5 dataset and the 10 models available.

superior performance compared to other imputation methods. However, it is important to note that, overall, there are no significant and clear differences among the various imputation techniques, thus requiring extensive experimentation to find the optimal configuration for the data and task at hand.

Upon observing the superior performance of NAIM across all missing data scenarios, we examine its performance robustness to varying levels of missing data (Figure 2.8). To this end, we measure the performance drop presented in experiments with missing data either in the training or in the test set compared to the optimal scenario of 0% of missing data in training and test sets. Subsequently, we average these values to present them in a single plot. To quantify the robustness of the models when dealing with increasing missing values in testing, we evaluate the performance differences across scenarios with complete training data and various missing data percentages in the test sets. In other words, to compute the y coordinates of Figure 2.8, we measure the average performance drop in the experiments reported in the first chart of the first row of panels A and B of Figure 2.5 and in Figure 2.6. Conversely, to assess the robustness of the models when dealing with increasing missing values in training, we examine the performance differences in scenarios with various missing data percentages in training and complete test data. Therefore, we measure the differences among the first performance reported in each of the 6 charts shown in panels A and B of Figure 2.5 and in Figure 2.6 and report their average as x coordinates in Figure 2.8. To facilitate interpretation and effectively rank models based on their robustness, we also plot

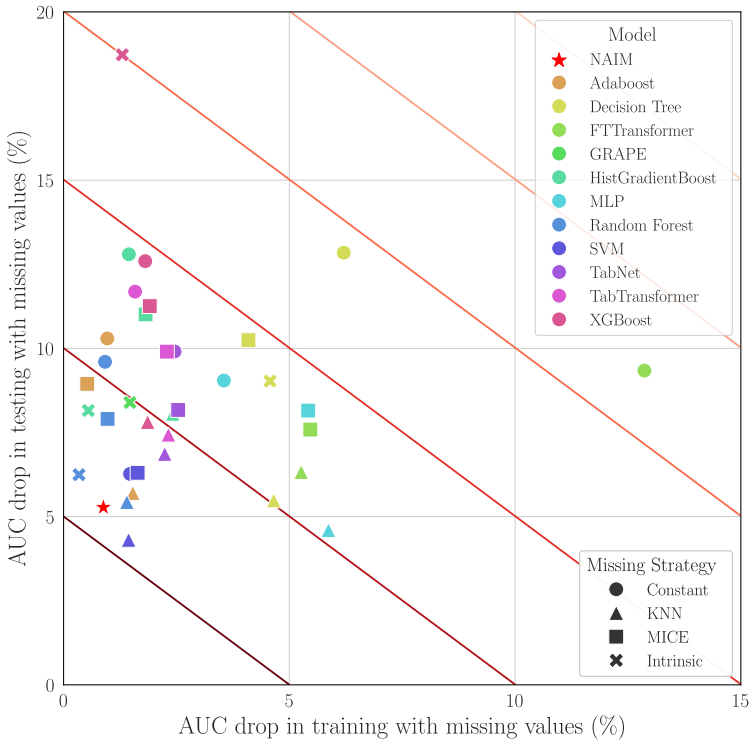

Figure 2.8: Analysis of the robustness of the models to increasing levels of missing data. Each experiment (combination of model and missing strategy) is positioned based on the average drop in performance obtained in scenarios with increasing percentages of missing values in the training and testing sets, separately.

equi-drop lines connecting data points that exhibit equivalent overall performance reductions from the no-missing data state (0% of missing values in both sets). For instance, an experiment exhibiting an average 10% drop in the presence of missing data in the test set and 0% drop in the training set, would be ranked equivalently to a model with the opposite behavior of 0% drop in the presence of missing data in the training set and 10% drop in the set set. Straightforwardly, a model completely robust to missing data in both training and testing sets would be placed on the point $(0, 0)$. We note that NAIM's robustness to missing data is indicative of minimal performance degradation with an increasing number of missing values, since it presents a 0.88% drop when the rate of missing values increases in the training set and a 5.27% drop when the rate of missing values increases in the test set. Notably, despite this analysis positions NAIM just behind the SVM model paired with the KNN imputer, when considering the comprehensive performance evaluations presented in panel A of Figure 2.5 and in Table C.1, NAIM significantly surpasses the competitor. Furthermore, looking

at Table D.1, we can see that NAIM performance is statistically larger in 67.8% of the cases, whereas it is never inferior. This means that SVM is a more robust model but with lower performance compared to NAIM, confirming that our approach presents a superior capacity to maintain high-performance levels despite the presence of missing data. Moreover, from Figure 2.8 we can see that the third model in terms of robustness in the presence of missing values is the Random Forest that exploits the MIA strategy, which also performs statistically worse than NAIM as depicted in Table D.1. These observations confirm the robustness of the proposed model and regularization technique in handling missing values.

As a further analysis to evaluate the contributions of the regularization technique and the masking mechanism, we conducted ablation studies in which the model was trained without regularization, both in scenarios with missing values (NAIM w/o reg) and when combined with imputation techniques (NAIM w/o reg + Imputer). Figure 2.9 shows the average performance of these experiments conducted across the five datasets, while a comprehensive presentation of the results is provided in Tables C.1C.6 in Appendix C. Notably, the contribution of the regularization technique is evident throughout the 6 plots. Indeed, in the first plot (top left), NAIM w/o reg exhibits a performance trend comparable to that of XGBoost in Figure 2.6, suggesting that exposure to missing data during training is essential for the model to learn effectively how to handle them. Conversely, in the final plot (bottom right), where the missing rate is highest, the benefit of regularization becomes less pronounced. When evaluating NAIM combined with the three imputation strategies, it is evident that the masking self-attention mechanism consistently outperforms the imputation-based baselines across all epochs. These findings indicate that each component of our approach, particularly the missing data regularization, contributes significantly to the overall performance of NAIM. Finally, to complement the performance analysis, we also conducted a computational assessment of the DL models considered in this study. Specifically, we report in E the number of floating-point operations (FLOPs) and trainable parameters for each model, highlighting the balance achieved by NAIM between representational power and computational efficiency.

2.6 Conclusion

In this work, we introduced NAIM, an innovative transformer-based architecture for modeling tabular data in supervised learning environments, specifically designed to handle missing values. This architecture simplifies the analysis process, essentially avoiding the need for traditional imputation strategies. Our empirical analysis demonstrates NAIM's superiority over existing state-of-the-art solutions in handling missing data within tabular datasets. Our experiments spanned across NAIM and a combination of 11 ML and DL models, each inte-

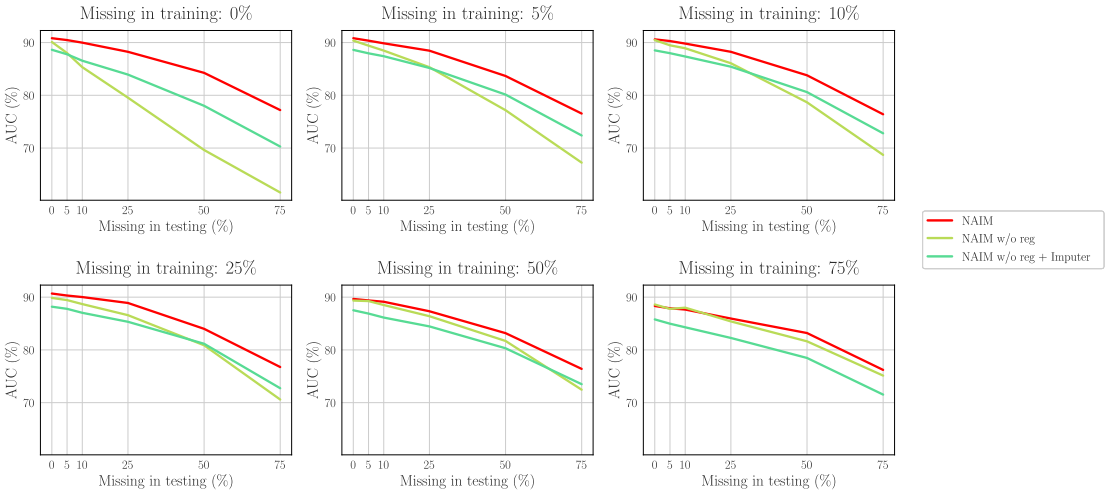

Figure 2.9: Ablation analysis evaluating the impact of the regularization technique (NAIM w/o reg) and the masked self-attention mechanism (NAIM w/o reg + Imputer). The plots report test performance averaged over the 5 datasets. For NAIM w/o reg + Imputer, results are additionally averaged across the 3 different imputers.

grated with 3 distinct imputation methods, and also against those 5 models that intrinsically handle missing values. We performed the analyses on 5 publicly available datasets, which differ in the number of samples and features.

NAIM leverages specialized embedding mechanisms for both categorical and numerical features, coupled with an innovative self-attention mechanism, to maximize the information of the available data at hand. Additionally, we introduced a novel regularization strategy designed to address a significant challenge identified in our research: the need for state-of-the-art approaches to learn how to handle missing values in the training process. This strategy, which involves random masking of each sample in every epoch, equips NAIM with the robust capability to handle missing data under all circumstances, including scenarios where there are no missing values in the training set.

While we have attained promising results, this work has some limitations that are now discussed introducing the related future directions:

- Extension to MAR and MNAR mechanisms: In this work we focused on the MCAR setting, which is the most commonly adopted evaluation regime in the literature and allowed us to systematically explore a wide range of scenarios. Nevertheless, real-world applications often involve more complex missingness mechanisms, such as Missing At Random (MAR) or Missing Not At Random (MNAR), where the probability of missingness depends on observed or unobserved variables. Future work will therefore extend

the evaluation of NAIM to MAR and MNAR conditions, in order to further validate its robustness and generalizability in more realistic settings.

- Multimodal learning with missing modalities: We plan to extend NAIM beyond the tabular domain by developing a multimodal variant capable of handling heterogeneous data types and the complete absence of entire modalities (e.g., missing imaging or tabular information). This extension would enable a unified framework that leverages available data across different domains while degrading as little as possible in performance when some sources are absent.

- Efficient attention mechanisms: Although transformers are powerful, they are known for their high computational cost. Future investigations should explore alternative attention formulations that can reduce memory and computational requirements. This would make NAIM more scalable, particularly for large datasets with high-dimensional features.

- Model interpretability and transparency: To foster trust and adoption in critical domains like healthcare and finance, future studies should deepen the interpretability of NAIM. This includes leveraging attention weights to understand feature importance, analyzing learned embeddings to uncover latent data structure [28], and developing attribution methods tailored to transformers with missing inputs. These efforts would help demystify the internal reasoning process of the model, especially in the presence of incomplete data.

- Self-supervised and semi-supervised learning: Since labeled tabular data is often scarce, particularly in specialized domains, future works should investigate self-supervised pretraining strategies for NAIM. Techniques such as contrastive learning, masked feature modeling, or denoising objectives could help the model learn robust representations from raw data, improving downstream performance when only limited supervision is available.

- Temporal modeling for incomplete time-series: Another natural extension of our framework involves adapting NAIM to handle longitudinal or sequential data. Time-series datasets, especially in domains like sensor monitoring or electronic health records, frequently exhibit irregular sampling and missingness. Future work should aim to develop a time-aware version of NAIM that integrates temporal encodings.

- Domain-specific evaluation and adaptation: While our evaluation included diverse public datasets, future work should assess NAIM in domain-specific real-world appli-

cations. Particular attention should be given to healthcare datasets, where missingness is often informative and non-random. This would not only validate the models utility in complex scenarios but also drive the development of domain-adaptive mechanisms that incorporate domain knowledge into the learning process.

These potential directions not only underscore the versatility and expansiveness of our approach but also highlight the fertile ground for future advancements in the domain of tabular data analysis. In conclusion, when faced with tabular datasets exhibiting missing data either in the training or testing phase, a promising and competitive choice among the tested methods is NAIM, as it enables the extraction of useful information from the available data without any need for imputation.

In this chapter, we presented NAIM, a transformer-based model specifically designed to handle missingness in tabular data by integrating a masked attention mechanism and a missingness-aware regularization strategy. The proposed approach demonstrated strong performance and robustness across various benchmark datasets, outperforming existing methods in most scenarios. However, the experiments were conducted on synthetic or structured datasets and focused solely on tabular data. These limitations hinder the clinical transferability of the model, as real-world medical datasets often include complex temporal dynamics, heterogeneous patient populations, and partially observed clinical histories. To address this gap, the next chapter focuses on applying machine learning models to real clinical data for survival prediction in cancer patients, emphasizing the interpretability and stability of predictions in the presence of naturally occurring missing information.

External evidence. In a subsequent study [29], NAIM was evaluated in the Pembroreal 5Y registry, a global real-world cohort of 1050 patients with PD-L1$\geq$50% treated with first-line pembrolizumab across 61 institutions in 14 countries. NAIM handled heterogeneous missingness without imputation and supported time-dependent risk modelling; reported performance included a c-index around 0.63 (risk of death) and an $AUC \simeq 0.61$ for 5-year survival, accompanied by SHAP-based explanations highlighting evolving prognostic factors. This multicentre use case substantiates NAIM's clinical robustness while also showing headroom for further optimisation and calibration.

Chapter 3

Survival Prediction from Incomplete Clinical Data

Building upon the findings of the previous chapter, we now shift our attention from generic benchmark datasets to real-world clinical scenarios. In particular, we explore the prediction of overall survival in oncology patients using routinely collected clinical data, where missingness naturally arises due to irregular follow-up schedules and incomplete records. Unlike the controlled settings of Chapter 2, this application highlights the practical challenges of modeling clinical features in the presence of partial observations. Furthermore, the emphasis moves beyond pure performance to include interpretability, a crucial aspect in clinical decision-making. This chapter thus represents a key step towards translational applications of robust modeling strategies in healthcare settings.

3.1 Introduction

In recent years, Artificial Intelligence (AI) has rapidly become a part of our daily lives, including healthcare [30]. There have been notable advances in applying quantitative methods in clinical practice, which have paved the way for precision medicine. Cancer research is one of the most promising areas where AI can be applied [31]. According to the World Health Organization, cancer is responsible for nearly 10 million yearly deaths globally, with lung cancer accounting for 18% of these [32] and non-small cell lung cancer (NSCLC) is the most frequent type, being approximately 82% of all cases [33].

In cancer research, a key factor is the overall survival (OS) of the patient, which refers to the time from the initial cancer diagnosis to the time of death. Identifying subgroups of patients with a higher or lower chance of survival is critical in developing effective strategies

to improve OS rates. For example, in the case of NSCLC, the 5-year survival rate is only 26%, and this rate drops to a mere 7% when cancer returns locally or spreads to distant organs [33]. Therefore, although some AI-based methods have been proposed in healthcare [34], accurately predicting OS remains a major challenge.

In medicine, it can be quite challenging to obtain a dataset, i.e., tabular dataset, with no missing features: how to handle this situation effectively is a key question in AI research. In fact, traditional methods require complete data samples, so standard practices involve either excluding samples with missing values or utilizing imputation strategies. However, it is important to note that these methods can compromise the findings, introducing bias and reducing statistical power. Therefore, research should go towards the development of models which can cope with missing features without any imputation.

To address these issues, we introduce a novel approach that tackles the problem of handling missing features in tabular data in the study of OS analysis for patients affected by NSCLC. Indeed, tabular data are the most simple and diffused data type in AI applications [35]. Our method stems from the transformer architecture [13] and leverages the idea of the mask inside the self-attention module to learn from incomplete input data. By masking missing features, it learns only from available data, avoiding any type of imputation of missing features, with the goal of improving the performance of survival prediction of patients affected by NSCLC.

To validate the proposed model, we compared its predictive performance with feature imputation and OS prediction state-of-the-art approaches. Indeed, even though some methods are capable of handling missing features in the test phase, to the best of our knowledge, none has yet addressed the issue of how to deal with them during training without the need for any imputation method in the context of OS prediction [34].

The main contributions of this work are the following:

- We propose a specialized transformer-based decision support system on tabular data, for predicting lung cancer OS using clinical data.
- Our approach efficiently handles missing data, eliminating the need for imputation strategies.
- Our approach uses ad-hoc designed loss functions allowing the use of both censored and uncensored patients, without the need to exclude any patient.
- We assessed the performance of the proposed model at various time granularities, confirming its robustness across different time windows. The results obtained on the

OS prediction in NSCLC patients outperform OS state-of-the-art models, regardless of the imputation method used.

The manuscript is organized as follows: section 3.2 presents the state-of-the-art of survival analysis and data imputation techniques; section 3.3 reports the details about the data employed in the analyses; section 3.4 introduces the proposed model and explains the metrics employed; section 3.5 discusses the experimental results; section 3.6 provides concluding remarks.

3.2 Background

Survival analysis, also known as time-to-event analysis, plays a crucial role in various fields, especially medicine [36]. It aims to understand the relationship between covariates, such as patient features, and the distribution of survival times. In the lung cancer setting, this type of analysis helps to identify risk factors that affect survival and to compare risks among different subjects, with the aim of tailoring the right therapy for the right patient".

One important aspect to consider in this particular field is censored data, specifically right-censoring. This occurs when a patient withdraws from the study, is lost to follow-up, or is still alive without experiencing the event of interest at the last follow-up. In such cases, a special analysis is required since it is not known when and if the event occurred, and thus these samples cannot be considered together with uncensored data, i.e., patients for which it is known when they have experienced the event. However, most of the studies that predict OS in NSCLC have approached the survival problem as a classification task, dividing patients into high and low risks based on a survival time threshold [37, 38, 39, 40, 41, 42]. This approach does not fully exploit the information of the censored patients, since it excludes from the analysis those patients with a survival time shorter than the threshold. In contrast, other studies predict the risks faced by patients, which can be used to evaluate the correct ordering of the patients through the *C-index* metric [43, 44, 45, 46, 47, 48]. The downside of these approaches is that they do not make full use of the available temporal information, not taking into account possible changes in risk over time. On the contrary, another possible approach is to use several output nodes for each time interval, which represent the risk the patient experienced at the specific time interval. Indeed, in [49] the authors present ad-hoc designed loss functions to exploit information from both uncensored and censored patients, enabling learning from the latter that the event of interest did not occur up until the last follow-up. This was achieved by defining the survival problem as identifying the first time an underlying stochastic process hits a specific boundary, also known as the first hitting time.

Although this approach still does not enable predictions to be made about censored samples, it does permit learning from them.

To tackle the survival task, a few models in the literature have been proposed. The most standard used approach is the Kaplan-Meier estimator, which can take censored samples into account, but does not incorporate patient covariates, making it useful to estimate the survival rate at the population level but not at the patient level. Instead, the most commonly used method is the Cox proportional hazard (CPH), which incorporates the features of the patient, but assumes that the hazard rate, i.e., the probability of experiencing the event within a short time interval, is constant and that the log of the hazard rate is a linear function of the covariates. These two assumptions are known as the proportional hazard assumption, and a few methods have been proposed to address the limitations it introduces. The Survival Tree (ST) and the Random Survival Forest (RSF) are extensions of the CART and Random Forest algorithms that deal with censored survival data. ST accomplished this by a recursive partitioning procedure based on maximizing the dissimilarity in the survival distributions of patients between different regions of the covariate space, thus taking into account not only the homogeneity of predictor variables but also the survival probability and time of the events. RSF estimates the cumulative hazard function for each case's terminal nodes by leveraging out-of-bag data and then averages the cumulative hazard functions of all trees in the forest. The cumulative hazard function is a measure that describes the cumulative probability of experiencing a particular event, such as death or failure, up to a specified time point. Instead, DeepHit (DH) [49] offers a new approach that predicts the probability of the first hitting time of an event using a deep neural network, employing a loss function that exploits survival times and relative risks to predict the cumulative incidence function, a measure useful in understanding the risk of developing a particular health outcome, as it takes into account both the risk of developing the outcome and the time period over which the risk is evaluated.

When working with medical data, an important aspect to consider is the presence of incomplete records. This phenomenon may be due to different reasons and can hinder the employment of AI models. Although some strategies have been proposed to address this issue, only a few methods are commonly used in a healthcare setting [34]: complete case analysis, overall mean imputation, k-nearest neighbors (kNN) imputation, multiple imputations by chained equations (MICE) and MissForest. The first two methods for dealing with missing data, complete case analysis, and overall mean imputation, involve either discarding incomplete samples or imputing the missing values with the mean value of the corresponding feature computed from the available data, which are straightforward but have some drawbacks. Specifically, the former may discard too many samples, which can prevent the learning

of deep learning techniques, whereas the latter has a high likelihood of introducing bias into the model's final outcome. As a result, kNN, MICE and MissForest have been proposed as alternative solutions. The first method identifies similar samples to the one being analyzed by calculating the distance based on other non-missing features in order to impute a more appropriate value. The second one imputes missing values by modeling each feature with missing values as a function of other features in a round-robin fashion. The third one uses a Random Forest model to predict missing data, which starts by imputing missing values using a simple strategy, e.g., mean and mode imputation, and then iteratively updates the imputations using the model predictions, refining these estimates until the changes between iterations are minimal.

In this context, we present our approach, which combines the strengths of DH and transformer architecture [13]. We use the transformer architecture to exploit the power of the mask in the self-attention module, which allows us to avoid any imputation of missing values in the survival analysis task.

3.3 Materials

To validate our clinical decision support system, presented in the next section, we used clinical data from the CLARO dataset [50]. This consists of 297 patients affected by NSCLC, who underwent concurrent chemoradiation for locally advanced NSCLC and systemic treatment for metastatic disease. The OS of the entire population, which included 184 censored and 113 uncensored patients, has a mean of 20.74±42.45 months (95% CI). The population was enrolled in the study under two separate Ethical Committee approvals, including a retrospective phase that was approved on October 30th, 2012, and registered on ClinicalTrials.gov on July 12th, 2018 with the identifier NCT03583723. The prospective phase was approved with the identifier 16/19 OSS. The Institutional Review Board approved this analysis, and all patients provided written informed consent.

The clinical data that we have collected contains 8 clinical descriptors, as outlined in Table 3.1, which reports the distribution and the amount of missing values for each feature. As mentioned above, our features are composed of personal information, i.e., *Age* and *Sex* (assigned at birth), and details about tumor histopathology, i.e., clinical target volume (CTV), *Overall Stage*, tumor (T), nodule (N), metastasis (M) stages and *Histology*. To determine the stage of the tumor, we had two radiation oncologists with extensive experience independently review CT scans and assign staging scores for the tumor: *Overall Stage*, T, N, M. If there was any disagreement between the two experts, they would review the patient's CT images together until they reached a consensus. It is worth noting that some patients in

our study did not undergo a histopathological examination, this is why the unknown" class is included as one of the categories for the *Histology* feature.

To prepare the data for analysis, we applied one-hot encoding for categorical features and z-score normalization for continuous features[1]. For our approach, we generated empty vectors for the one-hot encoding of the missing features.

Feature	**Missing Data**	**Categories**	**Distribution**
*Age**	0 (0.0%)	< 68 *years*	138 (46.46%)
		≥ 68 *years*	159 (53.54%)
Sex	0 (0.0%)	F	97 (32.66%)
		M	200 (67.34%)
*CTV**	112 (37.71%)	$< 146.51\ cm^3$	115 (38.72%)
		$\geq 146.51\ cm^3$	70 (23.57%)
Overall Stage	0 (0.0%)	II	8 (2.69%)
		III	188 (63.31%)
		IV	96 (32.32%)
		Recurrence	5 (1.68%)
T	116 (39.06%)	1	11 (3.70%)
		2	40 (13.46%)
		3	74 (24.92%)
		4	56 (18.86%)
N	104 (35.02%)	0	17 (5.72%)
		1	39 (13.13%)
		2	112 (37.71%)
		3	18 (6.06%)
		Recurrence	7 (2.36%)
M	94 (31.65%)	0	201 (67.68%)
		1	2 (0.67%)
Histology	3 (1.01%)	Adenocarcinoma	153 (51.52%)
		Squamous	73 (24.58%)
		Other	20 (6.73%)
		Unknown	48 (16.16%)

Table 3.1: Patients' characteristics. For each feature is reported the number and percentage of missing values, along with the distribution (number and percentage) of the possible categorical values. Note that the variables marked with *, are continuous, but, for the sake of clarity, we have presented their distribution using their mean values as thresholds. It is important to note that the model used the original continuous values of these variables.

[1]This operation is performed on the test set based on the parameters computed using the training set.

3.4 Methods

In this section, we first describe the architecture of the proposed approach with an example of its main blocks, next we illustrate the loss function used during training and the evaluation metric employed, and finally we report the experimental setup used to perform the experiments.

3.4.1 Model

Given that AI methods for OS prediction require complete data [34], and that in healthcare it can be quite challenging to obtain a complete data set without missing values, we need to effectively handle the data without the necessity of imputing or removing any information. Taking inspiration from the cutting-edge transformer architecture [13], we present here a novel model that takes into account only the available features. Our approach involves adapting the transformer's encoder architecture to tabular data, via a novel positional encoding for tabular features, and utilizing padding to mask any missing features within the attention module, enabling the model to ignore them effectively.

The schematic representation of the proposed model is shown in Figure 3.1: left panel, denoted by the letter A, represents the overall architecture from the input (the orange block shown at the bottom of the figure) to the output (the blue block at the top of the figure), whilst right panel, marked as B, offers an example of four blocks of panel A, which can be identified by the name as well as by the color.

The input to the model, shown in the first block represented in orange in Figure 3.1, is a feature vector $\mathbf{x}$ of dimension d composed of the preprocessed patient information. The second block, represented in yellow, is the positional encoding, which is used for identifying the feature itself without explicitly encoding the feature order. To achieve this, we explored the use of a one-hot encoding vector representing the position of each feature. In this respect panel B of Figure 3.1 shows an example in the case of the CLARO dataset: here the initial feature vector is turned into a $d \times (d + 1)$ matrix, where the columns from the first to the second-last represent the positional encoding, whilst the last column reports the feature vector.

The third block is the transformer encoder (light gray block), which consists of a stack of M identical layers. Each layer has two sub-layers: the first is a multi-head self-attention mechanism (red block), and the second is a position-wise fully connected feed-forward network (green block). As in the original architecture [13], we employ a layer normalization (purple block) and a residual connection (denoted with $\oplus$) around each of the two sub-layers.

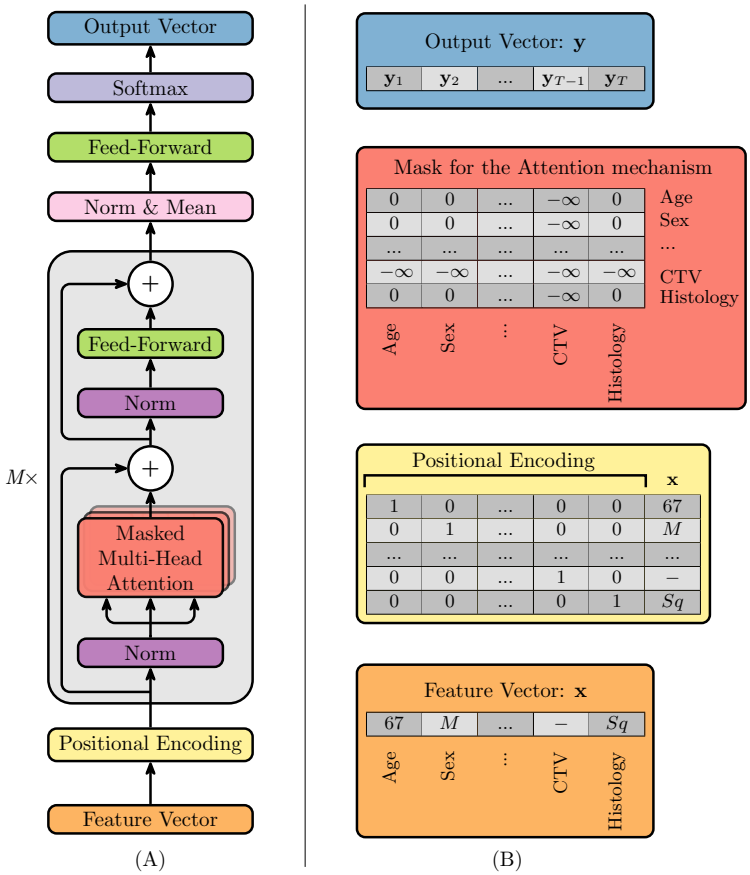

Figure 3.1: Schematic representation of the proposed model: (A) Architecture of the proposed approach and (B) example of positional encoding, mask and output vector, where the − symbol represents a missing feature. Note that for simplicity of representation, just a few features are reported and none of the preprocessing procedures are applied.

The normalization step precedes each of these blocks to reduce the risk of the vanishing or exploding gradients problem, which can occur during the training process.

To handle the presence of missing features, we drew inspiration from the padding mask technique used in natural language processing [13]. This technique extends the capabilities of the attention mechanism, preventing leftward information flow, which exploits the use of the mask within the scaled dot-product attention mechanism to mask out, setting to $-\infty$, any values that would result in illegal connections. In this regard, the red block of panel B of Figure 3.1 shows an example of the mask employed in the attention mechanism. Using such a mask, we were able to effectively ignore any missing features, without requiring any imputation strategy.

Next, we applied a normalization and averaging step (pink block) in order to get a latent representation vector, respective to the non-missing features, to be fed to the final classifier, composed of a feed-forward module (green block) and softmax function (violet block). The

feed-forward module that succeeds the encoder maps the encoder's embedding to the output vector's dimension T. Hence, each element $\mathbf{y}_t$ of the output vector $\mathbf{y}$ (blue block) represents the probability that the event occurs at the time point t.

3.4.2 Training and Testing

When training the model, the goal is to correctly predict the probability that the event of interest, formally denoted as $k = 1$ and corresponding to the patient's death, occurs at time t, given the patient's feature vector $\mathbf{x}$ and under the constraint that t is smaller or equal to s, the true time when the event $k = 1$ occurs. Straightforwardly, we identify with $k = 0$ the event of censoring for patients who did not experience the event. The cumulative incidence function $F(t|\mathbf{x})$ is therefore defined as:

$$F(t|\mathbf{x}) = P(s \leq t, k = 1|\mathbf{x}) = \sum_{s=0}^{t} P(s, k = 1|\mathbf{x})$$

However, since the true cumulative incidence function $F(t|\mathbf{x})$ is not known, it can be estimated with $\hat{F}(s|\mathbf{x})$, which is the cumulative sum of the outputs of the model, defined as:

$$\hat{F}(s|\mathbf{x}) = \sum_{t=0}^{s} \mathbf{y}_t.$$

Therefore we can now compute this measure and thus compare the risks faced by different patients at a specific time.

We train the proposed model using the loss function L presented in [49], specifically designed to handle censored patients. It is composed of two terms, L_1 and L_2, so that

$$L = L_1 + L_2$$

where the first term captures the death and censoring times of patients, evaluating respectively the output $\mathbf{y}$ and the estimated cumulative incidence function $\hat{F}$ at patient-specific event times, whereas the second term measures the correct ordering of the patients based on their relative risk. Indeed, L_1 is the log-likelihood of the distribution of the first hitting time of an event, defined as:

$$\begin{aligned} L_1 = -\sum_{i=1}^{N} [&\mathbb{1}(k^{(i)} = 1) \cdot \log(\mathbf{y}^{(i)}_{s^{(i)}}) + \\ &+ \mathbb{1}(k^{(i)} = 0) \cdot \log(1 - \hat{F}(s^{(i)}|\mathbf{x}^{(i)}))] \end{aligned}$$

where $\mathbb{1}(\cdot)$ is an indicator function, and the summation is performed on all N samples, where the apex (i) indicates that the information is related to the specific patient i. Through two terms, it exploits the information of both uncensored and censored patients: the first term captures the first hitting time to maximize the patient's risk at the specific time of occurrence of the event; the second one maximizes the survival function, defined as $1 - F(t)$ [51], evaluated at the patient's last follow-up.

L_2 employs a ranking loss function based on the concept of concordance. Essentially, it indicates that a patient who has passed away at a certain time should be considered at a greater risk than a patient who is still alive at that time. It is defined as:

$$L_2 = \sum_{i=1}^{N} \sum_{j \neq i}^{N} A_{i,j} \cdot \exp\left(-\frac{\hat{F}(s^{(i)}|\mathbf{x}^{(i)}) - \hat{F}(s^{(i)}|\mathbf{x}^{(j)})}{0.1} \right)$$

where $A_{i,j} = \mathbb{1}\left(k^{(i)} = 1, s^{(i)} < s^{(j)}\right)$. Using these terms the loss function penalizes the incorrect ordering of pairs. Indeed, on the one hand, $A_{i,j}$ is an indicator function that identifies pairs of patients whose comparison is meaningful, known as acceptable pairs. These pairs consist of patients in whom the first patient experienced the event at a specific time point, while the second patient has a longer survival time. On the other hand, the exponential function compares the risks faced by the two patients, i and j, at the time the first one experienced the event. In particular, it assumes small values in the case of a correct ordering, i.e. if the first patient has a higher risk than the second, whilst it assumes large values in the case of a wrong ordering.

3.4.3 Evaluation Metric

Once a survival analysis model is trained, we need to evaluate its outputs taking time into account, as does the time-dependent concordance index (*Ct-index*) does. It is an evolution of the commonly used *C-index*; this latter is based on the assumption that patients who lived longer should be assigned lower risks than those who lived a shorter period. However, the *C-index* does not account for any potential changes in risk over time, whereas the *Ct-index* is calculated by comparing pairs of patients, where one patient has experienced the event at a specific time while the other patient has neither experienced the event nor been censored

by that time. Formally, *Ct-index* is defined as:

$$\begin{aligned} \textit{Ct-index} &= P\left(\hat{F}(s^{(i)}|\mathbf{x}^{(i)}) > \hat{F}(s^{(i)}|\mathbf{x}^{(j)})|s^{(i)} < s^{(j)}\right) \\ &\approx \frac{\sum_{i=1}^{N}\sum_{j\neq i}^{N} A_{i,j}\cdot \mathbb{1}\left(\hat{F}(s^{(i)}|\mathbf{x}^{(i)}) > \hat{F}(s^{(i)}|\mathbf{x}^{(j)})\right)}{\sum_{i=1}^{N}\sum_{j\neq i}^{N} A_{i,j}} \end{aligned}$$

This index is not based on a fixed time, unlike the *C-index*, which considers only the ordering of subjects at a fixed time point. Instead, the *Ct-index* takes into account the timing of events for each subject over time. In fact, using $A_{i,j}$ and the indicator function that verifies whether, at the time the first patient experienced the event, he/she presents a higher risk than the second one, this metric computes the fraction of correctly ordered patients out of the total number of acceptable pairs.

3.4.4 Experimental setup

Our architecture adopts 12 consecutive encoder layers (M), with each layer utilizing 17 attention heads and a feed-forward module composed of one hidden layer of 3072 neurons. We do not further investigate any other hyperparameters configuration, since their tuning is out of the scope of this manuscript. Nevertheless, the No Free Lunch" Theorem for optimization states that there is no universal set of hyperparameters that will optimize the performance of a model across all possible datasets [52].

To have a fair comparison between the models employed in the analysis, we applied 5-fold stratified cross-validation, maintaining the distribution of censored and uncensored patients among the different folds. Thus, we divided the data into test set (20%) and train set (80%), part of which was used as validation set (20%).

We compared our approach with each pair of state-of-the-art imputation strategies and models for OS analysis presented in section 3.2. Regarding the imputation methods, we employed overall mean imputation strategy, kNN imputer, MICE and MissForest with their default parameters [53, 54]. More specifically, we opted for 5 neighbors and the euclidean distance for the kNN imputer, for a maximum number of iterations equal to 10 and a mean initial strategy for the MICE strategy, and for a maximum number of iterations equal to 10, 100 estimators and the squared error as a criterion for the growth of the MissForest. We opted not to use the complete case analysis since in our application the number of patients would be almost halved, passing from 297 to 158. Focusing on the models, we tested the CPH, the ST, the RSF and the DH. For training the CPH, ST and RSF, we used their default parameters [55]. More specifically, we opted for the breslow method to handle tied

event times and 100 iterations in the training of CPH, for the best splitter, a minimum of 6 samples to split a node and 3 samples to define leaf nodes in the growth of the ST, and 100 estimators, a minimum of 6 samples to split a node and 3 samples to define leaf nodes in the growth of the RSF. In training both our model and DH we opted for the Adam optimizer, a batch size of 32, an initial learning rate of 10^{-4} and a Xavier initialization. The training was set to a maximum of 1500 epochs, in conjunction with an early stopping criterion and a learning rate scheduler both based on the validation loss with patience of 200 and 100 epochs, respectively.

To conduct the comparisons, we used different units of time, i.e., one month, one year, and two years, covering a period of six years. Note that we defined this time limit in order to include at least 95% of the patients' survival times without any modification, whereas we considered all the patients with a longer survival time as censored. Thus each element of the output vector indicates the level of risk that the patient faces within the corresponding time interval. It is worth emphasizing that the output vector's size N, changes across the different experiments based on the specific unit of time employed: 72 elements for the 1-month, 6 for the 1-year and 3 for the 2-year time unit.

3.5 Results and Discussions

As reported in section 3.4.4, we compared the performance of our approach with the state-of-the-art OS models, trained on the CLARO dataset [50], using different imputation strategies. The results, in terms of *Ct-index*, are presented in Table 3.2, where the first two columns report the combination of model and imputation method employed, whereas the rest of the columns represent the units of time applied. The mean value and standard error reported are calculated on the different folds and the best mean performance for each unit of time is marked in bold. As we can see, our approach always outperforms the benchmarks.

These results, independently from the imputation strategy applied, confirm the considerations made in section 3.2. It appears that CPH performs the worst, likely due to the limitations of the proportional hazard assumption in this context. Conversely, all other methods, which eliminate this constraint, perform better than CPH. Notably, DH and our approach outperform RSF, showcasing the potential of deep learning once again. We deem that the improvement in the performance of our approach compared with DH is due to its ability to handle a high degree of missing data without the need for imputation, which has the potential to bias the final prediction. Indeed, the missing data imputation itself poses some challenges in the selection of the most appropriate approach to the task at hand; furthermore, most of the existing imputation methods struggle to handle high levels of missing

Model	Imputation	1-month	1-year	2-year
CPH	Mean	61.10 ± 3.05*	60.92 ± 3.59*	47.64 ± 8.72
	kNN	60.72 ± 3.90	59.72 ± 5.46	48.51 ± 9.14
	MICE	60.80 ± 3.16*	60.01 ± 4.29*	47.16 ± 8.74
	MissForest	61.34 ± 3.05*	58.95 ± 4.11*	44.99 ± 9.44
ST	Mean	23.46 ± 4.24*	39.27 ± 4.02*	43.82 ± 2.78*
	kNN	27.05 ± 3.14*	22.51 ± 3.20*	35.28 ± 4.29*
	MICE	22.03 ± 4.17*	45.51 ± 4.02*	37.29 ± 7.47*
	MissForest	22.31 ± 3.48*	28.45 ± 8.12*	37.15 ± 10.88*
	–	26.49 ± 2.05*	37.61 ± 7.32*	42.71 ± 6.19*
RSF	Mean	65.88 ± 2.37	72.16 ± 3.02	57.59 ± 7.08
	kNN	61.92 ± 3.27*	68.24 ± 1.34*	54.66 ± 8.10
	MICE	67.10 ± 3.31	71.53 ± 4.06	57.45 ± 8.21
	MissForest	62.86 ± 2.66	67.69 ± 5.99	58.79 ± 11.16
DH	Mean	69.72 ± 3.67	75.19 ± 5.38	78.23 ± 4.11
	kNN	68.24 ± 3.94	71.28 ± 6.54	77.94 ± 3.66
	MICE	71.04 ± 2.92	75.26 ± 5.81	78.39 ± 3.54
	MissForest	68.39 ± 3.41	72.64 ± 5.42	74.64 ± 5.34
Ours	–	$\mathbf{71.97 \pm 2.39}$	$\mathbf{77.58 \pm 1.82}$	$\mathbf{80.72 \pm 9.11}$

Table 3.2: Performance of tested models in terms of *Ct-index* (mean ± standard error). The asterisks (*) denote experiments with performances that are statistically different (*p-value*< 0.05) from our proposed model.

data, leading to a drop in performance. On the contrary, our approach smoothly copes with this situation by learning only from the available features.

When we examine the *Ct-index* of the different models at various units of time, we observe that those of DH and our model improve as unit time increases. This can be attributed to the reduction of complexity for the task of correctly ordering pairs, as the granularity of the problem reduces, suggesting that our method effectively interprets the available features to estimate the OS time.

Focusing on the differences between the various imputation strategies, Table 3.2 highlights how difficult is to determine the most appropriate method, since the outcome depends both on the data and the models employed in the analysis. Therefore, our approach not only proves to be the best in terms of achieving the highest performance, but it also enables us to eliminate one of the variables from the problem by simply disregarding the missing features instead of searching for the most appropriate imputation strategy for the task at hand.

To further validate our results, we performed a paired t-test to evaluate the statistical

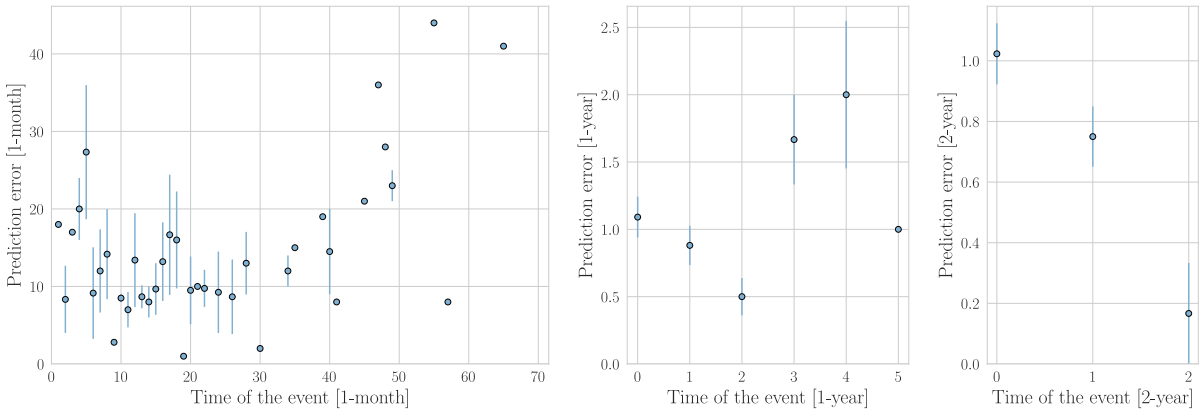

Figure 3.2: Prediction errors made for different patient groups by the proposed method, taking into account the actual times of occurrence of the event. The graph reports the mean and standard error of the prediction of uncensored patients, as for censored patients no valid prediction can be made since the event did not occur.

difference between the performance obtained in different folds by our model compared with all competitors. In Table 3.2, denoted by the asterisks (*), we report the results with a *p-value*< 0.05. These results show that our approach is significantly better than ST for all combinations with imputers and with all time granularities. Most notable, among the various combinations, is the case without imputers, showing that our model can handle missing values better. With regards to CPH, our model is statistically better than the configurations in combination with overall mean imputation, MICE and MissForest. Moreover, when the time granularity is increased to 2 years, the performance is no longer statistically distinguishable from those of our model, again indicating the reduction in complexity of the problem as granularity increases. In contrast, looking at the results obtained by RSF and DH, only the combination of RSF with kNN imputer for the time granularities of 1 month and 1 year are statistically worse than those of our model. Despite this, the results obtained from our model are still worthwhile since they are on average superior or comparable to the others, but having simplified the overall analysis, since no imputation of missing data is required.

Moreover, we performed an analysis to better understand on which patients the proposed model makes most errors. In Figure 3.2, in panels A, B, and C for the time granularities of 1 month, 1 year, and 2 years, respectively, we reported the mean errors and relative standard errors for uncensored patients grouped by unit of time in which the event occurred. From panel A we can observe that most uncensored patients, who present the event in the first few years after the diagnosis, have the lowest error, which is below 30 months, whereas, with regard to those patients who have survived longer, the errors are on average larger, even

surpassing 40 months. When the granularity of time increases, on the other hand, the trend reverses, leading to a minimum error of 4.8 months (0.2 of a 2-year period) in the case of patients who died in the third two-year period, as reported in panel C. This trend inversion, which probably also affects performance explaining its increase as time granularity increases, is perhaps due to the difference in the numbers of patients in the different groups of patients.

Additionally, we performed an explainability analysis to better understand which features most influence the model's decisions. We used the SHAP method [56], an explainability technique based on Shapley values, to estimate the contribution of each feature to the final output. Figure 3.3 shows the SHAP summary plots for the three models implemented with the 3 time units, i.e., 1 month, 1 year, and 2 years. In these plots, the order of the features is based on their importance in distinguishing the uncensored class. The higher the position of a feature in the plot, the more important it is. To represent the global feature importance, we averaged the absolute SHAP values obtained for each patient and time represented in the output vector. Moreover, to get a general overview of the importance of the categorical features, we averaged the contributions of each category. As we can see, in all time granularity models, the most relevant features for the task at hand refer to disease-related values such as the volume of the CTV and the T, N and M stages, which are measures of the severity of the tumour [57].

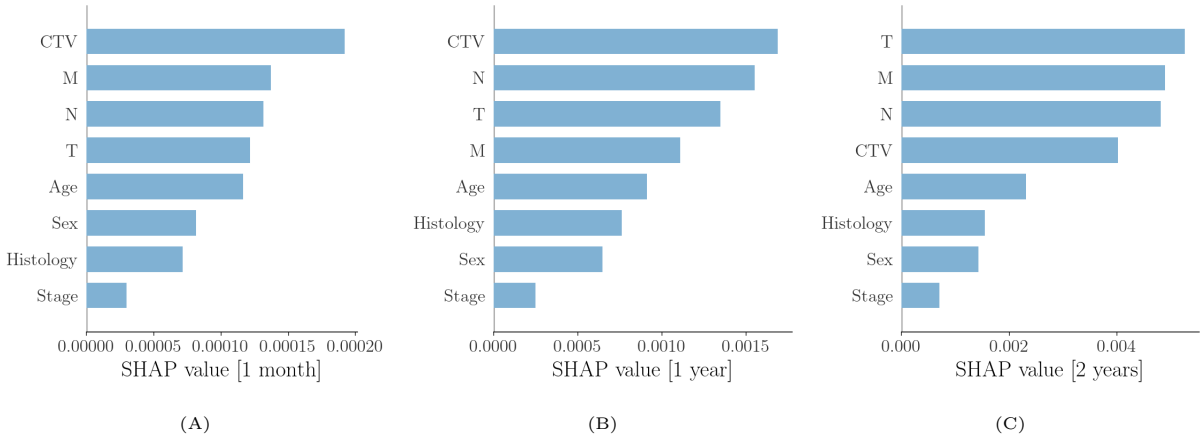

Figure 3.3: SHAP summary plots of features' contributions in the 3 models implemented with the 3 time units: Panel A) 1 month; Panel B) 1 year; Panel C) 2 years. The plots show the global feature importance by averaging the absolute SHAP values obtained for each patient and time represented in the output vector.

Furthermore, we conducted an ablation study to gain a better understanding of the individual contributions of its two terms, L_1 and L_2. We examined whether both terms played a role in the model's final performance (Figure 3.4.A) and in achieving convergence

(Figure 3.4.B). The figures depict the *Ct-index* and the number of epochs, respectively, which clearly demonstrate that utilizing both terms together positively impacted the model's performance and the convergence time.

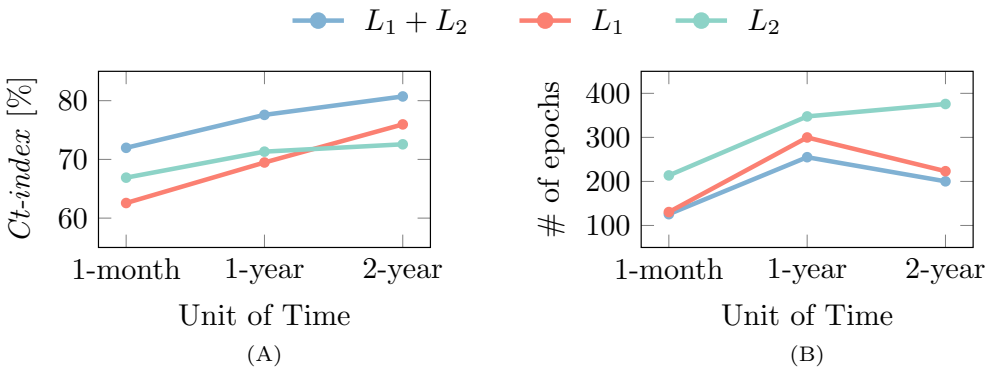

Figure 3.4: Ablation study of the two terms of the loss function proposed in [49]: (A) Average performance (*Ct-index*) and (B) mean number of epochs to achieve convergence.

Finally, considering that the model is designed for a medical application, we analyzed its computational burden and inference speed. We evaluated the former in terms of the model's weight count, approximately 2.7 million, which implies significantly longer training times compared to ML competitors, despite not requiring imputation. However, since training would occur prior to deployment, the critical factor for practical application is the inference time, approximately 5×10^{-3} seconds per sample, making it feasible for clinical practice.

3.6 Conclusion

In this manuscript, we proposed a novel approach to address the problem of missing data in the context of survival analysis of NSCLC. This task is usually tackled, in the literature, either by discarding the incomplete data samples or employing some imputation strategy, which could bias the final prediction. Conversely, our method is able to ignore those values, taking into account the available ones only. All the experiments are performed on the in-house dataset, CLARO [50], consisting of the clinical information of 297 patients suffering from NSCLC.

From a clinical perspective, the development of reliable and accurate prognosis prediction tools, to be used prior to treatment initiation, is a critical and unmet need in clinical practice. The availability of such tools would allow clinicians to tailor treatment strategies to the anticipated response, intensifying or descaling therapy as necessary based on the patient's prognosis.

Experimental results show that our model outperforms conventional OS methods at various time units with different imputation techniques, bringing an increase in performance when missing features are present.

The results described so far and the limitations of this work suggest future directions worthy of investigation. These include conducting additional experiments of our method on other OS datasets, as we are aware that validating a model on one dataset only, even if it is representative of a large population since it comes from a large Italian metropolis, is somewhat limiting, but obtaining medical datasets, preferably multicentric, is not always a straightforward process, particularly the raw version of the data in which imputation or exclusion of missing samples and features is not performed. Therefore, in future works, we plan to extend the range of datasets, artificially generating different missing value scenarios on which to test our approach. Furthermore, this work considers clinical features only, thus having a reduced view of the patient's status, which could be expanded by taking into account other sources of information and types of data. Hence, another future work could expand our approach to include various types of data beyond tabular data (e.g., imaging data). For a more seamless integration in the clinical setting, it could also be interesting to conduct further development to attempt to reduce the training time, by reducing the number of trainable weights, so as to accommodate eventual updates over time as new patient data are acquired. Moreover, it could be interesting to explore the generalization ability of the proposed approach beyond the time-to-event analysis, thus directing future works in testing it in other task domains such as classification and regression. In the meanwhile, we opted to make publicly available our GitHub repository[2] in a way to make it possible to test our approach on other datasets.

In this chapter, we investigated the prediction of OS in oncology patients using real-world clinical data. We demonstrated how DL models can provide meaningful and interpretable prognostic insights, even in the presence of naturally occurring missing values. The analysis highlighted the challenges posed by incomplete follow-up, and heterogeneity in clinical records. While these models are informative and clinically relevant, their applicability is constrained by their monomodality, only tabular data were utilized. However, modern clinical practice increasingly relies on imaging data alongside structured information. To advance towards a more holistic modeling approach, the next chapter explores the integration of radiological and clinical data through multimodal pipelines for survival prediction.

[2] https://github.com/cosbidev/OSTransformer

Chapter 4

Fusing Imaging and Clinical Data for NSCLC Survival Prediction

The growing availability of both imaging and structured clinical data offers new opportunities to develop predictive models that reflect the full complexity of medical decision-making. In this chapter, we take a first step toward multimodal modeling by combining CT scans and clinical variables to predict overall survival in patients with non-small cell lung cancer (NSCLC). We investigate whether integrating heterogeneous modalities yields tangible benefits over single-modality models and explore fusion strategies to effectively combine imaging and tabular features. This study represents a preliminary but essential attempt to bridge the gap between unimodal robustness and multimodal expressiveness in real-world clinical tasks.

4.1 Introduction

Lung cancer is the second most common type of tumour worldwide, accounting for approximately 11.4% of all cases [32], and it is the first in terms of number of deaths. Non-small-cell lung cancer (NSCLC) is the most frequent, with approximately 82% of all cases [33]. The most common treatment options, selected according to patients' characteristics, include radiotherapy, chemotherapy, surgical resection, and immunotherapy but also targeted therapy [33, 58].

Overall survival (OS), a measure of the time elapsed from the date of diagnosis until the patient's death, allows the identification of subgroups of patients with a better or worse prognosis. Nevertheless, the 5-year survival rate for NSCLC is 26%, and it drops further to 7% when local recurrence or distant metastases occur [33]; in this respect, strategies to

improve OS are urgently needed.

Over the last few years, there has been a growing interest in the development and application of Artificial Intelligence (AI) methods to oncology to help personalised medicine make further progress by facilitating the identification of the correct treatment for each patient. This has fostered the emergence of radiomics, which represents the bridge between medical imaging and personalised medicine since it computes, in a non-invasive manner, quantitative characteristics from medical images, such as CT, MRI, X-ray, and PET, representing tumour phenotype [59, 60, 61, 62]. In addition to radiomics, researchers have attempted to extract prognostic information from other modalities, e.g., genome sequencing, whole-slide images (WSI), etc. [63, 64, 65]. For example, genomics data from a tumour allow the identification of cancer driver genes, whilst a WSI from a biopsy provides insight into the morphology and microenvironment of the tumour.

Several learning methods exist to perform these prognostic tasks, which can be roughly divided into model-based and data-based approaches. The former assume a model to describe the data trend, whilst the latter, exploiting the current large availability of digital repositories and using increasingly high-performance AI algorithms, learn directly from the data. In lung cancer predictive applications, such learning methods usually exploit one modality only [66, 67, 68, 69, 70, 71], but the availability of multimodal data, which provide complementary information about the phenomenon under investigation, has led to the development of multimodal learning techniques able to cope with different information and to perform significantly better than unimodal models [72, 73, 74, 75, 76]. From an AI perspective, early, joint, and late fusion are the three main fusion techniques to merge different modalities' information. In the first technique, the features of each modality are merged according to a rule into a feature vector to be given to the learner; in the second, the different modalities are merged at hidden and embedded levels, whilst in the last technique, the predictions made using the individual modalities are aggregated according to an aggregation rule.

In NSCLC, several studies have searched for a set of quantitative biomarkers, also referred to as a signature, to predict the overall survival. Among them, Table 4.1 summarises those using multimodal approaches [77, 78, 79, 80], which are also now shortly overviewed.

In [77], the authors used the NSCLC dataset available on The Cancer Imaging Archive (TCIA) [81] to present an early fusion-like approach which fuses PET and CT images, using a technique based on 3D discrete wavelet transform to combine spatial and frequency features, and then it extracts radiomic features (first-order, textural, and moment invariant features). After performing feature selection via univariate Cox analysis, the authors applied the Kaplan–Meier method. The proposed approach obtained a concordance index (C-index) of 0.708, measured with 1000-time bootstraps, which is higher than the results they achieved

from unimodal and traditional early fusion approaches (concatenation and averaging of the feature vectors separately extracted for each modality).

In [78], the authors used another NSCLC dataset also available on TCIA [82], and they performed an early fusion of deep features extracted from CT images and clinical data. The former were extracted using a 3D-ResNet34, whilst the latter using a Multilayer Perceptron (MLP). The concatenation of these features fed an MLP. In 5-fold cross-validation with a patient-level split, the authors tested different configurations by varying the structure of the ResNet, the depth of the final MLP, and the ratio between the number of the two types of deep features, achieving a C-index equal to 0.658 as best result.

In [79], the authors developed a hierarchical multicriterion fusion strategy to combine the predictions made by various classifiers working with different modalities. Even this study is based on the same data available on TCIA [82] used by [78], and it only takes into account 316 patients in whom the gross tumour volume was delineated. This permitted to extract clinical features and radiomic features (textural and non-textural) for each patient that, after a feature selection step separately performed for the two modalities were fed into the system. The modular architecture allows each modality to be analysed separately with a set of classifiers (Support Vector Machine, k-Nearest Neighbours, Decision Tree, Random Forest, and Extreme Gradient Boosting). By means of a sequence of aggregation rules that weight the contribution of each classifier to the output probability of each modality and then combine the probabilities of each modality, the system produces the final prediction. The experiments, run in 5-fold cross-validation, return an Area Under the ROC Curve (AUC) equal to 0.81.

In [80], the authors used the data in the National Cancer Institute's Genomic Data Commons database [83] to develop a multimodal deep learning method for long-term pan-cancer survival prediction, called MultiSurv, which works with six different modalities, namely clinical data, gene expression, microRNA expression, DNA methylation, gene copy number variation data, and WSI. In this modular architecture, each input data modality is handled by a dedicated submodel. For the clinical and omics submodels, they used an MLP, whilst for the imaging submodel a ResNeXt-50. The data fusion layer aggregates the multimodal feature representations by taking the element-wise maxima across the set of representation vectors, allowing any missing modalities to be handled as well. The fusion vector is the input to an MLP, which returns as output a vector of probabilities, one for each time interval of a set of predefined follow-up time intervals. This system was trained in an end-to-end fashion, applying an holdout cross-validation stratified by cancer type. The authors evaluated the model with different numbers and combinations of the six modalities, and the best performance was obtained with bimodal inputs combining clinical data with gene expression

Author	Modalities	Study Population	Number of Patients	Data Representation	Fusion Modality	Learning Model	Performance
[77]	CT, PET	NSCLC I-IV stages	182	Radiomic features extracted from an image obtained by merging PET and CT scans through a technique based on 3D discrete wavelet transform	Early	Kaplan–Meier method	C-index: 0.708
[78]	CT, clinical data	NSCLC I-III stages	422	Concatenation of deep features extracted by a 3D-ResNet34 and an MLP for CT images and clinical data, respectively	Early	MLP	C-index: 0.658
[79]	CT, clinical data	NSCLC I-III stages	316	Clinical data and radiomic features	Late	Modular architecture with SVM, DT, KNN, RF, and XGBoost as base classifiers	AUC: 0.81
[80]	Clinical data, gene expression, microRNA expression, DNA methylation, gene copy number variation data, and WSI	33 different cancer types	11.081	Element-wise maxima across the set of representation vectors of single-modality submodels	Joint	Modular architecture, with dedicated input data modality submodels, a data fusion layer, and a final survival prediction MLP submodel	Time-dependent C-index: best 0.822 lung squamous cell carcinoma 0.554
Putting our work in the background	CT, clinical data	NSCLC II-IV stages	191	Clinical data and CT slices	Optimisation-driven late	Multimodal ensemble of learners trained on different modalities and selected by a multiobjective optimisation algorithm	ACC: 0.75

Table 4.1: Summary of the background on the multimodal learning to predict the overall survival in NSCLC. For the sake of completeness, the last section puts our contribution in the context of the literature.

(time-dependent C-index: 0.822).

Although the works in the literature achieved promising results, they are few in number, despite the importance of predicting the overall survival in NSCLC cancer that, in turn, may open the chance to develop personalised therapeutic approaches. Furthermore, two out four of such contributions explored early fusion, one investigated late fusion, and the other joint fusion. In particular, the one using late fusion computes handcrafted features from CT images that feed well-established classifiers. Nevertheless, in the last decade, deep learning

has shown its potential in several fields, medical imaging included [84, 85, 86], to automatically learn discriminative features directly from images, without being limited to using predefined features or other descriptors whose definition come from researchers' experience. In particular, Convolutional Neural Networks (CNNs) are a well-established set of network architectures exploiting convolutional layers (and their variations) to learn a compact hierarchical representation of the input that well fits the specific task to solve. In this respect, and as an evolution of the state-of-the-art shown in Table 4.1, in this work, we present a method to algorithmically optimise the way to set up a multimodal ensemble of deep networks, which are then combined by a late fusion approach. Such an ensemble uses image and clinical data to tackle the challenge to predict the overall survival in a cohort of 191 patients affected by NSCLC cancer. Exploiting the classifications of different unimodal models, we propose an optimised multimodal late fusion approach, whose performance is shown in Section 4.4. In particular, our method addresses a key and open question in multimodal deep learning [73, 87], i.e., which should be the deep networks for each modality to be combined in the ensemble among the many available.

The manuscript is organised as follows: the next section describes the materials, and Section 4.3 introduces the methods. Section 4.4 presents and discusses the experimental results; finally, Section 4.5 provides concluding remarks.

4.2 Materials

Our clinical decision support system uses image and clinical data available within the CLARO dataset, which includes 191 NSCLC patients treated with concurrent chemoradiation for locally advanced NSCLC (86% of cases) and systemic treatment in the metastatic setting (14%). During treatment, all patients underwent weekly chest Computed Tomography (CT) scans, without intravenous contrast, to assess acute toxicity and tumour shrinkage, which were reviewed by two radiation oncologists independently. For all CTs, each physician was able to judge whether reduction was: (a) present and clinically significant, (b) present and clinically non-significant, or (c) absent. In the case of physician agreement for the (a) category, a contrast-enhanced CT was performed to better visualise node reduction, a new target volume was delineated, and a new treatment plan performed. Patients were treated without any time break.

The population was enrolled under two different approvals (the retrospesctive and prospective phases) of the Ethical Committee. The former was approved on 30 October 2012 and registered at ClinicalTrials.gov on 12 July 2018 with Identifier NCT03583723, whilst the latter was approved on 16 April 2019 with Identifier 16/19 OSS, and it was closed on April

2022. The Institutional Review Board approved this review. Written informed consent was obtained in all patients. The authors confirm that all ongoing and related trials for this intervention are registered.

The median OS for the entire population was 15.64 months, with a mean of 23.85 ± 77.22 (95% CI). The patients were then clinically followed until they were divided into two classes based on the median OS of all the patients: 95 dead and 96 alive.

4.2.1 Imaging

The characteristics investigated were extracted from CT scans collected at the time of patient diagnosis, on which expert radiation oncologists delineated the Clinical Target Volume (CTV).

For each patient, the CT images were acquired before the treatment using a Siemens Somatom Emotion, with 140 Kv, 80 mAs, and 3 mm for slice thickness. The scans were preprocessed applying a lung filter (kernel B70) and a mediastinum filter (kernel B31).

4.2.2 Clinical Features

Clinical data contained different information, which are listed in Table 4.2 together with the number of missing values and the distribution for each tabular feature among the different discrete values. To define the stage of the tumour, two experienced radiation oncologists (ROs) independently reviewed CT scans and assigned the staging scores of the tumour (T, N, and tumour stage); in case of disagreement, they reviewed the CT images together until consensus was reached. In addition to staging, age, and sex, Table 4.2 shows that we also collected features describing the histology of the tumour and the initial CTV, so that the clinical data account for seven descriptors in total.

In the imputation of missing values, the median value and the mode of the training set data were assigned for the numerical and categorical features, respectively. Furthermore, it should be noted that not all patients underwent a histopathological examination. Nevertheless, since on the one side it was not possible to impute the histology of the tumour and, on the other side, this feature could be informative, we add a virtual category named *unknown*.

4.3 Methods

To predict the prognosis in terms of binary classification task over the OS, we exploited both the images and clinical data described before, which were processed by a multimodal DL pipeline that, in the training phase, finds the optimal combination of models of different

Feature	Missing Data	Categories	Distribution
Age *	26 (13.62%)	<71 years	82 (42.93%)
		≥71 years	83 (43.46%)
CTV *	37 (19.37%)	<114.88 cm^3	77 (40.31%)
		≥114.88 cm^3	77 (40.31%)
Sex	0 (0.00%)	Male	133 (69.63%)
		Female	58 (30.37%)
Histology	0 (0.00%)	Adenocarcinoma	95 (49.74%)
		Squamous	59 (30.89%)
		Other	11 (5.76%)
		Unknown	26 (13.61%)
Stage	0 (0.00%)	II	4 (2.09%)
		III	160 (83.77%)
		IV	27 (14.14%)
T stage	36 (18.85%)	T0	1 (0.52%)
		T1	9 (4.71%)
		T2	32 (16.75%)
		T3	65 (34.03%)
		T4	48 (25.13%)
N stage	26 (13.61%)	N0	15 (7.85%)
		N1	33 (17.28%)
		N2	93 (48.69%)
		recurrence N2	6 (3.14%)
		N3	18 (9.42%)

Table 4.2: Patients' characteristics. As marked by *, note that, although *age* and *CTV* are continuous variables, for the sake of synthesis we report here their distribution considering their median values as thresholds, whilst the model used the continuous values. The division into stages is further defined by letters (a, b, and c), which are not reported for the sake of brevity, but the model uses the actual stages.

modalities via multiobjective optimisation. The idea stems from observing that today many deep neural networks are available, both in terms of architectures as well as of pretrained weights. This allows researchers to train or fine-tune them to search for the most suitable for the task at hand. Furthermore, it is well-known that, in many cases, ensembles of classifiers combined in late fusion provide better performance than unimodal models [88], but, at the same time, the learners in the ensemble have to complement each other, i.e., they have to make wrong decisions on different samples. Therefore, the abundance of available models asks for methods to support researchers in determining which is the best multimodal ensemble, a challenge that we address using an algorithmic and multimodal approach, schematically represented in Figure 4.1. It works with m different modalities and M different models, so that M_m is the number of models available for the mth modality. Furthermore, we denote with E an ensemble built using one or more models per modality, whose outputs are

combined by majority voting. Figure 4.1 shows that our method essentially consists of three main steps:

- Training all the available models for every single modality using the training sets defined by the bootstrap validation approach;
- Finding the multimodal set of unimodal models solving a multiobjective optimisation problem working with evaluation and diversity scores, which are computed on the validation sets defined by the same bootstrap approach;
- Computing the performance on the test sets defined by bootstrap, which are then averaged out (block "Average performance evaluation").

These steps are now detailed in the next subsections.

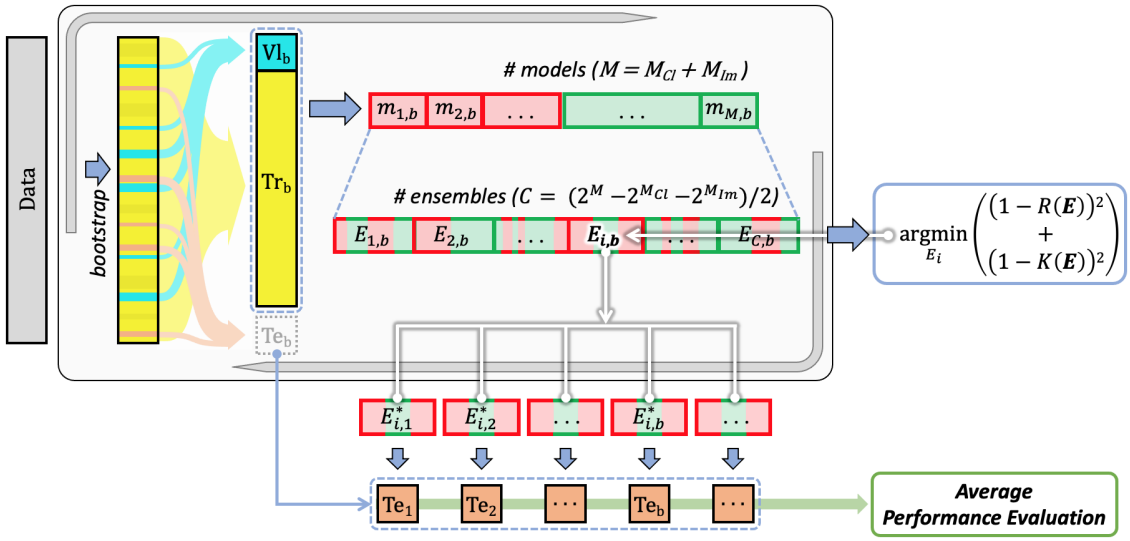

Figure 4.1: Schematic view of the pipeline. Symbols: Tr: training set, Vl: validation set, Te: test set, m: model, M: number of models ($\cdot_{Cl}$: for clinical data and $\cdot_{Im}$: for data from images), $\cdot_b$: a generic bootstrap fold, E: (models') ensemble, $\cdot_i$: a generic ensemble, C: number of ensembles, R: function of recall, K: function of diversity.

4.3.1 Training

To obtain the optimal ensemble E^* of models, the first step is to independently train and evaluate the different M unimodal models on the respective m modalities. In our scenario, we had $M = M_{Cl} + M_{Im}$, where M_{Cl} and M_{Im} denote the number of models for the clinical data and the imaging modality, respectively.

With respect to the clinical data, we worked with $M_{Cl} = 7$ different ML and DL models, which are acknowledged in the literature as those that best work with this modality [89]. In alphabetical order they are:

- AdaBoost as a cascade of classifiers;
- Decision Tree (DT) as tree model;
- Multilayer perceptron (MLP) as neural architecture with one hidden layer with 13 neurons and 1 neuron in the output layer, which use the ReLU and Sigmoid activation functions, respectively;
- Random forest (RF) as an ensemble of trees;
- Support Vector Machine (SVM) as a kernel machine;
- TABNET [5] as a neural architecture;
- XGBoost a variation of the AdaBoost that uses a gradient descent procedure to minimise the loss when adding weak learners.

Let us now turn the attention to see image modality. We worked with $M_{Im} = 30$ different CNNs from 8 architecture families, which have proved to have promising results in many biomedical applications [90]. They are:

- AlexNet [91];
- VGG [92]: VGG11, VGG11-BN, VGG13, VGG13-BN, VGG16, VGG16-BN, VGG19, VGG19-BN, where the suffix BN means that batch normalization is used;
- ResNet [93]: ResNet18, ResNet34, ResNet50, ResNet101, ResNet152, ResNeXt50, ResNeXt101, Wide-ResNet50-2, Wide-ResNet101-2;
- DenseNet [94]: DenseNet121, DenseNet169, DenseNet161, DenseNet201;
- GoogLeNet [95];
- ShuffleNet [96]: ShuffleNet-v2-x0-5, ShuffleNet-v2-x1-0, ShuffleNet-v2-x1-5, ShuffleNet-v2-x2-0;
- MobileNetV2 [97];
- MNasNet [98]: MNasNet0-5, MNasNet1-0.

All the CNNs were pretrained on the ImageNet dataset [99]. The architectures, layer organisation, and complexity of such models gave us the opportunity to investigate how different models perform on the task at hand.

4.3.2 Optimisation

To answer the question of which architectures should be used to construct the best multimodal ensemble, we solved a multiobjective optimisation problem that works with two scores capturing different views of the ensemble performance. Indeed, given an ensemble E, on one side we measured its recall (R) using, straightforwardly, the labels computed by applying the aforementioned majority voting scheme. R is defined as

$$R = \frac{TP}{P} \tag{4.1}$$

where TP is the number of true positive classifications and P is the number of positive instances, and it measures the sensitivity of the model, a desirable property in our application ensuring that no positive patients get excluded before treatment. On the other side, the optimisation algorithm also works with the kappa diversity (K), a pairwise score measuring to what extent two models provide the same errors. It is defined as

$$K = 1 - \frac{2(N_{11}N_{00} - N_{01}N_{10})}{(N_{11} + N_{10})(N_{01} + N_{00}) + (N_{11} + N_{01})(N_{10} + N_{00})} \tag{4.2}$$

where N_{11} and N_{00} are the number of instances classified correctly and incorrectly by each of the two models under consideration, respectively, and N_{10} and N_{01} are the number of instances classified correctly by the first model and incorrectly by the second and vice versa, respectively. The overall ensemble diversity is given by

$$\frac{2}{|E|(|E| - 1)} \sum_{i=1}^{|E|-1} \sum_{j=i+1}^{|E|} k \tag{4.3}$$

where $|E|$ is the number of models in E. Given these premises, let us notice that both R and K range in $[0, 1]$, and the higher the values, the more accurate and diverse the models. Hence, our algorithm solves the following multiobjective problem to determine the best ensemble E^*:

$$E^* = \arg\min_E [(1 - R(E))^2 + (1 - K(E))^2] \tag{4.4}$$

s.t.

$$\begin{cases} |E^*| > 1 \\ |E^*| mod 2 = 1 \\ |E^*|_{Cl} \geq 1 \\ |E^*|_{Im} \geq 1 \end{cases} \tag{4.5}$$

where $R(E)$, $K(E)$ represent the average values of R and K, respectively, of an ensemble E computed across all the validation sets given by bootstrap, a choice that avoids any bias. Looking at the constraints, $|E^*|$ denotes the number of models in E^*, whilst $|E^*|_{Cl}$ and $|E^*|_{Im}$ stand for the number of models in E^* working with clinical and imaging data, respectively; finally, *mod* is the modulo operation. The first two conditions imply that the number of models in E^* is odd to prevent ties in the majority voting, whilst the third and fourth conditions ensure that at least one model for each modality is present in E^*. Note also that finding E^* is equivalent to finding the Pareto optimum of this optimisation problem, as we showed in [100, 101, 102]; nevertheless, here, we are extending our previous unimodal approach [100] to multimodal learning, as guaranteed by the last two conditions in Equation (4.5).

Hence, this optimisation algorithm performs an exhaustive search for the ensemble E^* that, among the $C = \frac{2^M - 2^{M_{Cl}} - 2^{M_{Im}}}{2}$ combinations of learners, returns the best classification performance and reduces the incidence and effect of coincident errors among its members, thus considering possible relationships between models and modalities. Furthermore, the simple minimisation of only one of the objective functions (R or K) is not the best approach, since some models may degrade the performance of the ensemble, and they may have redundant classifications between each other, not exploiting the trade-off between performance and diversity [103].

Finally, in the test phase, each input instance is given to all the learners in E^*, whose outputs are combined by majority voting to obtain the final prediction.

4.3.3 Preprocessing

Before feeding the data to the models, a preprocessing phase was executed for both modalities.

With reference to the clinical data, we applied one hot encoding to categorical features, so that the original 7 features were mapped to 27 descriptors, which in practice were used as input to all the classifiers mentioned before for the clinical data. Furthermore, numerical features were normalised in $[0, 1]$. No data augmentation was applied to the clinical data. For all clinical models listed in section 4.3.1, the default parameters of the libraries were

used.

With reference to the imaging modality, we used a U-Net to automatically align the images by detecting the region of interest of the scans by including the bounding cuboid segmenting the lungs. The U-Net architecture has proved to obtain good performance in many biomedical applications [104]. We trained this network on the TCIA publicly available dataset [82], which comprises 422 patients, and on a subset of our dataset, 125 patients whose lungs had already been delineated, with the goal of segmenting the lung pixels of each 2D slice. From this segmentation, we extracted the minimum bounding cuboid of the segmented volume, preventing any deformation once re-scaled . As input, the U-Net received 224 Œ 224 images, and it was trained with an Adam optimiser and with a Dice loss function. The batch size was set to 32, and the number of epochs was equal to 50, but an early stop criterion was triggered at 13th epoch. We assessed the performance of this network in holdout cross-validation, obtaining a Dice score and an intersection over union equal to 98.5 and 97.0, respectively, which we considered satisfactory for our task.

Let us now focus on the image classification stage. All the CNNs work with 2D images, i.e., at CT slice level, and they need an input of size 224x224. To this end, each slice of the segmented lungs was resized to 224x224 and normalised with a min–max scaler, bringing the pixel values between 0 and 1. Random data augmentation was applied to prevent overfitting of the CNNs: horizontal or vertical shift ($-22 \leq$ pixels ≤ 22), random zoom ($0.9 \leq$ factor ≤ 1.1), vertical flip, random rotation ($-15° \leq$ angle $\leq 15°$), and elastic transform ($20 \leq \alpha \leq 40$, $\sigma = 7$). The cross-entropy loss was used and was regulated by an Adam optimiser with an initial learning rate of 0.001, which is scheduled to reduce by an order of magnitude every time the minimum validation loss does not change for 10 consecutive epochs. For all the nets, a maximum of 300 epochs was fixed, with an early stopping of 25 epochs following the validation loss.

Given the fact that the CNNs work at slice level and the clinical data at patient level, to uniform the classifications, we aggregated the predictions of the slices of each patient via a majority-voting rule, thus obtaining a final outcome for each modality at patient level.

All the training processes were executed using an NVIDIA TESLA V100 GPU with 16 GB of memory, using PyTorch and Scikit-learn as the main coding library.

4.4 Results and Discussion

All the experiments were performed in bootstrap, performing five random extractions of the samples, where in each fold the proportions between the training, validation, and testing sets are 80%-10%-10%, respectively. Straightforwardly, in the imaging modality, all the slices

coming from the same patient were always in the same set.

Table 4.3 shows the results: each row corresponds to a classifier in the case of unimodal learners reported in the uppermost section; it corresponds to a multimodal ensemble in the middle section, and it corresponds to a competitor in the bottom most section. The columns report the performance measured in terms of accuracy, F-score, and recall to have a complete view of how the different models perform on the test sets. With reference to the unimodal learners, the values in Table 4.3 show that the best classifier working with clinical data is the AdaBoost, whilst in the case of image data, the best CNN is the VGG11-BN. Both achieve the largest accuracy and F-score among the pool of unimodal models, whilst the latter is also the best in terms of recall.

The ensemble returned by our algorithm, denoted by E^* in the table, achieves larger performance in terms of accuracy, F-score, and recall with respect to the unimodal classifiers. Whilst this could be expected in the case of the recall, as it is built maximising a function including this metric, it is interesting to note that this happens also in the case of the accuracy and F-score.

It is worth noting that the Pareto optimum E^* is composed of three models (two from the imaging modality and one from the clinical modality): ResNet34, VGG11-BN, and TAB-NET, which belong to different families, suggesting that each model interprets its modality in a different way to address the classification task. We notice that E^* has better performance (for all metrics) than the two best unimodal models. This finding implies that it is useful to fuse different modalities, each carrying useful and distinct information for the prognosis task whilst, at the same time, it is important to consider the diversity also, since it offers complementary points of view to the ensemble.

To assess the optimisation function, we also investigated which are the performances of the ensembles maximising only R or K, denoted as E^R and E^K, respectively (middle section of Table 4.3). The former consists of DT, RF, and AlexNet, whereas the latter comprises DT, RF, and DenseNet161. Moreover, in this case, the class predictions on the test set revealed that the outputs provided by E^* are better than those returned by E^R and E^K. This finding supports the importance of satisfying the proposed multiobjective optimisation condition. This agrees with the literature that in other fields, and in the case of the majority voting rule, reports that a necessary and sufficient condition for an ensemble to be more accurate than any of its models is if the models are accurate and diverse [105].

To further prove the efficacy of the proposed approach, the middle section of Table 4.3 presents the performance of the following other experiments:

- $\overline{E}$: it denotes the average performance for all the possible ensembles;

Classifier	Modality	Accuracy	F-Score	Recall
AdaBoost	Clinical	**65.00 ± 5.00**	**67.35 ± 6.53**	**74.00 ± 16.73**
DT	Clinical	60.00 ± 3.54	59.42 ± 9.15	62.00 ± 20.49
MLP	Clinical	61.00 ± 5.48	54.37 ± 23.57	60.00 ± 38.08
RF	Clinical	60.00 ± 6.12	60.72 ± 9.74	64.00 ± 16.73
SVM	Clinical	59.00 ± 2.24	55.46 ± 10.29	54.00 ± 18.17
TABNET	Clinical	63.00 ± 10.37	64.68 ± 11.69	70.00 ± 22.36
XGBoost	Clinical	54.00 ± 8.22	49.67 ± 16.74	50.00 ± 24.49
AlexNet	Imaging	50.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
DenseNet121	Imaging	62.00 ± 19.24	59.97 ± 27.79	66.00 ± 35.07
DenseNet161	Imaging	69.00 ± 6.52	68.28 ± 8.88	70.00 ± 20.00
DenseNet169	Imaging	71.00 ± 17.82	72.28 ± 17.44	76.00 ± 20.74
DenseNet201	Imaging	63.00 ± 16.05	65.95 ± 16.37	74.00 ± 23.02
GoogLeNet	Imaging	60.00 ± 6.12	50.04 ± 19.69	48.00 ± 31.14
MNasNet0-5	Imaging	51.00 ± 13.42	45.65 ± 19.37	44.00 ± 23.02
MNasNet1-0	Imaging	62.00 ± 7.58	65.11 ± 9.94	74.00 ± 20.74
MobileNetV2	Imaging	67.00 ± 17.18	68.61 ± 17.17	74.00 ± 23.02
ResNet101	Imaging	51.00 ± 5.48	49.97 ± 20.44	60.00 ± 38.08
ResNet152	Imaging	71.00 ± 7.42	63.65 ± 19.16	60.00 ± 30.82
ResNet18	Imaging	64.00 ± 18.84	58.74 ± 29.30	60.00 ± 33.91
ResNet34	Imaging	70.00 ± 11.73	71.71 ± 10.51	78.00 ± 22.80
ResNet50	Imaging	69.00 ± 11.40	69.45 ± 17.58	78.00 ± 27.75
ResNeXt101	Imaging	69.00 ± 7.42	68.95 ± 8.46	70.00 ± 15.81
ResNeXt50	Imaging	63.00 ± 10.37	64.35 ± 19.82	78.00 ± 33.47
ShuffleNet-v2-x0-5	Imaging	74.00 ± 10.25	74.66 ± 11.07	78.00 ± 16.43
ShuffleNet-v2-x1-0	Imaging	67.00 ± 17.18	67.14 ± 20.74	72.00 ± 26.83
ShuffleNet-v2-x1-5	Imaging	74.00 ± 13.87	72.3 ± 19.53	74.00 ± 27.02
ShuffleNet-v2-x2-0	Imaging	73.00 ± 9.08	71.23 ± 11.99	70.00 ± 20.00
VGG11	Imaging	50.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
VGG11-BN	Imaging	**74.00 ± 16.36**	**75.03 ± 16.37**	**78.00 ± 19.24**
VGG13	Imaging	50.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
VGG13-BN	Imaging	64.00 ± 8.22	61.58 ± 25.24	72.00 ± 35.64
VGG16	Imaging	50.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
VGG16-BN	Imaging	71.00 ± 13.42	72.19 ± 10.95	74.00 ± 13.42
VGG19	Imaging	50.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
VGG19-BN	Imaging	59.00 ± 15.17	51.68 ± 32.01	58.00 ± 38.99
Wide-ResNet101-2	Imaging	68.00 ± 10.95	69.84 ± 9.55	76.00 ± 20.74
Wide-ResNet50-2	Imaging	64.00 ± 13.87	66.02 ± 12.41	70.00 ± 18.71
$\boldsymbol{E^*}$	Multimodal	**75.00 ± 16.20**	**77.70 ± 13.83**	**84.00 ± 15.17**
E^R	Multimodal	60.00 ± 6.12	58.15 ± 9.40	58.00 ± 17.89
E^K	Multimodal	61.00 ± 5.48	62.02 ± 9.58	66.00 ± 16.73
$\overline{E}$	Multimodal	66.58 ± 11.30	61.44 ± 15.13	62.35 ± 22.00
E^3_{post}	Multimodal	72.00 ± 12.04	75.41 ± 10.68	83.00 ± 15.17
$\overline{E^{2+*}_{post}}$	Multimodal	70.94 ± 10.90	71.79 ± 10.21	74.91 ± 13.86
$E^{\not\downarrow}$	Multimodal	61.00 ± 2.24	61.09 ± 8.11	64.00 ± 18.17
DeepMMSA [78]	Multimodal	59.00 ± 6.52	58.07 ± 12.32	52.00 ± 32.71
MCF [79]	Multimodal	62.00 ± 2.74	61.04 ± 10.53	64.00 ± 23.02

Table 4.3: Performance of all the tested models with the best for each modality reported in bold. Each column shows the mean value of a performance metric followed by the standard deviation. E^* is our optimum ensemble; E^R and E^K are the ensembles which maximise R and K, respectively; $\overline{E}$ is the average performance for all the possible ensembles; E^3_{post} is the ensemble consisting of the unimodal models with the largest recall, $\overline{E^{2+*}_{post}}$ is the ensemble with the two unimodal classifiers with the largest recall per modality whilst varying the remaining experts included in the ensemble; $E^{\not\downarrow}$ is the ensemble obtained relaxing the multimodality constraints.

- E^3_{post}: it denotes the performance of the ensemble consisting of the unimodal models with the largest recall, i.e., AdaBoost, ResNet34, and VGG11-BN. In this case, we

adopt the subscript *post* to specify that such three models were *a posteriori* selected, i.e., they provide the largest performance on the test set, and not on the validation set;

- $\overline{E^{2+*}_{post}}$: it denotes the average performance attained by all the possible ensembles, including the two unimodal classifiers with the largest *a posteriori* recall, i.e., Adaboost and VGG11-BN, whilst varying the remaining experts included in the ensemble;
- $E^{\not\phi}$: it denotes the performance of the ensemble obtained relaxing the multimodality constraints, and it is composed of AdaBoost, DT and RF.

It is worth noting that the performances returned by such four experiments are always lower than the performance of E^* and even lower than several unimodal learners. This confirms, again, that maximising recall and diversity together is a useful driver to guide the ensemble set-up, i.e., to select which are the unimodal learners to be included. Furthermore, the fact that such ensembles in some cases provide lower performance than some unimodal learners confirms that handcrafted ensemble definitions can lead to sub-optimal results.

The last section of Table 4.3 presents a direct comparison of our approach with two state-of-the-art studies [78, 79], which are denoted as DeepMMSA and MCF. As described in section 4.1, they work with clinical and imaging modalities so that we can apply them to our data, computing the same scores we used for the other architectures under consideration. Note also that we do not experimentally evaluate [77, 80] because, on the one side, [77] works with CT and PET images and it is not designed to handle clinical data, whereas on the other side, [80] does not work with CT images. The results show that such competitors perform worse than our method; we deem that this happens because such papers manually define the composition of the multimodal architectures, whilst our solution relies on an optimisation process.

As a further issue in our discussion, let us recall that three types of fusion exist in multimodal learning: early, joint, and late fusion. The latter is the one we used in this work, whilst the other two are other possible ways to proceed, which we considered as possible competitors for our method. To this end, we set up an early fusion learner using the best model per modality, i.e., AdaBoost and VGG11-BN, as already mentioned. Furthermore, we used the VGG11-BN as a feature extractor from the CT images, which we then concatenated with the clinical features and feed to the AdaBoost. We tested both slice-level and patient-level early fusion. The former consists of repeating the clinical features for each individual patient slice, whilst in the latter we averaged the CNN output of each slice to obtain a single feature vector per patient. These approaches got an accuracy equal to 62.92±8.31% and 70.00±9.35%, and an F-score equal to 62.69±6.41% and 69.83±12.25%, respectively, for the

slice- and patient-level fusion, which are lower than the proposed approach. We did not perform any joint fusion since the best unimodal model (AdaBoost) has larger performance than a fully connected network (MLP). This, in turn, makes it not possible to apply joint fusion between the adaptive boosted ensembles and the VGG11-BN, although this is an issue worthy of investigation in a future work.

4.5 Conclusions

In this manuscript, we proposed a multimodal method for survival analysis of NSCLC. NSCLC has been already studied in a few other works employing multimodal learning but, differently from the literature, we propose an algorithm able to identify the optimal set of classifiers to be added to the multimodal ensemble in a late fusion approach. Our study is based on two modalities, clinical and CT imaging data, of a cohort of 191 patients suffering from locally advanced non-small-cell lung cancer.

From a clinical point of view, the possibility of having prognosis prediction tools in addition to clinical data, and especially before starting treatment, represents an unmet need of particular interest. If this data are available at the start of therapy, the treatment itself could be modified, adapting it to the expected response, thus intensifying or descaling therapy in patients with poor or good prognosis, respectively.

Indeed, we presented an optimised late fusion ensemble search method that finds the optimal combination of multimodal models considering both a metric of performance and a diversity score. Experimental results show that our method outperforms conventional unimodal models, bringing significant increase in performance in the multimodal ensemble. Among the different combinations of classification algorithms, the proposed approach achieves an accuracy of 75.00%, an F-score of 77.70%, and a recall of 84.00%, achieved using a ResNet34 and a VGG11-BN for the imaging modality and a TABNET for the clinical modality. A limitation of our approach is the need to train all models before the optimal set can be selected, which certainly represents a high computational cost.

The results described so far suggest four future directions worthy of investigation:

- Retrieving data at 1-, 2-, and 3-year time points as well as the progression free survival, which would add useful information;
- Provide more complementary information by adding other modalities to improve performance, such as WSI, genome sequencing, etc.;
- Perform different multimodality fusion approaches, such as joint fusion to obtain a

end-to-end trainable system able to exploit the inherent correlations between multiple modalities;

- Search for an approach that a priori selects the models to be included in the ensemble, without the need to train them all individually;
- Switch from a classification to a regression task, which will allow predicting the actual survival time, also integrating the "Input doubling method" [106] as a preprocessing tool to augment the training set size.

In this chapter, we explored the feasibility and benefits of combining imaging and clinical data through multimodal learning strategies for survival prediction in NSCLC patients. The experimental results confirmed that multimodal models can outperform unimodal baselines, especially when the complementary information between modalities is effectively exploited. However, the proposed architectures assumed the availability of all modalities at both training and inference time, a condition that is rarely satisfied in real-world clinical scenarios. Missing data, including entire missing modalities, remains a major obstacle to the deployment of multimodal AI systems in healthcare. To address this challenge, the next chapter introduces MARIA, a novel transformer-based architecture specifically designed to enable robust multimodal fusion even in the presence of partially missing information.

Chapter 5

MARIA: a Multimodal Transformer Resilient to Missing Modalities

The deployment of multimodal models in clinical practice requires architectures that are not only accurate but also resilient to missing modalities. As highlighted in our systematic review of multimodal learning in healthcare [107], current strategies often struggle with real-world constraints such as partial observability and data heterogeneity. Inspired by the limitations observed in both literature and previous experiments, this chapter presents MARIA (Multimodal Attention Resilient to Incomplete datA), a novel architecture specifically designed to address incomplete multimodal data scenarios. MARIA leverages intermediate fusion and a masked self-attention mechanism to selectively integrate information from available modalities while ignoring missing components during training and inference. We evaluate its performance across diverse settings, including "missing modality" and "all missing" scenarios, and benchmark it against state-of-the-art strategies for both data imputation and modality fusion. This work establishes a robust foundation for multimodal learning under the imperfect conditions typical of real-world clinical applications.

5.1 Introduction

In recent years, multimodal learning has emerged as a powerful approach for leveraging diverse data sources to achieve a comprehensive understanding of complex systems [108]. This is particularly relevant in domains such as healthcare, where integrating multiple data modalities, such as clinical assessments, imaging, laboratory tests, and patient histories, can significantly enhance diagnostic accuracy and treatment outcomes [109]. The human experience itself exemplifies multimodality, as it relies on diverse sensory inputs to form a

unified perception of the environment. Similarly, deep learning (DL) models have been developed to synthesize and analyze disparate data sources, thereby enhancing their predictive capabilities and enabling more informed decision-making in multifaceted domains such as healthcare [110, 111].

Despite the promise of multimodal learning, integrating multiple data sources presents unique challenges, with data incompleteness being one of the most significant [112]. Missing data is a common feature of real-world datasets, arising from issues such as sensor failures, patient non-compliance, technical limitations during data collection, or privacy restrictions [8]. Whether the missing information relates to features within a modality or the complete absence of a modality, such gaps can severely degrade the performance of machine learning models unless effectively addressed [113]. Thus, the development of multimodal learning models resilient to incomplete data is critical to ensuring reliability and robustness, especially in critical fields like healthcare [114].

To address these challenges, multimodal fusion strategies such as early fusion, late fusion, and intermediate fusion have been extensively studied [108, 111]. Early fusion, which combines features at the raw data level into a unified representation, is straightforward but highly susceptible to the effects of missing data, as it requires the availability of all feature vectors. Late fusion, which merges outputs from independently trained models, offers flexibility when modalities are missing but often fails to capture the intricate interactions across modalities. Intermediate fusion strikes a balance by integrating modality-specific features after initial processing. This forms a shared representation that enhances the ability to capture cross-modal dependencies, ultimately improving performance [107]. Therefore, especially in healthcare, it is essential to develop methods that leverage the potential of intermediate fusion while maintaining robustness to missing data [115].

The MARIA (Multimodal Attention Resilient to Incomplete datA) model introduced in this work is designed to address the challenges of incomplete multimodal data. It builds upon our previous work, NAIM [116], by extending its masked attention mechanism to a multimodal scenario, where each modality is processed through a dedicated NAIM module and subsequently integrated via a shared transformer encoder. By employing an intermediate fusion strategy, MARIA combines modality-specific NAIM-based encoders with a shared attention-based encoder, enabling the model to effectively handle both missing features within each modality and entirely missing modalities without relying on data imputation. Unlike traditional methods that rely on data imputation to fill in missing entries, MARIA focuses exclusively on the available features, utilizing a modified masked self-attention mechanism to process observed information only, while completely masking out the contribution of missing information. This approach enhances both robustness and accuracy, while reducing

biases typically introduced by imputation techniques.

To summarize, the main contributions of this work are as follows:

- We propose MARIA, a novel multimodal transformer architecture specifically designed to handle incomplete tabular data without relying on imputation techniques.
- MARIA employs an intermediate fusion strategy with masked self-attention mechanisms at both the modality-specific and shared encoder levels, allowing the model to dynamically ignore missing features and modalities.
- We introduce regularization strategies tailored for missing data, including stochastic masking at both the feature and modality levels during training, to enhance generalization under incomplete conditions.
- We conduct an extensive benchmark comparison with 32 competing configurations across 8 diagnostic and prognostic tasks using two real-world clinical datasets, demonstrating MARIA's robustness and superior performance under various missingness scenarios.

The manuscript is organized as follows: Section 5.2 reviews related work on multimodal learning and data handling methods; Section 5.3 introduces the MARIA model and its architecture; Section 5.4 explains the experimental setup and evaluation methodology; Section 5.5 presents the obtained results, comparing MARIA with other models under various missing data conditions; Section 5.6 summarizes the key findings and suggests directions for future research.

5.2 State of the Art

Multimodal learning combines information from diverse data sources to achieve a more comprehensive understanding of complex systems. This mirrors the inherently multimodal nature of human perception: we rely on multiple senses, e.g., sight, sound, and touch, to develop a complete understanding of our environment. Similarly, DL models must be designed to integrate diverse data sources to better comprehend intricate systems. This is particularly relevant in healthcare, where clinicians utilize multimodal data, including patient histories, imaging, laboratory results, and physical examinations, to make informed decisions. By effectively integrating such diverse information, multimodal learning models can enhance decision-making and predictive accuracy, leading to improved diagnostic outcomes and more effective treatment plans [115, 117, 118, 119, 120].

However, one of the primary challenges in multimodal learning is handling missing data, which frequently arises due to factors such as sensor failures, survey non-responses, or technical issues during data collection [121]. Effectively managing missing data, whether it involves incomplete features within a modality or entirely absent modalities, is critical to ensuring the reliability and robustness of multimodal models.

Multimodal fusion techniques play a vital role in successfully integrating diverse data sources. These techniques are typically categorized into three main strategies: early fusion, late fusion, and intermediate fusion (Figure 5.1). Each approach has distinct characteristics, making it suitable for different scenarios [115, 122].

Early fusion integrates features at the raw data level, combining them into a unified representation before any significant processing. Late fusion merges outputs from independently trained models at the decision level, offering flexibility when dealing with missing modalities. Intermediate fusion takes a balanced approach, integrating modality-specific features after initial processing to create a shared representation. Each of these fusion strategies has specific advantages and limitations in terms of performance, computational complexity, and their ability to manage missing data. This is particularly significant in healthcare, where data quality and completeness often vary. The subsequent sections provide a detailed analysis of these fusion techniques, focusing on their applications and limitations in healthcare scenarios.

5.2.1 Early Fusion

Early fusion, as illustrated in Figure 5.1.a, involves integrating multiple data modalities at the feature level. In this approach, the raw features X_i from each modality are concatenated ($\oplus$ in the figure) to form a single feature vector, which is then fed into the learning model. This method facilitates the early combination of information from all available modalities, making it particularly advantageous when different data sources are highly complementary. Early fusion is conceptually straightforward and often enables the learning model to effectively exploit cross-modal correlations [115, 122].

However, early fusion presents several challenges, especially when dealing with incomplete data. Because this approach relies on the availability of all feature vectors, missing data from even a single modality can significantly degrade model performance. Imputation is a common strategy for addressing these gaps, but it introduces potential risks such as bias and information loss [123]. Additionally, early fusion typically requires extensive preprocessing to harmonize features from different modalities, which often vary in scale and distribution.

In healthcare, early fusion can be particularly beneficial when modalities are guaran-

teed to be complete or when missing data is minimal and can be addressed through robust preprocessing techniques. However, given the variability and incompleteness commonly encountered in real-world medical datasets, early fusion may struggle to perform effectively without sophisticated data-handling strategies [115, 122].

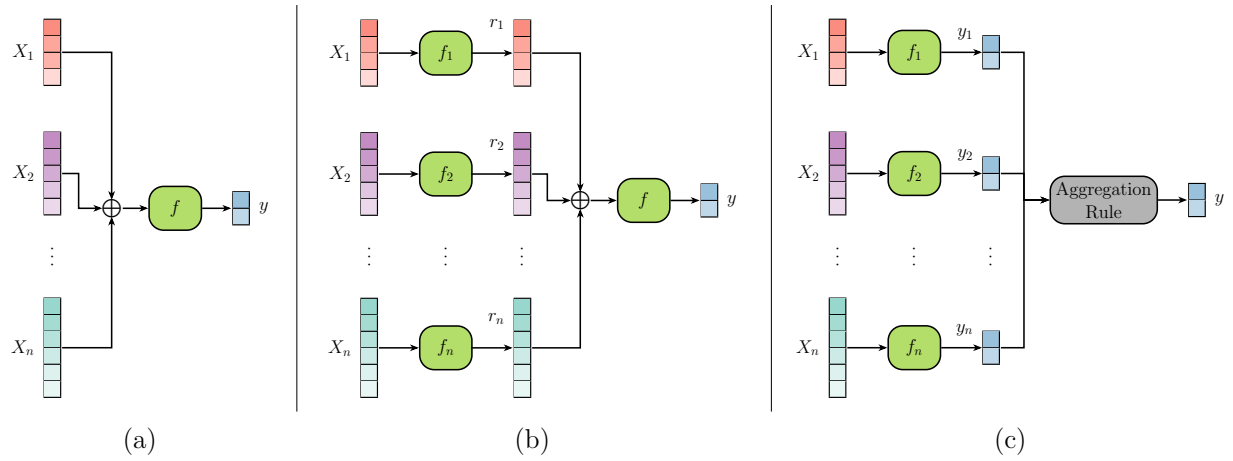

Figure 5.1: Overview of multimodal fusion strategies in DL: (a) Early fusion; (b) Intermediate fusion; (c) Late fusion. $X_1, \ldots, X_n$ represent the modalities; $f_{(i)}$ is a generic function representing a module or an entire model; $r_1, \ldots, r_n$ stand for the latent representations of the modalities; $y_{(i)}$ indicates an output of a model; $\oplus$ represent a fusion, e.g. concatenation or average, of the inputs.

5.2.2 Late Fusion

Late fusion, in contrast to early fusion, integrates modalities at the decision level, as illustrated in Figure 5.1.c. In this approach, separate models f_i are trained for each modality X_i, and their predictions y_i are subsequently aggregated using a predefined rule to generate the final output y. This method is particularly advantageous when the modalities differ significantly in data type or exhibit varying levels of reliability. By training separate models, late fusion allows each modality to be optimally utilized before combining their outputs.

One of the primary advantages of late fusion is its flexibility in handling missing modalities. Since each model operates independently, missing data from one modality does not prevent predictions from being made using the available modalities. However, late fusion has a notable limitation: it fails to fully exploit cross-modal interactions. Because modalities are only integrated at the decision level using a static, predefined rule, rather than a dynamically learned process, potentially rich correlations between features from different data sources may remain untapped. This drawback makes late fusion less suitable for tasks that require deep integration of modality-specific features, particularly in scenarios demanding

high levels of diagnostic precision.

5.2.3 Intermediate Fusion

Intermediate fusion, also known as joint or hybrid fusion, strikes a balance between early and late fusion. This approach combines modality-specific features at an intermediate stage of the learning process (Figure 5.1.b), typically after each modality has undergone initial, independent processing. In this setup, modality-specific modules f_i generate latent representations r_i for each modality. These representations are then fused using a defined technique (denoted by $\oplus$ in Figure 5.1.b) to form a shared representation $r_s h$. Finally, this shared representation is processed by a final module f to produce the desired output y. This approach facilitates a richer integration of features, retaining modality-specific information while capturing inter-modal relationships during the feature extraction phase.

One of the major advantages of intermediate fusion is its ability to handle incomplete data more flexibly and effectively. Various techniques are available for fusing latent representations from different modalities, and we recommend readers refer to the review [107] for an in-depth exploration of these methods. However, intermediate fusion comes with certain challenges, including increased computational complexity and training difficulty. The model must learn both unimodal and multimodal representations simultaneously, requiring significant computational and data resources. These demands can pose a barrier in resource-constrained environments, such as many healthcare settings.

Despite these challenges, the dynamic nature of intermediate fusion offers significant advantages. By allowing the model to learn how to fuse information from different sources dynamically, it enhances robustness and adaptability. This integration of complementary information from multiple modalities enables the model to leverage the strengths of each data source while mitigating their individual weaknesses. Such adaptability is particularly valuable in real-world healthcare scenarios, where the quality and availability of data often vary across modalities. Intermediate fusions ability to handle partially missing or noisy modalities can result in more reliable predictions. Moreover, the shared representation created through intermediate fusion fosters a deeper understanding of correlations between different data types, which is critical for complex tasks such as medical diagnosis and prognosis [115, 122].

Despite its resource demands, intermediate fusion represents a promising direction for the development of DL models that are both comprehensive and resilient. This makes it a powerful approach for enhancing decision-making in healthcare environments [107, 115, 122].

5.2.4 Handling Incomplete Data

Real-world multimodal data are often imperfect due to missing features or modalities. Therefore, there is a pressing need for multimodal models robust in the presence of incomplete data. Missing data, whether involving individual features or entire modalities, is a common challenge across various fields and is often caused by issues, such as human error, survey non-responses, data corruption, or systematic loss. Traditional approaches to address missing data typically rely on imputation techniques, which attempt to fill these gaps but can introduce biases or fail to capture underlying complexities. For instance, we employed the k-Nearest Neighbors (kNN) imputer, which has demonstrated effectiveness in handling missing values in tabular data [123, 124, 116]. Additionally, we tested the Missing in Attributes (MIA) strategy, used by tree-based models to dynamically manage missing features without requiring imputation.

Healthcare settings are particularly vulnerable to the problem of incomplete data, as patients may follow different treatment plans, discontinue care for reasons such as transferring facilities, voluntarily ceasing treatment, or even passing away. Moreover, privacy concerns further exacerbate data incompleteness in these settings [124]. Many methods simply exclude patients with missing values, which can significantly reduce data availability and compromise the robustness of analyses. Other approaches often involve imputing missing information using data from available modalities for the same subject or from other patients with similar characteristics.

Several advanced methodologies have been proposed to address the issue of missing data in multimodal contexts:

The Contrastive Masked-Attention Model integrates a Generative Adversarial Network (GAN)-based augmentation mechanism to synthesize data for missing modalities and employs contrastive learning to enhance cross-modal representations. Masked attention ensures that only interactions between observed modalities are captured, thereby minimizing the introduction of extraneous noise [125].

The Cascaded Multi-Modal Mixing Transformers implement a cascaded cross-attention architecture to effectively integrate multiple available modalities, enabling robust performance even when some modalities are missing. This approach offers flexibility and adaptability in fusing modality-specific information [126].

The Missing Modalities in Multimodal healthcare framework employs task-guided, modality-adaptive similarity metrics to identify similar patients and impute missing data. By leveraging auxiliary information from comparable patients, this method preserves the underlying relationships in multimodal healthcare data [127].

Shared-Specific Feature Modeling disentangles shared features from modality-specific ones, enabling efficient handling of missing data during both training and inference. By learning shared features across all available modalities, this approach ensures the retention of essential representations while maintaining model performance [128].

The Severely Missing Modality model uses a Bayesian meta-learning framework to approximate latent representations for missing modalities. This method is designed to handle incomplete data during both training and testing, offering robust generalization capabilities even when data availability is severely limited [121].

These methodologies highlight diverse strategies for compensating for missing information, including identifying shared latent representations, generating synthetic data, and leveraging auxiliary patient information. By reducing dependence on complete multimodal datasets, these approaches improve the practicality of multimodal models in real-world clinical and resource-constrained settings. However, both traditional and DL-based approaches share a common limitation: they rely on artificially filling data gaps, which can introduce bias and compromise task accuracy.

To address this limitation, we propose a model that exclusively utilizes the available features and modalities, avoiding the generation of synthetic data. By focusing solely on effectively leveraging observed information, our approach aims to enhance robustness and reliability, even in scenarios with severely missing data.

5.3 Methods

In this work, we propose MARIA (Multimodal Attention Resilient to Incomplete datA), a multimodal model specifically designed to address the challenges of incomplete features and modalities in multimodal healthcare data. The model effectively integrates data modalities that may be incomplete or entirely absent, offering a robust solution for predictive analysis without relying on traditional data imputation techniques or synthetic data generation.

This section first provides an overview of the MARIA model design. We then focus on the architecture of the modality-specific encoders, the strategies employed for handling missing data, and the regularization techniques implemented during training to enhance generalizability under incomplete input conditions.

5.3.1 Model

MARIA is specifically designed to be resilient to incomplete data and modalities without relying on imputation. It employs intermediate fusion, using modality-specific encoders and

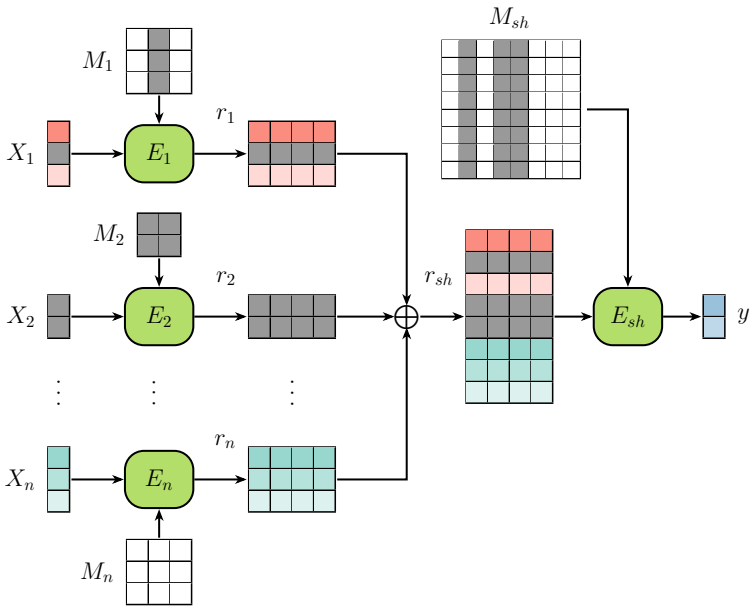

Figure 5.2: MARIA architecture. Each modality-specific encoder E_i takes the modality X_i as input to generate the latent representation r_i. Then, the shared encoder E_{sh} elaborates the concatenation of the latent representations r_{sh} to get the final output y. In the figure, a gray square represents a missing feature or its respective element in the masking matrix.

a shared encoder with a masked self-attention mechanism to combine latent representations while effectively managing missing data. The architecture incorporates multiple modality-specific encoders for each data modality. In this work, we focus on multimodal problems where the modalities represent tabular data describing various aspects of a patients condition. Thus, MARIA utilizes separate NAIM [116] modules as modality-specific encoders (Figure 5.2). These encoders integrate a modified masked multi-head attention mechanism that selectively focuses on available features within each modality while completely ignoring missing ones. This mechanism ensures robustness by excluding absent features from attention computations.

Each tabular modality X_i, where $i = 1, \ldots, n$, is encoded into embeddings using look-up tables [116], which represent missing entries with a specific non-trainable embedding. The modality embeddings are then processed by their corresponding encoders E_i, which compute query, key, and value matrices, denoted as Q_i, K_i, and V_i, using linear transformations:

$$\begin{cases} Q_i = X_i \cdot W_i^Q & d^h = d^e/h \\ K_i = X_i \cdot W_i^K & W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d^e \times d^h} \\ V_i = X_i \cdot W_i^V & Q_i, K_i, V_i \in \mathbb{R}^{|X_i| \times d^h} \end{cases} \tag{5.1}$$

where W_i^Q, W_i^K and W_i^V are learnable weight matrices. These transformations reduce di-

mensionality to d^h, determined by the token dimensions d^e and the number of heads h in the model. Next, a modified masked self-attention mechanism is applied:

$$MSA(Q_i, K_i, V_i) = ReLU\left(softmax\left(\frac{Q_i K_i^T}{\sqrt{d^h}} + M_i\right) + M_i^T\right) V_i \tag{5.2}$$

where the masking matrix M_i ensures that missing features do not influence the latent representation r_i. The elements of M_i are defined as follows:

$$M_i^{kj} = \begin{cases} -\infty & \text{if } X_i^j \text{ is missing} \\ 0 & \text{otherwise} \end{cases}, \quad M_i = \begin{bmatrix} 0 & \text{-}\infty & \dots & 0 & 0 \\ 0 & \text{-}\infty & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \text{-}\infty & \dots & 0 & 0 \\ 0 & \text{-}\infty & \dots & 0 & 0 \end{bmatrix} \tag{5.3}$$

This operation effectively zeroes out weights associated with missing features after applying *softmax* and *ReLU*. Each modality-specific encoder E_i generates a latent representation r_i, of dimensions $|X_i| \times d^e$, where $|X_i|$ represents the number of tokens in the modality, e.g., the number of features in the $i-$th modality. These latent representations are then concatenated to form a joint representation r_{sh}, composed only of available information, with null vectors representing missing features. This multimodal representation is passed to the shared encoder E_{sh}, which computes its own query, key, and value matrices, denoted as Q_{sh}, K_{sh}, and V_{sh}, as follows:

$$\begin{cases} Q_{sh} = r_{sh} \cdot W_{sh}^Q \\ K_{sh} = r_{sh} \cdot W_{sh}^K \\ V_{sh} = r_{sh} \cdot W_{sh}^V \end{cases} \quad \begin{matrix} W_{sh}^Q, W_{sh}^K, W_{sh}^V \in \mathbb{R}^{d^e \times d^h} \\ Q_{sh}, K_{sh}, V_{sh} \in \mathbb{R}^{\sum_i |X_i| \times d^h} \end{matrix} \tag{5.4}$$

where W_{sh}^Q, W_{sh}^K and W_{sh}^V are weights matrices learned during training. As with the modality-specific encoders, dimensionality is reduced to d^h through linear transformations.

A similar modified masked self-attention mechanism is then applied:

$$MSA(Q_{sh}, K_{sh}, V_{sh}) = ReLU\left(softmax\left(\frac{Q_{sh} K_{sh}^T}{\sqrt{d^h}} + M_{sh}\right) + M_{sh}^T\right) V_{sh} \tag{5.5}$$

where the masking matrix M_{sh} ensures that missing modalities do not impact the final shared representation. This matrix operates in the same manner as M_i, zeroing out weights associated with missing modalities. The elements of the masking matrix are defined as

follows:

$$M_{sh}^{kj} = \begin{cases} -\infty & \text{if } r_{sh}^{j} \text{ is missing} \\ 0 & \text{otherwise} \end{cases}, \quad M_{sh} = \begin{bmatrix} 0 & \text{-}\infty & \dots & 0 & 0 \\ 0 & \text{-}\infty & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \text{-}\infty & \dots & 0 & 0 \\ 0 & \text{-}\infty & \dots & 0 & 0 \end{bmatrix} \tag{5.6}$$

M_{sh} sums the $-\infty$ values to weights that need to be ignored, effectively zeroing them after applying the *softmax* and $ReLU$ operations.

The masking matrices M_i and M_{sh} play a crucial role in MARIAs ability to handle incomplete data. These matrices are incorporated into the attention mechanism to explicitly prevent the model from attending to missing values. Specifically, the matrix M_i is applied within each modality-specific encoder to suppress the influence of missing features. Similarly, the matrix M_{sh} is applied in the shared encoder to control attention across modalities. This dual-masking mechanism allows MARIA to dynamically adapt its attention based on the availability of data, without resorting to imputation or synthetic representations.

Finally, the joint representation is passed through a fully connected layer, which predicts the output y. The training process minimizes prediction error, updating the weights of both the modality-specific encoders (E_i) and the shared encoder (E_{sh}) via end-to-end backpropagation.

This architecture qualifies MARIA as an intermediate fusion model, performing fusion at the latent representation level. By dynamically optimizing all encoders, MARIA balances the contributions of different modalities and adapts to maximize their utility during training and inference. The use of a masked multi-head attention mechanism ensures the model focuses adaptively on informative parts of the input, completely ignoring missing data. This approach allows each modality to contribute based on its completeness, resulting in accurate and reliable multimodal representations.

5.3.2 Regularization Technique for Missing Data

To enhance the model's generalizability under incomplete input conditions, we employ regularization strategies during training that improve its resilience to varying degrees of data incompleteness. These strategies ensure that even when some modalities or features are unavailable, the model can still generate accurate and meaningful outputs [116, 126]. During training, we simulate a relaxed missing data setting where each modality or feature is treated as potentially missing, while maintaining at least one available data point per patient. This

approach allows the model to learn how to handle different levels of missingness effectively, making it particularly well-suited to the variability typically found in clinical datasets. By encouraging the model to extract meaningful representations from each available modality, these masking strategies promote robustness against incomplete information during both training and inference.

Modality Dropout During training, the model uses a stochastic masking procedure to simulate incomplete data scenarios. Given a sample $X = \{X_1, \ldots, X_n\}$, where n represents the number of modalities, let $v_m \leq n$ denote the number of non-missing modalities in the sample (where $v_m = n$ for fully populated data or $v_m < n$ for partially missing data). A binary decision variable determines whether masking will be applied to the sample X. If the sample is selected for masking, a random count c_m of modalities to mask is chosen uniformly from the set $\{1, 2, \ldots, v_m - 1\}$, ensuring that at least one modality remains unmasked. Finally, c_m non-missing modalities are randomly selected, and their values are set to *missing*, producing the augmented sample.

Feature Dropout Similarly, when a tabular modality X_i is set as present, a similar stochastic masking procedure is applied at the feature level. A binary decision variable determines whether masking will be applied to the features of the modality X_i. If masking is applied, a random count c_i of features to mask is chosen uniformly from the set $\{1, 2, \ldots, v_i - 1\}$, where v_i is the number of non-missing features of the modality X_i. This ensures that at least one feature remains unmasked. Finally, c_i non-missing features within X_i are randomly chosen and set to *missing*, resulting in the augmented modality.

5.4 Experimental Configuration

In this section, we first describe the datasets used in the experiments and the preprocessing applied (Section 5.4.1). We then outline the combinations of models and imputers employed as competitors (Section 5.4.2). Finally, we present the metrics used for evaluation (Section 5.4.5).

5.4.1 Data

We evaluated MARIA and its competitor models on two publicly available datasets across eight diagnostic and prognostic tasks (details in Table 5.1). These tabular datasets represent real-world scenarios where patient data is often incomplete, highlighting the need for methods that are resilient to missing information.

Dataset	Task	# of samples	Class distribution		Modalities Info (% of missing features and modalities)
ADNI [129]	Diagnosis Binary	953	**CN**: 542	**AD**: 411	Assessment: 37 (f: 35% - m: 0%) Biospecimen: 47 (f: 57% - m: 5%) Image Analysis: 14 (f: 38% - m: 1%) Subject Characteristics: 17 (f: 37% - m: 0%)
	Diagnosis Multiclass	2066	**CN**: 542 **LMCI**: 690	**EMCI**: 423 **AD**: 411	Assessment: 37 (f: 32% - m: 0%) Biospecimen: 47 (f: 56% - m: 7%) Image Analysis: 14 (f: 36% - m: 1%) Subject Characteristics: 17 (f: 38% - m: 0%)
	Prognosis 12 months	1340	**CN**: 427 **MCI**: 797	**Dementia**: 116	Assessment: 126 (f: 33% - m: 0%) Biospecimen: 51 (f: 47% - m: 0%) Image Analysis: 18 (f: 27% - m: 0%) Subject Characteristics: 21 (f: 29% - m: 0%)
	Prognosis 24 months	1159	**CN**: 428 **MCI**: 535	**Dementia**: 196	Assessment: 126 (f: 34% - m: 0%) Biospecimen: 51 (f: 46% - m: 0%) Image Analysis: 18 (f: 27% - m: 0%) Subject Characteristics: 21 (f: 28% - m: 0%)
	Prognosis 36 months	856	**CN**: 239 **MCI**: 420	**Dementia**: 197	Assessment: 126 (f: 27% - m: 0%) Biospecimen: 51 (f: 37% - m: 0%) Image Analysis: 18 (f: 27% - m: 0%) Subject Characteristics: 21 (f: 23% - m: 0%)
	Prognosis 48 months	693	**CN**: 269 **MCI**: 280	**Dementia**: 144	Assessment: 126 (f: 36% - m: 0%) Biospecimen: 51 (f: 47% - m: 0%) Image Analysis: 18 (f: 28% - m: 0%) Subject Characteristics: 21 (f: 30% - m: 0%)
AIforCOVID [130]	Mild/Severe	1585	**Mild**: 839	**Severe**: 746	Blood Analysis: 14 (f: 31% - m: 1%) Clinical History: 13 (f: 24% - m: 9%) Personal Info: 2 (f: 0% - m: 0%) Admission State: 5 (f: 8% - m: 0%)
	Death	1585	**Censored**: 1336	**Uncensored**: 249	

Table 5.1: Datasets' details consisting of: dataset name and reference, the task name, the number of samples, the classes' distribution, and the different tabular modalities, with the corresponding number of features and the respective rates of original missing features (f) and modalities (m).

The first dataset was obtained from the Alzheimers Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu) [129]. The ADNI was launched in 2003 as a public-private partnership, led by Principal Investigator Michael W. Weiner, MD. The primary goal of ADNI has been to test whether serial magnetic resonance imaging, positron emission tomography, other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment (MCI) and early Alzheimers disease (AD), with respect to the Cognitively Normal (CN) group. For our study, we used tabular data derived from four baseline modalities: Assessment (cognitive and neuropsychological scores), Biospecimen (CSF, ApoE genotyping, and lab data), Image analysis (MRI and PET neuroimaging biomarkers), and Subject Characteristics (family history, demographics). We analyzed data from ADNI 1, GO, 2, and 3 phases. Diagnostic tasks included binary classification (CN vs. AD) and ternary classification (CN vs. AD vs. MCI), reflecting real clinical differentiation scenarios. Additionally, prognostic tasks aimed to predict whether treatment intervention might be necessary at specific future points (12, 24, 36, and 48 months post-recruitment). These tasks classified patients into CN, MCI, and

Dementia categories. Table 5.1 provides details on patient distributions for both baseline and follow-up classifications.

The second dataset, AIforCOVID [130], contains clinical data from six Italian hospitals, collected during the first wave of the COVID-19 pandemic (MarchJune 2020). Data was recorded at the time of hospitalization of symptomatic patients and subsequently anonymized and reviewed. All patients tested positive for SARS-CoV-2 via RT-PCR, with 5% confirmed only after a second test. Each patient was classified as either mild (discharged or hospitalized without ventilatory support) or severe (requiring non-invasive ventilation, ICU care, or deceased). Additionally, we evaluated the proposed approach on a death prediction task, classifying patients as either censored (alive) or uncensored (deceased).

MARIA's performance across diverse medical tasks provides valuable insights into its resilience and adaptability in real-world healthcare scenarios. By leveraging the ADNI and AIforCOVID datasets, we demonstrate the model's ability to handle challenges such as missing modalities and heterogeneous data distributions.

These experiments emphasize the importance of multimodal fusion techniques that are not only robust to missing data but also capable of learning from complex, interrelated medical datasets. Such advancements provide the foundation for more resilient and flexible DL models in healthcare, ultimately supporting clinicians in making better-informed decisions under real-world constraints.

5.4.2 Competitors

We conduct an extensive comparison of our methodology against early, late, and intermediate fusion approaches that use missing data imputation as a preprocessing step before model training. Additionally, we benchmark against tree-based models that manage missing values using the Missing In Attributes (MIA) strategy. We choose not to include generative approaches, as these are primarily developed for imaging modalities rather than tabular data, and they introduce additional complexity and randomness, which can hinder reproducibility. Instead, we focus on interpretable, efficient approaches widely adopted in clinical settings, where model simplicity and reliability are crucial. Our analysis includes a total of 10 distinct competitor models, each combined with the k-Nearest Neighbors (kNN) imputation technique, as this method outperformed other imputation strategies in prior studies [123, 116, 1]. For all models, we used default hyperparameters. Indeed, as shown in [52], in line with the "No Free Lunch" theorem, the authors empirically observe that, in many cases, tuning hyperparameters does not lead to significantly better performance compared to the default values suggested in the literature.

Table 5.2 provides an overview of the competitors. The first column categorizes models as ML or DL approaches, the second specifies the base learners, the subsequent columns indicate the techniques used for handling missing data, and the final columns describe the fusion strategies employed. This results in 32 competitor configurations, each marked by an "×" in the respective columns.

Type	**Model**	**Imputer**		**Multimodal Strategies**		
		With	Without	Early	Intermediate	Late
Machine Learning	AdaBoost	×		×		×
	Decision Tree	×	×	×		×
	HistGradientBoost	×	×	×		×
	Random Forest	×	×	×		×
	SVM	×		×		×
	XGBoost	×	×	×		×
Deep Learning	MLP	×		×	×	×
	TabNet	×		×	×	×
	TabTransformer	×		×	×	×
	FTTransformer	×		×	×	×

Table 5.2: Combinations of models, missing techniques and fusion strategies used as competitors, represented by an "×", in the experiments.

To thoroughly evaluate the performance of our proposed methodology, we designed experiments comparing it against a diverse set of baseline models. These experiments involve both ML models, which may rely on imputation or employ the MIA strategy, and DL models paired with imputation techniques.

The ML models include AdaBoost, Decision Trees, HistGradientBoost, Random Forests, Support Vector Machines (SVM), and XGBoost. AdaBoost [21] is a cascading ensemble model that prioritizes hard-to-classify instances, offering robustness across diverse datasets. Decision Trees [22] are highly interpretable models that visually represent decision-making processes, providing insights into complex data relationships. HistGradientBoost [1] offers an efficient variation of gradient boosting, optimized for handling large datasets with improved speed and memory usage. Random Forests [23] is an ensemble of decision trees known for robustness against overfitting and enhanced reliability. SVM [24], equipped with an RBF kernel, is included for its versatility in handling non-linear data separations, providing a contrast to tree-based models. XGBoost [25], an advanced boosting model, employs a gradient descent procedure to minimize loss and is highly effective for tabular datasets.

Additionally, we evaluate DL models paired with imputation methods. These approaches include Multilayer Perceptron (MLP), TabNet, TabTransformer, and FTTransformer. MLP [26] is a foundational DL model that captures complex relationships between features, offering a

baseline for comparison. TabNet [5] leverages self-attention to dynamically select features, improving interpretability and decision-making. TabTransformer [6] uses transformer-based self-attention mechanisms to embed categorical features and capture complex inter-feature relationships. FTTransformer [7] further explores transformers' potential, using distinct embedding strategies for numerical and categorical features.

The selected DL models were evaluated using both early and late fusion approaches. In addition, we developed intermediate fusion variations of these models, where the respective architectures were employed for both modality-specific encoders and shared encoder settings. These intermediate fusion configurations assess the models' ability to concurrently handle multiple input types, leveraging the shared encoder to effectively combine information from various modalities.

These experiments were designed to comprehensively assess the strengths and limitations of each competitor model across various settings. Our comparisons aim to benchmark the performance of our intermediate fusion methodology against both traditional ML approaches and advanced DL competitors. By exploring a wide range of techniques, we highlight the effectiveness of our approach in handling incomplete and heterogeneous multimodal healthcare data.

5.4.3 Preprocessing

For each dataset, we normalize the numerical features using a Min-Max scaler and apply one-hot encoding to the categorical features before feeding them into the models. However, for models such as MARIA, NAIM, HistGradientBoost, TabNet, TabTransformer, FTTransformer, and XGBoost, one-hot encoding is not applied, as their implementations can directly handle categorical features. The preprocessing steps are calibrated using the training data and then applied to the validation and testing sets.

5.4.4 Missingness Evaluation

Our experiments center on generating *Missing Completely At Random* (MCAR) values artificially as it represents the broadest class of missing data type without the introduction of any bias. Our goal is to test our model under diverse missing data conditions by introducing missing values and modalities at various rates, denoted as p, across both the training and testing sets. Specifically, we generate separate missing rates for the training and test sets, set to 0%, 5%, 10%, 30%, 50%, and 75%. No additional missing values were introduced if the generated rate was lower than the dataset's pre-existing level of missingness, resulting in a variable number of experiments per dataset.

Moreover, we performed two types of experiments to evaluate our model under different missingness scenarios:

- **Missing Modalities**: where entire modalities for each patient within the dataset are masked, simulating scenarios where certain data sources were absent.
- **All Missing**: where a certain percentage of individual elements across the entire dataset are masked, thereby affecting all modalities simultaneously and eventually obtaining both missing features and modalities.

This approach allows us to explore a wide range of data completeness scenarios, starting from the dataset's original missing rate (Ω) to extreme cases where up to 75% of features or modalities are missing. Given a target missing rate p, the total number of samples N in the dataset, and the number of elements per sample (either the number of features $|X_i|$ or the number of modalities n), the total number of values to be masked is calculated as $N \cdot |X_i| \cdot p$ or $N \cdot n \cdot p$, respectively. This calculation also takes into account any pre-existing missing entries m_j for each sample j, representing either the number of missing features or missing modalities. Thus, the adjusted number of values to be masked was computed as $N \cdot |X_i| \cdot p - \sum_j m_j$ or $N \cdot n \cdot p - \sum_j m_j$. We then generated a random masking matrix of dimensions $N \times |X_i|$ or $N \times n$ to reflect the structure of the dataset, ensuring that at least one value in any fully masked row or column was replaced to avoid complete data loss in a specific dimension. As a result, the samples and features exhibited varying degrees of missingness, all conforming to the MCAR framework.

5.4.5 Evaluation Metrics

To evaluate the models, each dataset is divided into five stratified cross-validation splits, ensuring the original class distribution is preserved. Within each fold, 20% of the training samples is reserved for validation. The performance of each experiment is assessed by averaging the values of the Area Under the Receiver Operating Characteristic Curve (AUC) or the Matthews Correlation Coefficient (MCC) computed across the cross-validation folds.

The AUC is a widely used metric for evaluating classification tasks, as it measures the model's ability to distinguish between positive and negative instances across all possible thresholds. This makes it a comprehensive indicator of performance, especially for tasks with balanced or slightly imbalanced class distributions. We employed the AUC for tasks such as the ADNI diagnostic and AIforCOVID prognostic evaluations, where the class distributions were relatively balanced. The AUC effectively captures the trade-off between true positive and false positive rates, providing a robust measure of classifier performance.

For highly imbalanced datasets, the AUC may not reliably reflect model performance, as it can be disproportionately influenced by the majority class. In such cases, we use the MCC, which accounts for all four categories of the confusion matrix (true positives, false positives, true negatives, and false negatives). The MCC provides a more balanced view, making it particularly suitable for scenarios with significantly unbalanced positive and negative classes. We employ the MCC for tasks such as the ADNI prognostic evaluations and the AIforCOVID death prediction task, where class imbalance is pronounced.

5.4.6 Fusion Analysis

As an additional analysis of our proposed approach, we also compared MARIA with NAIM [116], our unimodal baseline model, which serves as the foundation for our intermediate fusion methodology. This comparison illustrates the progression from unimodal analysis to the more sophisticated fusion mechanism that supports our proposed methodology. By conducting this comparative analysis, we provide deeper insights into how our approach builds on and improves upon the unimodal baseline, emphasizing the advantages and advancements made through the intermediate fusion technique.

Specifically, we evaluated our approach against the NAIM model under both early and late fusion configurations, which are representative of different strategies for leveraging multimodal information. In the early fusion scenario, we concatenated the features from all modalities before inputting them into the model, effectively treating all available information as a unified input space. This allowed us to explore the interactions between different modalities at an early stage of the modeling process. In the late fusion scenario, we train separate models for each modality, subsequently combining their predictions by averaging the corresponding decision profiles. This approach enables each modality to be independently modeled, allowing the strengths of each individual modality to contribute to the final decision in an aggregated manner.

The results of this extended comparison help demonstrate the value of our proposed fusion strategies and their ability to extract meaningful insights from multimodal data. By examining the performance differences between early and late fusion configurations, we better understand the unique strengths of each strategy and how our intermediate fusion approach effectively balances them. This balance allows MARIA to extract meaningful insights from multimodal data, achieving optimal performance by leveraging both modality-specific information and cross-modal interactions.

5.5 Results and Discussions

As described in the previous section, we compare MARIA with 32 leading competitor models across both ML and DL approaches for tabular data. We evaluated performance under two distinct experimental configurations: missing modalities and all missing. Each configuration involved 36 combinations of missing value percentages in the training and testing sets across 8 tasks, resulting in a total of 18432 experiments (2304 per task). However, due to the pre-existing missing rates in the datasets, the final results were derived from 12192 experiments across all tasks.

To analyze the results, we grouped the competitors into categories for clearer comparisons and visualized the average performance achieved during five-fold cross-validation. The results are presented in eight separate plots (Figures 5.3, 5.4, 5.5, 5.6), each corresponding to a specific task. Each task-specific plot includes charts, one for each level of missing data in the training set, separately showing performance metrics (y-axis) as the missing rate in the testing set increased (x-axis). Note that the number of charts and points along the x-axis may vary due to the initial missing rates (Ω). These charts allow for a detailed comparison of MARIA against different groups of competitors, specifically ML (Figures 5.3 and 5.4) and DL models (Figures 5.5 and 5.6).

In addition to the aggregate performance visualizations, we conducted a statistical significance analysis using the Wilcoxon signed-rank test to assess whether the differences in prediction performance between MARIA and its competitors were statistically meaningful. This analysis was carried out across all experimental conditions, comparing the binary correctness vectors (i.e., correct vs. incorrect predictions) on the concatenated test sets of the five folds, i.e., the entire dataset. The number of experiments in which MARIA significantly outperformed or underperformed each competitor was computed and summarized. The results, presented in tables G.1 and G.2 in Appendix G, further support the robustness of MARIA, particularly in scenarios involving missing modalities, where MARIA outperforms all the competitors in at least 40% of the experiments, while underperforming in no more than 2%. In the more challenging all missing scenario, MARIA still achieves statistically significant wins in at least 30% of the experiments and loses in no more than 10%, confirming its ability to handle severe data incompleteness.

5.5.1 MARIA vs. ML

As an initial analysis, we compared MARIA with traditional ML approaches, as depicted in Figures 5.3 and 5.4. These figures present the average performance of the models under

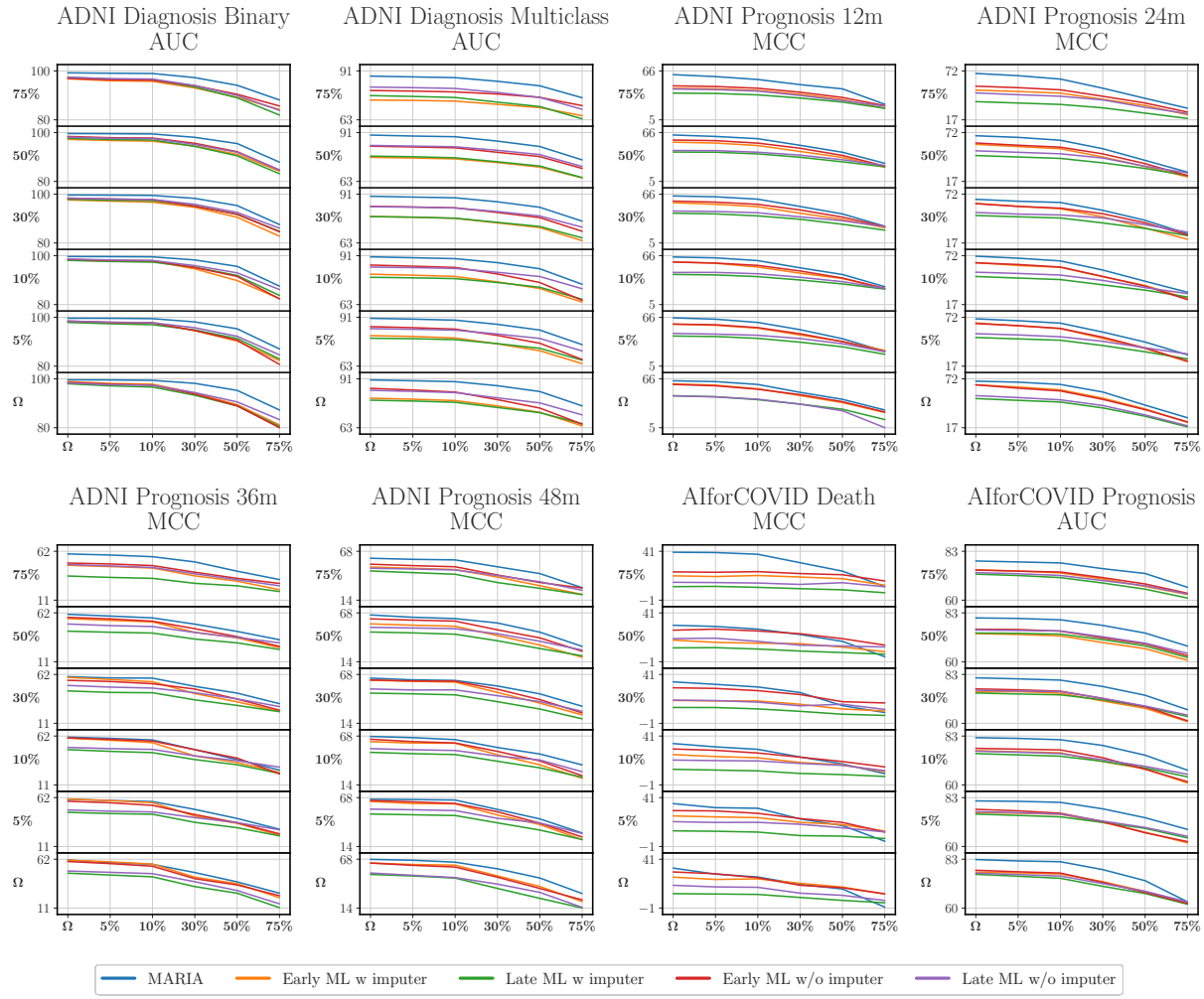

Figure 5.3: MARIA vs. ML in the "missing modalities" scenario. Each plot, one for each task, reports different charts, one for each of the missing rates in the training set, showing how the performance (y-axis) changes as the missing rate in the test set increases.

investigation, categorized by their use of either an imputer or the MIA strategy (without an imputer) and further grouped into early and late fusion approaches. Notably, MARIA consistently demonstrates superior average performance across all levels of missing data in both the training and testing phases, maintaining its advantage even under ideal conditions where no additional missing data is introduced.

This consistent superiority highlights not only MARIA's distinct advantages over traditional ML models but also underscores the largely unexplored potential of DL methods in addressing incomplete data. Moreover, the performance gap between MARIA and its competitors widens as the missing data rate during training increases, whether in the "missing modalities" or "all missing" configurations. This trend suggests that MARIA is particu-

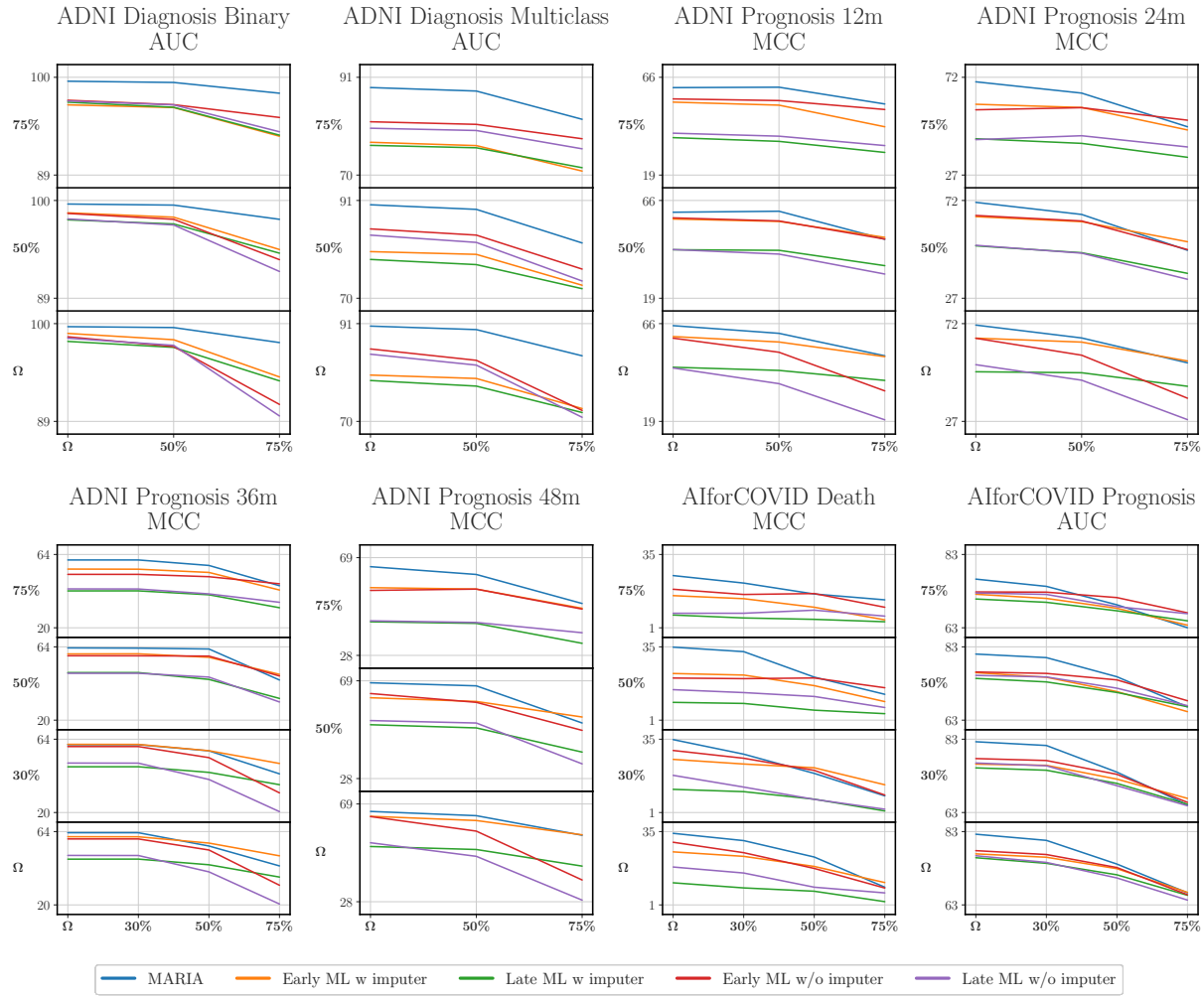

Figure 5.4: MARIA vs. ML in the "all missing" scenario. Each plot, one for each task, reports different charts, one for each of the missing rates in the training set, showing how the performance (y-axis) changes as the missing rate in the test set increases.

larly resilient to varying missing data scenarios. We attribute the improved performance of MARIA to its robust regularization techniques, which enable the model to effectively learn how to handle diverse missing rates during training.

Additionally, an analysis of Figure 5.4 reveals that early fusion approaches generally outperform late fusion ones. This finding underscores the limitations of late fusion in capturing intercorrelations between features from different modalities. Indeed, while late fusion can be compared to an ensemble of many experts, each specialized in a single modality and making independent predictions without communicating with each other, early fusion has a comprehensive view of the patient's information, similar to how a physician integrates all available data about a patient, allowing it to achieve superior performance as a result.

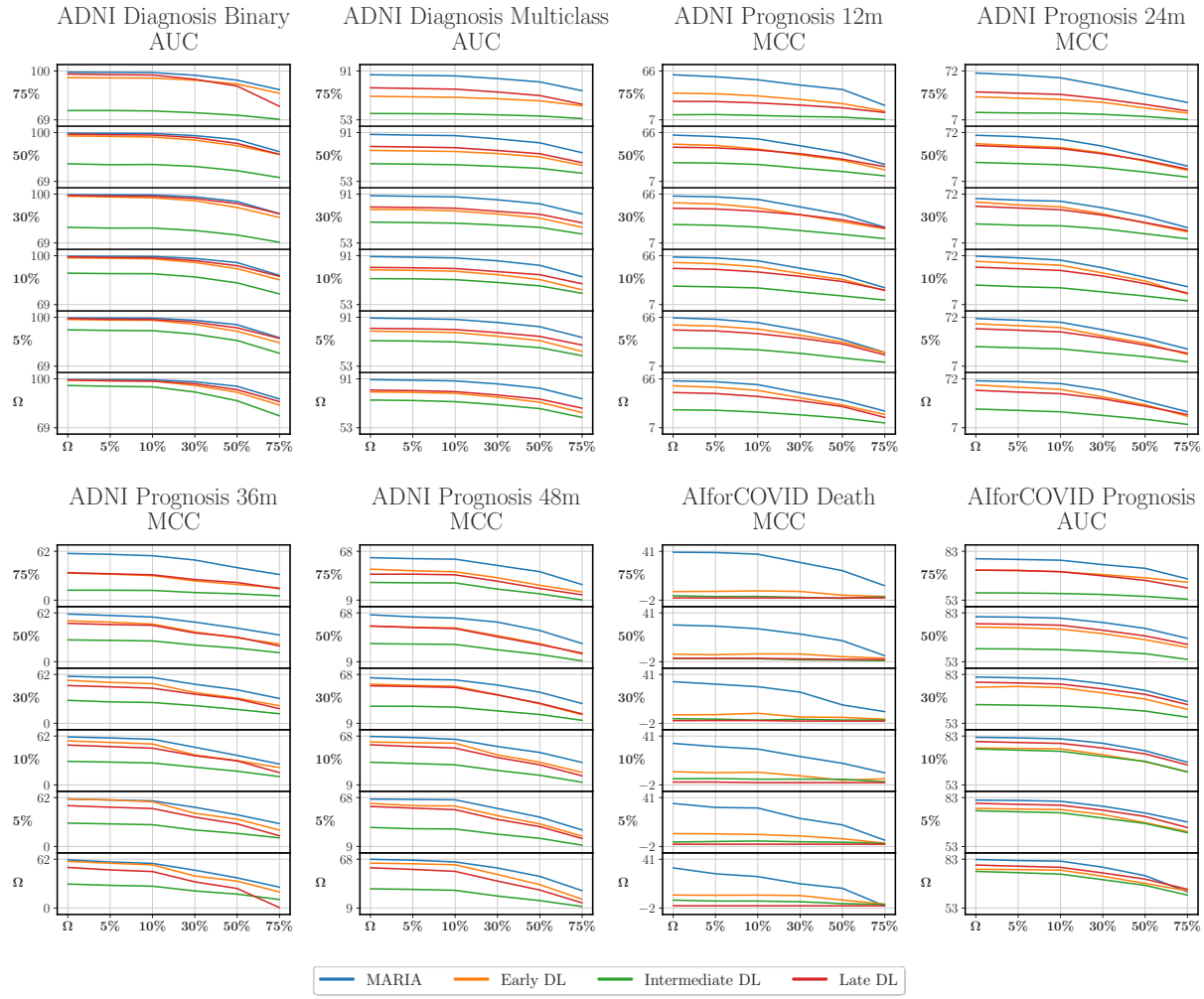

Figure 5.5: MARIA vs. DL in the "missing modalities" scenario. Each plot, one for each task, reports different charts, one for each of the missing rates in the training set, showing how the performance (y-axis) changes as the missing rate in the test set increases.

5.5.2 MARIA vs. DL

We also compared MARIA to leading competitors from the DL domain, specifically designed to analyze tabular data. In Figures 5.5 and 5.6, we reported the average performance of these models when paired with an imputer and grouped into early and late fusion approaches. Additionally, since no existing multimodal approaches are tailored for tabular data, we compared our method to intermediate fusion versions based on these models, as described in section 5.4.2. Once again, MARIA demonstrated superior performance across all levels of missing data in both experimental configurations. Similar to the observations made in comparisons with ML-based approaches, the performance gap between MARIA and its competitors widened as the missing rate in the training set increased. However, unlike

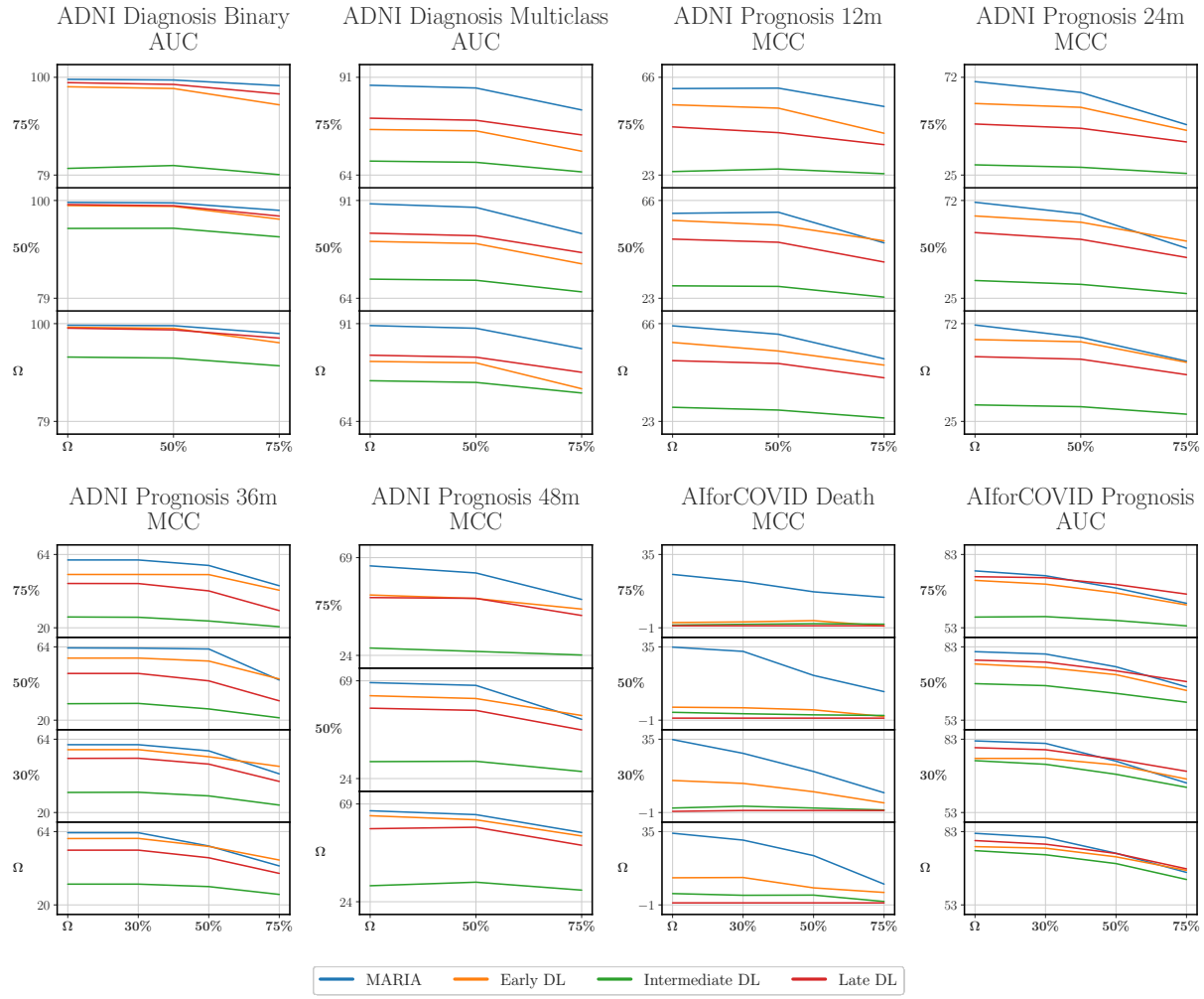

Figure 5.6: MARIA vs. DL in the "all missing" scenario. Each plot, one for each task, reports different charts, one for each of the missing rates in the training set, showing how the performance (y-axis) changes as the missing rate in the test set increases.

previous experiments, early and late fusion approaches exhibited comparable performance, indicating that both methodologies possess similar robustness to missing data. This outcome highlights the capacity of DL techniques to derive meaningful and informative representations from data, even in incomplete scenarios.

In contrast, the intermediate fusion approaches struggled to match the performance of other methods, consistently failing to perform as well as early fusion models. This suggests that intermediate fusion learning is unlikely to outperform its corresponding early and late fusion configurations when all modalities are tabular, particularly in scenarios with high missing rates.

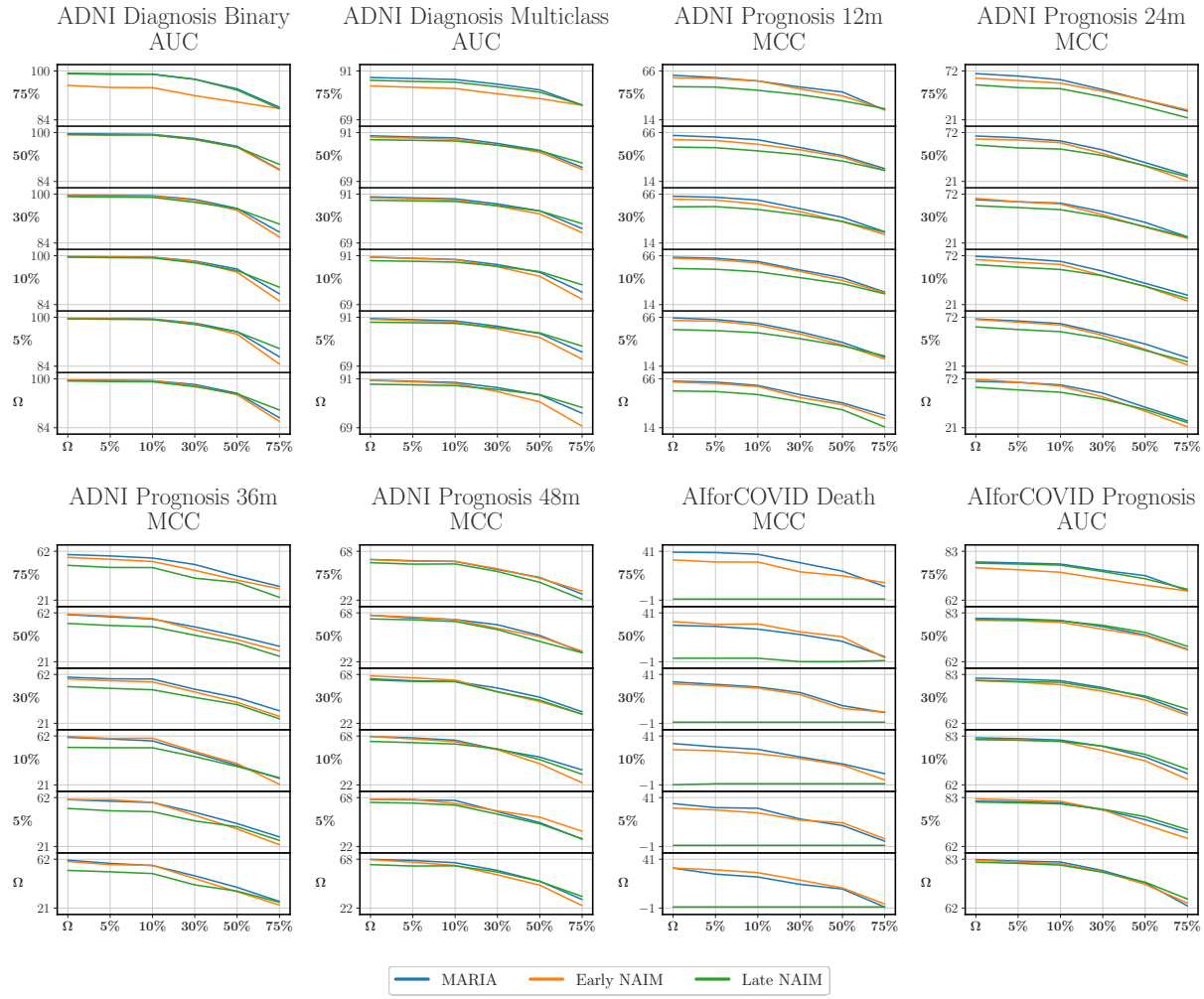

Figure 5.7: MARIA vs. NAIM in the "missing modalities" scenario. Each plot, one for each task, reports different charts, one for each of the missing rates in the training set, showing how the performance (y-axis) changes as the missing rate in the test set increases.

5.5.3 MARIA vs. NAIM

As a final analysis, we compared the proposed intermediate fusion approach with its respective early and late fusion counterparts, all based on the NAIM model [116]. As in the previous analyses, we used the same evaluation framework to compare the different approaches (Figures 5.7 and 5.8).

Interestingly, and as previously noted but now more pronounced, the intermediate fusion approach struggles to consistently outperform the early fusion approach, particularly in the context of tabular data. This suggests that early fusion may offer specific advantages when handling highly structured tabular datasets, where the model benefits from a unified representation of all input features from the beginning. In such contexts, the expected benefits of

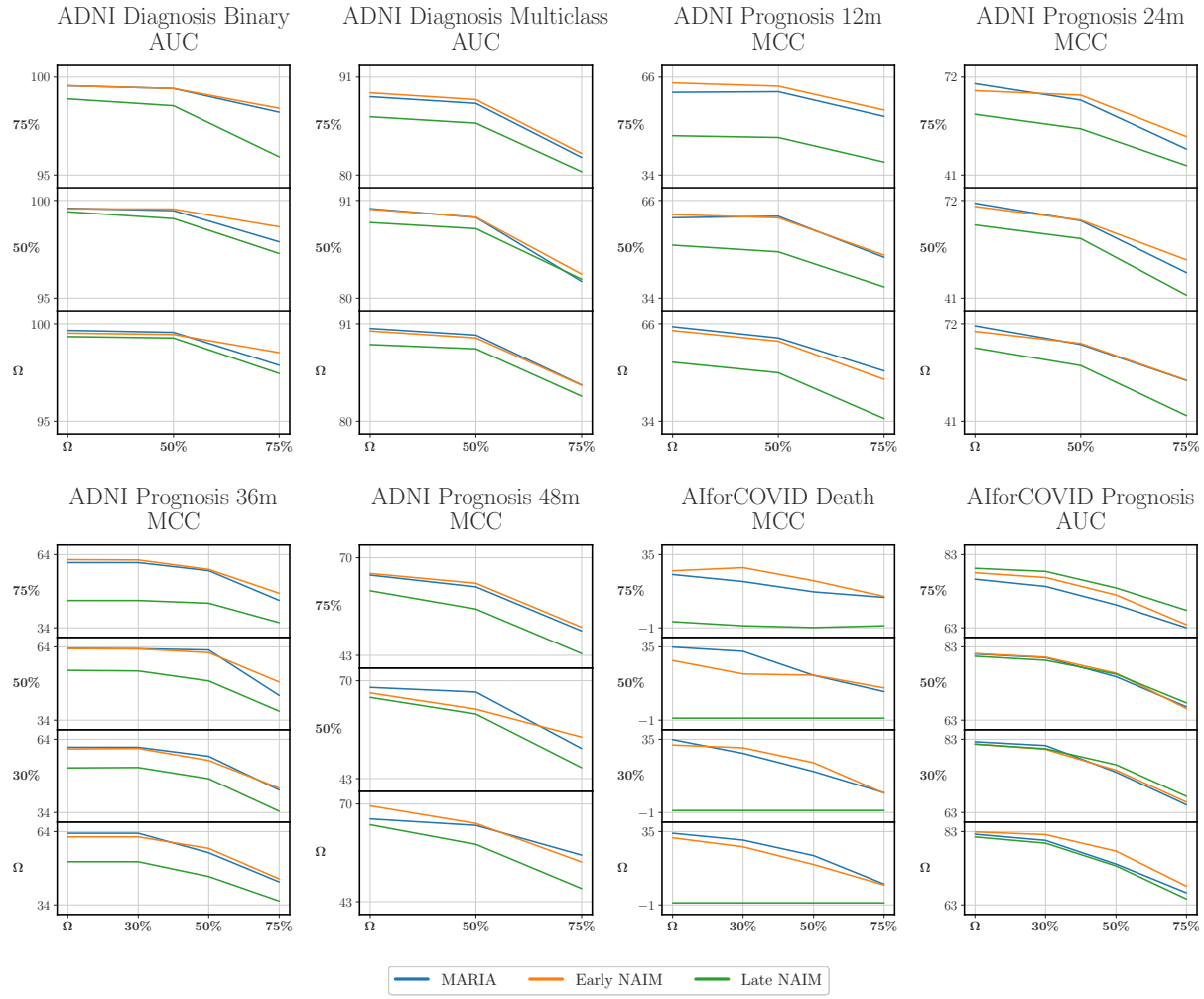

Figure 5.8: MARIA vs. NAIM in the "all missing" scenario. Each plot, one for each task, reports different charts, one for each of the missing rates in the training set, showing how the performance (y-axis) changes as the missing rate in the test set increases.

intermediate fusion appear less significant, potentially due to the inherent heterogeneity of tabular features, which may not require the additional representational flexibility provided by intermediate fusion.

By contrast, the late fusion approach, especially in the "all missing" configuration, rarely achieves performance comparable to the other two approaches. This limitation highlights the challenges faced by late fusion in capturing intercorrelations between features from different modalities, particularly when data completeness is severely compromised.

Overall, these results underscore the trade-offs between fusion strategies, suggesting that the choice between early, intermediate, and late fusion should be guided by the specific characteristics of the data modalities. For structured tabular data, early fusion seems to

provide an optimal balance between simplicity and performance. By contrast, intermediate and late fusion approaches may be more beneficial in scenarios involving heterogeneous or unstructured data sources.

In Appendix F, we present detailed tables for both types of experiments, showing the average performance in terms of AUC and MCC computed across the respective tasks under consideration. The tables include results at different percentages of missing values (reported in the columns). The first rows indicate the percentage of missing values used during training and testing, while the subsequent rows report the performance for each combination of fusion strategy, model, and missing data handling technique. To facilitate readability, we highlighted in bold the best performance for each combination of missing rates.

As presented in Table F.1, which summarizes the results for the "missing modalities" setting, the first table shows the average AUC scores for the ADNI diagnosis and AIforCOVID prognosis tasks. In this context, MARIA outperforms its competitors in 69.4% of cases (25 out of 36). Similarly, the second table reports the average MCC scores for the ADNI prognosis and AIforCOVID death tasks, where MARIA achieves superior performance in 61.1% of cases (22 out of 36).

In contrast, Table F.2, which details the results for the "all missing" setting, indicates that MARIA exhibits stronger performance in fewer instances: 37.5% (6 out of 16) for tasks evaluated using AUC (first table) and 12.5% (2 out of 16) for tasks assessed with MCC (second table). This difference is likely due to the "all missing" setting generally involving less severe information loss compared to the "missing modalities" setting. These findings suggest that the advantages of MARIA are particularly pronounced in scenarios characterized by significant modality loss.

These analyses demonstrate the robustness of the MARIA model, particularly under conditions with high missing rates, a common challenge in medical applications where patient data are often incomplete or inconsistently available. By effectively leveraging only the available information, the MARIA model enhances diagnostic accuracy and decision support in healthcare, ultimately leading to improved patient outcomes. Across a wide range of experimental configurations and degrees of missingness, the model consistently outperforms competing approaches, including both traditional ML and DL models.

5.6 Conclusions

In this study, we introduced MARIA, a novel transformer-based model designed to tackle the challenges of incomplete multimodal data, especially in healthcare. MARIA employs an intermediate fusion strategy, integrating data from multiple incomplete modalities through

a masked self-attention mechanism that selectively focuses on available information while ignoring missing parts. This approach not only avoids the disadvantages of synthetic imputation but also ensures a robust predictive performance even in the presence of severe data missingness. The results show that MARIA is effective across multiple diagnostic and prognostic tasks, consistently outperforming traditional ML and DL models by adapting to various missing data scenarios during both training and inference.

Despite these promising results, some limitations need to be addressed. One of the key challenges of MARIA lies in its computational complexity. Although the model eliminates the need for imputers, the use of masked attention mechanisms and the intermediate fusion strategy requires substantial computational resources, which may limit its scalability in low-resource environments or when applied to extremely large datasets. To mitigate this, future work will focus on enhancing MARIA's scalability and efficiency, potentially by incorporating more lightweight attention mechanisms or by developing hybrid techniques that blend intermediate fusion with other fusion strategies in a computationally feasible manner.

Furthermore, while MARIA's design is effective for tabular data, its generalization to other types of multimodal inputs, such as imaging or textual data, has not been fully explored [131, 132]. To address this, future research will aim to expand MARIA's applicability to other forms of multimodal data, including medical images and clinical notes [118, 133, 87, 134, 135], which would ensure broader usability across diverse healthcare datasets.

Moreover, a limitation of the current work lies in the exclusive focus on MCAR patterns during training and evaluation. While this setting allows for unbiased benchmarking and controlled experimentation, it does not capture the complexity of Missing At Random (MAR) or Missing Not At Random (MNAR) scenarios, which are more reflective of real-world clinical datasets. In future work, we plan to extend the masking strategies used in training to emulate more realistic missingness mechanisms, by introducing feature- or outcome-dependent dropout schemes. This would allow us to better assess and enhance MARIA's robustness under structured or biased missingness conditions.

Chapter 6

Conclusions and Future Perspectives on Resilient Clinical AI

The development of methods resilient to incomplete clinical data is a crucial step toward ensuring that AI can be safely, effectively, and meaningfully integrated into real-world healthcare systems. In modern clinical environments, where data is collected under varying protocols and with inconsistent frequency, the presence of missing or incomplete data is not a rare anomaly, it is a defining characteristic. These gaps often stem from differences in clinical workflows, limitations in data acquisition tools, patient variability, or institutional constraints. Consequently, designing AI models that can operate under these imperfect conditions is essential to building systems that can perform reliably in real-world clinical practice.

In this thesis, we have taken a structured and progressive approach to tackling the challenges posed by data incompleteness. We began with foundational research on structured tabular data and gradually advanced to more complex scenarios involving multimodal inputs. Each step in this progression has aimed to build a deeper understanding of how deep learning systems can be designed to handle the uncertainty and fragmentation that are inherent to healthcare datasets.

In Chapter 2, we introduced NAIM, a transformer-based model that incorporates masked self-attention to learn from tabular datasets with incomplete features. Unlike conventional approaches that rely heavily on data imputation techniques before modeling, NAIM directly attends to the available inputs, ignoring missing elements in a principled way. This method was benchmarked against multiple existing approaches and demonstrated consistent performance across varying missingness levels. The findings highlighted that integrating resilience directly into the models design, rather than treating it as an external correction, can lead to better adaptability and more trustworthy predictions, especially in real-world scenarios

where clean, complete datasets are rarely available.

Chapter 3 extended this idea to a more complex and realistic clinical task: predicting OS in NSCLC patients using routinely collected clinical data. We employed a variant of NAIM tailored for time-dependent outputs, incorporating custom loss functions capable of handling censored survival data across multiple prediction horizons. Although the presence of missing values posed clear difficulties, the model demonstrated strong predictive stability and reliability. These results show that transformer-based models, when properly adapted, can serve as powerful tools for survival analysis in routine clinical environments where complete data is rarely the norm.

The thesis then transitioned to the domain of multimodal learning in Chapter 4, where we investigated whether combining structured clinical data with medical imaging (specifically CT scans) could enhance predictive performance. This initial attempt employed a late fusion strategy, in which each modality was processed separately before combining the resulting representations at the decision level. While this design offers robustness to missing modalities, since decisions can still be made from the available components, it lacks the ability to capture deep inter-modality relationships. As such, it limits the models ability to fully leverage the complementary nature of different data types. These observations align with the findings of our prior systematic review on multimodal fusion techniques in clinical AI, which identified late fusion as a commonly adopted yet fundamentally limited approach [107]. In contrast, intermediate fusion approaches appear more promising, as they allow for richer interactions between modalities during the learning process, enabling the model to better capture complementary information and dependencies across data types.

To address these limitations, Chapter 5 introduced MARIA, a novel transformer-based architecture designed to perform masked integration across multiple modalities. MARIA employs an intermediate fusion strategy, which enables interaction between modalities, enhancing the models capacity to capture cross-modal synergies while also accounting for missing data. The architecture uses masks to attend selectively to available inputs, making it resilient to various forms of structural incompleteness. Our experiments, conducted across numerous simulated missing-modality scenarios, demonstrated that MARIA outperformed traditional fusion models in both performance and stability. This advancement underscores the importance of designing fusion architectures that are not only effective in ideal conditions but also robust enough to handle the inherent variability of clinical data.

From a methodological standpoint, the central argument of this thesis is that resilience to missing data must be treated as a core design principle in AI for healthcare. Rather than viewing incomplete data as a minor problem to be fixed before analysis, we propose that models should be built with an awareness of what is missing. This perspective leads

to architectures that are inherently more flexible and adaptive, capable of making the most out of whatever information is available without introducing additional bias or uncertainty through arbitrary imputations.

Clinically, the implications of this work are substantial. AI tools that fail when data is missing are unlikely to gain acceptance in day-to-day practice, where clinicians often make decisions based on incomplete records. The models presented in this thesis provide a framework for developing AI systems that are both technically sound and practically deployable. By embracing the reality of missing data rather than working around it, these systems demonstrate a level of robustness and adaptability that is essential for meaningful integration into clinical workflows. Moreover, their ability to operate under uncertainty makes them more aligned with the realities of medical decision-making, where perfect information is rarely available.

6.1 Limitations and Future Works

Despite the promising outcomes and contributions presented in this thesis, several important limitations remain that shape the boundaries of the work and highlight opportunities for future investigation. Addressing these limitations will be essential to broaden the clinical utility, adaptability, and impact of resilient AI models developed for healthcare applications.

Evaluation in a Truly Multimodal Context Currently, the evaluation of MARIA has been limited to tasks involving only tabular data coming from different sources. Although the architecture was explicitly designed to generalize across heterogeneous modalities, such as radiological images and structured clinical features, its capabilities in genuinely multimodal settings have yet to be tested. A logical and necessary extension of this work would be to validate MARIA using datasets that include both imaging and tabular components. Such testing would reflect more realistic clinical environments where multiple types of patient data must be processed and interpreted simultaneously, often under conditions of partial or missing modalities. Demonstrating MARIA's ability to learn meaningful representations across different data types in these scenarios will be critical to assessing its broader clinical applicability.

Expanding Across Modalities and Clinical Domains While the models in this thesis are tested on structured tabular data and CT imaging, healthcare data often spans a far broader spectrum. Modalities such as clinical free-text notes, genomics, laboratory time series, biosensor streams, and electronic health records introduce new challenges and oppor-

tunities. Evaluating the proposed models, and especially MARIA, on tasks involving diverse data types will offer insights into their generalizability, flexibility, and capacity for transfer learning. Furthermore, applying these models in different clinical domains beyond oncology, such as cardiology, neurology, and intensive care, will be crucial for establishing their real-world relevance.

Incorporating Temporal and Longitudinal Information Much of the work in this thesis treats patient data as static inputs, without fully modeling how information evolves over time. In clinical reality, however, healthcare decision-making often relies on interpreting sequences of events, such as changes in clinical measurements, the timing of treatments, or disease progression over multiple encounters. The integration of temporal dynamics across various modalities, especially when faced with missing or irregularly sampled data, remains an underexplored but essential area of investigation. Extending architectures like MARIA to handle temporal data streams, such as visit sequences, diagnostic timelines, or evolving treatment patterns, could enable more accurate modeling of patient trajectories and support early detection, risk stratification, and personalized interventions based on longitudinal trends.

Improving Scalability and Efficiency A further point concerns computational efficiency. As previously discussed, transformer-based models inherently demand greater training resources than classical ML pipelines coupled with imputation. Nonetheless, this cost is partially offset by the elimination of extensive imputation and hyperparameter optimization procedures, which can themselves be computationally and methodologically demanding. Future research should therefore focus on designing lightweight attention mechanisms and hybrid fusion strategies that preserve predictive performance while reducing training and memory overhead. Such advances would help make NAIM and MARIA more practical for deployment in large-scale or resource-constrained clinical environments, where scalability and efficiency are as critical as accuracy.

Broadening the Types of Missingness Considered This thesis primarily addresses missing completely at random (MCAR) conditions, which are analytically tractable but often overly simplistic for real-world clinical data. In practice, clinical missingness frequently follows more complex mechanisms, such as missing at random (MAR) or missing not at random (MNAR), where the probability of missingness depends on observed or unobserved variables, respectively. These forms of missingness introduce both statistical and computational challenges that must be addressed for models to be resilient in operational healthcare

settings. Future work should consider how models can be explicitly trained under MAR and MNAR conditions, possibly by modifying loss functions or introducing auxiliary variables that encode the likelihood of missingness. A promising direction involves embedding these more realistic missingness mechanisms into the models regularization schemes, allowing the system to internalize and adapt to various missingness patterns during training. This could enable more detailed and accurate reasoning in the presence of biased or structured data gaps.

Leveraging Self-Supervised and Semi-Supervised Learning The current models are trained under fully supervised paradigms, relying on labeled datasets curated for specific tasks. However, in many healthcare contexts, annotated data is limited or prohibitively expensive to obtain. Large portions of available data remain unlabeled, especially for rare conditions or outcomes. Incorporating self-supervised learning approaches, such as masked prediction tasks or contrastive learning, could allow the models to pretrain on unlabeled data and adapt to new tasks with minimal supervision. Similarly, semi-supervised methods that leverage both labeled and unlabeled samples may improve data efficiency, accelerate model development, and enable better generalization in low-resource scenarios.

Toward Federated and Privacy-Preserving Learning A significant limitation of the current thesis lies in its implicit assumption of centralized data access. In practice, healthcare data is distributed across multiple institutions, each bound by legal, ethical, and infrastructural constraints on data sharing. Future research should explore how the architectural principles introduced here can be translated into federated learning settings. In such frameworks, models can be trained collaboratively across sites without exchanging raw patient data, thus preserving privacy. Moreover, federated models can be adapted to heterogeneous data environments, where each institution may have different patient populations and data collection practices. This adaptation is essential for building fair and inclusive clinical AI systems.

Enhancing Interpretability and Clinical Trust Interpretability remains a central challenge for deploying AI in medicine. Although the transformer models developed in this thesis incorporate attention mechanisms, which can provide insight into feature importance, the clinical validity of such interpretations is not well established. Indeed, attention maps may not align with human clinical reasoning, and their interpretive value remains context-dependent. Future efforts should investigate whether attention weights can be reliably used for explanation in healthcare tasks, or whether additional techniques, such as saliency maps, SHAP values, or counterfactual example generation, are necessary. Embedding these expla-

nations into clinical workflows could enhance transparency, support clinical validation, and ultimately increase trust and acceptance among healthcare professionals.

Beyond these specific directions, the general theme of this thesis supports the vision of resilient, multimodal AI as a transformative force in healthcare. By designing models that can intelligently integrate incomplete and heterogeneous information, we move closer to systems that reflect the complexity and imperfection of real-world medicine. Such systems have the potential to offer not only higher accuracy but also increased fairness and adaptability, qualities that are essential for clinical tools deployed in diverse and unpredictable environments. As research in this field advances, resilient multimodal learning could become a cornerstone of next-generation clinical decision support tools, capable of handling noisy data, adapting to new settings, and empowering healthcare professionals to make more informed, context-sensitive decisions.

The limitations identified in this chapter, rather than undermining the value of the work, serve to highlight where future efforts should be directed. They outline a clear direction for future research aimed at developing models that are better suited to the complexities of real-world medical data and more aligned with the practical challenges of clinical AI applications.

Ultimately, this thesis argues that embracing the imperfections of clinical data is not a compromise, but a necessary evolution in the development of AI for healthcare. By shifting the emphasis from data completeness to model-level resilience, we move toward systems that are grounded in the realities of clinical decision-making, where information is often partial, but the need to act remains urgent.

> *"In medicine, waiting for complete data is often not an option. Learning to act with what we have: thats where AI must rise."*

Bibliography

[1] Alexandre Perez-Lebel et al. “Benchmarking missing-values approaches for predictive models on health databases”. In: *GigaScience* 11 (2022), giac013.

[2] Ravid Shwartz-Ziv and Amitai Armon. “Tabular data: Deep learning is not all you need”. In: *Information Fusion* 81 (2022), pp. 84–90.

[3] Alexey Dosovitskiy et al. “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *arXiv preprint arXiv:2010.11929* (2020).

[4] Alec Radford et al. “Robust Speech Recognition via Large-Scale Weak Supervision”. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 28492–28518.

[5] Sercan Ö Arik and Tomas Pfister. “TabNet: Attentive Interpretable Tabular Learning”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 35. 2021, pp. 6679–6687.

[6] Xin Huang et al. “TabTransformer: Tabular Data Modeling Using Contextual Embeddings”. In: *arXiv preprint arXiv:2012.06678* (2020).

[7] Yury Gorishniy et al. “Revisiting Deep Learning Models for Tabular Data”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 18932–18943.

[8] Mingxuan Liu et al. “Handling missing values in healthcare data: A systematic review of deep learning-based imputation techniques”. In: *Artificial Intelligence in Medicine* (2023), p. 102587.

[9] H Cevallos Valdiviezo and Stefan Van Aelst. “Tree-based prediction on incomplete data using imputation or surrogate decisions”. In: *Information Sciences* 311 (2015), pp. 163–181.

[10] Stef Van Buuren and Karin Groothuis-Oudshoorn. “mice: Multivariate Imputation by Chained Equations in R”. In: *Journal of statistical software* 45 (2011), pp. 1–67.

[11] Olga Troyanskaya et al. “Missing value estimation methods for DNA microarrays”. In: *Bioinformatics* 17.6 (2001), pp. 520–525.

[12] Jiaxuan You et al. "Handling missing data with graph representation learning". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 19075–19087.

[13] Ashish Vaswani et al. "Attention is All you Need". In: *Advances in neural information processing systems* 30 (2017).

[14] Terrance DeVries and Graham W Taylor. "Improved Regularization of Convolutional Neural Networks with Cutout". In: *arXiv preprint arXiv:1708.04552* (2017).

[15] D.; Dua and C. Graff. *UCI Machine Learning Repository*. 2017. URL: http://archive.ics.uci.edu/ml.

[16] Barry Becker and Ronny Kohavi. *Adult*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5XW20. 1996. DOI: https://doi.org/10.24432/C5XW20.

[17] S. Moro, P. Rita, and P. Cortez. *Bank Marketing*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5K306. 2012. DOI: https://doi.org/10.24432/C5K306.

[18] C. Sakar and Yomi Kastro. *Online Shoppers Purchasing Intention Dataset*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5F88Q. 2018. DOI: https://doi.org/10.24432/C5F88Q.

[19] Marek Sikora and Lukasz Wrobel. *seismic-bumps*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5W902. 2013. DOI: https://doi.org/10.24432/C5W902.

[20] Mark Hopkins et al. *Spambase*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C53G6X. 1999. DOI: https://doi.org/10.24432/C53G6X.

[21] Yoav Freund, Robert Schapire, and Naoki Abe. "A Short Introduction to Boosting". In: *Journal-Japanese Society For Artificial Intelligence* 14.771-780 (1999), p. 1612.

[22] Leo Breiman et al. "Classification and regression trees". In: *Pacific Grove, Wadsworth* (1984).

[23] Tin Kam Ho. "Random Decision Forests". In: *Proceedings of 3rd international conference on document analysis and recognition*. Vol. 1. IEEE. 1995, pp. 278–282.

[24] Corinna Cortes and Vladimir Vapnik. "Support-Vector Networks". In: *Machine learning* 20.3 (1995), pp. 273–297.

[25] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016, pp. 785–794.

[26] Hassan Ramchoun et al. “Multilayer Perceptron: Architecture Optimization and Training”. In: *International Journal of Interactive Multimedia and Artificial Intelligence* 4.1 (Special Issue on Artificial Intelligence Underpinning 2016), pp. 26–30.

[27] Philipp Thölke et al. “Class imbalance should not throw you off balance: Choosing the right classifiers and performance metrics for brain decoding with imbalanced data”. In: *NeuroImage* 277 (2023), p. 120253.

[28] Guy Dar et al. “Analyzing Transformers in Embedding Space”. In: *Annual Meeting of the Association for Computational Linguistics*. 2023, pp. 16124–16170.

[29] A Cortellini et al. “Transformer-based AI approach to unravel long-term, time-dependent prognostic complexity in patients with advanced NSCLC and PD-L1≥50%: insights from the pembrolizumab 5-year global registry”. In: *J Immunother Cancer* 13 (2025). DOI: 10.1136/jitc-2025-012423.

[30] Abdullah Alanazi. “Using machine learning for healthcare challenges and opportunities”. In: *Informatics in Medicine Unlocked* (2022), p. 100924.

[31] Muhammad Javed Iqbal et al. “Clinical applications of artificial intelligence and machine learning in cancer diagnosis: looking into the future”. In: *Cancer cell international* 21.1 (2021), pp. 1–11.

[32] Word Health Organisation. *LUNG*. Available online: https://gco.iarc.fr/today/data/factsheets/cancers/15-Lung-fact-sheet.pdf. 2020.

[33] Cancer.net. *Lung CancerNon-Small Cell: Statistics*. Available online: https://www.cancer.net/cancer-types/lung-cancer-non-small-cell/statistics. 2022.

[34] Georgios Kantidakis, Audinga-Dea Hazewinkel, Marta Fiocco, et al. “Neural Networks for Survival Prediction in Medicine Using Prognostic Factors: A Review and Critical Appraisal”. In: *Computational and Mathematical Methods in Medicine* 2022 (2022).

[35] Daochen Zha et al. “Data-centric artificial intelligence: A survey”. In: *arXiv preprint arXiv:2303.10158* (2023).

[36] Mogana Darshini Ganggayah et al. “Predicting factors for survival of breast cancer patients using machine learning techniques”. In: *BMC medical informatics and decision making* 19 (2019), pp. 1–17.

[37] Sumeet Hindocha et al. “A comparison of machine learning methods for predicting recurrence and death after curative-intent radiotherapy for non-small cell lung cancer: Development and validation of multivariable clinical prediction models”. In: *EBioMedicine* 77 (2022), p. 103911.

[38] Jason C Hsu et al. “Development and Validation of Novel Deep-Learning Models Using Multiple Data Types for Lung Cancer Survival”. In: *Cancers* 14.22 (2022), p. 5562.

[39] Yang Yang et al. “Machine learning application in personalised lung cancer recurrence and survivability prediction”. In: *Computational and Structural Biotechnology Journal* 20 (2022), pp. 1811–1820.

[40] Yash Dagli, Saumya Choksi, and Sudipta Roy. “Prediction of two year survival among patients of non-small cell lung cancer”. In: *Computer Aided Intervention and Diagnostics in Clinical and Medical Images*. Springer. 2019, pp. 169–177.

[41] Yougen Wu et al. “Using machine learning for mortality prediction and risk stratification in atezolizumab-treated cancer patients: Integrative analysis of eight clinical trials”. In: *Cancer Medicine* 12.3 (2023), pp. 3744–3757.

[42] Magdalena Oguszka et al. “Evaluate Cutpoints: Adaptable continuous data distribution system for determining survival in Kaplan-Meier estimator”. In: *Computer Methods and Programs in Biomedicine* 177 (2019), pp. 133–139.

[43] Cary Oberije et al. “A validated prediction model for overall survival from stage III non-small cell lung cancer: toward survival prediction for individual patients”. In: *International Journal of Radiation Oncology* Biology* Physics* 92.4 (2015), pp. 935–944.

[44] Hugo Loureiro et al. “Artificial intelligence for prognostic scores in oncology: a benchmarking study”. In: *Frontiers in Artificial Intelligence* 4 (2021), p. 625573.

[45] Bora Lee et al. “DeepBTS: prediction of recurrence-free survival of non-small cell lung cancer using a time-binned deep neural network”. In: *Scientific reports* 10.1 (2020), pp. 1–10.

[46] Qianyu Yuan et al. “Performance of a machine learning algorithm using electronic health record data to identify and estimate survival in a longitudinal cohort of patients with lung cancer”. In: *JAMA Network Open* 4.7 (2021), e2114723–e2114723.

[47] Gaetano Manzo et al. “Breast cancer survival analysis agents for clinical decision support”. In: *Computer Methods and Programs in Biomedicine* 231 (2023), p. 107373.

[48] Yiling Wang et al. “Deep learning based time-to-event analysis with PET, CT and joint PET/CT for head and neck cancer prognosis”. In: *Computer Methods and Programs in Biomedicine* 222 (2022), p. 106948.

[49] Changhee Lee et al. "DeepHit: A Deep Learning Approach to Survival Analysis With Competing Risks". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32.1 (Apr. 2018). DOI: 10.1609/aaai.v32i1.11842. URL: https://ojs.aaai.org/index.php/AAAI/article/view/11842.

[50] *CLARO - CoLlAborative multi-sources Radiopathomics approach for personalized Oncology in non-small cell lung cancer*. http://www.cosbi-lab.it/claro/. Accessed: 2023-03-20.

[51] Peter C Austin, Douglas S Lee, and Jason P Fine. "Introduction to the analysis of survival data in the presence of competing risks". In: *Circulation* 133.6 (2016), pp. 601–609.

[52] A. Arcuri and G. Fraser. "Parameter tuning or default values? An empirical investigation in search-based software engineering". In: *Empirical Software Engineering* 18.3 (2013), pp. 594–623.

[53] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[54] Daniel J Stekhoven and Peter Bühlmann. "MissForestnon-parametric missing value imputation for mixed-type data". In: *Bioinformatics* 28.1 (2012), pp. 112–118.

[55] Sebastian Pölsterl. "scikit-survival: A Library for Time-to-Event Analysis Built on Top of scikit-learn". In: *Journal of Machine Learning Research* 21.212 (2020), pp. 1–6. URL: http://jmlr.org/papers/v21/20-729.html.

[56] Scott M Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 4765–4774.

[57] Myra Van Laar et al. "Prognostic factors for overall survival of stage III non-small cell lung cancer patients on computed tomography: A systematic review and meta-analysis". In: *Radiotherapy and Oncology* 151 (2020), pp. 152–175.

[58] National Cancer Institute. *Non-Small Cell Lung Cancer Treatment*. Available online: https://www.cancer.gov/types/lung/hp/non-small-cell-lung-treatment-pdq. (accessed on 18 May 2022). 2022.

[59] Philippe Lambin et al. "Radiomics: Extracting more information from medical images using advanced feature analysis". en. In: *European Journal of Cancer* 48.4 (Mar. 2012), pp. 441–446. ISSN: 09598049. DOI: 10.1016/j.ejca.2011.11.036. URL: https://linkinghub.elsevier.com/retrieve/pii/S0959804911009993 (visited on 05/18/2022).

[60] Madeleine Scrivener et al. "Radiomics applied to lung cancer: a review". en. In: *Translational Cancer Research* 5.4 (Aug. 2016), pp. 398–409. ISSN: 2218676X, 22196803. DOI: 10.21037/tcr.2016.06.18. URL: http://tcr.amegroups.com/article/view/8536/7742 (visited on 05/24/2022).

[61] Sara Ramella et al. "A radiomic approach for adaptive radiotherapy in non-small cell lung cancer patients". en. In: *PLOS ONE* 13.11 (Nov. 2018). Ed. by Aamir Ahmad, e0207455. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0207455. URL: https://dx.plos.org/10.1371/journal.pone.0207455 (visited on 04/03/2022).

[62] Rosa Sicilia et al. "Exploratory radiomics for predicting adaptive radiotherapy in non-small cell lung cancer". In: *2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE. 2018, pp. 250–255.

[63] Irantzu Anzar et al. "NeoMutate: An ensemble machine learning framework for the prediction of somatic mutations in cancer". In: *BMC Medical Genomics* 12.1 (2019). DOI: 10.1186/s12920-019-0508-5. URL: https://link.springer.com/article/10.1186/s12920-019-0508-5.

[64] Rajarsi Gupta et al. "The Emergence of Pathomics". en. In: *Current Pathobiology Reports* 7.3 (Sept. 2019), pp. 73–84. ISSN: 2167-485X. DOI: 10.1007/s40139-019-00200-x. URL: http://link.springer.com/10.1007/s40139-019-00200-x (visited on 05/24/2022).

[65] Charles Z. Liu et al. "Exploring Deep Pathomics in Lung Cancer". en. In: *2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)*. Aveiro, Portugal: IEEE, June 2021, pp. 407–412. ISBN: 978-1-66544-121-6. DOI: 10.1109/CBMS52027.2021.00092. URL: https://ieeexplore.ieee.org/document/9474667/ (visited on 04/04/2022).

[66] Hugo J. W. L. Aerts et al. "Decoding tumour phenotype by noninvasive imaging using a quantitative radiomics approach". en. In: *Nature Communications* 5.1 (Sept. 2014), p. 4006. ISSN: 2041-1723. DOI: 10.1038/ncomms5006. URL: http://www.nature.com/articles/ncomms5006 (visited on 04/04/2022).

[67] Elizabeth Huynh et al. "CT-based radiomic analysis of stereotactic body radiation therapy patients with lung cancer". In: *Radiotherapy and Oncology* 120.2 (2016), pp. 258–266. ISSN: 0167-8140. DOI: https://doi.org/10.1016/j.radonc.2016.05.024. URL: https://www.sciencedirect.com/science/article/pii/S0167814016311380.

[68] Xenia Fave et al. "Delta-radiomics features for the prediction of patient outcomes in nonsmall cell lung cancer". en. In: *Scientific Reports* 7.1 (Dec. 2017), p. 588. ISSN: 2045-2322. DOI: 10.1038/s41598-017-00665-z. URL: http://www.nature.com/articles/s41598-017-00665-z (visited on 05/20/2022).

[69] Hongming Li et al. "Unsupervised machine learning of radiomic features for predicting treatment response and overall survival of early stage non-small cell lung cancer patients treated with stereotactic body radiation therapy". In: *Radiotherapy and Oncology* 129.2 (2018). Special Issue: Radiotherapy in Asia - Part 2, pp. 218–226. ISSN: 0167-8140. DOI: https://doi.org/10.1016/j.radonc.2018.06.025. URL: https://www.sciencedirect.com/science/article/pii/S0167814018333401.

[70] Chintan Parmar et al. "Machine Learning methods for Quantitative Radiomic Biomarkers". In: *Scientific reports* 5 (Aug. 2015), p. 13087. DOI: 10.1038/srep13087.

[71] Natascha Claudia DAmico et al. "Radiomics-Based Prediction of Overall Survival in Lung Cancer Using Different Volumes-Of-Interest". en. In: *Applied Sciences* 10.18 (Sept. 2020), p. 6425. ISSN: 2076-3417. DOI: 10.3390/app10186425. URL: https://www.mdpi.com/2076-3417/10/18/6425 (visited on 04/04/2022).

[72] Lars Heiliger et al. "Beyond Medical Imaging: A Review of Multimodal Deep Learning in Radiology". en. In: (), p. 13.

[73] Sören Richard Stahlschmidt, Benjamin Ulfenborg, and Jane Synnergren. "Multimodal deep learning for biomedical data fusion: a review". en. In: *Briefings in Bioinformatics* 23.2 (Mar. 2022), bbab569. ISSN: 1467-5463, 1477-4054. DOI: 10.1093/bib/bbab569. URL: https://academic.oup.com/bib/article/doi/10.1093/bib/bbab569/6516346 (visited on 04/04/2022).

[74] Yiying Zhang et al. "A Novel Multimodal Radiomics Model for Preoperative Prediction of Lymphovascular Invasion in Rectal Cancer". en. In: *Frontiers in Oncology* 10 (Apr. 2020), p. 457. ISSN: 2234-943X. DOI: 10.3389/fonc.2020.00457. URL: https://www.frontiersin.org/article/10.3389/fonc.2020.00457/full (visited on 04/04/2022).

[75] Matteo Tortora et al. "RadioPathomics: Multimodal Learning in Non-Small Cell Lung Cancer for Adaptive Radiotherapy". In: *arXiv preprint arXiv:2204.12423* (2022).

[76] Stefano Cipollari et al. "Convolutional neural networks for automated classification of prostate multiparametric magnetic resonance imaging based on image quality". In: *Journal of Magnetic Resonance Imaging* 55.2 (2022), pp. 480–490.

[77] Mehdi Amini et al. "Multi-Level PET and CT Fusion Radiomics-based Survival Analysis of NSCLC Patients". en. In: *2020 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*. Boston, MA, USA: IEEE, Oct. 2020, pp. 1–4. ISBN: 978-1-72817-693-2. DOI: 10.1109/NSS/MIC42677.2020.9507759. URL: https://ieeexplore.ieee.org/document/9507759/ (visited on 04/07/2022).

[78] Yujiao Wu et al. "DeepMMSA: A Novel Multimodal Deep Learning Method for Non-small Cell Lung Cancer Survival Analysis". en. In: *arXiv:2106.06744 [cs, eess]* (June 2021). arXiv: 2106.06744. URL: http://arxiv.org/abs/2106.06744 (visited on 04/04/2022).

[79] Qiang He et al. "Feasibility study of a multi-criteria decision-making based hierarchical model for multi-modality feature and multi-classifier fusion: Applications in medical prognosis prediction". en. In: *Information Fusion* 55 (Mar. 2020), pp. 207–219. ISSN: 15662535. DOI: 10.1016/j.inffus.2019.09.001. URL: https://linkinghub.elsevier.com/retrieve/pii/S1566253518309059 (visited on 04/04/2022).

[80] Luís A. Vale-Silva and Karl Rohr. "Long-term cancer survival prediction using multimodal deep learning". en. In: *Scientific Reports* 11.1 (Dec. 2021), p. 13505. ISSN: 2045-2322. DOI: 10.1038/s41598-021-92799-4. URL: http://www.nature.com/articles/s41598-021-92799-4 (visited on 04/04/2022).

[81] Shaimaa Bakr et al. "Data for NSCLC Radiogenomics Collection". In: The Cancer Imaging Archive, 2017. DOI: 10.7937/K9/TCIA.2017.7hs46erv.

[82] H. J. W. L. Aerts et al. "Data From NSCLC-Radiomics". In: The Cancer Imaging Archive, 2019. DOI: 10.7937/K9/TCIA.2015.PF0M9REI.

[83] Robert L. Grossman et al. "Toward a Shared Vision for Cancer Genomic Data". In: *New England Journal of Medicine* 375.12 (2016). PMID: 27653561, pp. 1109–1112. DOI: 10.1056/NEJMp1607591. URL: https://doi.org/10.1056/NEJMp1607591.

[84] Ravi Aggarwal et al. "Diagnostic accuracy of deep learning in medical imaging: A systematic review and meta-analysis". In: *NPJ digital medicine* 4.1 (2021), pp. 1–23.

[85] Mehmet A Gulum, Christopher M Trombley, and Mehmed Kantardzic. "A review of explainable deep learning cancer detection models in medical imaging". In: *Applied Sciences* 11.10 (2021), p. 4573.

[86] Matteo Tortora et al. "Deep Reinforcement Learning for Fractionated Radiotherapy in Non-Small Cell Lung Carcinoma". In: *Artificial Intelligence in Medicine* 119 (2021), p. 102137.

[87] Valerio Guarrasi and Paolo Soda. "Multi-objective optimization determines when, which and how to fuse deep networks: An application to predict COVID-19 outcomes". In: *Computers in Biology and Medicine* 154 (2023), p. 106625.

[88] Víctor Aceña et al. "Minimally overfitted learners: A general framework for ensemble learning". In: *Knowledge-Based Systems* 254 (2022), p. 109669. ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2022.109669. URL: https://www.sciencedirect.com/science/article/pii/S0950705122008450.

[89] Vadim Borisov et al. "Deep neural networks and tabular data: A survey". In: *arXiv preprint arXiv:2110.01889* (2021).

[90] Jiuxiang Gu et al. "Recent advances in convolutional neural networks". In: *Pattern Recognition* 77 (2018), pp. 354–377.

[91] Alex Krizhevsky. "One weird trick for parallelizing convolutional neural networks". In: *arXiv preprint arXiv:1404.5997* (2014).

[92] Karen Simonyan et al. "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (2014).

[93] Kaiming He et al. "Deep residual learning for image recognition". In: *IEEE Conf. on computer vision and pattern recognition*. 2016, pp. 770–778.

[94] Gao Huang et al. "Densely connected convolutional networks". In: *IEEE Conf. on computer vision and pattern recognition*. 2017, pp. 4700–4708.

[95] Christian Szegedy et al. "Going deeper with convolutions". In: *IEEE Conf. on computer vision and pattern recognition*. 2015, pp. 1–9.

[96] Ningning Ma et al. "ShuffleNet V2: Practical guidelines for efficient CNN architecture design". In: *European Conference on Computer Vision (ECCV)*. 2018, pp. 116–131.

[97] Mark Sandler et al. "MobileNetV2: Inverted residuals and linear bottlenecks". In: *IEEE Conf. on computer vision and pattern recognition*. 2018, pp. 4510–4520.

[98] Mingxing Tan et al. "MnasNet: Platform-aware neural architecture search for mobile". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 2820–2828.

[99] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[100] Valerio Guarrasi et al. "Pareto optimization of deep networks for COVID-19 diagnosis from chest X-rays". In: *Pattern Recognition* 121 (2022), p. 108242.

[101] Valerio Guarrasi et al. “A Multi-Expert System to Detect COVID-19 Cases in X-ray Images”. In: *2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE. 2021, pp. 395–400.

[102] Valerio Guarrasi and Paolo Soda. “Optimized Fusion of CNNs to Diagnose Pulmonary Diseases on Chest X-Rays”. In: *International Conference on Image Analysis and Processing*. Springer. 2022, pp. 197–209.

[103] Thomas G Dietterich. “An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization”. In: *Machine learning* 40.2 (2000), pp. 139–157.

[104] Olaf Ronneberger et al. “U-Net: Convolutional networks for biomedical image segmentation”. In: *International Conf. on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[105] Gavin Brown et al. “Diversity creation methods: a survey and categorisation”. In: *Information fusion* 6.1 (2005), pp. 5–20.

[106] Ivan Izonin et al. “The Additive Input-Doubling Method Based on the SVR with Nonlinear Kernels: Small Data Approach”. In: *Symmetry* 13.4 (2021). ISSN: 2073-8994. DOI: 10.3390/sym13040612. URL: https://www.mdpi.com/2073-8994/13/4/612.

[107] Valerio Guarrasi et al. “A systematic review of intermediate fusion in multimodal deep learning for biomedical applications”. In: *Image and Vision Computing* 158 (2025), p. 105509. ISSN: 0262-8856. DOI: https://doi.org/10.1016/j.imavis.2025.105509.

[108] Tadas Baltruaitis, Chaitanya Ahuja, and Louis-Philippe Morency. “Multimodal machine learning: A survey and taxonomy”. In: *IEEE transactions on pattern analysis and machine intelligence* 41.2 (2018), pp. 423–443.

[109] Michael Moor et al. “Foundation models for generalist medical artificial intelligence”. In: *Nature* 616.7956 (2023), pp. 259–265.

[110] Jiquan Ngiam et al. “Multimodal deep learning.” In: *ICML*. Vol. 11. 2011, pp. 689–696.

[111] Fei Zhao, Chengcui Zhang, and Baocheng Geng. “Deep multimodal data fusion”. In: *ACM computing surveys* 56.9 (2024), pp. 1–36.

[112] Renjie Wu et al. “Deep multimodal learning with missing modality: A survey”. In: *arXiv preprint arXiv:2409.07825* (2024).

[113] Md Kaykobad Reza, Ashley Prater-Bennette, and M Salman Asif. "Robust multimodal learning with missing modalities via parameter-efficient adaptation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).

[114] Zhihui Zhang et al. "Synergistic Prompting for Robust Visual Recognition with Missing Modalities". In: *arXiv preprint arXiv:2507.07802* (2025).

[115] Sören Richard Stahlschmidt, Benjamin Ulfenborg, and Jane Synnergren. "Multimodal deep learning for biomedical data fusion: a review". In: *Briefings in Bioinformatics* 23.2 (2022), bbab569.

[116] Camillo Maria Caruso, Paolo Soda, and Valerio Guarrasi. "Not Another Imputation Method: A Transformer-based Model for Missing Values in Tabular Datasets". In: *arXiv preprint arXiv:2407.11540* (2024).

[117] Arianna Francesconi et al. "Class balancing diversity multimodal ensemble for Alzheimers disease diagnosis and early detection". In: *Computerized Medical Imaging and Graphics* 123 (2025), p. 102529.

[118] Filippo Ruffini et al. "Benchmarking Foundation Models and Parameter-Efficient Fine-Tuning for Prognosis Prediction in Medical Imaging". In: *arXiv preprint arXiv:2506.18434* (2025).

[119] Giulia Di Teodoro et al. "A graph neural network-based model with Out-of-Distribution Robustness for enhancing Antiretroviral Therapy Outcome Prediction for HIV-1". In: *Computerized Medical Imaging and Graphics* (2025), p. 102484.

[120] Valerio Guarrasi et al. "Multimodal explainability via latent shift applied to COVID-19 stratification". In: *Pattern Recognition* (2024), p. 110825.

[121] Mengmeng Ma et al. "Smil: Multimodal learning with severely missing modality". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 2021, pp. 2302–2310.

[122] Fatemeh Behrad and Mohammad Saniee Abadeh. "An overview of deep learning methods for multimodal medical data mining". In: *Expert Systems with Applications* 200 (2022), p. 117006.

[123] Marziyeh Afkanpour, Elham Hosseinzadeh, and Hamed Tabesh. "Identify the most appropriate imputation method for handling missing values in clinical structured datasets: a systematic review". In: *BMC Medical Research Methodology* 24.1 (2024), p. 188.

[124] Javier E Flores et al. "Missing data in multi-omics integration: Recent advances through artificial intelligence". In: *Frontiers in Artificial Intelligence* 6 (2023), p. 1098308.

[125] Shuwei Qian and Chongjun Wang. "COM: Contrastive Masked-attention model for incomplete multimodal learning". In: *Neural Networks* 162 (2023), pp. 443–455.

[126] Linfeng Liu et al. "Cascaded multi-modal mixing transformers for alzheimers disease classification with incomplete data". In: *NeuroImage* 277 (2023), p. 120267.

[127] Chaohe Zhang et al. "M3care: Learning with missing modalities in multimodal healthcare data". In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2022, pp. 2418–2428.

[128] Hu Wang et al. "Multi-modal learning with missing modality via shared-specific feature modelling". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 15878–15887.

[129] Clifford R Jack Jr et al. "The Alzheimer's disease neuroimaging initiative (ADNI): MRI methods". In: *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine* 27.4 (2008), pp. 685–691.

[130] Paolo Soda et al. "AIforCOVID: Predicting the clinical outcomes in patients with COVID-19 applying AI to chest-X-rays. An Italian multicentre study". In: *Medical Image Analysis* 74 (2021), p. 102216. ISSN: 1361-8415. DOI: https://doi.org/10.1016/j.media.2021.102216.

[131] Filippo Ruffini et al. "Multi-dataset multi-task learning for COVID-19 prognosis". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2024, pp. 251–261.

[132] Alice Natalina Caragliano et al. "Doctor-in-the-Loop: An explainable, multi-view deep learning framework for predicting pathological response in non-small cell lung cancer". In: *Image and Vision Computing* (2025), p. 105630.

[133] Camillo Maria Caruso et al. "A multimodal ensemble driven by multiobjective optimisation to predict overall survival in non-small-cell lung cancer". In: *Journal of Imaging* 8.11 (2022), p. 298.

[134] Tao Wang et al. "Multi-view imputation and cross-attention network based on incomplete longitudinal and multimodal data for conversion prediction of mild cognitive impairment". In: *Expert Systems with Applications* 231 (2023), p. 120761.

[135] Wei Xiong et al. "Disentanglement and codebook learning-induced feature match network to diagnose neurodegenerative diseases on incomplete multimodal data". In: *Pattern Recognition* 165 (2025), p. 111597.

Appendices

Appendix A

Masking Example

In this section, we aim to illustrate in more depth how the proposed masked self-attention mechanism is able to mask and ignore the contribution of missing values via an example.

Let the feature vector x be composed of 4 different features, out of which the third feature is missing:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{A.1}$$

Let us now compute the embedding matrix e of the features, using equations 2.4 and 2.5 for the categorical and numerical features respectively, using a dimension of the embeddings $d_e = 4$:

$$e = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ d_1 & d_2 & d_3 & d_4 \end{bmatrix} \tag{A.2}$$

As x_3 was missing, we can remark that embedding c denotes the missing feature and the

mask M can be defined:

$$M = \begin{bmatrix} 0 & 0 & \text{-}\infty & 0 \\ 0 & 0 & \text{-}\infty & 0 \\ 0 & 0 & \text{-}\infty & 0 \\ 0 & 0 & \text{-}\infty & 0 \end{bmatrix} \tag{A.3}$$

Then, following the steps in the transformer pipeline, we compute Q, K and V using Eq. 2.6 and a number of heads $h = 2$:

$$Q = \begin{bmatrix} a^Q \\ b^Q \\ c^Q \\ d^Q \end{bmatrix} = \begin{bmatrix} a_1^Q & a_2^Q \\ b_1^Q & b_2^Q \\ c_1^Q & c_2^Q \\ d_1^Q & d_2^Q \end{bmatrix}, \quad K = \begin{bmatrix} a^K \\ b^K \\ c^K \\ d^K \end{bmatrix} = \begin{bmatrix} a_1^K & a_2^K \\ b_1^K & b_2^K \\ c_1^K & c_2^K \\ d_1^K & d_2^K \end{bmatrix}, \quad V = \begin{bmatrix} a^V \\ b^V \\ c^V \\ d^V \end{bmatrix} = \begin{bmatrix} a_1^V & a_2^V \\ b_1^V & b_2^V \\ c_1^V & c_2^V \\ d_1^V & d_2^V \end{bmatrix} \tag{A.4}$$

We now compute the QK^T product to better understand how the contributions from various features (a, b, c and d) are distributed across the attention matrix.

$$QK^T = \begin{bmatrix} a^Q a^K & a^Q b^K & a^Q c^K & a^Q d^K \\ b^Q a^K & b^Q b^K & b^Q c^K & b^Q d^K \\ c^Q a^K & c^Q b^K & c^Q c^K & c^Q d^K \\ d^Q a^K & d^Q b^K & d^Q c^K & d^Q d^K \end{bmatrix} \tag{A.5}$$

Now, using the QK^T product, the vector V and the mask M, we can calculate the output of an attention head for the classical masked self-attention mechanism applied to

tabular data:

$$
\begin{aligned}
&softmax(QK^T + M)V = \\
&= softmax\left(\begin{bmatrix} a^Q a^K & a^Q b^K & a^Q c^K & a^Q d^K \\ b^Q a^K & b^Q b^K & b^Q c^K & b^Q d^K \\ c^Q a^K & c^Q b^K & c^Q c^K & c^Q d^K \\ d^Q a^K & d^Q b^K & d^Q c^K & d^Q d^K \end{bmatrix} + \begin{bmatrix} 0 & 0 & \text{-}\infty & 0 \\ 0 & 0 & \text{-}\infty & 0 \\ 0 & 0 & \text{-}\infty & 0 \\ 0 & 0 & \text{-}\infty & 0 \end{bmatrix}\right)\begin{bmatrix} a_1^V & a_2^V \\ b_1^V & b_2^V \\ c_1^V & c_2^V \\ d_1^V & d_2^V \end{bmatrix} = \\
&= \begin{bmatrix} softmax(a^Q a^K & a^Q b^K & \text{-}\infty & a^Q d^K) \\ softmax(b^Q a^K & b^Q b^K & \text{-}\infty & b^Q d^K) \\ softmax(c^Q a^K & c^Q b^K & \text{-}\infty & c^Q d^K) \\ softmax(d^Q a^K & d^Q b^K & \text{-}\infty & d^Q d^K) \end{bmatrix}\begin{bmatrix} a_1^V & a_2^V \\ b_1^V & b_2^V \\ c_1^V & c_2^V \\ d_1^V & d_2^V \end{bmatrix} = \\
&= \begin{bmatrix} (a^Q a^K)_S & (a^Q b^K)_S & 0 & (a^Q d^K)_S \\ (b^Q a^K)_S & (b^Q b^K)_S & 0 & (b^Q d^K)_S \\ (c^Q a^K)_S & (c^Q b^K)_S & 0 & (c^Q d^K)_S \\ (d^Q a^K)_S & (d^Q b^K)_S & 0 & (d^Q d^K)_S \end{bmatrix}\begin{bmatrix} a_1^V & a_2^V \\ b_1^V & b_2^V \\ c_1^V & c_2^V \\ d_1^V & d_2^V \end{bmatrix} = \\
&= \begin{bmatrix} (a^Q a^K)_S \cdot a_1^V + (a^Q b^K)_S \cdot b_1^V + (a^Q d^K)_S \cdot d_1^V + 0 & (a^Q a^K)_S \cdot a_2^V + (a^Q b^K)_S \cdot b_2^V + (a^Q d^K)_S \cdot d_2^V + 0 \\ (b^Q a^K)_S \cdot a_1^V + (b^Q b^K)_S \cdot b_1^V + (b^Q d^K)_S \cdot d_1^V + 0 & (b^Q a^K)_S \cdot a_2^V + (b^Q b^K)_S \cdot b_2^V + (b^Q d^K)_S \cdot d_2^V + 0 \\ (c^Q a^K)_S \cdot a_1^V + (c^Q b^K)_S \cdot b_1^V + (c^Q d^K)_S \cdot d_1^V + 0 & (c^Q a^K)_S \cdot a_2^V + (c^Q b^K)_S \cdot b_2^V + (c^Q d^K)_S \cdot d_2^V + 0 \\ (d^Q a^K)_S \cdot a_1^V + (d^Q b^K)_S \cdot b_1^V + (d^Q d^K)_S \cdot d_1^V + 0 & (d^Q a^K)_S \cdot a_2^V + (d^Q b^K)_S \cdot b_2^V + (d^Q d^K)_S \cdot d_2^V + 0 \end{bmatrix}
\end{aligned}
\tag{A.6}
$$

As can be seen, some contributions of the missing feature are still present on the third line of the output matrix. Using as a starting point, for the sake of simplicity, the attention matrix computed on the third step of the Eq. A.6, let us compute the output matrix obtained

using the presented masked self-attention mechanism:

$$
\begin{aligned}
&ReLU(softmax(QK^T + M) + M^T)V = \\
&= ReLU\left(\begin{bmatrix} (a^Q a^K)_S & (a^Q b^K)_S & 0 & (a^Q d^K)_S \\ (b^Q a^K)_S & (b^Q b^K)_S & 0 & (b^Q d^K)_S \\ (c^Q a^K)_S & (c^Q b^K)_S & 0 & (c^Q d^K)_S \\ (d^Q a^K)_S & (d^Q b^K)_S & 0 & (d^Q d^K)_S \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\infty & -\infty & -\infty & -\infty \\ 0 & 0 & 0 & 0 \end{bmatrix}\right)\begin{bmatrix} a_1^V & a_2^V \\ b_1^V & b_2^V \\ c_1^V & c_2^V \\ d_1^V & d_2^V \end{bmatrix} = \\
&= ReLU\left(\begin{bmatrix} (a^Q a^K)_S & (a^Q b^K)_S & 0 & (a^Q d^K)_S \\ (b^Q a^K)_S & (b^Q b^K)_S & 0 & (b^Q d^K)_S \\ -\infty & -\infty & -\infty & -\infty \\ (d^Q a^K)_S & (d^Q b^K)_S & 0 & (d^Q d^K)_S \end{bmatrix}\right)\begin{bmatrix} a_1^V & a_2^V \\ b_1^V & b_2^V \\ c_1^V & c_2^V \\ d_1^V & d_2^V \end{bmatrix} = \\
&= \begin{bmatrix} (a^Q a^K)_S & (a^Q b^K)_S & 0 & (a^Q d^K)_S \\ (b^Q a^K)_S & (b^Q b^K)_S & 0 & (b^Q d^K)_S \\ 0 & 0 & 0 & 0 \\ (d^Q a^K)_S & (d^Q b^K)_S & 0 & (d^Q d^K)_S \end{bmatrix}\begin{bmatrix} a_1^V & a_2^V \\ b_1^V & b_2^V \\ c_1^V & c_2^V \\ d_1^V & d_2^V \end{bmatrix} = \\
&= \begin{bmatrix} (a^Q a^K)_S \cdot a_1^V + (a^Q b^K)_S \cdot b_1^V + (a^Q d^K)_S \cdot d_1^V + 0 & (a^Q a^K)_S \cdot a_2^V + (a^Q b^K)_S \cdot b_2^V + (a^Q d^K)_S \cdot d_2^V + 0 \\ (b^Q a^K)_S \cdot a_1^V + (b^Q b^K)_S \cdot b_1^V + (b^Q d^K)_S \cdot d_1^V + 0 & (b^Q a^K)_S \cdot a_2^V + (b^Q b^K)_S \cdot b_2^V + (b^Q d^K)_S \cdot d_2^V + 0 \\ 0 & 0 \\ (d^Q a^K)_S \cdot a_1^V + (d^Q b^K)_S \cdot b_1^V + (d^Q d^K)_S \cdot d_1^V + 0 & (d^Q a^K)_S \cdot a_2^V + (d^Q b^K)_S \cdot b_2^V + (d^Q d^K)_S \cdot d_2^V + 0 \end{bmatrix}
\end{aligned}
\tag{A.7}
$$

By comparing the final outputs, we can note that our method allows the contributions of the missing feature to be completely masked, making the most of the available data.

Appendix B

Training of the Models

For NAIM, we specified its architecture with an embedding dimension $d_e = 6$, encoder layers $L = 6$, and attention heads $h = 3$ [13]. We configured the feed-forward layer size to 1000 neurons, and we eliminated the bias in the embedding of numerical and categorical features, to represent missing values as zero vectors.

During the training of DL models, since all the datasets work on a classification task, we opted to use the cross-entropy as loss function. To enhance the training process, we employed a dynamic learning rate schedule that reduced the rate by a factor of 10 whenever the loss stagnates for 25 epochs. The training incorporated the Adam optimization algorithm with a consistent batch size of 32 across all models and datasets. For all the DL models we used a Glorot uniform initialization of the weights, which were trained all the models for 1500 epochs, using an early stopping rule with patience of 50 epochs. We set the initial 50 epochs as a warm-up period for parameter tuning for all models, except GRAPE for which 500 warm-up epochs are assigned due to its lack of batch sampling. Furthermore, we incorporated L1 and L2 regularization techniques to prevent overfitting.

All models and techniques under consideration are evaluated using their publicly available implementations. For XGBoost[1], GRAPE[2], TabNet[3], TabTransformer[3], and FTTransformer[3], these implementations can be found on GitHub, whereas for the remaining models and imputation techniques, we used their implementations available in the Sklearn library[4]. For all the competitors, we set their default parameters or those provided in the public implementations by authors because we are not interested in fine-tuning the models. Therefore, for the same reason, even in the NAIM implementation we did not perform parameter

[1] https://github.com/dmlc/xgboost/tree/master
[2] https://github.com/maxiaoba/GRAPE/tree/master
[3] https://github.com/dreamquark-ai/tabnet
[3] https://github.com/lucidrains/tab-transformer-pytorch
[4] https://scikit-learn.org/stable

tuning, but we selected them according to the dimensionality of the data used. Indeed, in [52], confirming the "No Free Lunch" theorem, the authors empirically observe that in many cases the use of tuned parameters cannot significantly outperform the default values of a classifier suggested in the literature.

Appendix C

NAIM Complete Results

		Train missing percentage: 0%						Train missing percentage: 5%						Train missing percentage: 10%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
Model	**Strategy**	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	90.82 (3.67)	**90.47 (3.69)**	**89.98 (3.71)**	**88.24 (3.78)**	**84.25 (3.78)**	**77.20 (3.43)**	**90.82 (3.60)**	**90.36 (3.64)**	**89.85 (3.70)**	**88.46 (3.62)**	**83.66 (4.17)**	**76.54 (4.32)**	90.60 (3.71)	**90.28 (3.67)**	**89.80 (3.71)**	**88.24 (3.77)**	**83.79 (4.33)**	76.40 (4.19)
NAIM w/o regularization	Intrinsic	90.11 (3.97)	88.01 (4.06)	85.33 (4.46)	79.55 (4.40)	69.63 (3.90)	61.57 (2.00)	90.37 (3.72)	89.41 (4.01)	88.47 (4.28)	85.32 (4.18)	77.17 (4.57)	67.23 (2.78)	90.46 (3.52)	89.48 (3.90)	88.91 (3.76)	86.07 (4.11)	78.67 (4.89)	68.70 (3.56)
	Constant	88.57 (3.91)	87.35 (3.72)	85.65 (3.77)	82.28 (3.00)	74.83 (2.43)	66.27 (2.21)	88.73 (3.63)	87.98 (3.62)	87.21 (3.55)	84.39 (3.43)	78.13 (2.63)	69.64 (1.85)	88.65 (3.65)	88.03 (3.54)	87.24 (3.53)	85.09 (3.27)	79.37 (2.75)	70.76 (1.58)
	KNN	88.70 (3.87)	88.23 (3.75)	87.33 (3.72)	85.46 (3.47)	81.37 (3.39)	75.41 (3.47)	88.54 (3.78)	87.93 (3.79)	87.46 (3.64)	85.51 (3.36)	81.21 (3.34)	73.27 (3.09)	88.54 (3.74)	88.03 (3.67)	87.41 (3.63)	85.47 (3.35)	80.88 (3.07)	72.60 (2.64)
	MICE	88.62 (3.87)	87.75 (3.69)	86.69 (3.56)	84.04 (3.21)	77.83 (2.27)	69.16 (1.86)	88.55 (3.89)	87.96 (3.81)	87.56 (3.69)	85.63 (3.42)	81.06 (3.04)	74.22 (2.87)	88.36 (3.82)	87.88 (3.71)	87.42 (3.57)	85.70 (3.33)	81.56 (3.15)	75.02 (2.24)
Adaboost	Constant	88.60 (5.22)	87.44 (4.96)	85.95 (4.90)	82.42 (4.12)	74.43 (4.52)	67.16 (2.29)	88.71 (5.05)	88.16 (5.05)	87.55 (5.17)	86.03 (4.57)	80.77 (4.71)	72.11 (4.40)	88.61 (5.07)	88.04 (5.21)	87.64 (5.04)	85.96 (4.96)	80.81 (5.40)	72.98 (4.06)
	KNN	88.60 (5.21)	88.15 (5.02)	87.35 (4.91)	85.75 (4.24)	81.39 (4.39)	75.11 (3.64)	88.55 (5.18)	87.92 (5.15)	87.35 (5.09)	85.72 (4.40)	81.05 (4.44)	72.51 (3.89)	88.59 (5.08)	88.16 (4.96)	87.55 (4.88)	85.75 (4.49)	80.94 (4.10)	72.65 (3.21)
	MICE	88.60 (5.22)	87.82 (4.99)	86.84 (4.79)	84.03 (3.99)	76.47 (3.87)	68.21 (2.31)	88.92 (4.89)	88.18 (4.88)	87.12 (5.33)	84.87 (5.11)	79.46 (5.27)	72.34 (4.55)	88.70 (4.91)	88.12 (4.91)	87.57 (4.90)	85.41 (4.93)	80.72 (5.03)	73.00 (4.60)
Decision Tree	Constant	73.24 (5.50)	71.24 (5.04)	69.53 (4.34)	64.47 (3.50)	59.76 (1.86)	54.17 (1.33)	72.68 (5.98)	71.55 (5.99)	70.51 (5.83)	67.51 (5.15)	62.70 (4.31)	57.43 (2.46)	73.60 (4.86)	72.60 (5.10)	71.71 (4.88)	69.44 (4.64)	65.04 (4.04)	58.78 (3.16)
	KNN	73.31 (5.50)	72.78 (5.34)	72.01 (5.22)	70.00 (5.18)	67.56 (4.31)	64.10 (3.72)	73.08 (5.63)	72.23 (5.57)	71.67 (5.39)	69.59 (5.01)	65.81 (4.61)	60.51 (3.32)	72.73 (5.12)	72.00 (4.97)	71.53 (4.98)	69.26 (4.41)	65.73 (3.95)	59.94 (2.94)
	MICE	73.26 (5.50)	72.35 (5.17)	70.79 (4.72)	67.40 (3.98)	61.99 (2.48)	56.24 (1.69)	73.26 (5.25)	72.35 (5.11)	71.45 (4.95)	68.62 (4.49)	63.57 (3.68)	58.52 (3.11)	73.20 (5.15)	72.14 (5.00)	71.06 (4.87)	68.74 (4.74)	64.58 (4.05)	58.63 (3.37)
	Intrinsic	74.59 (5.73)	73.76 (5.62)	72.75 (5.46)	69.96 (5.05)	64.69 (3.85)	58.11 (2.69)	76.21 (5.61)	74.06 (5.53)	72.57 (5.26)	67.39 (4.75)	61.50 (3.16)	55.63 (1.86)	75.33 (5.61)	73.86 (5.59)	72.28 (5.60)	68.65 (5.15)	62.32 (4.04)	56.07 (2.33)
FTTransformer	Constant	87.60 (3.10)	85.63 (2.69)	84.61 (2.67)	82.17 (2.20)	76.32 (1.74)	68.36 (1.49)	81.37 (6.17)	80.82 (6.04)	80.39 (5.89)	78.63 (5.36)	74.10 (4.31)	67.44 (2.79)	79.52 (7.84)	79.02 (7.71)	78.55 (7.55)	76.69 (7.07)	72.98 (5.99)	66.98 (4.33)
	KNN	85.52 (3.75)	83.83 (3.20)	83.36 (2.92)	81.84 (2.43)	78.65 (1.63)	72.87 (0.96)	87.04 (2.85)	86.49 (2.75)	86.07 (2.62)	84.49 (2.25)	80.39 (2.01)	72.94 (1.94)	85.09 (3.24)	84.68 (3.10)	84.21 (2.92)	82.24 (2.55)	78.27 (1.73)	70.80 (0.90)
	MICE	85.79 (3.57)	83.99 (3.01)	83.40 (2.84)	81.60 (2.41)	77.06 (1.44)	70.36 (0.34)	81.43 (6.14)	81.02 (6.02)	80.69 (5.88)	79.15 (5.46)	75.37 (4.61)	69.49 (3.24)	84.91 (3.36)	84.50 (3.24)	84.13 (3.02)	82.52 (2.69)	78.85 (1.93)	73.31 (0.59)
GRAPE	Intrinsic	86.41 (2.74)	85.55 (2.71)	81.62 (2.54)	81.39 (2.73)	78.62 (3.30)	68.62 (2.85)	84.48 (3.00)	84.55 (2.83)	80.90 (3.72)	79.55 (3.01)	75.85 (2.72)	69.02 (2.59)	85.89 (2.81)	85.16 (4.12)	81.59 (3.02)	77.92 (3.54)	78.01 (2.36)	70.37 (2.93)
HistGradientBoost	Constant	88.81 (5.04)	86.19 (5.03)	84.53 (5.23)	80.67 (4.83)	71.97 (4.89)	63.87 (3.42)	87.73 (5.34)	87.26 (5.31)	86.47 (5.51)	84.51 (5.41)	79.59 (5.31)	71.01 (5.25)	88.04 (5.07)	87.00 (5.55)	86.74 (5.30)	84.88 (5.30)	79.89 (5.57)	71.99 (4.78)
	KNN	88.81 (5.04)	86.93 (4.85)	86.14 (4.81)	83.93 (4.59)	78.92 (5.01)	72.36 (5.03)	87.72 (5.26)	86.84 (5.52)	86.17 (5.49)	84.56 (5.09)	79.67 (5.26)	72.40 (4.33)	87.62 (5.24)	87.09 (5.27)	86.52 (5.16)	84.38 (5.20)	80.39 (4.44)	71.57 (4.01)
	MICE	88.81 (5.04)	86.39 (5.04)	85.50 (4.83)	82.42 (4.50)	74.61 (4.96)	66.24 (4.12)	88.03 (4.91)	87.26 (5.12)	86.67 (5.13)	84.70 (5.13)	79.55 (5.26)	70.86 (5.68)	87.61 (5.25)	87.04 (5.32)	86.46 (5.31)	84.83 (5.22)	80.21 (5.23)	72.59 (4.70)
	Intrinsic	89.51 (5.10)	88.64 (5.29)	87.63 (5.30)	85.74 (4.80)	79.09 (6.10)	69.95 (6.41)	89.19 (5.46)	88.93 (5.21)	88.33 (5.25)	86.36 (5.25)	81.65 (4.79)	73.11 (4.61)	89.62 (5.04)	88.92 (5.30)	88.74 (5.00)	86.62 (5.36)	82.01 (5.53)	74.59 (4.44)
MLP	Constant	84.09 (5.90)	82.92 (6.21)	81.85 (6.11)	79.49 (5.31)	73.33 (4.62)	64.83 (4.11)	82.86 (6.28)	82.05 (6.43)	81.77 (5.99)	79.35 (5.88)	74.67 (5.17)	66.58 (4.97)	83.14 (5.84)	82.57 (5.82)	81.59 (5.97)	79.17 (5.93)	73.90 (6.07)	66.67 (4.76)
	KNN	84.12 (5.91)	83.71 (5.84)	83.07 (5.83)	82.15 (5.11)	78.84 (5.12)	73.49 (4.27)	83.08 (5.92)	82.56 (5.95)	81.89 (5.91)	81.22 (5.00)	77.03 (5.05)	69.96 (4.21)	82.81 (5.60)	82.58 (5.33)	81.62 (5.51)	80.11 (5.19)	75.95 (5.05)	68.12 (4.41)
	MICE	84.12 (5.91)	83.27 (6.06)	82.46 (5.84)	80.02 (5.19)	74.13 (4.83)	66.43 (4.94)	83.23 (5.97)	82.79 (5.93)	82.22 (5.89)	80.51 (5.88)	76.37 (5.84)	69.67 (5.81)	83.01 (5.84)	82.74 (5.65)	82.24 (5.64)	79.95 (6.10)	76.47 (5.85)	70.09 (5.09)
Random Forest	Constant	89.80 (4.12)	88.95 (4.16)	87.76 (4.36)	84.92 (3.86)	76.40 (4.90)	67.86 (3.36)	89.68 (4.19)	89.03 (4.30)	88.69 (4.15)	86.72 (4.21)	81.64 (4.51)	73.43 (4.40)	89.22 (4.51)	88.80 (4.45)	88.03 (4.69)	87.26 (3.90)	82.17 (4.48)	73.97 (4.16)
	KNN	89.82 (4.12)	89.39 (4.02)	88.78 (3.91)	86.90 (3.63)	82.92 (3.81)	76.71 (3.81)	89.54 (4.26)	88.99 (4.18)	88.50 (4.07)	86.74 (3.85)	82.26 (3.99)	74.87 (3.76)	89.57 (4.15)	89.06 (4.11)	88.40 (4.11)	86.51 (3.90)	82.12 (3.65)	73.73 (3.39)
	MICE	89.80 (4.12)	89.20 (4.10)	88.46 (4.03)	85.93 (3.82)	79.75 (3.98)	70.17 (3.98)	89.44 (4.37)	88.93 (4.36)	88.70 (4.07)	86.65 (4.19)	81.86 (4.26)	74.41 (4.14)	89.49 (4.23)	89.05 (4.19)	88.35 (4.33)	86.91 (3.98)	82.50 (4.08)	74.69 (3.93)
	Intrinsic	89.88 (4.41)	89.28 (4.48)	88.49 (4.34)	86.76 (4.00)	82.38 (3.92)	74.42 (3.99)	90.15 (4.21)	89.76 (4.18)	89.34 (4.13)	87.32 (4.54)	82.91 (4.64)	75.35 (4.44)	90.23 (4.11)	89.73 (4.20)	89.14 (4.27)	87.62 (4.24)	83.56 (4.40)	75.51 (4.28)
SVM	Constant	87.07 (4.12)	86.49 (4.19)	85.74 (4.15)	84.19 (3.79)	79.64 (3.83)	72.00 (3.93)	86.96 (4.12)	86.49 (4.14)	86.16 (4.07)	84.91 (3.81)	80.89 (3.90)	74.31 (3.94)	87.05 (3.98)	86.70 (3.95)	86.37 (3.86)	84.58 (4.02)	81.13 (3.89)	74.85 (3.27)
	KNN	87.08 (4.12)	86.74 (4.08)	86.33 (4.02)	85.03 (3.77)	82.29 (3.51)	76.24 (3.52)	86.88 (4.19)	86.42 (4.17)	86.12 (4.07)	85.06 (3.63)	81.33 (3.79)	74.21 (3.65)	87.19 (3.90)	86.89 (3.80)	86.51 (3.69)	84.84 (3.76)	81.26 (3.49)	73.82 (3.11)
	MICE	87.07 (4.12)	86.61 (4.08)	85.97 (3.98)	84.06 (3.72)	79.25 (3.80)	72.02 (3.94)	86.90 (4.19)	86.54 (4.14)	86.45 (3.94)	85.34 (3.65)	81.82 (3.69)	75.90 (3.62)	87.03 (4.04)	86.80 (3.94)	86.44 (3.88)	84.99 (3.88)	81.94 (3.67)	**76.49 (2.99)**
TabNet	Constant	88.66 (4.24)	86.40 (4.09)	85.80 (3.84)	82.49 (4.09)	76.44 (3.41)	68.25 (2.80)	88.07 (4.15)	87.43 (4.20)	86.75 (4.19)	84.76 (3.98)	79.14 (4.07)	70.42 (4.29)	88.00 (4.27)	87.43 (4.28)	86.56 (4.46)	84.67 (4.30)	79.79 (4.32)	71.64 (3.74)
	KNN	88.68 (4.24)	86.47 (4.24)	86.22 (3.92)	84.58 (3.66)	81.18 (3.57)	74.51 (3.92)	87.88 (4.49)	87.22 (4.50)	86.42 (4.75)	85.34 (3.75)	80.82 (4.09)	73.29 (3.73)	88.09 (4.12)	87.62 (4.11)	87.01 (3.98)	85.12 (3.76)	80.35 (3.81)	72.23 (3.26)
	MICE	88.66 (4.24)	86.71 (4.01)	86.24 (3.87)	83.84 (4.04)	78.51 (3.95)	71.79 (3.37)	87.62 (4.71)	87.22 (4.60)	86.62 (4.66)	85.32 (4.06)	81.36 (3.76)	74.01 (4.09)	88.27 (3.95)	87.83 (3.91)	87.20 (3.94)	85.37 (4.01)	81.62 (3.79)	75.56 (2.99)
TabTransformer	Constant	88.34 (4.36)	85.80 (4.35)	84.19 (4.63)	81.06 (4.22)	73.93 (4.06)	65.11 (3.89)	87.72 (4.48)	86.74 (4.64)	86.13 (4.46)	83.36 (4.41)	76.91 (4.18)	67.78 (4.25)	87.96 (4.08)	87.34 (3.97)	86.45 (4.02)	83.94 (4.07)	78.21 (3.98)	69.32 (3.62)
	KNN	88.42 (4.37)	86.48 (4.07)	85.74 (4.07)	84.08 (3.75)	79.95 (3.71)	72.97 (3.94)	87.98 (4.27)	87.33 (4.25)	86.80 (4.08)	85.10 (3.66)	80.25 (3.65)	71.57 (3.50)	87.70 (4.45)	87.34 (4.20)	86.57 (4.23)	84.38 (4.01)	80.27 (3.34)	71.11 (3.13)
	MICE	88.46 (4.26)	86.20 (4.15)	85.52 (4.02)	82.95 (3.72)	76.43 (3.81)	67.39 (4.26)	88.02 (4.28)	87.48 (4.14)	86.82 (4.11)	84.93 (3.77)	79.10 (3.97)	69.86 (4.61)	88.01 (4.07)	87.47 (3.97)	86.93 (3.90)	84.83 (3.93)	79.92 (3.94)	71.62 (3.93)
XGBoost	Constant	90.17 (4.06)	87.92 (3.84)	86.27 (3.95)	81.76 (3.71)	73.21 (3.61)	64.93 (2.85)	89.26 (4.34)	88.60 (4.51)	87.86 (4.70)	85.99 (4.49)	80.38 (4.85)	71.45 (4.51)	89.16 (4.41)	88.70 (4.41)	88.15 (4.39)	86.68 (4.12)	81.04 (4.88)	72.55 (4.59)
	KNN	90.17 (4.06)	88.56 (3.84)	87.63 (3.91)	85.48 (3.72)	80.22 (4.17)	73.74 (4.36)	89.58 (4.09)	89.05 (4.02)	88.30 (4.06)	86.53 (3.74)	81.53 (3.99)	73.70 (3.53)	89.50 (4.05)	89.05 (3.95)	88.39 (3.96)	86.32 (3.85)	81.67 (3.52)	72.62 (3.42)
	MICE	90.17 (4.06)	88.19 (3.80)	86.88 (3.88)	83.38 (3.66)	75.09 (3.96)	66.56 (3.26)	89.41 (4.11)	88.73 (4.23)	88.19 (4.19)	86.01 (4.28)	80.71 (4.36)	72.80 (4.25)	89.51 (3.90)	89.04 (3.85)	88.42 (3.92)	86.37 (4.15)	81.73 (4.12)	73.47 (3.88)
	Intrinsic	**90.86 (4.08)**	87.47 (4.43)	84.90 (3.79)	75.41 (3.14)	64.81 (1.31)	56.63 (1.61)	90.31 (4.54)	89.77 (4.58)	89.08 (4.70)	86.71 (5.01)	81.26 (4.83)	71.27 (4.63)	**90.64 (4.22)**	89.92 (4.48)	89.54 (4.35)	87.81 (4.32)	82.78 (4.73)	74.91 (3.60)

		Train missing percentage: 25%						Train missing percentage: 50%						Train missing percentage: 75%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
Model	**Strategy**	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	**90.69 (3.43)**	**90.31 (3.47)**	**90.02 (3.36)**	**88.90 (3.18)**	83.99 (4.12)	76.75 (3.97)	89.66 (3.79)	89.38 (3.76)	89.13 (3.69)	87.32 (4.14)	83.17 (4.79)	76.40 (4.34)	88.32 (4.06)	**87.89 (4.23)**	87.65 (4.17)	**85.94 (4.65)**	83.20 (4.34)	76.20 (5.04)
NAIM w/o regularization	Intrinsic	89.85 (4.08)	89.44 (4.06)	88.68 (4.16)	86.58 (4.20)	80.88 (4.24)	70.59 (3.61)	89.35 (4.03)	89.25 (3.75)	88.51 (4.04)	86.39 (4.34)	81.69 (4.65)	72.45 (4.48)	**88.61 (4.06)**	87.77 (4.50)	**88.01 (3.90)**	85.40 (4.97)	81.62 (5.23)	75.13 (4.66)
	Constant	88.26 (3.70)	87.92 (3.58)	87.20 (3.61)	85.18 (3.66)	80.90 (3.32)	72.30 (2.74)	87.67 (3.74)	87.24 (3.74)	86.52 (3.80)	84.69 (3.82)	80.59 (3.84)	74.47 (3.18)	85.63 (3.70)	85.16 (3.55)	84.27 (3.74)	82.42 (3.79)	78.87 (4.23)	72.26 (4.40)
	KNN	88.23 (3.72)	87.85 (3.57)	87.17 (3.54)	85.36 (3.26)	80.37 (2.90)	71.02 (2.35)	87.59 (3.59)	87.15 (3.54)	86.55 (3.48)	84.36 (3.52)	79.09 (3.44)	70.75 (2.12)	85.94 (3.25)	85.28 (3.27)	84.90 (3.15)	82.61 (3.11)	77.59 (3.19)	68.55 (3.08)
	MICE	88.08 (3.68)	87.58 (3.50)	86.72 (3.62)	85.46 (3.34)	82.22 (2.73)	74.87 (2.79)	87.29 (3.41)	86.27 (3.13)	85.32 (3.21)	84.31 (3.07)	81.25 (3.10)	75.27 (2.72)	85.78 (3.46)	84.46 (3.23)	83.69 (3.08)	81.76 (3.16)	78.96 (3.57)	73.76 (3.53)
Adaboost	Constant	88.07 (5.50)	87.61 (5.59)	87.24 (5.47)	86.34 (4.68)	82.92 (4.29)	75.34 (3.60)	88.00 (4.94)	87.70 (4.99)	87.42 (4.80)	86.42 (4.57)	82.82 (4.64)	76.84 (3.74)	85.33 (6.24)	85.25 (6.06)	85.05 (5.93)	83.71 (6.14)	81.50 (5.53)	76.73 (4.32)
	KNN	87.81 (5.59)	87.26 (5.66)	86.92 (5.35)	85.59 (4.67)	81.01 (4.26)	72.49 (2.75)	87.32 (4.94)	87.01 (4.86)	86.45 (4.90)	84.88 (4.72)	80.55 (4.43)	72.43 (2.65)	83.93 (6.18)	83.72 (5.97)	83.24 (6.06)	81.86 (5.73)	78.25 (5.41)	71.87 (3.58)
	MICE	88.70 (4.80)	87.99 (4.61)	87.24 (4.78)	85.91 (4.54)	81.89 (4.68)	75.05 (3.81)	88.30 (4.49)	86.43 (4.70)	85.76 (4.49)	84.66 (4.30)	81.95 (4.25)	75.99 (3.56)	86.07 (5.57)	83.11 (5.42)	81.67 (5.40)	81.02 (4.54)	78.92 (5.46)	75.64 (4.16)
Decision Tree	Constant	71.30 (5.14)	70.92 (4.87)	70.09 (5.08)	69.05 (4.86)	66.34 (3.99)	62.39 (2.35)	67.09 (4.00)	67.25 (4.05)	67.39 (3.95)	67.22 (4.07)	66.19 (3.94)	63.14 (3.37)	58.81 (2.89)	59.35 (2.86)	59.57 (2.92)	60.33 (3.08)	61.90 (3.40)	62.33 (3.40)
	KNN	71.69 (4.81)	70.97 (4.70)	70.20 (4.91)	68.46 (4.85)	64.98 (3.84)	58.63 (2.50)	68.40 (4.85)	68.11 (4.80)	67.60 (4.64)	65.91 (4.37)	62.46 (3.81)	58.04 (1.99)	63.59 (3.91)	63.31 (3.90)	62.78 (3.94)	61.55 (3.27)	59.44 (2.60)	56.84 (2.15)
	MICE	72.70 (4.91)	71.58 (4.32)	70.59 (4.41)	69.23 (3.91)	65.68 (3.80)	59.97 (3.13)	69.28 (4.52)	67.72 (4.02)	67.17 (3.98)	65.98 (3.70)	63.87 (3.91)	61.44 (3.22)	62.86 (2.86)	61.43 (2.53)	61.17 (1.97)	60.10 (1.97)	60.11 (2.55)	59.83 (3.52)
	Intrinsic	75.10 (4.69)	74.49 (4.65)	73.48 (4.78)	71.54 (4.71)	67.40 (4.26)	61.35 (2.84)	68.83 (4.57)	68.69 (4.66)	69.02 (4.49)	69.01 (4.13)	66.99 (4.11)	63.40 (3.33)	60.43 (3.90)	60.69 (3.96)	61.22 (3.74)	62.36 (3.70)	63.46 (4.41)	64.73 (3.35)
FTTransformer	Constant	79.26 (7.77)	79.04 (7.67)	78.23 (7.52)	76.87 (7.09)	73.77 (6.12)	66.92 (4.40)	77.20 (7.29)	76.88 (7.20)	76.52 (7.08)	75.00 (6.69)	71.85 (5.81)	67.01 (4.44)	64.30 (7.79)	64.22 (7.67)	63.80 (7.62)	63.20 (7.24)	61.43 (6.41)	59.15 (4.66)
	KNN	83.07 (4.47)	82.74 (4.33)	81.93 (4.24)	80.66 (3.71)	76.51 (2.67)	68.03 (1.78)	78.26 (7.56)	77.84 (7.45)	77.38 (7.29)	75.58 (6.76)	71.75 (5.72)	65.77 (4.11)	71.63 (7.72)	71.41 (7.64)	71.14 (7.51)	69.88 (7.12)	66.89 (6.16)	61.85 (4.57)
	MICE	84.74 (3.20)	83.93 (3.22)	83.35 (3.05)	82.71 (2.39)	79.51 (1.66)	72.63 (0.94)	84.21 (2.74)	81.94 (2.50)	81.62 (2.38)	81.18 (2.41)	78.55 (2.41)	73.68 (1.91)	70.22 (6.24)	69.58 (5.97)	68.83 (5.80)	67.85 (5.54)	66.01 (5.22)	62.76 (4.19)
GRAPE	Intrinsic	83.72 (2.24)	82.85 (3.39)	82.78 (2.25)	81.66 (2.60)	77.00 (2.66)	67.16 (3.73)	86.38 (2.66)	83.17 (2.20)	83.59 (2.86)	80.73 (2.33)	78.25 (3.29)	71.05 (2.72)	85.22 (3.19)	84.67 (3.67)	83.57 (3.80)	83.82 (3.81)	79.95 (3.84)	74.79 (4.59)
HistGradientBoost	Constant	88.15 (4.66)	87.82 (4.59)	87.44 (4.43)	85.58 (4.62)	81.11 (5.02)	74.10 (4.49)	87.26 (4.98)	86.82 (5.08)	86.49 (4.87)	85.02 (4.80)	81.02 (4.95)	74.20 (4.78)	86.46 (4.66)	86.02 (4.64)	85.47 (4.80)	83.89 (4.82)	79.55 (5.53)	73.56 (5.31)
	KNN	87.38 (4.87)	86.77 (4.97)	86.40 (4.73)	83.98 (5.14)	79.76 (4.60)	70.91 (3.75)	86.25 (4.88)	85.63 (5.01)	85.30 (4.78)	83.01 (5.09)	79.04 (4.65)	71.09 (3.49)	84.34 (4.56)	83.94 (4.52)	83.63 (4.44)	81.59 (4.54)	76.74 (4.87)	69.87 (4.19)
	MICE	88.30 (4.21)	87.47 (4.30)	87.28 (3.95)	85.54 (4.23)	81.60 (4.14)	74.16 (3.83)	86.76 (4.60)	85.87 (4.41)	85.03 (4.49)	83.77 (4.45)	81.22 (4.25)	74.50 (4.26)	85.28 (4.53)	84.11 (4.30)	83.31 (4.21)	81.68 (4.46)	79.46 (4.45)	74.16 (4.71)
	Intrinsic	89.62 (4.86)	89.29 (4.85)	89.15 (4.49)	87.44 (4.70)	83.51 (4.73)	75.72 (4.62)	89.57 (4.35)	89.22 (4.33)	88.64 (4.51)	87.71 (4.11)	83.95 (4.41)	76.54 (4.72)	87.10 (5.21)	86.77 (5.15)	86.19 (5.34)	84.94 (5.25)	82.27 (5.08)	76.93 (4.67)
MLP	Constant	81.00 (6.49)	80.50 (6.46)	79.97 (6.47)	78.03 (6.35)	74.90 (5.50)	68.11 (4.68)	80.62 (5.21)	80.05 (5.32)	79.69 (5.28)	77.96 (5.51)	74.79 (5.37)	69.60 (4.70)	77.90 (5.13)	77.54 (5.12)	76.93 (5.18)	75.71 (5.29)	74.52 (4.90)	71.56 (3.83)
	KNN	79.62 (6.91)	79.15 (6.84)	78.84 (6.70)	77.80 (5.91)	73.33 (5.37)	65.68 (4.11)	77.53 (6.14)	77.17 (6.16)	76.09 (6.62)	75.30 (5.77)	71.26 (5.00)	63.10 (4.09)	72.89 (6.84)	72.50 (6.82)	72.04 (6.82)	70.03 (7.02)	67.13 (5.76)	61.79 (4.45)
	MICE	80.54 (6.23)	78.74 (5.21)	78.81 (5.43)	78.41 (5.94)	75.68 (5.63)	69.91 (4.82)	76.97 (5.87)	72.94 (4.27)	73.67 (4.42)	74.22 (5.65)	73.84 (5.43)	70.71 (4.34)	74.07 (5.58)	68.94 (4.24)	68.41 (3.80)	68.31 (4.57)	69.72 (5.61)	69.66 (4.24)
Random Forest	Constant	89.44 (4.05)	88.93 (4.09)	88.62 (3.93)	87.10 (3.82)	83.29 (3.96)	75.51 (3.59)	89.28 (3.65)	88.96 (3.54)	88.59 (3.51)	86.87 (3.68)	82.98 (3.71)	75.33 (3.77)	87.26 (4.44)	87.02 (4.36)	86.21 (4.58)	84.97 (4.19)	81.06 (4.32)	73.93 (4.02)
	KNN	89.45 (3.92)	89.09 (3.74)	88.40 (3.77)	86.58 (3.62)	81.78 (3.61)	73.41 (2.86)	88.50 (3.73)	87.89 (3.80)	87.58 (3.57)	86.11 (3.14)	81.35 (3.34)	72.58 (2.85)	85.74 (3.95)	85.28 (3.91)	84.61 (4.11)	83.07 (3.83)	78.92 (3.69)	70.40 (3.31)
	MICE	89.83 (3.52)	89.37 (3.47)	89.03 (3.20)	87.57 (3.13)	83.32 (3.47)	75.62 (3.36)	88.89 (3.44)	88.41 (3.22)	87.60 (3.41)	86.02 (3.52)	82.79 (3.27)	75.58 (3.47)	86.98 (3.60)	85.73 (3.73)	85.49 (3.35)	83.51 (3.62)	80.10 (4.19)	73.70 (4.02)
	Intrinsic	90.23 (3.90)	89.83 (3.90)	89.47 (3.78)	87.97 (3.90)	**84.69 (3.52)**	**77.41 (3.48)**	89.49 (4.07)	89.32 (3.90)	88.73 (4.02)	87.87 (3.63)	84.42 (3.57)	77.08 (3.88)	87.76 (4.82)	87.63 (4.53)	86.97 (4.73)	85.85 (4.58)	82.73 (4.45)	76.67 (4.06)
SVM	Constant	86.61 (3.97)	86.46 (3.85)	85.84 (3.99)	84.70 (3.86)	81.78 (3.68)	74.85 (3.69)	85.23 (3.99)	85.15 (3.88)	84.84 (3.87)	83.81 (3.84)	80.95 (3.83)	75.92 (3.32)	83.09 (3.59)	82.78 (3.63)	82.31 (3.82)	81.55 (3.81)	79.80 (3.81)	74.48 (3.82)
	KNN	86.54 (4.12)	86.34 (4.04)	85.72 (4.08)	84.56 (3.85)	80.42 (3.80)	72.29 (2.99)	85.29 (4.02)	85.23 (3.85)	84.68 (3.91)	83.32 (3.81)	79.15 (3.69)	71.63 (2.87)	83.23 (3.84)	82.84 (3.86)	82.12 (4.16)	80.51 (4.14)	77.07 (3.96)	69.81 (3.41)
	MICE	86.55 (4.13)	85.95 (3.62)	85.47 (3.93)	85.07 (3.72)	82.19 (3.54)	76.04 (3.37)	85.23 (4.16)	83.73 (3.35)	83.56 (3.55)	83.35 (3.72)	81.47 (3.56)	76.70 (3.05)	82.51 (4.10)	79.00 (3.59)	77.91 (3.51)	79.05 (3.55)	78.36 (4.09)	74.80 (3.60)
TabNet	Constant	87.05 (4.44)	86.71 (4.29)	86.11 (4.37)	84.22 (4.36)	80.22 (4.06)	71.91 (3.92)	85.47 (5.23)	85.26 (5.03)	84.79 (4.99)	83.23 (4.87)	79.50 (4.59)	72.74 (4.12)	83.82 (4.57)	83.23 (4.60)	82.20 (5.13)	80.26 (5.46)	77.58 (4.46)	71.88 (3.45)
	KNN	87.46 (4.34)	86.99 (4.24)	86.13 (4.43)	84.74 (3.94)	79.82 (3.87)	69.99 (3.87)	86.25 (4.36)	85.57 (4.48)	85.11 (4.33)	83.20 (3.95)	77.99 (3.95)	70.84 (2.41)	83.79 (4.59)	82.98 (4.66)	82.79 (4.39)	80.75 (3.93)	75.17 (4.13)	67.62 (3.32)
	MICE	87.26 (4.49)	86.14 (3.93)	85.64 (4.06)	84.96 (3.96)	81.47 (3.90)	74.92 (3.36)	85.86 (4.65)	83.59 (3.80)	82.83 (4.25)	82.59 (4.19)	80.81 (3.49)	75.80 (2.88)	83.04 (5.24)	78.77 (4.53)	77.79 (3.91)	78.05 (3.57)	77.74 (3.94)	73.83 (3.15)
TabTransformer	Constant	87.28 (4.03)	86.88 (3.92)	85.76 (4.20)	83.76 (4.19)	78.81 (4.12)	70.11 (3.91)	86.50 (3.83)	86.15 (3.77)	85.72 (3.65)	83.40 (3.95)	79.42 (3.65)	72.05 (3.51)	85.24 (3.60)	84.79 (3.56)	84.16 (3.64)	82.65 (3.62)	79.19 (3.79)	72.92 (3.53)
	KNN	86.98 (4.55)	86.63 (4.36)	85.88 (4.32)	84.12 (4.00)	78.88 (3.75)	69.72 (3.08)	85.95 (3.78)	85.34 (3.90)	84.92 (3.56)	83.05 (3.30)	77.41 (3.60)	69.44 (2.06)	83.22 (3.85)	82.76 (3.68)	82.25 (3.74)	80.34 (3.22)	75.47 (3.24)	67.61 (2.80)
	MICE	87.02 (4.08)	85.54 (3.51)	85.16 (3.69)	84.14 (3.96)	80.40 (3.87)	72.61 (3.77)	85.50 (3.53)	81.77 (2.51)	81.62 (2.53)	81.55 (3.53)	79.51 (3.72)	73.59 (3.19)	83.64 (3.96)	78.17 (2.23)	76.76 (1.98)	77.55 (2.56)	77.42 (3.65)	73.41 (3.41)
XGBoost	Constant	89.43 (4.03)	88.98 (4.07)	88.53 (4.02)	86.70 (4.15)	82.34 (4.59)	76.02 (3.48)	88.44 (4.21)	88.18 (4.15)	87.79 (4.10)	86.64 (3.82)	83.43 (3.65)	76.70 (3.67)	86.41 (4.33)	86.12 (4.30)	85.80 (4.30)	84.84 (4.14)	81.96 (4.28)	76.20 (4.06)
	KNN	89.20 (3.98)	88.73 (3.91)	88.15 (3.94)	85.77 (4.21)	80.87 (4.03)	71.64 (3.08)	87.96 (3.97)	87.55 (3.90)	86.88 (3.97)	84.76 (4.08)	80.15 (3.95)	71.66 (2.97)	86.21 (3.68)	85.88 (3.51)	85.16 (3.67)	83.16 (3.68)	78.41 (3.81)	71.26 (3.23)
	MICE	89.09 (3.94)	88.45 (3.87)	87.90 (3.80)	86.11 (4.04)	82.10 (4.11)	74.48 (3.71)	88.00 (3.86)	87.03 (3.73)	86.26 (3.65)	85.34 (3.42)	81.81 (3.99)	76.01 (3.27)	86.24 (3.94)	84.76 (3.81)	83.81 (3.72)	82.44 (3.64)	79.18 (4.75)	76.03 (3.59)
	Intrinsic	90.33 (4.37)	90.01 (4.35)	89.64 (4.19)	88.15 (4.21)	84.15 (4.40)	76.33 (4.18)	**90.01 (4.00)**	**89.91 (3.83)**	**89.34 (3.98)**	**88.30 (3.79)**	**85.44 (3.46)**	**78.10 (3.81)**	87.11 (4.82)	86.86 (4.83)	86.35 (5.00)	85.70 (4.70)	**83.28 (4.57)**	**78.68 (3.67)**

Table C.1: Average AUC performance and standard error (in brackets) of the experiments across the 5 different datasets. To facilitate the analysis we highlighted the best performance in each column in bold.

		Train missing percentage: 0%						Train missing percentage: 5%						Train missing percentage: 10%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
Model	**Strategy**	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	91.60 (0.21)	91.30 (0.20)	90.98 (0.19)	**89.58 (0.14)**	**86.19 (0.17)**	**79.02 (0.25)**	91.61 (0.17)	91.29 (0.14)	90.93 (0.18)	89.68 (0.17)	**86.30 (0.23)**	**79.19 (0.25)**	91.50 (0.19)	91.27 (0.20)	90.97 (0.21)	89.69 (0.21)	86.05 (0.18)	**79.18 (0.13)**
NAIM w/o regularization	Intrinsic	91.46 (0.17)	88.79 (0.42)	85.69 (0.71)	78.18 (1.38)	68.92 (0.59)	62.13 (1.21)	91.36 (0.25)	90.97 (0.28)	90.47 (0.26)	88.56 (0.33)	83.62 (0.57)	74.29 (0.80)	91.32 (0.12)	90.94 (0.08)	90.40 (0.09)	88.44 (0.20)	83.65 (0.67)	74.88 (0.95)
	Constant	86.15 (0.30)	85.29 (0.23)	84.02 (0.25)	80.70 (0.15)	73.27 (0.30)	64.34 (0.57)	86.23 (0.26)	85.60 (0.27)	84.68 (0.24)	81.66 (0.40)	75.30 (0.42)	65.85 (0.36)	86.03 (0.24)	85.28 (0.21)	84.58 (0.14)	81.74 (0.29)	76.38 (0.14)	67.38 (0.43)
	KNN	86.57 (0.24)	86.25 (0.22)	85.75 (0.29)	84.15 (0.12)	79.54 (0.19)	70.93 (0.21)	86.22 (0.35)	85.96 (0.35)	85.50 (0.34)	83.77 (0.31)	78.96 (0.36)	70.42 (0.15)	86.35 (0.18)	85.98 (0.17)	85.44 (0.13)	83.53 (0.22)	78.92 (0.29)	69.80 (0.16)
	MICE	86.21 (0.36)	85.50 (0.31)	84.53 (0.43)	81.61 (0.34)	74.24 (1.15)	65.46 (1.76)	86.34 (0.23)	85.93 (0.24)	85.35 (0.27)	82.92 (0.42)	78.08 (0.52)	70.86 (0.76)	86.09 (0.25)	85.57 (0.23)	85.10 (0.22)	83.05 (0.38)	78.89 (0.25)	72.30 (0.16)
Adaboost	Constant	91.91 (0.14)	91.35 (0.11)	90.51 (0.15)	87.86 (0.19)	81.93 (0.14)	72.31 (0.35)	91.85 (0.14)	91.45 (0.13)	90.93 (0.13)	88.98 (0.17)	84.17 (0.32)	75.54 (0.17)	91.81 (0.14)	91.49 (0.13)	90.95 (0.11)	89.27 (0.09)	85.16 (0.13)	76.45 (0.37)
	KNN	91.88 (0.12)	91.53 (0.12)	91.06 (0.13)	89.54 (0.18)	84.84 (0.27)	75.89 (0.17)	91.80 (0.14)	91.46 (0.14)	91.09 (0.13)	89.45 (0.21)	84.70 (0.31)	74.79 (0.04)	91.78 (0.14)	91.51 (0.13)	90.99 (0.10)	89.40 (0.19)	84.19 (0.23)	74.67 (0.09)
	MICE	91.91 (0.14)	91.30 (0.16)	90.32 (0.26)	87.53 (0.62)	81.27 (1.60)	72.88 (1.89)	91.76 (0.14)	91.33 (0.14)	90.82 (0.12)	88.96 (0.17)	84.43 (0.29)	76.35 (0.10)	91.67 (0.15)	91.33 (0.15)	90.76 (0.14)	89.12 (0.11)	85.15 (0.17)	77.15 (0.26)
Decision Tree	Constant	74.49 (0.14)	73.80 (0.06)	72.90 (0.18)	70.08 (0.20)	65.30 (0.17)	58.38 (0.46)	74.72 (0.19)	73.96 (0.16)	73.26 (0.17)	70.51 (0.36)	65.12 (0.35)	57.94 (0.48)	75.20 (0.25)	74.55 (0.21)	73.60 (0.19)	71.59 (0.25)	66.86 (0.36)	59.41 (0.59)
	KNN	74.80 (0.18)	74.45 (0.19)	74.12 (0.30)	72.75 (0.20)	69.44 (0.20)	62.73 (0.39)	75.01 (0.24)	74.51 (0.27)	74.02 (0.19)	72.06 (0.38)	68.11 (0.43)	61.02 (0.28)	74.58 (0.30)	74.06 (0.27)	73.63 (0.23)	71.93 (0.25)	66.93 (0.25)	60.77 (0.21)
	MICE	74.56 (0.11)	74.06 (0.06)	73.16 (0.14)	70.05 (0.27)	64.14 (1.12)	58.75 (1.09)	74.76 (0.21)	74.18 (0.12)	73.62 (0.21)	70.89 (0.18)	65.94 (0.59)	59.45 (0.46)	74.67 (0.37)	74.09 (0.28)	73.27 (0.47)	71.61 (0.28)	66.60 (0.39)	59.43 (0.21)
	Intrinsic	81.23 (0.39)	79.77 (0.40)	77.59 (0.55)	72.55 (1.00)	64.28 (1.40)	56.97 (0.99)	77.57 (0.42)	76.26 (0.39)	74.90 (0.48)	70.24 (0.16)	63.97 (0.60)	57.91 (0.81)	77.80 (0.20)	76.89 (0.25)	75.78 (0.11)	72.18 (0.21)	65.26 (0.45)	57.52 (0.43)
FTTransformer	Constant	87.38 (0.18)	86.66 (0.12)	85.58 (0.18)	82.81 (0.15)	76.06 (0.51)	67.42 (0.23)	87.38 (0.20)	86.82 (0.19)	85.97 (0.17)	83.27 (0.36)	77.16 (0.38)	67.94 (0.30)	87.31 (0.19)	86.64 (0.18)	85.95 (0.13)	83.32 (0.21)	78.03 (0.16)	69.18 (0.14)
	KNN	87.34 (0.21)	87.05 (0.20)	86.62 (0.25)	85.30 (0.08)	81.03 (0.17)	73.15 (0.16)	87.23 (0.09)	87.01 (0.10)	86.51 (0.12)	84.91 (0.19)	80.19 (0.27)	71.40 (0.20)	87.01 (0.21)	86.67 (0.21)	86.21 (0.16)	84.33 (0.23)	79.55 (0.24)	71.27 (0.34)
	MICE	87.39 (0.24)	86.78 (0.19)	85.98 (0.29)	83.59 (0.15)	77.69 (0.38)	70.42 (0.49)	87.27 (0.16)	86.82 (0.15)	86.11 (0.17)	83.62 (0.26)	78.63 (0.30)	71.45 (0.35)	87.30 (0.15)	86.72 (0.14)	86.17 (0.10)	83.77 (0.23)	79.10 (0.25)	71.90 (0.26)
GRAPE	Intrinsic	90.57 (0.19)	89.67 (0.21)	89.58 (0.21)	88.17 (0.16)	84.20 (0.27)	75.42 (0.46)	89.97 (0.22)	89.54 (0.24)	89.23 (0.35)	87.64 (0.27)	84.22 (0.37)	75.49 (0.48)	89.78 (0.27)	89.51 (0.25)	89.20 (0.25)	87.70 (0.23)	84.33 (0.04)	75.91 (0.39)
HistGradientBoost	Constant	89.27 (0.14)	87.29 (0.14)	84.63 (0.18)	77.35 (0.55)	67.94 (0.38)	61.01 (0.25)	89.14 (0.17)	88.58 (0.15)	87.75 (0.13)	84.84 (0.36)	78.45 (0.37)	68.28 (0.40)	88.99 (0.14)	88.35 (0.15)	87.67 (0.07)	85.09 (0.17)	79.57 (0.14)	70.27 (0.20)
	KNN	89.27 (0.14)	88.34 (0.11)	87.13 (0.23)	83.38 (0.11)	75.81 (0.21)	65.75 (0.39)	89.03 (0.17)	88.66 (0.16)	88.06 (0.18)	86.05 (0.21)	80.74 (0.38)	70.97 (0.14)	88.57 (0.13)	88.20 (0.14)	87.62 (0.13)	85.65 (0.18)	80.29 (0.10)	70.93 (0.30)
	MICE	89.27 (0.14)	87.29 (0.10)	84.83 (0.13)	77.58 (0.32)	67.23 (0.29)	59.59 (0.52)	88.35 (0.16)	87.82 (0.17)	86.95 (0.17)	84.13 (0.30)	78.79 (0.32)	69.28 (0.50)	88.07 (0.17)	87.52 (0.18)	87.00 (0.13)	84.85 (0.25)	80.19 (0.24)	71.94 (0.19)
	Intrinsic	**92.79 (0.13)**	**92.28 (0.08)**	**91.57 (0.16)**	89.54 (0.15)	84.50 (0.34)	76.20 (0.44)	**92.77 (0.12)**	**92.40 (0.13)**	**92.01 (0.12)**	**90.33 (0.13)**	85.79 (0.29)	76.56 (0.51)	**92.71 (0.13)**	**92.47 (0.11)**	**92.07 (0.09)**	**90.57 (0.10)**	86.74 (0.12)	78.32 (0.35)
MLP	Constant	87.45 (0.13)	86.48 (0.12)	85.05 (0.13)	80.96 (0.40)	73.49 (0.76)	65.58 (0.93)	87.61 (0.13)	87.17 (0.10)	86.36 (0.14)	84.13 (0.17)	79.20 (0.36)	71.28 (0.51)	87.25 (0.24)	86.93 (0.21)	86.46 (0.17)	84.36 (0.17)	79.93 (0.38)	72.08 (0.67)
	KNN	87.59 (0.25)	87.29 (0.28)	87.11 (0.23)	86.08 (0.19)	82.15 (0.42)	74.01 (0.16)	87.10 (0.10)	86.88 (0.14)	86.41 (0.17)	85.02 (0.14)	80.15 (0.25)	71.23 (0.20)	86.56 (0.10)	86.28 (0.12)	85.84 (0.12)	84.27 (0.17)	79.47 (0.13)	70.45 (0.24)
	MICE	87.58 (0.13)	86.65 (0.20)	84.87 (0.67)	79.96 (2.01)	72.04 (3.90)	65.24 (3.36)	87.47 (0.10)	86.98 (0.07)	86.44 (0.11)	84.58 (0.10)	80.72 (0.21)	73.77 (0.34)	87.12 (0.16)	86.92 (0.15)	86.40 (0.17)	84.62 (0.18)	80.57 (0.44)	73.24 (0.60)
Random Forest	Constant	90.05 (0.11)	89.51 (0.10)	88.75 (0.15)	86.16 (0.24)	80.23 (0.21)	70.08 (0.50)	90.20 (0.13)	89.95 (0.15)	89.46 (0.13)	87.73 (0.18)	83.09 (0.38)	73.45 (0.30)	90.17 (0.16)	89.87 (0.13)	89.57 (0.13)	87.85 (0.15)	83.08 (0.16)	73.37 (0.99)
	KNN	90.15 (0.10)	89.85 (0.12)	89.48 (0.15)	87.99 (0.11)	83.71 (0.34)	75.25 (0.29)	90.09 (0.08)	89.79 (0.09)	89.41 (0.13)	87.97 (0.18)	83.57 (0.21)	74.15 (0.15)	90.06 (0.13)	89.78 (0.13)	89.44 (0.08)	87.83 (0.15)	82.87 (0.17)	73.50 (0.31)
	MICE	90.06 (0.11)	89.62 (0.09)	88.89 (0.17)	86.35 (0.23)	80.71 (0.73)	70.36 (0.65)	90.12 (0.12)	89.89 (0.13)	89.48 (0.13)	87.92 (0.17)	83.70 (0.37)	74.42 (0.44)	90.11 (0.13)	89.81 (0.09)	89.67 (0.09)	88.08 (0.13)	83.94 (0.09)	74.98 (0.80)
	Intrinsic	91.17 (0.13)	90.81 (0.11)	90.21 (0.13)	88.71 (0.11)	84.37 (0.27)	76.93 (0.30)	91.46 (0.13)	91.17 (0.13)	90.68 (0.14)	89.14 (0.13)	85.13 (0.25)	76.74 (0.20)	91.43 (0.15)	91.12 (0.12)	90.74 (0.12)	89.22 (0.14)	85.24 (0.21)	76.89 (0.17)
SVM	Constant	89.41 (0.22)	88.66 (0.18)	87.52 (0.17)	84.62 (0.20)	78.69 (0.17)	70.82 (0.26)	89.41 (0.21)	89.10 (0.21)	88.78 (0.21)	87.18 (0.21)	82.90 (0.22)	74.66 (0.19)	89.34 (0.21)	89.15 (0.18)	88.80 (0.18)	87.29 (0.21)	82.96 (0.32)	74.60 (0.33)
	KNN	89.46 (0.21)	89.24 (0.21)	89.00 (0.21)	87.89 (0.20)	84.12 (0.35)	75.71 (0.22)	89.37 (0.21)	89.17 (0.22)	88.91 (0.20)	87.53 (0.24)	83.48 (0.30)	74.33 (0.18)	89.25 (0.23)	89.07 (0.21)	88.70 (0.17)	87.30 (0.25)	82.55 (0.24)	73.26 (0.15)
	MICE	89.41 (0.22)	88.50 (0.25)	87.09 (0.59)	82.81 (1.86)	74.55 (3.90)	67.70 (3.69)	89.41 (0.21)	89.09 (0.22)	88.83 (0.21)	87.31 (0.21)	83.40 (0.25)	76.11 (0.18)	89.32 (0.21)	89.15 (0.18)	88.82 (0.18)	87.35 (0.21)	83.42 (0.33)	76.03 (0.36)
TabNet	Constant	86.48 (0.20)	85.62 (0.15)	84.45 (0.18)	81.38 (0.19)	74.25 (0.30)	65.91 (0.32)	86.31 (0.15)	85.72 (0.14)	84.87 (0.13)	82.00 (0.27)	75.99 (0.30)	67.08 (0.27)	86.25 (0.21)	85.56 (0.19)	84.88 (0.12)	82.21 (0.29)	77.12 (0.28)	68.49 (0.25)
	KNN	86.61 (0.24)	86.35 (0.23)	86.00 (0.28)	84.65 (0.10)	80.37 (0.14)	72.21 (0.22)	86.46 (0.21)	86.29 (0.20)	85.85 (0.21)	84.27 (0.23)	79.54 (0.43)	70.65 (0.25)	86.25 (0.28)	85.95 (0.27)	85.48 (0.22)	83.68 (0.30)	78.92 (0.34)	70.50 (0.29)
	MICE	86.48 (0.20)	85.77 (0.16)	84.89 (0.24)	82.23 (0.20)	75.26 (0.33)	66.93 (0.35)	86.34 (0.19)	85.88 (0.17)	85.11 (0.21)	82.60 (0.26)	77.45 (0.27)	69.46 (0.44)	86.13 (0.17)	85.50 (0.17)	85.02 (0.11)	82.60 (0.24)	78.03 (0.17)	70.83 (0.20)
TabTransformer	Constant	85.89 (0.18)	85.05 (0.13)	83.78 (0.23)	80.55 (0.16)	72.58 (0.42)	63.13 (0.26)	85.77 (0.20)	85.13 (0.19)	84.15 (0.17)	81.01 (0.27)	74.25 (0.38)	64.65 (0.28)	85.61 (0.21)	84.85 (0.21)	84.08 (0.13)	81.04 (0.33)	75.15 (0.21)	66.75 (0.34)
	KNN	85.92 (0.18)	85.65 (0.19)	85.29 (0.21)	83.91 (0.13)	79.39 (0.17)	70.76 (0.17)	85.88 (0.17)	85.71 (0.17)	85.18 (0.17)	83.57 (0.15)	78.33 (0.39)	68.71 (0.31)	85.79 (0.21)	85.46 (0.21)	84.94 (0.15)	82.86 (0.33)	77.54 (0.18)	68.80 (0.29)
	MICE	85.90 (0.17)	85.19 (0.13)	84.26 (0.25)	81.18 (0.15)	73.12 (0.38)	63.14 (0.21)	85.81 (0.16)	85.32 (0.14)	84.48 (0.18)	81.52 (0.30)	75.02 (0.47)	64.94 (0.59)	85.69 (0.22)	85.06 (0.21)	84.45 (0.18)	81.69 (0.36)	75.77 (0.34)	67.16 (0.55)
XGBoost	Constant	89.34 (0.13)	87.61 (0.13)	85.30 (0.22)	78.84 (0.48)	69.71 (0.45)	62.14 (0.32)	89.19 (0.15)	88.64 (0.13)	87.77 (0.11)	84.95 (0.32)	78.51 (0.35)	68.37 (0.44)	89.14 (0.15)	88.51 (0.15)	87.81 (0.07)	85.18 (0.19)	79.65 (0.16)	70.40 (0.22)
	KNN	89.33 (0.14)	88.48 (0.10)	87.48 (0.20)	84.13 (0.13)	77.00 (0.19)	67.10 (0.61)	89.26 (0.13)	88.91 (0.11)	88.27 (0.14)	86.38 (0.19)	80.91 (0.37)	71.21 (0.12)	88.99 (0.15)	88.60 (0.17)	88.00 (0.13)	86.00 (0.21)	80.66 (0.14)	71.19 (0.23)
	MICE	89.34 (0.13)	87.68 (0.13)	85.59 (0.23)	79.32 (0.51)	69.87 (0.50)	61.77 (0.48)	88.67 (0.15)	88.14 (0.13)	87.36 (0.15)	84.58 (0.23)	78.99 (0.23)	69.87 (0.36)	88.49 (0.17)	87.94 (0.16)	87.42 (0.13)	85.24 (0.25)	80.50 (0.27)	72.29 (0.24)
	Intrinsic	92.78 (0.11)	89.51 (0.18)	85.48 (0.20)	75.34 (0.24)	62.11 (0.32)	53.94 (0.32)	92.75 (0.12)	92.38 (0.12)	91.96 (0.12)	90.32 (0.14)	85.55 (0.34)	76.07 (0.69)	92.70 (0.09)	92.45 (0.08)	92.04 (0.07)	90.49 (0.11)	**86.76 (0.12)**	78.50 (0.20)

		Train missing percentage: 25%						Train missing percentage: 50%						Train missing percentage: 75%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
Model	**Strategy**	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	91.44 (0.18)	91.23 (0.17)	90.91 (0.17)	89.65 (0.14)	86.27 (0.31)	79.36 (0.29)	91.00 (0.25)	90.77 (0.27)	90.41 (0.25)	89.44 (0.26)	86.27 (0.21)	79.46 (0.24)	90.50 (0.16)	90.28 (0.17)	89.97 (0.16)	88.79 (0.23)	85.88 (0.27)	79.56 (0.25)
NAIM w/o regularization	Intrinsic	91.31 (0.22)	91.04 (0.23)	90.68 (0.21)	89.34 (0.27)	85.65 (0.27)	77.90 (0.36)	90.95 (0.21)	90.68 (0.22)	90.23 (0.25)	88.86 (0.29)	85.02 (0.63)	77.22 (0.92)	90.49 (0.19)	90.24 (0.20)	89.84 (0.23)	88.63 (0.23)	85.53 (0.35)	78.95 (0.39)
	Constant	85.74 (0.30)	85.25 (0.30)	84.47 (0.31)	82.01 (0.37)	77.16 (0.42)	68.59 (0.26)	85.35 (0.24)	84.78 (0.24)	84.18 (0.28)	81.78 (0.26)	76.91 (0.24)	69.05 (0.41)	85.08 (0.23)	84.57 (0.22)	83.81 (0.25)	81.65 (0.26)	76.88 (0.26)	69.14 (0.37)
	KNN	85.78 (0.21)	85.46 (0.20)	84.96 (0.13)	83.02 (0.24)	78.27 (0.30)	70.22 (0.25)	85.16 (0.34)	84.65 (0.35)	84.12 (0.38)	81.93 (0.32)	77.13 (0.38)	69.81 (0.50)	84.57 (0.29)	84.03 (0.29)	83.33 (0.33)	81.08 (0.34)	75.96 (0.48)	69.08 (0.34)
	MICE	85.90 (0.23)	85.71 (0.24)	85.30 (0.16)	83.77 (0.25)	79.96 (0.36)	72.53 (0.20)	84.82 (0.37)	83.24 (0.32)	82.23 (0.38)	81.04 (0.19)	78.72 (0.27)	72.83 (0.34)	83.63 (0.51)	82.01 (0.61)	81.00 (0.72)	79.26 (0.92)	77.43 (0.67)	72.35 (0.97)
Adaboost	Constant	91.65 (0.17)	91.37 (0.17)	90.99 (0.18)	89.55 (0.17)	85.53 (0.20)	77.16 (0.28)	91.46 (0.16)	91.18 (0.16)	90.76 (0.19)	89.33 (0.10)	85.67 (0.29)	78.18 (0.28)	**91.15 (0.22)**	**90.87 (0.21)**	**90.48 (0.25)**	89.14 (0.20)	85.71 (0.34)	78.40 (0.31)
	KNN	91.52 (0.18)	91.26 (0.18)	90.85 (0.16)	89.22 (0.17)	84.66 (0.20)	74.95 (0.15)	91.33 (0.18)	90.99 (0.18)	90.52 (0.21)	88.86 (0.16)	84.32 (0.29)	75.22 (0.32)	90.40 (0.26)	89.99 (0.25)	89.60 (0.30)	87.87 (0.25)	83.55 (0.29)	75.03 (0.35)
	MICE	91.47 (0.17)	91.18 (0.16)	90.76 (0.17)	89.35 (0.16)	85.61 (0.17)	77.86 (0.28)	90.85 (0.18)	90.38 (0.17)	89.94 (0.21)	88.30 (0.15)	84.97 (0.24)	78.50 (0.30)	90.12 (0.24)	89.59 (0.21)	88.91 (0.22)	87.24 (0.20)	84.40 (0.31)	78.43 (0.40)
Decision Tree	Constant	73.95 (0.17)	73.56 (0.15)	73.04 (0.11)	71.55 (0.24)	68.07 (0.36)	62.88 (0.76)	70.87 (0.48)	70.82 (0.47)	70.61 (0.37)	69.90 (0.30)	68.55 (0.54)	66.38 (0.14)	66.76 (0.56)	66.69 (0.52)	66.55 (0.51)	66.22 (0.29)	66.49 (0.31)	66.93 (0.37)
	KNN	73.96 (0.30)	73.62 (0.28)	72.83 (0.29)	70.98 (0.27)	66.51 (0.40)	59.87 (0.29)	72.12 (0.39)	71.70 (0.31)	70.92 (0.30)	68.95 (0.34)	64.83 (0.30)	59.56 (0.24)	68.08 (0.17)	67.78 (0.28)	67.37 (0.18)	65.37 (0.22)	61.67 (0.36)	60.45 (0.21)
	MICE	74.07 (0.21)	73.55 (0.28)	73.20 (0.21)	71.35 (0.04)	67.23 (0.15)	60.71 (0.26)	69.34 (0.36)	68.96 (0.30)	68.84 (0.21)	68.12 (0.27)	65.44 (0.43)	61.45 (0.29)	63.59 (0.64)	63.45 (0.69)	63.29 (0.46)	62.73 (0.37)	62.71 (0.34)	63.42 (0.37)
	Intrinsic	76.38 (0.53)	76.10 (0.51)	75.54 (0.51)	73.51 (0.30)	69.75 (0.36)	63.66 (0.64)	72.63 (0.59)	72.41 (0.58)	72.12 (0.51)	71.15 (0.21)	70.37 (0.14)	67.51 (0.40)	67.30 (0.28)	67.23 (0.20)	67.15 (0.21)	67.01 (0.21)	68.26 (0.42)	68.96 (0.42)
FTTransformer	Constant	86.70 (0.14)	86.27 (0.13)	85.53 (0.16)	83.18 (0.23)	78.10 (0.30)	69.49 (0.14)	86.24 (0.19)	85.71 (0.17)	85.11 (0.20)	82.93 (0.19)	78.14 (0.27)	70.36 (0.33)	85.74 (0.24)	85.29 (0.25)	84.68 (0.27)	82.65 (0.27)	77.96 (0.28)	70.27 (0.50)
	KNN	87.06 (0.27)	86.74 (0.25)	86.17 (0.26)	84.11 (0.30)	79.36 (0.24)	71.03 (0.22)	86.33 (0.31)	85.87 (0.29)	85.14 (0.31)	82.92 (0.32)	78.03 (0.42)	70.32 (0.44)	85.59 (0.25)	85.10 (0.25)	84.34 (0.26)	82.27 (0.26)	77.20 (0.37)	69.87 (0.29)
	MICE	86.68 (0.19)	86.35 (0.21)	85.76 (0.16)	83.88 (0.26)	79.87 (0.24)	72.36 (0.10)	84.91 (0.16)	83.41 (0.16)	82.44 (0.25)	81.28 (0.19)	79.17 (0.24)	73.39 (0.30)	83.45 (0.75)	82.01 (0.89)	81.11 (1.04)	79.77 (1.02)	78.19 (0.64)	73.18 (0.72)
GRAPE	Intrinsic	89.67 (0.21)	89.52 (0.17)	89.16 (0.18)	87.70 (0.27)	83.99 (0.48)	76.57 (0.21)	89.48 (0.19)	89.39 (0.24)	89.02 (0.26)	87.47 (0.22)	84.42 (0.25)	77.30 (0.29)	88.76 (0.27)	88.03 (0.62)	87.56 (0.46)	87.07 (0.32)	83.78 (0.25)	77.81 (0.41)
HistGradientBoost	Constant	88.62 (0.16)	88.19 (0.14)	87.42 (0.21)	85.22 (0.22)	80.12 (0.20)	70.95 (0.14)	87.76 (0.21)	87.29 (0.20)	86.66 (0.28)	84.46 (0.12)	79.57 (0.28)	71.53 (0.21)	86.45 (0.08)	85.96 (0.08)	85.50 (0.16)	83.57 (0.14)	78.78 (0.22)	71.05 (0.38)
	KNN	88.06 (0.16)	87.68 (0.17)	87.12 (0.19)	85.00 (0.23)	80.04 (0.21)	70.92 (0.20)	86.93 (0.31)	86.51 (0.30)	85.74 (0.32)	83.55 (0.26)	78.45 (0.39)	70.39 (0.38)	84.98 (0.28)	84.51 (0.25)	83.81 (0.30)	81.78 (0.23)	76.74 (0.32)	70.28 (0.18)
	MICE	87.36 (0.16)	87.16 (0.18)	86.63 (0.15)	85.21 (0.26)	81.30 (0.26)	73.23 (0.12)	85.29 (0.22)	84.08 (0.22)	83.36 (0.30)	82.31 (0.17)	80.41 (0.23)	74.32 (0.33)	83.25 (0.40)	81.74 (0.57)	80.94 (0.65)	80.01 (0.62)	79.18 (0.48)	75.27 (0.68)
	Intrinsic	**92.55 (0.11)**	**92.36 (0.13)**	**92.01 (0.10)**	**90.96 (0.14)**	**87.64 (0.13)**	**80.33 (0.14)**	**92.08 (0.14)**	**91.86 (0.14)**	**91.53 (0.18)**	**90.42 (0.11)**	**87.45 (0.20)**	80.58 (0.33)	90.82 (0.26)	90.60 (0.24)	90.40 (0.29)	**89.44 (0.21)**	**86.78 (0.26)**	**80.46 (0.38)**
MLP	Constant	86.73 (0.17)	86.40 (0.18)	86.12 (0.20)	84.66 (0.27)	80.87 (0.25)	73.45 (0.41)	85.54 (0.21)	85.34 (0.24)	85.19 (0.18)	84.04 (0.13)	81.13 (0.23)	75.29 (0.22)	83.68 (0.68)	83.50 (0.66)	83.24 (0.59)	82.54 (0.44)	80.86 (0.29)	75.55 (0.40)
	KNN	85.99 (0.20)	85.73 (0.24)	85.16 (0.18)	83.36 (0.31)	78.64 (0.17)	69.99 (0.17)	84.08 (0.39)	83.85 (0.35)	83.14 (0.38)	81.51 (0.34)	77.11 (0.31)	69.53 (0.55)	80.13 (0.36)	79.82 (0.37)	79.34 (0.30)	77.64 (0.49)	74.07 (0.24)	69.15 (0.11)
	MICE	86.03 (0.16)	85.85 (0.20)	85.44 (0.17)	84.57 (0.18)	81.59 (0.06)	74.70 (0.13)	83.68 (0.36)	83.39 (0.27)	83.03 (0.31)	82.01 (0.15)	80.37 (0.19)	75.81 (0.34)	81.73 (0.80)	81.36 (0.82)	80.74 (0.81)	79.87 (0.64)	79.08 (0.35)	75.89 (0.32)
Random Forest	Constant	89.99 (0.17)	89.75 (0.16)	89.37 (0.16)	87.68 (0.20)	82.80 (0.27)	73.37 (0.39)	89.84 (0.20)	89.49 (0.17)	88.93 (0.17)	87.02 (0.09)	81.88 (0.26)	72.46 (0.30)	89.11 (0.28)	88.68 (0.29)	88.15 (0.31)	86.09 (0.23)	80.80 (0.18)	72.62 (0.23)
	KNN	89.81 (0.16)	89.59 (0.14)	89.15 (0.13)	87.35 (0.13)	82.49 (0.20)	72.59 (0.24)	89.40 (0.19)	89.06 (0.21)	88.42 (0.23)	86.52 (0.12)	81.43 (0.29)	71.40 (0.42)	87.72 (0.16)	87.24 (0.14)	86.84 (0.16)	84.66 (0.21)	79.47 (0.21)	70.46 (0.26)
	MICE	89.77 (0.14)	89.58 (0.16)	89.26 (0.13)	88.08 (0.13)	84.62 (0.25)	76.66 (0.23)	89.17 (0.13)	88.70 (0.13)	88.14 (0.11)	86.68 (0.06)	83.26 (0.20)	74.79 (0.21)	88.07 (0.34)	87.55 (0.28)	86.95 (0.42)	85.47 (0.26)	81.85 (0.31)	73.46 (0.37)
	Intrinsic	91.20 (0.17)	91.00 (0.18)	90.67 (0.13)	89.41 (0.14)	85.51 (0.24)	77.31 (0.22)	90.89 (0.16)	90.63 (0.16)	90.23 (0.18)	89.02 (0.08)	85.37 (0.25)	77.62 (0.38)	90.05 (0.15)	89.81 (0.14)	89.48 (0.19)	88.21 (0.16)	84.78 (0.29)	78.11 (0.43)
SVM	Constant	89.08 (0.25)	88.90 (0.26)	88.61 (0.23)	87.45 (0.25)	83.53 (0.28)	76.03 (0.17)	88.41 (0.21)	88.19 (0.20)	87.95 (0.21)	86.75 (0.17)	83.47 (0.31)	76.14 (0.34)	86.58 (0.31)	86.48 (0.33)	86.37 (0.29)	85.65 (0.23)	82.66 (0.27)	75.95 (0.28)
	KNN	89.04 (0.25)	88.87 (0.25)	88.43 (0.21)	86.98 (0.24)	82.26 (0.29)	72.62 (0.21)	88.54 (0.22)	88.28 (0.21)	87.86 (0.20)	86.19 (0.18)	81.62 (0.23)	72.57 (0.43)	87.32 (0.21)	86.98 (0.19)	86.61 (0.16)	84.84 (0.28)	80.71 (0.23)	72.36 (0.19)
	MICE	89.06 (0.23)	88.88 (0.24)	88.62 (0.21)	87.59 (0.23)	83.96 (0.24)	77.19 (0.14)	87.89 (0.26)	87.64 (0.22)	87.36 (0.24)	86.17 (0.14)	83.41 (0.27)	77.06 (0.28)	86.07 (0.45)	85.92 (0.45)	85.68 (0.43)	84.86 (0.36)	82.23 (0.23)	76.39 (0.30)
TabNet	Constant	85.61 (0.29)	85.17 (0.30)	84.48 (0.29)	82.08 (0.38)	77.27 (0.38)	68.64 (0.33)	83.93 (0.36)	83.44 (0.36)	82.90 (0.37)	80.78 (0.36)	76.70 (0.30)	69.39 (0.34)	81.47 (0.71)	81.17 (0.67)	80.69 (0.63)	79.34 (0.48)	75.70 (0.33)	69.02 (0.48)
	KNN	86.01 (0.23)	85.68 (0.21)	85.16 (0.22)	83.16 (0.25)	78.72 (0.21)	70.33 (0.21)	85.03 (0.30)	84.59 (0.28)	83.93 (0.29)	81.76 (0.29)	77.18 (0.33)	69.82 (0.38)	83.87 (0.18)	83.34 (0.19)	82.67 (0.17)	80.69 (0.17)	75.85 (0.31)	69.26 (0.20)
	MICE	85.66 (0.07)	85.28 (0.09)	84.68 (0.07)	82.77 (0.17)	79.05 (0.25)	71.92 (0.13)	83.90 (0.17)	82.35 (0.23)	81.46 (0.34)	80.68 (0.19)	78.81 (0.22)	73.21 (0.31)	81.34 (0.33)	78.64 (0.48)	77.45 (0.38)	76.67 (0.53)	76.94 (0.50)	72.92 (0.72)
TabTransformer	Constant	85.21 (0.24)	84.69 (0.25)	83.91 (0.25)	81.38 (0.32)	75.98 (0.21)	67.11 (0.25)	83.95 (0.55)	83.46 (0.51)	82.74 (0.50)	80.55 (0.35)	75.78 (0.27)	67.96 (0.15)	80.18 (0.56)	79.81 (0.58)	79.36 (0.51)	77.84 (0.40)	74.13 (0.17)	67.84 (0.30)
	KNN	85.44 (0.21)	85.06 (0.21)	84.57 (0.21)	82.29 (0.27)	77.17 (0.18)	68.73 (0.11)	83.24 (0.56)	82.82 (0.53)	82.06 (0.55)	79.93 (0.46)	75.04 (0.48)	67.97 (0.38)	80.95 (0.27)	80.49 (0.28)	79.84 (0.25)	77.98 (0.33)	73.47 (0.46)	67.14 (0.21)
	MICE	85.16 (0.27)	84.77 (0.29)	84.15 (0.21)	82.20 (0.33)	77.56 (0.25)	68.86 (0.09)	81.47 (0.27)	79.61 (0.23)	78.48 (0.31)	77.83 (0.29)	76.13 (0.42)	70.35 (0.41)	77.28 (1.03)	75.25 (0.82)	74.07 (0.67)	73.52 (0.68)	73.67 (0.31)	70.09 (0.31)
XGBoost	Constant	88.79 (0.18)	88.35 (0.19)	87.60 (0.21)	85.38 (0.25)	80.15 (0.27)	70.97 (0.12)	87.97 (0.21)	87.51 (0.21)	86.85 (0.26)	84.75 (0.13)	79.80 (0.24)	71.78 (0.23)	86.18 (0.32)	85.78 (0.33)	85.35 (0.30)	83.55 (0.15)	79.16 (0.23)	71.33 (0.39)
	KNN	88.51 (0.15)	88.11 (0.17)	87.56 (0.18)	85.35 (0.21)	80.24 (0.17)	71.13 (0.16)	87.01 (0.29)	86.53 (0.29)	85.77 (0.33)	83.65 (0.28)	78.40 (0.36)	70.33 (0.42)	85.55 (0.25)	84.99 (0.25)	84.35 (0.26)	82.27 (0.30)	77.17 (0.39)	70.20 (0.28)
	MICE	87.64 (0.16)	87.41 (0.16)	86.90 (0.15)	85.46 (0.28)	81.42 (0.22)	73.11 (0.27)	85.35 (0.28)	84.15 (0.31)	83.32 (0.39)	82.26 (0.28)	80.29 (0.22)	74.09 (0.32)	83.03 (0.51)	81.41 (0.62)	80.48 (0.65)	79.66 (0.67)	78.86 (0.52)	74.98 (0.71)
	Intrinsic	92.48 (0.14)	92.29 (0.15)	91.88 (0.13)	90.85 (0.14)	87.50 (0.13)	79.99 (0.19)	91.98 (0.14)	91.81 (0.13)	91.47 (0.18)	90.35 (0.12)	87.40 (0.15)	**80.59 (0.29)**	89.98 (0.24)	89.87 (0.23)	89.68 (0.23)	88.91 (0.24)	86.57 (0.27)	80.41 (0.40)

Table C.2: Average AUC performance and standard error (in brackets) of the experiments on the ADULT dataset. To facilitate the analysis we highlighted the best performance in each column in bold.

		Train missing percentage: 0%						Train missing percentage: 5%						Train missing percentage: 10%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
Model	**Strategy**	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	94.32 (0.13)	93.89 (0.14)	**93.23 (0.17)**	**91.35 (0.13)**	**86.01 (0.24)**	76.72 (0.14)	94.27 (0.11)	93.77 (0.15)	93.21 (0.13)	**91.25 (0.19)**	85.72 (0.08)	77.64 (0.42)	94.00 (0.16)	93.43 (0.11)	92.95 (0.21)	90.97 (0.25)	86.03 (0.25)	**77.55 (0.50)**
NAIM w/o regularization	Intrinsic	93.56 (0.21)	90.66 (0.77)	87.73 (1.25)	80.38 (2.37)	70.07 (2.96)	61.86 (2.44)	93.62 (0.19)	92.91 (0.17)	92.12 (0.21)	88.89 (0.36)	80.61 (0.58)	69.72 (0.68)	93.53 (0.13)	92.81 (0.17)	91.83 (0.24)	89.08 (0.35)	82.51 (0.64)	71.18 (0.78)
	Constant	93.28 (0.21)	92.55 (0.23)	91.13 (0.39)	87.96 (0.45)	81.26 (0.51)	71.58 (1.02)	93.25 (0.19)	92.35 (0.24)	91.70 (0.19)	88.75 (0.44)	82.77 (0.60)	73.90 (0.49)	93.30 (0.22)	92.53 (0.21)	91.54 (0.33)	89.13 (0.37)	83.51 (0.31)	73.99 (0.39)
	KNN	93.43 (0.25)	92.86 (0.27)	91.74 (0.34)	88.95 (0.34)	84.16 (0.45)	76.70 (0.34)	93.35 (0.27)	92.42 (0.28)	91.71 (0.25)	88.76 (0.52)	83.91 (0.48)	74.86 (0.51)	93.09 (0.30)	92.24 (0.29)	91.27 (0.41)	88.93 (0.53)	83.73 (0.47)	73.61 (0.76)
	MICE	93.43 (0.22)	92.91 (0.25)	91.92 (0.33)	88.91 (0.45)	81.52 (0.55)	70.88 (1.04)	93.26 (0.25)	92.49 (0.26)	91.98 (0.24)	89.38 (0.49)	83.76 (0.53)	75.32 (0.66)	93.07 (0.21)	92.37 (0.21)	91.54 (0.29)	89.44 (0.31)	84.34 (0.28)	74.81 (0.37)
Adaboost	Constant	93.98 (0.15)	93.18 (0.16)	91.98 (0.25)	88.29 (0.26)	80.79 (0.41)	69.51 (0.23)	93.95 (0.17)	93.23 (0.17)	92.57 (0.20)	89.78 (0.30)	83.91 (0.45)	74.44 (0.53)	93.85 (0.13)	93.28 (0.10)	92.46 (0.15)	90.19 (0.23)	85.00 (0.18)	75.36 (0.40)
	KNN	93.98 (0.15)	93.41 (0.18)	92.41 (0.23)	89.67 (0.29)	85.06 (0.40)	77.38 (0.47)	93.98 (0.14)	93.16 (0.16)	92.44 (0.12)	89.71 (0.38)	84.69 (0.37)	75.58 (0.92)	93.90 (0.18)	93.22 (0.17)	92.24 (0.23)	89.95 (0.36)	84.85 (0.25)	74.54 (0.46)
	MICE	93.98 (0.15)	93.50 (0.16)	92.59 (0.26)	89.25 (0.35)	81.14 (0.65)	67.84 (0.70)	93.96 (0.17)	93.19 (0.20)	92.63 (0.17)	89.37 (0.34)	82.98 (0.27)	73.96 (0.44)	93.81 (0.17)	93.15 (0.19)	92.31 (0.22)	89.85 (0.48)	84.39 (0.46)	74.73 (0.34)
Decision Tree	Constant	71.95 (0.34)	70.60 (0.29)	68.81 (0.45)	64.89 (0.75)	59.39 (0.35)	54.30 (0.56)	72.62 (0.41)	71.58 (0.34)	69.86 (0.60)	66.35 (0.55)	60.64 (0.56)	55.15 (0.45)	72.01 (0.42)	71.09 (0.34)	70.22 (0.54)	67.40 (0.37)	62.84 (0.79)	57.03 (0.57)
	KNN	71.95 (0.34)	71.64 (0.40)	70.41 (0.39)	68.98 (0.54)	66.09 (0.33)	62.67 (0.24)	71.76 (0.48)	71.04 (0.66)	69.95 (0.26)	67.87 (0.39)	64.90 (0.46)	58.61 (0.39)	71.80 (0.40)	71.00 (0.42)	70.00 (0.49)	68.00 (0.40)	64.60 (0.24)	58.86 (0.34)
	MICE	71.95 (0.34)	71.39 (0.35)	70.21 (0.25)	66.50 (0.32)	59.61 (0.53)	53.20 (0.17)	72.63 (0.34)	71.76 (0.45)	70.30 (0.44)	67.28 (0.44)	61.24 (0.38)	56.45 (0.46)	72.60 (0.35)	71.66 (0.34)	70.34 (0.09)	67.31 (0.25)	62.84 (0.25)	56.43 (0.54)
	Intrinsic	71.95 (0.34)	71.14 (0.39)	70.36 (0.56)	67.89 (0.61)	62.51 (0.33)	57.56 (0.25)	80.92 (0.77)	77.66 (0.60)	74.73 (0.62)	67.77 (0.92)	59.81 (1.19)	54.81 (1.39)	77.69 (1.69)	75.36 (1.74)	73.43 (1.85)	67.92 (1.74)	59.99 (1.48)	54.91 (0.97)
FTTransformer	Constant	93.95 (0.16)	91.00 (0.68)	89.56 (0.64)	85.88 (0.73)	78.00 (1.22)	68.00 (1.53)	93.42 (0.08)	92.57 (0.09)	91.88 (0.17)	89.00 (0.13)	83.01 (0.51)	74.13 (0.36)	93.28 (0.15)	92.51 (0.19)	91.55 (0.20)	89.09 (0.22)	83.69 (0.44)	74.26 (0.13)
	KNN	94.25 (0.20)	91.85 (0.16)	90.87 (0.13)	88.09 (0.08)	83.34 (0.29)	76.07 (0.46)	93.73 (0.16)	92.84 (0.14)	92.19 (0.14)	89.48 (0.38)	84.61 (0.37)	75.18 (0.57)	93.43 (0.24)	92.65 (0.25)	91.70 (0.28)	89.34 (0.38)	83.97 (0.36)	73.33 (0.55)
	MICE	94.05 (0.13)	91.77 (0.26)	90.81 (0.26)	88.03 (0.46)	81.56 (0.68)	71.63 (0.58)	93.58 (0.21)	92.81 (0.21)	92.28 (0.19)	89.85 (0.40)	84.46 (0.47)	76.33 (0.49)	93.27 (0.28)	92.62 (0.27)	91.75 (0.35)	89.72 (0.48)	84.51 (0.55)	75.17 (0.76)
GRAPE	Intrinsic	89.54 (2.64)	88.65 (1.39)	84.49 (2.49)	78.33 (2.67)	77.93 (1.51)	72.09 (1.24)	85.28 (2.30)	82.43 (2.15)	87.15 (2.36)	84.90 (2.30)	77.57 (1.16)	73.08 (1.33)	86.19 (2.54)	87.48 (2.83)	86.14 (2.74)	81.90 (2.32)	77.63 (2.77)	74.06 (0.93)
HistGradientBoost	Constant	**94.82 (0.10)**	92.82 (0.19)	91.53 (0.29)	87.76 (0.33)	80.21 (0.85)	69.82 (0.80)	94.76 (0.09)	94.18 (0.09)	**93.66 (0.09)**	91.20 (0.26)	85.72 (0.18)	75.93 (0.20)	94.71 (0.11)	94.19 (0.13)	93.52 (0.16)	91.50 (0.31)	86.28 (0.25)	76.54 (0.17)
	KNN	**94.82 (0.10)**	93.08 (0.22)	92.06 (0.28)	89.32 (0.33)	84.57 (0.49)	77.13 (0.34)	94.75 (0.10)	93.87 (0.12)	93.19 (0.09)	90.55 (0.38)	85.39 (0.32)	76.05 (0.32)	94.72 (0.13)	93.93 (0.12)	92.95 (0.21)	90.76 (0.29)	85.60 (0.23)	74.83 (0.47)
	MICE	**94.82 (0.10)**	93.09 (0.21)	92.12 (0.29)	89.08 (0.22)	83.22 (0.43)	73.17 (0.38)	**94.81 (0.10)**	94.10 (0.11)	93.59 (0.10)	91.09 (0.31)	85.16 (0.22)	75.02 (0.44)	94.74 (0.13)	94.15 (0.15)	93.39 (0.20)	91.31 (0.31)	85.82 (0.35)	75.39 (0.20)
	Intrinsic	**94.82 (0.10)**	**93.96 (0.10)**	92.61 (0.20)	88.99 (0.26)	82.51 (0.49)	72.69 (0.42)	94.76 (0.09)	**94.20 (0.11)**	93.52 (0.13)	91.07 (0.27)	84.55 (0.28)	74.05 (0.54)	**94.77 (0.08)**	**94.28 (0.10)**	**93.57 (0.13)**	91.48 (0.25)	85.85 (0.12)	75.19 (0.32)
MLP	Constant	88.63 (0.28)	87.57 (0.21)	85.65 (0.35)	81.48 (0.49)	73.50 (0.52)	65.03 (0.58)	86.47 (0.75)	85.39 (0.81)	84.35 (0.63)	79.99 (0.83)	73.69 (0.74)	63.77 (1.06)	85.79 (0.12)	84.98 (0.05)	83.75 (0.21)	79.93 (0.34)	74.03 (0.46)	64.59 (0.63)
	KNN	88.63 (0.28)	87.99 (0.31)	86.83 (0.30)	84.33 (0.30)	80.27 (0.61)	73.47 (0.24)	86.48 (0.34)	85.69 (0.25)	84.80 (0.29)	82.18 (0.37)	77.51 (0.33)	68.98 (0.80)	84.99 (0.34)	84.28 (0.31)	83.26 (0.40)	81.09 (0.33)	76.17 (0.38)	65.77 (0.55)
	MICE	88.63 (0.28)	87.95 (0.20)	86.47 (0.40)	82.13 (0.43)	73.06 (0.30)	63.00 (1.05)	87.00 (0.37)	86.27 (0.46)	85.66 (0.31)	83.05 (0.45)	78.24 (0.76)	69.08 (0.73)	86.42 (0.16)	85.68 (0.17)	84.87 (0.38)	82.41 (0.66)	77.73 (0.73)	68.74 (1.18)
Random Forest	Constant	94.05 (0.10)	93.18 (0.16)	91.91 (0.23)	88.62 (0.38)	82.75 (0.39)	72.91 (0.18)	94.09 (0.06)	93.28 (0.08)	92.60 (0.10)	89.82 (0.25)	85.06 (0.28)	76.98 (0.39)	94.06 (0.17)	93.37 (0.12)	92.42 (0.20)	90.55 (0.27)	85.62 (0.32)	76.52 (0.34)
	KNN	94.05 (0.10)	93.39 (0.14)	92.23 (0.20)	88.95 (0.29)	84.45 (0.44)	**77.76 (0.38)**	93.84 (0.08)	92.85 (0.13)	92.05 (0.07)	89.07 (0.34)	84.31 (0.20)	**77.70 (0.46)**	93.90 (0.15)	93.02 (0.14)	91.82 (0.13)	89.33 (0.24)	84.42 (0.18)	75.99 (0.12)
	MICE	94.05 (0.10)	93.36 (0.14)	92.17 (0.25)	88.85 (0.30)	83.42 (0.40)	73.47 (0.34)	94.01 (0.10)	93.17 (0.16)	92.29 (0.08)	89.31 (0.33)	84.19 (0.15)	75.83 (0.56)	94.04 (0.12)	93.22 (0.09)	92.21 (0.11)	89.80 (0.25)	84.60 (0.35)	76.24 (0.24)
	Intrinsic	94.43 (0.08)	93.79 (0.13)	92.39 (0.25)	89.02 (0.20)	82.71 (0.42)	72.02 (0.67)	94.61 (0.10)	94.01 (0.12)	93.44 (0.06)	91.21 (0.24)	**85.95 (0.13)**	76.85 (0.27)	94.54 (0.09)	94.04 (0.08)	93.29 (0.15)	91.31 (0.28)	**86.61 (0.29)**	76.87 (0.29)
SVM	Constant	93.08 (0.12)	92.38 (0.12)	91.08 (0.19)	88.11 (0.19)	80.57 (0.45)	70.81 (0.68)	92.94 (0.15)	92.20 (0.18)	91.38 (0.16)	88.78 (0.40)	82.80 (0.47)	73.45 (0.45)	92.78 (0.13)	92.02 (0.11)	91.25 (0.23)	88.72 (0.38)	83.48 (0.21)	74.00 (0.38)
	KNN	93.08 (0.12)	92.52 (0.16)	91.52 (0.21)	88.91 (0.20)	84.58 (0.36)	77.26 (0.30)	92.97 (0.13)	92.20 (0.17)	91.48 (0.19)	88.95 (0.35)	84.11 (0.21)	75.90 (0.45)	92.87 (0.17)	92.04 (0.20)	91.19 (0.25)	88.81 (0.32)	84.02 (0.25)	74.35 (0.42)
	MICE	93.08 (0.12)	92.44 (0.13)	91.35 (0.20)	88.43 (0.21)	80.82 (0.33)	70.15 (1.01)	93.01 (0.14)	92.31 (0.19)	91.62 (0.15)	89.34 (0.41)	84.43 (0.45)	76.37 (0.47)	92.87 (0.12)	92.11 (0.09)	91.42 (0.21)	89.30 (0.35)	84.50 (0.34)	76.46 (0.49)
TabNet	Constant	94.42 (0.08)	91.43 (0.53)	90.18 (0.45)	86.31 (0.68)	78.54 (0.95)	69.05 (1.03)	94.12 (0.17)	93.26 (0.14)	92.50 (0.19)	89.30 (0.28)	82.70 (0.47)	73.16 (0.79)	94.13 (0.10)	93.28 (0.08)	92.47 (0.16)	89.66 (0.32)	83.58 (0.39)	73.46 (0.75)
	KNN	94.42 (0.08)	91.78 (0.55)	90.84 (0.53)	88.04 (0.69)	83.64 (0.77)	76.62 (0.36)	94.31 (0.16)	93.39 (0.16)	92.75 (0.16)	89.98 (0.38)	85.28 (0.16)	76.50 (0.36)	94.23 (0.15)	93.44 (0.16)	92.58 (0.18)	90.24 (0.32)	85.26 (0.21)	75.21 (0.25)
	MICE	94.42 (0.08)	91.92 (0.56)	90.95 (0.50)	88.14 (0.65)	81.19 (0.74)	70.90 (1.07)	94.32 (0.21)	93.56 (0.21)	93.08 (0.17)	90.51 (0.33)	84.97 (0.32)	76.44 (0.51)	94.24 (0.16)	93.56 (0.17)	92.78 (0.19)	90.61 (0.39)	85.29 (0.32)	75.19 (0.42)
TabTransformer	Constant	94.42 (0.12)	90.75 (0.38)	88.51 (0.61)	82.77 (0.85)	72.96 (1.26)	63.39 (0.62)	93.81 (0.20)	92.73 (0.20)	91.87 (0.17)	88.15 (0.36)	80.38 (0.24)	68.18 (0.41)	93.56 (0.19)	92.71 (0.17)	91.56 (0.26)	88.70 (0.39)	81.47 (0.53)	70.11 (0.30)
	KNN	94.37 (0.14)	90.72 (0.59)	89.38 (0.79)	85.95 (0.89)	80.31 (1.10)	71.53 (1.28)	94.05 (0.15)	93.01 (0.13)	92.16 (0.13)	89.13 (0.39)	83.08 (0.22)	70.13 (0.53)	93.90 (0.20)	93.02 (0.22)	92.04 (0.25)	89.32 (0.41)	82.97 (0.40)	69.92 (0.59)
	MICE	94.35 (0.15)	91.68 (0.44)	90.52 (0.42)	85.81 (0.87)	76.05 (1.34)	64.48 (0.89)	94.05 (0.17)	93.17 (0.19)	92.55 (0.14)	89.41 (0.42)	81.47 (0.32)	67.42 (0.82)	93.91 (0.15)	93.13 (0.16)	92.19 (0.18)	89.71 (0.34)	82.71 (0.52)	69.52 (0.49)
XGBoost	Constant	94.77 (0.08)	92.81 (0.16)	91.61 (0.30)	88.16 (0.15)	81.51 (0.41)	71.06 (0.82)	94.76 (0.10)	94.18 (0.08)	93.65 (0.11)	91.14 (0.30)	85.48 (0.26)	75.44 (0.56)	94.59 (0.11)	94.09 (0.10)	93.38 (0.15)	**91.53 (0.26)**	86.46 (0.25)	76.79 (0.34)
	KNN	94.77 (0.08)	93.01 (0.19)	91.99 (0.25)	89.22 (0.27)	84.35 (0.50)	77.15 (0.44)	94.75 (0.11)	93.85 (0.12)	93.19 (0.13)	90.49 (0.37)	85.48 (0.29)	76.32 (0.31)	94.62 (0.13)	93.83 (0.15)	92.90 (0.18)	90.64 (0.29)	85.67 (0.22)	75.18 (0.45)
	MICE	94.77 (0.08)	92.92 (0.17)	91.75 (0.26)	88.14 (0.24)	80.93 (0.55)	69.87 (0.69)	94.74 (0.12)	94.02 (0.13)	93.52 (0.12)	91.04 (0.32)	85.24 (0.34)	75.30 (0.51)	94.66 (0.14)	94.09 (0.17)	93.33 (0.17)	91.40 (0.30)	85.95 (0.34)	75.27 (0.26)
	Intrinsic	94.77 (0.08)	91.63 (0.20)	88.23 (0.35)	78.51 (0.37)	64.14 (0.89)	54.71 (0.79)	94.71 (0.16)	94.15 (0.18)	93.51 (0.21)	90.94 (0.29)	84.50 (0.25)	72.79 (0.53)	94.67 (0.13)	94.23 (0.13)	93.47 (0.16)	91.42 (0.29)	85.99 (0.24)	75.93 (0.49)

		Train missing percentage: 25%						Train missing percentage: 50%						Train missing percentage: 75%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
Model	**Strategy**	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	93.71 (0.13)	93.27 (0.13)	92.81 (0.18)	91.15 (0.15)	85.84 (0.30)	77.45 (0.18)	93.14 (0.11)	92.67 (0.11)	92.17 (0.17)	90.24 (0.11)	85.77 (0.30)	77.00 (0.31)	91.18 (0.24)	90.86 (0.22)	90.43 (0.14)	88.97 (0.24)	84.81 (0.23)	78.20 (0.42)
NAIM w/o regularization	Intrinsic	93.68 (0.12)	93.12 (0.14)	92.19 (0.19)	89.47 (0.44)	82.40 (0.75)	71.56 (1.81)	93.08 (0.19)	92.52 (0.18)	91.96 (0.25)	89.19 (0.30)	83.92 (0.47)	74.24 (0.77)	92.13 (0.26)	91.63 (0.30)	91.02 (0.28)	89.16 (0.25)	84.73 (0.35)	76.43 (0.85)
	Constant	92.70 (0.15)	92.06 (0.21)	90.98 (0.17)	88.49 (0.13)	82.68 (0.38)	74.49 (0.49)	92.29 (0.21)	91.68 (0.26)	90.77 (0.29)	88.13 (0.26)	82.72 (0.31)	74.09 (0.38)	91.42 (0.25)	90.71 (0.25)	89.95 (0.22)	87.53 (0.26)	82.97 (0.43)	74.11 (0.76)
	KNN	92.72 (0.17)	92.08 (0.23)	91.15 (0.21)	88.81 (0.11)	81.98 (0.25)	72.13 (0.38)	92.14 (0.19)	91.51 (0.29)	90.63 (0.26)	87.78 (0.24)	81.82 (0.16)	72.99 (0.61)	90.75 (0.23)	89.96 (0.29)	89.09 (0.29)	85.86 (0.26)	80.67 (0.42)	71.22 (0.71)
	MICE	92.77 (0.22)	91.65 (0.31)	90.79 (0.31)	88.98 (0.18)	83.63 (0.63)	75.74 (0.69)	92.62 (0.17)	90.94 (0.17)	89.94 (0.23)	88.04 (0.37)	83.73 (0.38)	76.04 (0.46)	89.10 (2.23)	87.67 (1.98)	86.64 (1.84)	84.05 (1.35)	80.21 (1.15)	73.36 (0.80)
Adaboost	Constant	93.65 (0.16)	93.19 (0.21)	92.46 (0.19)	90.35 (0.14)	85.10 (0.45)	77.28 (0.51)	92.94 (0.22)	92.58 (0.24)	92.08 (0.30)	90.16 (0.21)	86.03 (0.37)	78.48 (0.37)	91.46 (0.28)	91.06 (0.32)	90.65 (0.28)	89.23 (0.23)	86.10 (0.37)	79.61 (0.44)
	KNN	93.74 (0.13)	93.20 (0.17)	92.27 (0.24)	90.15 (0.22)	84.39 (0.56)	74.13 (0.44)	92.80 (0.22)	92.32 (0.20)	91.62 (0.32)	89.64 (0.24)	84.79 (0.31)	75.42 (0.27)	90.96 (0.30)	90.45 (0.34)	89.91 (0.34)	88.24 (0.25)	84.61 (0.50)	76.82 (0.45)
	MICE	93.67 (0.16)	93.22 (0.21)	92.45 (0.21)	90.29 (0.21)	85.21 (0.52)	76.67 (0.59)	93.23 (0.25)	91.93 (0.34)	90.99 (0.53)	89.04 (0.39)	85.16 (0.30)	77.97 (0.42)	92.16 (0.25)	88.47 (0.82)	86.73 (1.03)	84.60 (0.71)	83.71 (0.33)	78.40 (0.31)
Decision Tree	Constant	69.78 (0.44)	69.19 (0.34)	68.63 (0.38)	67.56 (0.42)	64.27 (0.43)	60.40 (0.68)	64.63 (0.59)	64.66 (0.52)	64.20 (0.49)	64.32 (0.55)	64.45 (0.49)	62.09 (0.44)	57.54 (1.04)	57.66 (0.73)	57.94 (0.96)	58.78 (0.63)	60.13 (0.60)	59.97 (0.36)
	KNN	70.52 (0.28)	69.84 (0.34)	68.89 (0.21)	67.40 (0.35)	62.93 (0.48)	56.67 (0.29)	67.45 (0.39)	67.34 (0.30)	66.99 (0.45)	65.28 (0.23)	62.93 (0.47)	57.99 (0.27)	62.03 (0.30)	61.87 (0.29)	61.55 (0.41)	61.42 (0.53)	60.40 (0.31)	57.25 (0.24)
	MICE	72.53 (0.58)	71.54 (0.68)	70.70 (0.55)	67.91 (0.57)	63.33 (0.49)	57.66 (0.81)	68.29 (0.66)	66.80 (0.51)	66.14 (0.64)	65.87 (0.55)	63.67 (0.40)	59.60 (0.57)	61.10 (1.23)	59.88 (1.05)	59.24 (0.99)	58.42 (0.87)	59.11 (0.46)	59.02 (0.18)
	Intrinsic	78.91 (1.17)	77.86 (1.00)	76.21 (0.92)	72.62 (0.49)	66.83 (0.81)	59.18 (0.73)	66.21 (1.44)	65.94 (1.40)	66.14 (1.34)	66.21 (0.78)	65.40 (0.88)	62.16 (0.48)	56.26 (2.60)	56.92 (2.24)	57.31 (2.43)	58.86 (1.44)	61.06 (0.97)	62.82 (0.61)
FTTransformer	Constant	92.97 (0.24)	92.43 (0.30)	91.58 (0.21)	89.30 (0.28)	83.99 (0.49)	75.31 (0.58)	92.09 (0.39)	91.62 (0.44)	90.97 (0.38)	88.85 (0.36)	84.14 (0.32)	75.82 (0.60)	80.51 (7.13)	80.21 (7.06)	79.84 (6.97)	78.85 (6.74)	76.15 (6.10)	70.42 (4.80)
	KNN	93.09 (0.19)	92.46 (0.26)	91.57 (0.32)	89.27 (0.22)	82.88 (0.52)	72.74 (0.68)	91.43 (0.22)	90.85 (0.30)	90.11 (0.27)	87.52 (0.33)	82.50 (0.22)	73.63 (0.54)	89.23 (0.30)	88.63 (0.35)	88.06 (0.36)	85.91 (0.31)	81.83 (0.34)	73.94 (0.79)
	MICE	93.06 (0.18)	91.94 (0.27)	91.15 (0.23)	89.31 (0.20)	84.20 (0.53)	75.66 (0.38)	92.23 (0.25)	90.72 (0.27)	89.74 (0.30)	87.99 (0.38)	83.75 (0.48)	76.01 (0.34)	83.05 (7.79)	81.54 (7.44)	80.56 (7.21)	78.90 (6.83)	76.43 (6.26)	70.04 (4.80)
GRAPE	Intrinsic	82.59 (1.18)	86.90 (2.35)	85.07 (2.86)	82.48 (2.85)	81.44 (1.73)	74.37 (0.74)	87.69 (1.03)	82.57 (1.55)	88.06 (1.10)	81.02 (1.49)	81.70 (1.39)	73.85 (0.73)	81.95 (1.34)	82.40 (1.55)	79.06 (1.20)	81.52 (0.64)	82.05 (0.92)	74.25 (0.37)
HistGradientBoost	Constant	94.59 (0.12)	94.19 (0.16)	93.58 (0.12)	91.70 (0.14)	86.64 (0.43)	78.36 (0.51)	93.92 (0.13)	93.48 (0.16)	93.05 (0.17)	91.29 (0.15)	87.25 (0.33)	79.75 (0.28)	92.66 (0.19)	92.12 (0.25)	**91.78 (0.22)**	89.89 (0.20)	86.72 (0.26)	80.07 (0.51)
	KNN	94.41 (0.15)	93.78 (0.19)	92.90 (0.23)	90.49 (0.25)	84.53 (0.62)	74.89 (0.27)	93.59 (0.15)	92.97 (0.18)	92.29 (0.22)	89.85 (0.18)	85.07 (0.16)	76.69 (0.30)	92.08 (0.21)	91.41 (0.26)	90.85 (0.28)	88.43 (0.26)	84.39 (0.41)	76.55 (0.57)
	MICE	**94.67 (0.13)**	93.53 (0.17)	92.69 (0.18)	90.86 (0.08)	85.83 (0.46)	76.79 (0.59)	93.85 (0.12)	92.28 (0.13)	91.25 (0.17)	89.52 (0.31)	85.84 (0.47)	78.34 (0.56)	92.62 (0.15)	90.41 (0.26)	88.97 (0.24)	86.59 (0.49)	84.41 (0.53)	78.69 (0.49)
	Intrinsic	94.60 (0.12)	**94.23 (0.16)**	**93.63 (0.14)**	**91.76 (0.15)**	86.39 (0.42)	77.58 (0.58)	**94.22 (0.11)**	**93.82 (0.17)**	**93.42 (0.16)**	**91.58 (0.13)**	**87.39 (0.29)**	**79.90 (0.27)**	92.75 (0.20)	**92.28 (0.25)**	91.74 (0.27)	89.98 (0.28)	**86.84 (0.30)**	**80.59 (0.56)**
MLP	Constant	83.16 (0.91)	82.57 (0.86)	81.76 (0.89)	77.99 (0.37)	72.27 (1.14)	63.85 (0.78)	78.47 (0.76)	77.98 (0.72)	77.17 (0.66)	75.41 (0.60)	72.08 (0.80)	66.93 (0.37)	75.13 (1.14)	74.35 (1.00)	73.26 (1.13)	71.41 (0.76)	69.56 (0.75)	67.89 (0.18)
	KNN	83.17 (0.57)	82.27 (0.64)	81.64 (0.55)	79.05 (0.26)	72.26 (0.58)	64.06 (0.63)	78.88 (0.80)	78.51 (0.74)	77.53 (0.77)	75.36 (0.39)	70.11 (0.51)	62.90 (0.51)	73.76 (0.71)	73.35 (0.59)	73.11 (0.94)	71.91 (0.56)	67.84 (0.52)	61.67 (0.70)
	MICE	82.82 (0.60)	82.01 (0.75)	81.36 (0.63)	79.31 (0.45)	74.51 (0.73)	67.30 (0.98)	77.17 (0.40)	75.79 (0.24)	75.02 (0.47)	74.32 (0.35)	72.65 (0.20)	68.88 (0.29)	72.83 (1.34)	70.35 (1.00)	69.23 (1.16)	67.86 (1.06)	68.13 (0.71)	67.51 (0.36)
Random Forest	Constant	93.97 (0.18)	93.28 (0.25)	92.65 (0.25)	90.44 (0.28)	85.74 (0.57)	76.78 (0.59)	93.41 (0.15)	92.85 (0.19)	92.28 (0.22)	90.24 (0.21)	85.68 (0.28)	76.69 (0.52)	92.24 (0.28)	91.67 (0.28)	91.01 (0.23)	88.69 (0.27)	84.52 (0.27)	73.62 (0.55)
	KNN	93.58 (0.17)	92.81 (0.20)	91.71 (0.30)	89.63 (0.20)	83.67 (0.54)	75.08 (0.51)	92.86 (0.18)	92.19 (0.21)	91.36 (0.30)	89.05 (0.25)	84.06 (0.39)	76.20 (0.57)	91.22 (0.22)	90.51 (0.27)	89.74 (0.31)	87.53 (0.33)	83.91 (0.43)	74.82 (0.50)
	MICE	93.96 (0.17)	93.28 (0.23)	92.40 (0.28)	90.00 (0.27)	84.75 (0.71)	76.05 (0.87)	93.16 (0.18)	92.28 (0.18)	91.37 (0.16)	89.29 (0.27)	84.98 (0.34)	76.91 (0.68)	91.49 (0.35)	89.80 (0.39)	89.04 (0.42)	86.44 (0.38)	84.13 (0.38)	76.20 (0.42)
	Intrinsic	94.31 (0.16)	93.91 (0.19)	93.30 (0.22)	91.52 (0.17)	86.56 (0.51)	77.83 (0.67)	93.62 (0.14)	93.24 (0.15)	92.79 (0.19)	90.94 (0.22)	86.76 (0.32)	78.90 (0.43)	**92.76 (0.14)**	92.21 (0.17)	91.61 (0.18)	89.71 (0.16)	86.15 (0.27)	78.10 (0.50)
SVM	Constant	92.10 (0.17)	91.67 (0.23)	90.78 (0.21)	88.75 (0.28)	83.83 (0.34)	75.52 (0.53)	90.04 (0.30)	89.67 (0.34)	89.11 (0.37)	87.47 (0.27)	83.57 (0.42)	76.73 (0.24)	86.35 (0.30)	85.98 (0.30)	85.74 (0.34)	84.56 (0.29)	82.17 (0.42)	76.52 (0.73)
	KNN	92.52 (0.15)	91.97 (0.18)	91.02 (0.20)	88.97 (0.25)	83.34 (0.49)	73.65 (0.58)	90.72 (0.37)	90.23 (0.35)	89.54 (0.44)	87.42 (0.28)	82.45 (0.36)	73.70 (0.39)	87.25 (0.45)	86.82 (0.46)	86.42 (0.54)	84.59 (0.47)	80.84 (0.63)	73.49 (0.51)
	MICE	92.50 (0.14)	92.00 (0.19)	91.14 (0.18)	89.21 (0.25)	84.40 (0.40)	77.06 (0.44)	90.49 (0.32)	89.40 (0.30)	88.84 (0.32)	87.27 (0.29)	83.81 (0.31)	77.26 (0.24)	87.20 (0.43)	83.84 (0.73)	82.30 (0.91)	81.66 (0.89)	81.44 (0.45)	76.23 (0.63)
TabNet	Constant	93.68 (0.13)	93.09 (0.16)	92.35 (0.18)	90.07 (0.18)	84.16 (0.36)	75.25 (0.39)	92.93 (0.20)	92.40 (0.25)	91.76 (0.29)	89.58 (0.32)	85.01 (0.46)	76.81 (0.59)	91.44 (0.25)	90.79 (0.26)	90.29 (0.19)	88.33 (0.25)	84.45 (0.44)	76.85 (0.60)
	KNN	93.96 (0.16)	93.28 (0.16)	92.44 (0.28)	90.15 (0.31)	84.07 (0.53)	74.32 (0.53)	93.03 (0.18)	92.45 (0.22)	91.69 (0.26)	89.13 (0.27)	84.05 (0.24)	75.33 (0.48)	91.21 (0.24)	90.49 (0.30)	89.86 (0.22)	87.15 (0.27)	82.99 (0.27)	74.45 (0.79)
	MICE	93.83 (0.17)	92.38 (0.25)	91.51 (0.20)	90.00 (0.17)	84.76 (0.32)	76.56 (0.38)	93.00 (0.21)	90.66 (0.21)	89.58 (0.31)	88.33 (0.34)	84.85 (0.47)	77.15 (0.27)	91.48 (0.25)	86.67 (0.47)	84.82 (0.48)	83.23 (0.93)	82.47 (0.69)	76.63 (0.66)
TabTransformer	Constant	92.66 (0.23)	92.03 (0.27)	91.21 (0.26)	88.80 (0.32)	82.77 (0.59)	72.75 (0.65)	91.61 (0.29)	91.08 (0.32)	90.39 (0.37)	88.21 (0.35)	83.50 (0.28)	74.63 (0.14)	89.87 (0.52)	89.21 (0.57)	88.73 (0.47)	87.04 (0.39)	83.91 (0.56)	76.27 (0.68)
	KNN	93.18 (0.21)	92.49 (0.23)	91.56 (0.33)	89.26 (0.31)	82.80 (0.58)	72.76 (0.71)	91.82 (0.22)	91.22 (0.28)	90.53 (0.29)	88.00 (0.33)	83.12 (0.20)	74.43 (0.42)	90.10 (0.59)	89.41 (0.60)	88.90 (0.57)	86.61 (0.48)	82.64 (0.58)	74.25 (0.58)
	MICE	93.03 (0.19)	91.44 (0.32)	90.74 (0.29)	89.05 (0.17)	83.35 (0.54)	72.90 (0.67)	92.04 (0.25)	88.86 (0.35)	87.40 (0.45)	87.08 (0.50)	83.63 (0.58)	75.45 (0.64)	90.73 (0.39)	84.53 (0.90)	82.47 (1.12)	82.50 (1.11)	82.35 (0.77)	75.96 (0.76)
XGBoost	Constant	94.55 (0.13)	94.19 (0.17)	93.59 (0.16)	91.61 (0.16)	**86.69 (0.47)**	**78.44 (0.55)**	93.97 (0.14)	93.56 (0.17)	93.09 (0.20)	91.43 (0.20)	87.07 (0.33)	79.48 (0.33)	92.46 (0.11)	92.01 (0.11)	91.60 (0.13)	**90.10 (0.11)**	86.62 (0.29)	79.80 (0.55)
	KNN	94.33 (0.12)	93.71 (0.16)	92.82 (0.20)	90.56 (0.26)	84.51 (0.62)	74.49 (0.31)	93.50 (0.12)	92.92 (0.18)	92.24 (0.19)	89.73 (0.18)	84.86 (0.13)	76.73 (0.41)	91.83 (0.19)	91.11 (0.25)	90.56 (0.24)	88.20 (0.34)	84.12 (0.40)	76.33 (0.52)
	MICE	94.53 (0.15)	93.27 (0.17)	92.51 (0.23)	90.71 (0.13)	85.61 (0.44)	76.88 (0.59)	93.61 (0.15)	91.95 (0.19)	90.95 (0.17)	89.19 (0.33)	85.67 (0.50)	78.43 (0.52)	92.33 (0.15)	90.05 (0.20)	88.60 (0.33)	86.11 (0.52)	84.15 (0.68)	78.60 (0.49)
	Intrinsic	94.44 (0.14)	94.08 (0.17)	93.50 (0.17)	91.58 (0.20)	86.39 (0.42)	78.18 (0.64)	93.94 (0.12)	93.53 (0.15)	93.17 (0.17)	91.41 (0.18)	87.25 (0.38)	79.51 (0.20)	91.95 (0.34)	91.71 (0.34)	91.33 (0.26)	89.81 (0.17)	86.51 (0.19)	79.63 (0.51)

Table C.3: Average AUC performance and standard error (in brackets) of the experiments on the BankMarketing dataset. To facilitate the analysis we highlighted the best performance in each column in bold.

Model	Strategy	Train missing percentage: 0%						Train missing percentage: 5%						Train missing percentage: 10%					
		Test missing percentage: 0%	5%	10%	25%	50%	75%	Test missing percentage: 0%	5%	10%	25%	50%	75%	Test missing percentage: 0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	92.80 (0.23)	**92.25 (0.27)**	**91.46 (0.25)**	**88.36 (0.40)**	**82.23 (0.49)**	**72.97 (0.95)**	92.64 (0.24)	**91.85 (0.22)**	**90.99 (0.22)**	**88.43 (0.06)**	**82.01 (0.42)**	**73.75 (0.53)**	92.52 (0.26)	**91.91 (0.30)**	91.15 (0.30)	**88.64 (0.36)**	**82.88 (0.42)**	72.57 (0.46)
NAIM w/o regularization	Intrinsic	92.49 (0.24)	90.79 (0.28)	88.78 (0.38)	83.42 (0.69)	72.75 (1.39)	63.15 (2.21)	92.43 (0.15)	91.38 (0.14)	90.56 (0.18)	86.43 (0.23)	77.30 (0.73)	64.94 (1.29)	92.11 (0.34)	91.14 (0.33)	90.04 (0.35)	86.97 (0.56)	79.15 (0.37)	67.77 (0.56)
	Constant	90.43 (0.42)	88.42 (0.44)	86.18 (0.36)	80.96 (0.66)	71.12 (1.09)	63.09 (1.50)	89.97 (0.37)	89.03 (0.30)	87.95 (0.31)	85.18 (0.59)	78.14 (0.89)	68.44 (0.71)	90.00 (0.54)	89.16 (0.54)	88.31 (0.54)	85.61 (0.66)	79.02 (0.62)	69.39 (0.33)
	KNN	90.05 (0.68)	88.82 (0.54)	87.24 (0.51)	83.89 (0.40)	78.09 (0.51)	69.73 (0.62)	89.93 (0.45)	88.85 (0.53)	87.84 (0.41)	84.91 (0.65)	78.04 (0.97)	68.91 (0.60)	89.86 (0.56)	88.93 (0.49)	88.18 (0.46)	84.53 (0.52)	77.80 (0.55)	67.47 (0.58)
	MICE	90.14 (0.64)	88.72 (0.51)	87.18 (0.34)	83.47 (0.30)	74.61 (1.09)	66.24 (1.06)	90.38 (0.42)	89.35 (0.40)	88.30 (0.34)	85.82 (0.57)	79.58 (0.49)	71.15 (0.58)	89.69 (0.58)	88.96 (0.53)	88.25 (0.51)	86.07 (0.52)	80.83 (0.35)	71.92 (0.30)
Adaboost	Constant	91.19 (0.28)	89.90 (0.34)	88.53 (0.34)	84.42 (0.51)	76.76 (0.54)	66.37 (0.70)	91.07 (0.34)	90.19 (0.40)	89.64 (0.39)	86.56 (0.58)	80.17 (0.43)	70.09 (0.51)	90.97 (0.23)	90.28 (0.13)	89.71 (0.30)	86.98 (0.36)	80.33 (0.38)	70.80 (0.38)
	KNN	91.19 (0.28)	89.96 (0.22)	88.52 (0.32)	84.86 (0.28)	79.79 (0.52)	70.29 (0.42)	91.06 (0.30)	89.89 (0.44)	88.84 (0.35)	85.45 (0.39)	78.76 (0.49)	69.12 (0.57)	90.80 (0.30)	89.83 (0.26)	88.98 (0.27)	84.78 (0.52)	77.85 (0.78)	66.91 (0.14)
	MICE	91.19 (0.28)	90.01 (0.33)	88.65 (0.35)	84.94 (0.39)	78.00 (0.63)	68.20 (0.74)	91.24 (0.25)	90.01 (0.33)	88.88 (0.12)	85.58 (0.30)	79.35 (0.29)	70.01 (0.64)	90.84 (0.38)	90.01 (0.36)	89.25 (0.37)	85.78 (0.36)	79.68 (0.19)	70.46 (0.38)
Decision Tree	Constant	73.44 (0.32)	71.63 (0.49)	68.86 (0.68)	64.30 (0.21)	58.90 (1.01)	54.51 (0.68)	73.07 (0.25)	71.45 (0.47)	70.27 (0.21)	66.94 (0.75)	61.01 (0.58)	56.45 (0.29)	73.77 (0.42)	72.41 (0.41)	72.13 (0.33)	68.04 (0.38)	62.15 (0.69)	56.29 (0.89)
	KNN	73.44 (0.32)	72.22 (0.49)	70.47 (0.59)	67.49 (0.29)	64.35 (0.21)	60.04 (0.51)	72.96 (0.42)	71.77 (0.42)	71.05 (0.21)	68.13 (0.19)	62.68 (0.38)	58.10 (0.58)	71.49 (0.80)	70.49 (0.89)	70.30 (0.55)	66.86 (0.56)	63.07 (0.41)	56.23 (0.21)
	MICE	73.44 (0.32)	72.62 (0.58)	71.02 (0.32)	67.79 (0.32)	63.06 (1.01)	57.85 (0.77)	72.33 (0.42)	71.15 (0.30)	70.33 (0.34)	67.89 (0.42)	62.70 (0.99)	56.42 (0.48)	71.88 (0.53)	70.99 (0.68)	69.90 (0.96)	67.75 (0.38)	62.46 (0.37)	57.42 (0.51)
	Intrinsic	73.44 (0.32)	72.82 (0.41)	71.27 (0.36)	68.63 (0.79)	63.23 (0.54)	57.28 (0.56)	75.75 (0.74)	72.96 (0.51)	71.55 (0.69)	65.81 (1.03)	58.52 (0.60)	52.26 (0.83)	75.05 (0.97)	72.96 (0.90)	71.59 (0.72)	67.42 (0.64)	59.11 (1.09)	53.88 (1.53)
FTTransformer	Constant	92.45 (0.21)	87.72 (0.59)	86.47 (0.50)	83.26 (0.32)	75.83 (0.94)	66.69 (0.84)	90.07 (0.56)	89.35 (0.49)	88.72 (0.43)	86.35 (0.61)	80.07 (0.50)	70.29 (0.34)	89.68 (0.59)	89.04 (0.57)	88.51 (0.55)	85.97 (0.61)	79.94 (0.40)	70.41 (0.39)
	KNN	92.34 (0.20)	87.44 (0.42)	86.13 (0.40)	83.27 (0.45)	78.40 (0.74)	70.25 (0.68)	89.86 (0.55)	88.88 (0.67)	87.95 (0.51)	85.58 (0.68)	78.61 (0.47)	69.74 (0.31)	89.87 (0.49)	89.13 (0.47)	88.36 (0.46)	84.70 (0.48)	78.34 (0.58)	67.81 (0.67)
	MICE	92.47 (0.23)	87.47 (0.48)	86.61 (0.37)	84.37 (0.56)	78.20 (0.33)	70.08 (0.90)	89.97 (0.57)	89.35 (0.54)	88.64 (0.52)	86.62 (0.58)	81.51 (0.41)	73.15 (0.64)	89.84 (0.56)	89.23 (0.55)	88.55 (0.58)	86.28 (0.55)	81.13 (0.42)	**73.01 (0.10)**
GRAPE	Intrinsic	84.92 (2.38)	84.08 (0.91)	81.45 (2.93)	80.38 (1.52)	71.82 (2.50)	70.20 (0.85)	80.79 (1.57)	84.30 (0.88)	83.76 (2.27)	79.25 (1.01)	72.44 (3.49)	69.53 (0.70)	85.16 (1.88)	83.41 (2.08)	81.06 (0.92)	80.43 (1.16)	79.42 (1.21)	66.42 (2.31)
HistGradientBoost	Constant	91.61 (0.25)	84.57 (0.66)	82.90 (0.53)	78.26 (0.78)	69.47 (0.48)	60.78 (0.90)	88.38 (0.46)	87.59 (0.47)	86.51 (0.54)	83.83 (0.68)	77.60 (0.59)	68.16 (0.40)	88.53 (0.25)	87.57 (0.26)	87.06 (0.34)	83.98 (0.26)	77.22 (0.84)	66.62 (0.56)
	KNN	91.61 (0.25)	85.07 (0.47)	83.93 (0.54)	80.10 (0.53)	74.98 (0.55)	66.77 (0.89)	88.20 (0.54)	87.14 (0.55)	85.60 (0.48)	83.03 (0.69)	75.61 (0.67)	66.96 (0.10)	88.09 (0.26)	87.24 (0.33)	86.36 (0.35)	82.35 (0.57)	75.47 (0.39)	64.84 (0.72)
	MICE	91.61 (0.25)	85.10 (0.66)	83.85 (0.52)	79.89 (0.60)	70.90 (0.20)	62.35 (0.47)	88.34 (0.60)	87.49 (0.64)	86.43 (0.56)	84.56 (0.59)	78.55 (0.72)	69.45 (0.78)	88.50 (0.13)	87.64 (0.11)	86.91 (0.16)	84.40 (0.13)	77.99 (0.51)	69.59 (0.25)
	Intrinsic	91.61 (0.25)	90.20 (0.23)	88.18 (0.36)	83.98 (0.33)	75.44 (0.51)	65.15 (0.38)	91.82 (0.33)	90.83 (0.30)	89.75 (0.44)	86.53 (0.71)	78.20 (0.83)	66.41 (1.06)	91.78 (0.24)	90.82 (0.17)	90.28 (0.33)	86.91 (0.42)	79.84 (0.55)	69.74 (0.43)
MLP	Constant	84.61 (0.50)	83.57 (0.46)	82.37 (0.42)	78.48 (0.65)	69.99 (0.75)	60.12 (0.95)	82.93 (0.52)	81.92 (0.47)	80.94 (0.56)	77.46 (0.28)	69.86 (0.65)	61.91 (0.51)	83.47 (0.66)	82.42 (0.59)	80.95 (0.72)	78.07 (0.48)	69.82 (0.82)	62.94 (0.83)
	KNN	84.61 (0.50)	83.75 (0.44)	82.57 (0.43)	79.52 (0.29)	76.30 (0.27)	68.03 (0.69)	82.83 (0.60)	82.03 (0.70)	80.74 (0.58)	79.06 (0.64)	73.52 (0.42)	65.37 (0.56)	82.30 (0.40)	81.45 (0.46)	80.02 (0.35)	76.75 (0.50)	71.43 (0.66)	63.10 (0.54)
	MICE	84.61 (0.50)	83.87 (0.47)	82.99 (0.32)	79.78 (0.80)	72.22 (0.75)	62.33 (0.91)	83.25 (0.61)	82.57 (0.54)	81.41 (0.59)	79.00 (0.23)	72.91 (0.48)	66.22 (0.43)	82.72 (0.62)	81.96 (0.59)	81.14 (0.59)	78.80 (0.42)	74.00 (0.42)	67.43 (0.63)
Random Forest	Constant	92.07 (0.26)	90.70 (0.34)	89.13 (0.32)	84.30 (0.38)	74.95 (0.32)	64.06 (0.54)	92.00 (0.30)	91.15 (0.33)	90.27 (0.34)	87.68 (0.65)	80.87 (0.39)	70.25 (0.28)	91.69 (0.29)	90.85 (0.29)	90.49 (0.26)	87.67 (0.41)	81.50 (0.42)	71.04 (0.28)
	KNN	92.07 (0.26)	90.89 (0.23)	89.59 (0.21)	85.63 (0.19)	79.84 (0.38)	70.64 (0.34)	91.90 (0.33)	90.82 (0.39)	89.38 (0.31)	86.50 (0.40)	79.02 (0.46)	69.51 (0.36)	91.59 (0.28)	90.55 (0.32)	89.76 (0.28)	85.83 (0.44)	78.28 (0.52)	67.41 (0.31)
	MICE	92.07 (0.26)	90.93 (0.28)	89.70 (0.23)	85.70 (0.31)	76.85 (0.35)	66.75 (0.53)	91.98 (0.26)	91.15 (0.24)	90.10 (0.29)	87.75 (0.36)	81.14 (0.41)	70.88 (0.50)	91.85 (0.34)	90.96 (0.31)	90.42 (0.31)	87.41 (0.31)	81.54 (0.27)	72.04 (0.14)
	Intrinsic	92.29 (0.22)	91.19 (0.21)	89.43 (0.20)	85.74 (0.21)	78.12 (0.55)	66.54 (0.85)	92.33 (0.21)	91.63 (0.30)	90.80 (0.32)	88.40 (0.35)	81.60 (0.35)	69.99 (0.75)	92.31 (0.26)	91.56 (0.25)	91.04 (0.21)	88.36 (0.47)	82.03 (0.56)	71.33 (0.46)
SVM	Constant	81.57 (0.29)	81.22 (0.31)	80.28 (0.15)	78.21 (0.41)	73.64 (0.38)	65.45 (0.64)	81.39 (0.41)	80.66 (0.50)	80.22 (0.43)	78.47 (0.31)	73.84 (0.34)	67.14 (0.37)	81.26 (0.44)	80.63 (0.42)	80.13 (0.40)	77.72 (0.68)	73.56 (0.56)	67.36 (0.32)
	KNN	81.57 (0.29)	80.96 (0.32)	80.07 (0.15)	78.11 (0.13)	75.70 (0.47)	68.78 (0.60)	81.41 (0.44)	80.70 (0.47)	79.84 (0.31)	78.63 (0.25)	74.33 (0.47)	66.51 (0.43)	81.43 (0.47)	80.62 (0.54)	80.24 (0.41)	77.67 (0.42)	73.52 (0.59)	65.32 (0.59)
	MICE	81.57 (0.29)	81.34 (0.30)	80.56 (0.17)	78.75 (0.37)	75.00 (0.23)	67.43 (0.37)	81.41 (0.41)	80.85 (0.48)	80.53 (0.45)	79.22 (0.34)	75.71 (0.26)	69.38 (0.42)	81.35 (0.45)	80.90 (0.47)	80.43 (0.43)	78.43 (0.63)	75.34 (0.40)	69.61 (0.31)
TabNet	Constant	91.94 (0.13)	85.92 (0.76)	84.94 (0.55)	82.00 (0.64)	74.59 (0.48)	65.40 (0.99)	89.48 (0.68)	88.73 (0.66)	88.07 (0.48)	85.80 (0.54)	79.40 (0.71)	69.45 (0.66)	89.46 (0.70)	88.82 (0.63)	88.11 (0.63)	85.55 (0.70)	79.58 (0.50)	70.89 (0.52)
	KNN	91.94 (0.13)	85.66 (0.65)	84.62 (0.64)	81.96 (0.48)	76.89 (0.60)	68.91 (0.69)	89.84 (0.47)	88.76 (0.48)	87.93 (0.29)	85.12 (0.40)	78.12 (0.69)	69.06 (0.58)	89.34 (0.61)	88.64 (0.56)	87.68 (0.53)	84.05 (0.43)	77.46 (0.79)	67.89 (0.49)
	MICE	91.94 (0.13)	86.04 (0.72)	85.20 (0.53)	82.75 (0.57)	76.47 (0.64)	67.49 (1.23)	89.97 (0.30)	89.32 (0.31)	88.59 (0.27)	86.53 (0.48)	81.53 (0.58)	72.62 (0.60)	89.88 (0.45)	89.28 (0.42)	88.51 (0.44)	86.16 (0.38)	80.82 (0.35)	72.23 (0.42)
TabTransformer	Constant	90.53 (0.46)	84.95 (0.61)	83.65 (0.64)	79.38 (0.31)	69.33 (0.84)	60.86 (0.87)	89.52 (0.47)	88.44 (0.45)	87.20 (0.35)	83.48 (0.61)	73.52 (0.89)	63.43 (0.44)	89.09 (0.50)	88.10 (0.39)	87.09 (0.55)	83.41 (0.83)	75.95 (0.62)	63.92 (1.06)
	KNN	90.85 (0.47)	85.24 (0.60)	84.50 (0.62)	81.62 (0.57)	75.80 (0.33)	66.97 (0.66)	89.43 (0.55)	88.35 (0.55)	87.22 (0.34)	84.26 (0.75)	77.07 (0.62)	66.92 (0.31)	89.25 (0.50)	88.29 (0.47)	87.48 (0.43)	83.25 (0.48)	76.80 (0.56)	65.43 (0.62)
	MICE	90.56 (0.48)	84.10 (0.54)	83.03 (0.60)	80.12 (0.25)	71.83 (0.76)	63.55 (0.64)	89.34 (0.47)	88.37 (0.49)	87.11 (0.36)	84.02 (0.52)	76.33 (0.64)	65.75 (0.97)	89.06 (0.48)	88.24 (0.44)	87.18 (0.51)	84.06 (0.55)	77.80 (0.33)	67.19 (0.77)
XGBoost	Constant	**92.92 (0.19)**	87.44 (0.56)	85.45 (0.46)	80.82 (0.53)	70.42 (0.78)	60.87 (1.21)	90.56 (0.44)	89.93 (0.38)	89.12 (0.36)	86.60 (0.51)	80.37 (0.48)	69.22 (0.52)	90.75 (0.38)	89.93 (0.36)	89.57 (0.43)	86.89 (0.46)	80.44 (0.47)	71.10 (0.37)
	KNN	**92.92 (0.19)**	87.72 (0.44)	86.37 (0.47)	82.69 (0.30)	76.62 (0.35)	67.82 (0.48)	90.74 (0.35)	89.69 (0.33)	88.33 (0.22)	85.30 (0.38)	77.92 (0.54)	68.60 (0.19)	90.62 (0.45)	89.67 (0.46)	88.92 (0.45)	84.78 (0.50)	77.49 (0.57)	66.97 (0.61)
	MICE	**92.92 (0.19)**	87.69 (0.59)	86.07 (0.43)	81.87 (0.40)	72.04 (0.76)	62.44 (1.02)	90.64 (0.38)	89.92 (0.34)	88.86 (0.39)	86.58 (0.53)	80.24 (0.55)	70.21 (0.48)	90.48 (0.42)	89.64 (0.37)	89.03 (0.47)	86.41 (0.48)	80.17 (0.38)	70.80 (0.60)
	Intrinsic	**92.92 (0.19)**	90.14 (0.23)	87.44 (0.23)	79.46 (0.47)	68.05 (0.79)	57.69 (1.06)	**92.80 (0.26)**	91.85 (0.29)	90.75 (0.35)	87.45 (0.19)	78.12 (0.49)	65.04 (1.19)	**92.90 (0.17)**	91.89 (0.15)	**91.28 (0.28)**	88.05 (0.39)	81.20 (0.47)	70.02 (0.84)

Model	Strategy	Train missing percentage: 25%						Train missing percentage: 50%						Train missing percentage: 75%					
		Test missing percentage: 0%	5%	10%	25%	50%	75%	Test missing percentage: 0%	5%	10%	25%	50%	75%	Test missing percentage: 0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	92.05 (0.32)	91.40 (0.37)	90.66 (0.28)	**88.56 (0.49)**	**83.16 (0.47)**	73.15 (0.75)	90.83 (0.47)	90.36 (0.45)	89.88 (0.53)	87.74 (0.32)	**82.92 (0.63)**	74.07 (0.69)	89.50 (0.74)	89.08 (0.67)	**88.65 (0.79)**	**86.59 (0.62)**	81.39 (0.70)	73.62 (0.64)
NAIM w/o regularization	Intrinsic	91.80 (0.39)	91.08 (0.50)	90.03 (0.52)	87.15 (0.62)	79.81 (0.66)	68.51 (0.79)	90.89 (0.34)	90.14 (0.34)	89.47 (0.38)	87.05 (0.46)	80.24 (0.34)	70.49 (0.53)	89.67 (0.55)	88.98 (0.47)	88.52 (0.54)	85.99 (0.39)	80.05 (0.64)	70.99 (0.48)
	Constant	89.56 (0.66)	89.01 (0.63)	88.17 (0.63)	85.93 (0.50)	80.93 (0.73)	71.50 (0.58)	89.01 (0.58)	88.54 (0.51)	87.68 (0.68)	85.78 (0.82)	80.34 (0.71)	72.38 (0.90)	80.78 (4.17)	80.21 (4.13)	79.70 (3.92)	77.04 (3.72)	73.24 (3.48)	65.57 (2.32)
	KNN	89.41 (0.66)	88.47 (0.67)	87.54 (0.58)	84.46 (0.69)	77.73 (0.67)	66.47 (0.51)	88.70 (0.68)	87.92 (0.70)	86.92 (0.66)	83.85 (0.65)	77.11 (0.62)	67.18 (0.88)	84.07 (2.33)	83.16 (2.33)	82.47 (2.38)	78.82 (2.29)	72.98 (1.87)	63.75 (1.39)
	MICE	89.53 (0.54)	88.90 (0.56)	87.80 (0.58)	85.85 (0.46)	81.60 (0.53)	72.58 (0.40)	88.36 (0.24)	86.87 (0.39)	85.84 (0.33)	83.93 (0.30)	78.84 (0.64)	72.26 (0.49)	87.08 (0.99)	85.36 (1.23)	83.86 (1.56)	80.68 (1.41)	76.63 (1.35)	68.28 (1.84)
Adaboost	Constant	90.75 (0.29)	90.08 (0.31)	89.41 (0.27)	86.92 (0.42)	81.83 (0.70)	71.68 (0.67)	89.74 (0.32)	89.38 (0.33)	88.56 (0.37)	87.19 (0.32)	81.94 (0.49)	73.59 (0.55)	86.95 (1.00)	86.69 (1.02)	86.35 (0.89)	84.49 (0.83)	81.08 (0.79)	73.97 (0.35)
	KNN	90.48 (0.34)	89.51 (0.27)	88.64 (0.23)	85.05 (0.44)	77.73 (0.37)	67.40 (0.57)	87.99 (0.51)	87.25 (0.57)	86.47 (0.50)	84.03 (0.23)	78.45 (0.58)	69.04 (0.76)	83.32 (1.16)	82.91 (1.10)	82.21 (0.91)	79.79 (1.06)	75.60 (0.97)	67.29 (0.58)
	MICE	90.72 (0.38)	89.53 (0.42)	88.72 (0.34)	86.14 (0.47)	80.83 (0.64)	71.67 (0.43)	90.06 (0.33)	87.74 (0.24)	86.15 (0.80)	84.32 (0.63)	80.11 (0.42)	72.95 (0.71)	87.50 (0.68)	83.73 (1.01)	82.22 (1.03)	79.08 (0.80)	77.09 (1.18)	71.52 (0.30)
Decision Tree	Constant	71.35 (0.47)	70.38 (0.50)	69.62 (0.22)	67.65 (0.69)	63.74 (0.48)	58.70 (0.73)	66.31 (0.57)	66.07 (0.60)	66.23 (0.64)	65.36 (0.59)	61.94 (0.70)	59.18 (0.86)	60.57 (1.02)	60.51 (0.89)	60.34 (0.85)	60.45 (1.10)	59.91 (0.67)	58.32 (0.56)
	KNN	68.93 (0.80)	67.53 (0.93)	67.59 (0.72)	65.06 (0.86)	59.78 (0.51)	55.41 (0.59)	64.60 (1.06)	64.12 (1.00)	64.42 (0.85)	62.30 (1.02)	58.55 (0.73)	55.13 (0.29)	59.91 (0.43)	59.52 (0.52)	58.68 (0.48)	57.95 (0.42)	55.69 (0.35)	54.27 (0.80)
	MICE	70.86 (0.57)	70.17 (0.51)	68.84 (0.56)	66.50 (0.72)	62.50 (0.54)	57.39 (0.75)	67.24 (0.80)	65.93 (0.75)	65.19 (0.54)	62.76 (0.97)	60.71 (0.53)	57.44 (0.34)	61.79 (1.15)	61.00 (0.84)	60.48 (0.59)	59.24 (0.61)	57.82 (0.69)	55.72 (0.51)
	Intrinsic	74.22 (0.53)	73.39 (0.67)	72.51 (0.44)	69.15 (0.72)	63.92 (1.17)	57.38 (0.97)	70.77 (2.97)	70.64 (2.69)	69.61 (2.35)	68.55 (1.53)	64.65 (1.31)	60.26 (0.92)	60.97 (1.78)	61.55 (1.54)	61.84 (1.23)	62.21 (1.14)	61.96 (1.17)	59.36 (0.68)
FTTransformer	Constant	89.48 (0.55)	89.01 (0.60)	88.11 (0.56)	86.25 (0.60)	81.36 (0.67)	71.67 (0.48)	81.39 (7.42)	81.11 (7.35)	80.60 (7.23)	78.93 (6.85)	74.58 (5.83)	67.96 (4.27)	49.99 (0.01)	49.99 (0.01)	49.99 (0.01)	49.99 (0.01)	49.99 (0.01)	49.99 (0.01)
	KNN	89.47 (0.51)	88.68 (0.50)	87.63 (0.43)	85.07 (0.51)	78.05 (0.32)	67.20 (0.70)	88.13 (0.59)	87.44 (0.58)	86.57 (0.47)	83.44 (0.44)	76.93 (0.45)	68.25 (0.78)	57.41 (7.00)	57.20 (6.80)	57.14 (6.74)	56.36 (6.01)	55.33 (5.04)	53.65 (3.45)
	MICE	89.28 (0.68)	88.64 (0.75)	87.48 (0.76)	85.85 (0.59)	81.85 (0.68)	73.03 (0.40)	80.82 (7.31)	79.75 (7.05)	78.71 (6.82)	77.51 (6.51)	73.10 (5.46)	68.68 (4.42)	64.66 (8.47)	63.97 (8.08)	63.50 (7.83)	62.30 (7.12)	60.50 (6.07)	57.65 (4.43)
GRAPE	Intrinsic	84.60 (3.23)	82.07 (1.79)	83.58 (0.89)	78.21 (3.76)	77.54 (2.12)	64.59 (4.15)	85.95 (1.05)	84.83 (1.45)	78.55 (6.28)	82.01 (1.14)	79.70 (0.76)	71.69 (0.97)	85.43 (0.93)	85.47 (1.55)	82.29 (2.25)	83.29 (1.01)	79.08 (1.48)	73.72 (0.38)
HistGradientBoost	Constant	87.72 (0.25)	86.97 (0.25)	85.94 (0.38)	83.46 (0.60)	77.92 (0.44)	68.66 (0.64)	87.41 (0.43)	86.93 (0.42)	85.89 (0.47)	83.54 (0.55)	76.89 (0.48)	68.72 (1.10)	86.34 (0.60)	85.40 (0.66)	84.92 (0.42)	82.15 (0.21)	76.59 (0.62)	67.83 (0.42)
	KNN	86.11 (0.23)	85.09 (0.24)	83.95 (0.37)	80.96 (0.59)	73.97 (0.13)	64.36 (1.06)	83.53 (0.81)	82.66 (0.83)	81.79 (0.77)	79.14 (0.47)	73.71 (0.65)	65.07 (0.91)	76.86 (1.76)	76.23 (1.48)	76.24 (1.30)	73.42 (1.10)	69.00 (0.85)	62.23 (0.27)
	MICE	87.67 (0.34)	86.64 (0.39)	85.75 (0.40)	83.47 (0.23)	77.97 (0.19)	69.30 (0.57)	86.17 (0.70)	84.85 (0.67)	83.93 (0.45)	81.45 (0.63)	76.37 (0.73)	69.84 (0.82)	84.53 (0.47)	82.90 (0.32)	82.09 (0.27)	79.87 (0.29)	75.06 (0.26)	67.00 (0.53)
	Intrinsic	91.47 (0.23)	90.70 (0.28)	89.63 (0.23)	86.77 (0.44)	80.58 (0.31)	69.20 (0.46)	90.23 (0.59)	89.28 (0.64)	88.59 (0.61)	86.32 (0.81)	79.90 (0.58)	70.77 (0.21)	87.12 (0.50)	86.26 (0.49)	85.60 (0.47)	82.89 (0.43)	78.10 (0.65)	69.19 (0.51)
MLP	Constant	80.37 (0.65)	79.10 (0.68)	78.35 (0.70)	75.36 (0.68)	69.13 (0.76)	61.78 (1.02)	78.32 (0.84)	77.27 (0.94)	76.63 (0.80)	74.12 (0.90)	69.96 (0.86)	63.57 (0.71)	73.96 (0.50)	73.28 (0.40)	72.46 (0.62)	69.91 (0.46)	70.09 (0.38)	66.19 (0.60)
	KNN	77.71 (0.76)	77.09 (0.76)	76.17 (0.86)	74.99 (0.41)	67.17 (0.95)	60.40 (0.87)	72.96 (0.50)	72.41 (0.37)	72.43 (0.49)	69.41 (0.20)	65.55 (0.84)	58.22 (1.26)	66.87 (1.62)	66.53 (1.44)	66.06 (1.29)	63.93 (1.35)	62.06 (1.24)	56.14 (0.32)
	MICE	79.59 (0.60)	79.02 (0.52)	77.75 (0.59)	75.95 (0.64)	71.55 (1.53)	62.82 (1.99)	72.66 (0.97)	71.81 (0.89)	70.76 (1.13)	70.46 (0.50)	68.62 (0.58)	64.44 (0.70)	68.99 (0.74)	67.23 (0.77)	66.52 (1.06)	63.71 (0.55)	65.12 (0.63)	64.18 (1.02)
Random Forest	Constant	91.23 (0.26)	90.44 (0.31)	89.69 (0.28)	87.58 (0.30)	82.47 (0.49)	72.00 (0.43)	90.34 (0.42)	89.48 (0.41)	89.16 (0.38)	87.09 (0.57)	81.53 (0.53)	71.71 (0.60)	89.25 (0.44)	88.83 (0.41)	88.05 (0.38)	85.00 (0.59)	79.63 (0.29)	68.80 (0.30)
	KNN	91.07 (0.31)	89.94 (0.31)	89.02 (0.43)	85.62 (0.25)	77.57 (0.49)	67.50 (0.71)	88.79 (0.51)	87.76 (0.52)	87.15 (0.51)	84.17 (0.31)	78.06 (0.41)	68.47 (0.71)	83.58 (0.52)	82.97 (0.42)	82.26 (0.32)	79.99 (0.70)	75.67 (0.87)	66.26 (0.67)
	MICE	91.30 (0.26)	90.54 (0.40)	89.44 (0.32)	87.33 (0.34)	81.72 (0.56)	72.32 (0.46)	89.46 (0.37)	88.29 (0.31)	87.61 (0.40)	85.42 (0.72)	79.95 (0.54)	72.49 (0.47)	86.97 (0.62)	85.48 (0.52)	84.40 (0.58)	81.33 (0.45)	77.10 (0.42)	69.33 (0.42)
	Intrinsic	92.13 (0.29)	91.26 (0.37)	90.31 (0.35)	88.49 (0.30)	82.55 (0.64)	73.04 (0.46)	91.24 (0.26)	90.60 (0.31)	89.77 (0.29)	87.81 (0.38)	82.58 (0.51)	74.11 (0.38)	**90.05 (0.57)**	**89.26 (0.54)**	88.48 (0.42)	86.56 (0.54)	81.39 (0.40)	72.14 (0.38)
SVM	Constant	80.45 (0.36)	80.04 (0.30)	79.17 (0.42)	77.96 (0.57)	73.76 (0.40)	67.45 (0.80)	78.55 (0.51)	78.31 (0.46)	77.74 (0.49)	77.04 (0.37)	73.78 (0.55)	68.11 (0.46)	76.35 (0.98)	76.12 (0.92)	75.49 (0.97)	73.68 (1.16)	72.80 (1.06)	67.04 (0.71)
	KNN	80.03 (0.30)	79.62 (0.34)	78.75 (0.31)	77.76 (0.52)	71.57 (0.33)	64.84 (0.56)	78.09 (0.74)	77.70 (0.81)	77.28 (0.61)	75.34 (0.53)	70.84 (0.68)	64.16 (0.87)	75.46 (1.05)	75.26 (0.98)	74.26 (0.97)	72.01 (1.00)	69.92 (1.37)	62.48 (0.68)
	MICE	80.47 (0.37)	80.20 (0.34)	79.44 (0.49)	78.71 (0.61)	75.38 (0.33)	69.15 (0.83)	78.34 (0.63)	77.82 (0.63)	77.47 (0.61)	77.17 (0.47)	74.78 (0.51)	69.79 (0.39)	74.75 (0.97)	73.33 (0.86)	71.80 (0.99)	70.52 (0.81)	71.29 (0.95)	66.99 (0.50)
TabNet	Constant	88.97 (0.64)	88.49 (0.61)	87.51 (0.61)	85.27 (0.81)	80.26 (0.47)	70.77 (0.86)	88.15 (0.61)	87.42 (0.61)	87.01 (0.60)	84.96 (0.76)	78.86 (0.71)	70.77 (1.36)	87.02 (0.68)	86.31 (0.67)	85.99 (0.69)	83.93 (0.72)	79.19 (0.48)	72.09 (0.33)
	KNN	88.89 (0.58)	88.03 (0.62)	86.78 (0.57)	84.31 (0.56)	77.49 (0.55)	67.16 (0.34)	88.09 (0.47)	87.23 (0.55)	86.78 (0.37)	83.17 (0.52)	76.40 (0.33)	68.14 (0.70)	85.13 (0.97)	84.31 (0.91)	83.83 (0.77)	80.39 (0.95)	75.16 (0.74)	66.24 (0.59)
	MICE	89.09 (0.52)	88.38 (0.61)	87.24 (0.65)	85.85 (0.56)	81.55 (0.59)	72.66 (0.32)	87.96 (0.71)	86.15 (0.69)	85.42 (0.64)	83.09 (0.80)	78.38 (0.61)	72.49 (0.82)	87.10 (0.80)	83.62 (1.52)	82.36 (1.60)	80.15 (1.35)	77.66 (0.72)	70.72 (0.95)
TabTransformer	Constant	88.05 (0.42)	87.25 (0.47)	86.16 (0.49)	83.61 (0.48)	76.93 (0.69)	65.61 (1.07)	87.10 (0.68)	86.41 (0.68)	85.60 (0.64)	83.15 (0.58)	77.30 (0.58)	69.62 (0.64)	85.30 (0.93)	84.69 (0.84)	83.88 (0.99)	81.41 (0.87)	76.04 (0.92)	69.08 (0.62)
	KNN	88.53 (0.63)	87.63 (0.67)	86.42 (0.59)	83.80 (0.48)	76.46 (0.35)	65.22 (0.44)	86.63 (0.80)	85.87 (0.79)	85.09 (0.72)	81.67 (0.51)	74.57 (0.40)	66.66 (0.83)	84.48 (0.89)	83.60 (0.95)	82.59 (0.91)	79.00 (0.98)	73.44 (0.76)	65.45 (0.69)
	MICE	87.83 (0.62)	87.18 (0.72)	86.03 (0.61)	83.66 (0.64)	78.94 (0.45)	68.83 (0.86)	85.73 (0.72)	83.23 (0.72)	82.00 (0.59)	80.23 (0.75)	76.08 (0.38)	69.99 (0.45)	83.83 (0.93)	78.97 (1.18)	77.87 (1.38)	75.57 (1.42)	74.56 (1.10)	69.26 (0.85)
XGBoost	Constant	90.67 (0.29)	89.95 (0.39)	89.16 (0.35)	86.98 (0.46)	82.07 (0.62)	**73.38 (0.54)**	89.15 (0.69)	88.75 (0.66)	88.22 (0.59)	86.02 (0.77)	81.58 (0.64)	73.47 (0.69)	87.77 (0.51)	87.29 (0.61)	87.02 (0.48)	84.81 (0.44)	81.46 (0.32)	73.72 (0.35)
	KNN	90.10 (0.35)	88.97 (0.38)	88.33 (0.43)	85.19 (0.50)	77.64 (0.30)	66.30 (0.35)	87.81 (0.75)	86.95 (0.76)	86.20 (0.61)	83.42 (0.68)	77.10 (0.80)	68.23 (0.95)	83.86 (0.94)	83.09 (0.89)	82.55 (0.79)	80.19 (0.81)	75.30 (1.02)	67.37 (0.48)
	MICE	90.29 (0.42)	89.51 (0.51)	88.52 (0.42)	86.17 (0.51)	81.30 (0.46)	71.88 (0.40)	88.94 (0.59)	87.60 (0.52)	86.55 (0.57)	84.67 (0.64)	79.77 (0.53)	72.74 (0.62)	87.52 (0.72)	85.90 (0.70)	84.90 (0.78)	82.29 (0.56)	78.90 (0.37)	72.25 (0.55)
	Intrinsic	**92.79 (0.23)**	**92.05 (0.32)**	**91.03 (0.27)**	88.50 (0.42)	82.72 (0.42)	72.80 (0.56)	**91.28 (0.48)**	**90.74 (0.48)**	**89.90 (0.49)**	**88.07 (0.49)**	82.68 (0.58)	**74.33 (0.55)**	88.86 (0.90)	88.28 (0.79)	87.84 (0.59)	85.89 (0.42)	**81.75 (0.62)**	**74.30 (0.36)**

Table C.4: Average AUC performance and standard error (in brackets) of the experiments on the OnlineShoppers dataset. To facilitate the analysis we highlighted the best performance in each column in bold.

Model	Strategy	Train missing percentage: 0%						Train missing percentage: 5%						Train missing percentage: 10%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
		0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	**76.89 (1.19)**	**76.53 (1.41)**	**76.05 (1.28)**	74.46 (1.70)	71.71 (1.69)	68.41 (1.90)	**77.19 (0.91)**	**76.63 (0.81)**	76.00 (0.80)	75.40 (1.22)	69.31 (1.57)	62.68 (2.01)	76.56 (1.04)	76.48 (0.96)	75.89 (1.16)	74.46 (1.36)	68.62 (0.61)	63.54 (2.41)
NAIM w/o regularization	Intrinsic	74.91 (1.67)	72.74 (2.10)	68.80 (3.44)	64.38 (1.09)	56.16 (2.04)	54.28 (2.73)	76.23 (1.06)	74.11 (1.37)	72.06 (1.24)	69.24 (1.22)	59.64 (2.67)	57.84 (2.11)	77.19 (1.06)	74.69 (1.48)	74.75 (0.99)	70.63 (2.01)	59.94 (2.32)	55.37 (0.72)
	Constant	74.97 (1.18)	74.38 (1.27)	72.46 (1.25)	72.54 (0.89)	68.79 (1.14)	60.90 (0.98)	76.39 (1.67)	75.70 (1.75)	75.32 (1.35)	73.07 (1.76)	69.91 (2.59)	65.89 (3.60)	76.22 (1.18)	76.16 (1.27)	75.44 (1.52)	74.88 (1.17)	71.08 (1.07)	68.01 (1.84)
	KNN	75.24 (1.76)	75.42 (1.90)	74.75 (1.50)	74.57 (1.93)	72.40 (0.97)	71.26 (1.28)	75.47 (1.34)	74.87 (1.67)	75.17 (1.12)	74.76 (1.33)	72.64 (1.97)	67.55 (1.48)	75.60 (0.41)	75.49 (0.42)	75.11 (0.47)	74.99 (0.89)	72.92 (0.97)	69.74 (2.33)
	MICE	75.26 (1.18)	75.04 (1.05)	74.54 (1.22)	73.77 (1.47)	73.85 (0.84)	67.64 (1.86)	74.92 (1.32)	74.69 (1.50)	75.08 (1.13)	74.74 (1.69)	72.84 (2.95)	68.88 (1.25)	75.16 (1.60)	75.21 (1.59)	75.43 (1.54)	74.97 (1.47)	72.31 (1.87)	72.30 (1.94)
Adaboost	Constant	68.23 (0.74)	67.88 (1.08)	66.55 (1.23)	66.21 (2.03)	56.97 (2.11)	58.81 (2.74)	69.02 (1.76)	68.56 (2.10)	67.50 (1.49)	68.84 (1.23)	63.56 (3.17)	56.85 (1.57)	68.86 (0.73)	67.77 (0.33)	68.11 (0.89)	67.07 (1.27)	60.74 (1.57)	58.90 (1.99)
	KNN	68.23 (0.74)	68.67 (0.84)	68.37 (0.69)	69.97 (0.34)	65.56 (0.91)	65.18 (0.64)	68.35 (1.29)	67.89 (1.61)	67.65 (1.26)	69.16 (0.40)	65.33 (2.55)	59.82 (2.02)	68.81 (1.02)	68.93 (0.80)	68.72 (1.26)	69.14 (2.07)	66.81 (0.77)	64.51 (1.99)
	MICE	68.23 (0.74)	68.27 (0.94)	68.08 (0.76)	68.46 (0.97)	61.19 (1.33)	59.91 (2.13)	69.91 (0.76)	69.23 (1.04)	66.45 (1.94)	65.38 (2.17)	59.71 (1.86)	56.81 (1.99)	69.64 (1.00)	69.05 (0.62)	68.65 (1.48)	66.73 (1.52)	62.25 (0.97)	57.35 (2.21)
Decision Tree	Constant	55.84 (0.36)	54.28 (0.76)	55.10 (0.68)	51.65 (1.14)	53.83 (0.78)	50.01 (0.56)	52.67 (0.59)	51.55 (0.75)	51.29 (0.58)	50.75 (0.89)	50.02 (0.56)	51.36 (1.62)	58.21 (0.86)	56.42 (0.87)	55.98 (1.03)	55.63 (1.71)	54.31 (1.44)	50.99 (0.17)
	KNN	55.84 (0.36)	55.98 (0.49)	56.16 (0.25)	54.23 (0.83)	55.73 (1.38)	56.75 (1.27)	55.11 (0.70)	54.40 (0.96)	54.76 (0.34)	54.29 (1.83)	52.41 (1.03)	52.30 (1.18)	56.79 (1.77)	56.62 (1.21)	56.24 (1.63)	56.05 (1.59)	54.79 (1.13)	53.29 (1.38)
	MICE	55.84 (0.36)	55.59 (0.47)	54.98 (0.71)	53.93 (1.08)	54.12 (0.52)	51.35 (0.80)	56.75 (1.19)	56.26 (1.05)	55.95 (0.38)	54.46 (1.06)	52.63 (0.89)	50.69 (0.89)	57.22 (0.87)	56.26 (0.87)	55.59 (0.63)	53.72 (1.00)	52.96 (1.37)	49.52 (1.10)
	Intrinsic	55.84 (0.36)	55.44 (0.28)	55.44 (1.09)	54.61 (0.72)	54.86 (0.68)	51.02 (0.69)	56.24 (1.52)	54.67 (1.60)	54.44 (1.20)	51.73 (1.00)	53.11 (1.19)	51.54 (1.31)	55.63 (1.52)	54.64 (1.26)	52.91 (1.48)	51.82 (1.85)	51.39 (1.04)	50.00 (0.87)
FTTransformer	Constant	76.25 (1.16)	75.30 (0.90)	74.29 (1.09)	73.70 (0.97)	70.52 (1.33)	65.57 (2.79)	76.49 (1.18)	75.96 (1.22)	76.00 (0.92)	75.30 (1.34)	71.66 (3.30)	67.50 (1.25)	**77.35 (1.02)**	**76.92 (1.06)**	76.74 (1.19)	75.09 (1.37)	73.22 (1.22)	71.07 (2.53)
	KNN	75.24 (1.56)	74.59 (1.85)	75.13 (1.57)	75.06 (1.19)	**74.27 (1.48)**	**71.85 (1.27)**	76.60 (1.23)	76.27 (1.52)	76.37 (1.24)	76.06 (1.42)	73.91 (1.72)	68.99 (2.49)	76.83 (0.86)	76.81 (0.91)	**76.80 (1.12)**	**75.57 (0.94)**	**74.21 (1.66)**	71.34 (1.65)
	MICE	76.44 (1.20)	75.92 (1.18)	75.76 (1.29)	**75.35 (1.11)**	73.63 (0.65)	69.59 (1.85)	76.74 (1.21)	76.61 (1.18)	**76.95 (1.08)**	**76.49 (1.68)**	**73.96 (1.72)**	**69.08 (1.84)**	75.82 (1.22)	75.75 (1.27)	76.13 (1.25)	75.30 (1.29)	73.34 (1.58)	**74.05 (2.16)**
GRAPE	Intrinsic	76.29 (1.39)	75.53 (1.44)	76.00 (1.33)	73.36 (1.18)	71.29 (1.23)	66.64 (2.44)	75.07 (0.88)	75.26 (1.42)	70.12 (4.62)	72.39 (1.63)	68.00 (4.01)	60.81 (2.70)	75.91 (1.11)	70.44 (4.75)	71.53 (2.88)	70.25 (5.38)	69.76 (1.95)	60.72 (4.39)
HistGradientBoost	Constant	69.68 (1.87)	68.31 (1.80)	66.30 (1.88)	65.87 (3.34)	57.20 (1.31)	54.36 (1.40)	67.72 (1.51)	67.53 (2.19)	66.14 (2.22)	65.21 (1.50)	62.02 (2.46)	55.36 (2.02)	69.23 (1.62)	66.32 (1.80)	67.19 (1.91)	66.12 (2.17)	61.34 (1.06)	59.11 (2.25)
	KNN	69.68 (1.87)	69.76 (2.23)	69.47 (2.44)	69.70 (2.56)	64.72 (0.93)	62.28 (0.47)	68.09 (1.84)	66.21 (1.78)	65.95 (1.71)	66.42 (1.72)	62.49 (2.16)	61.25 (3.05)	68.19 (1.57)	67.64 (1.67)	67.61 (1.63)	66.10 (1.94)	67.21 (1.38)	62.27 (2.02)
	MICE	69.68 (1.87)	68.38 (2.34)	69.01 (2.33)	69.90 (1.85)	62.69 (1.84)	57.41 (1.89)	70.02 (1.78)	68.48 (1.64)	68.10 (1.29)	66.55 (1.42)	61.62 (2.40)	52.61 (2.42)	68.15 (2.09)	67.43 (2.37)	66.95 (2.23)	66.22 (1.62)	62.65 (3.08)	58.43 (1.31)
	Intrinsic	69.68 (1.87)	68.23 (1.78)	67.45 (2.26)	68.63 (3.01)	58.10 (3.18)	48.56 (1.53)	67.86 (1.60)	68.71 (1.90)	68.07 (1.89)	66.53 (1.97)	65.45 (2.73)	60.76 (1.78)	70.05 (1.46)	68.39 (1.77)	69.45 (1.27)	66.36 (1.48)	62.22 (2.52)	61.52 (3.12)
MLP	Constant	62.15 (2.21)	59.82 (1.29)	59.45 (2.84)	61.62 (2.24)	60.50 (2.75)	54.31 (3.36)	59.71 (2.63)	58.48 (2.24)	60.32 (2.49)	59.46 (2.68)	59.67 (2.06)	53.16 (3.10)	61.81 (1.88)	61.51 (1.03)	60.17 (1.81)	58.55 (3.69)	54.39 (2.20)	52.62 (4.19)
	KNN	62.15 (2.21)	62.15 (2.21)	61.80 (2.35)	64.77 (2.03)	61.90 (2.93)	63.28 (3.35)	61.52 (1.71)	60.98 (1.79)	60.73 (1.95)	64.47 (0.64)	61.38 (1.74)	59.42 (3.42)	62.85 (2.34)	64.01 (1.91)	62.52 (2.65)	63.30 (2.62)	60.91 (1.61)	57.60 (2.46)
	MICE	62.15 (2.21)	60.75 (1.69)	61.15 (1.98)	62.81 (1.40)	61.75 (2.79)	56.28 (2.80)	61.21 (2.73)	61.12 (2.88)	60.89 (2.88)	59.91 (3.11)	57.06 (3.50)	51.67 (3.55)	61.67 (2.47)	62.39 (2.12)	62.16 (2.15)	58.29 (4.04)	56.94 (2.12)	54.73 (4.01)
Random Forest	Constant	74.29 (1.55)	73.35 (0.73)	71.50 (1.50)	71.06 (1.66)	58.17 (1.27)	56.78 (2.37)	73.84 (1.23)	72.75 (1.13)	73.18 (1.40)	71.33 (0.82)	65.73 (2.55)	59.73 (1.44)	71.99 (1.20)	71.91 (1.22)	70.12 (1.51)	73.22 (1.03)	66.53 (0.28)	61.60 (1.19)
	KNN	74.29 (1.55)	74.45 (1.14)	74.44 (1.38)	74.69 (1.33)	71.53 (0.46)	69.25 (0.46)	73.44 (1.31)	73.32 (1.43)	73.51 (1.16)	73.16 (0.79)	69.90 (0.96)	65.54 (1.77)	73.96 (1.17)	73.73 (1.60)	73.13 (1.57)	72.78 (1.09)	71.42 (1.87)	66.50 (2.45)
	MICE	74.29 (1.55)	73.88 (1.28)	73.62 (1.11)	72.57 (1.67)	66.79 (1.91)	58.00 (3.23)	72.80 (1.39)	72.40 (1.49)	73.58 (1.19)	71.33 (1.50)	66.95 (2.84)	62.71 (1.79)	73.41 (1.44)	73.24 (1.57)	71.92 (1.86)	72.46 (1.27)	68.51 (1.59)	62.86 (3.26)
	Intrinsic	72.97 (1.14)	72.22 (0.80)	72.25 (1.44)	72.82 (2.15)	71.44 (1.00)	67.95 (1.82)	74.01 (1.45)	73.79 (1.38)	73.70 (1.08)	70.34 (1.87)	66.66 (2.59)	63.19 (3.01)	74.51 (1.29)	73.71 (1.36)	72.82 (1.69)	71.87 (1.14)	68.39 (1.50)	63.14 (0.86)
SVM	Constant	74.16 (1.68)	73.27 (1.72)	73.03 (1.49)	74.17 (1.18)	71.76 (2.07)	65.89 (2.84)	74.06 (1.17)	73.71 (1.02)	73.80 (0.59)	74.37 (1.49)	71.46 (2.46)	67.48 (3.05)	74.90 (1.39)	74.88 (1.44)	75.15 (1.47)	73.35 (2.18)	72.08 (1.38)	71.41 (3.00)
	KNN	74.16 (1.68)	74.16 (1.48)	74.37 (1.77)	74.82 (1.45)	73.77 (1.16)	70.60 (1.74)	73.65 (1.78)	73.40 (1.94)	73.95 (1.54)	75.07 (1.43)	71.78 (1.20)	67.47 (2.60)	75.42 (1.53)	75.92 (1.43)	76.06 (1.27)	74.97 (1.38)	73.95 (1.53)	71.61 (2.21)
	MICE	74.16 (1.68)	73.90 (1.73)	74.08 (1.57)	74.53 (1.61)	72.46 (1.27)	67.22 (2.29)	73.65 (1.73)	73.67 (1.71)	74.65 (1.32)	75.19 (1.94)	72.22 (1.78)	68.76 (2.63)	74.57 (1.56)	74.95 (1.50)	74.97 (1.53)	74.11 (1.64)	72.88 (1.29)	72.96 (2.63)
TabNet	Constant	73.24 (1.33)	72.25 (1.60)	73.10 (1.12)	68.80 (1.39)	66.95 (3.44)	62.31 (3.62)	73.24 (0.73)	72.46 (1.20)	71.95 (0.48)	71.42 (0.50)	66.40 (1.61)	58.03 (1.36)	72.75 (1.90)	72.24 (2.06)	70.60 (1.70)	70.07 (3.29)	66.07 (3.65)	61.19 (3.52)
	KNN	73.24 (1.33)	71.59 (1.81)	72.99 (1.16)	72.95 (2.04)	71.86 (0.76)	66.25 (1.53)	71.50 (2.33)	70.78 (2.22)	68.95 (3.34)	72.42 (2.62)	68.22 (1.23)	64.24 (2.35)	73.44 (1.20)	73.04 (1.39)	73.00 (1.51)	72.64 (2.76)	68.74 (2.42)	64.34 (1.44)
	MICE	73.24 (1.33)	72.94 (1.44)	73.44 (1.02)	70.74 (1.94)	67.83 (2.06)	68.65 (2.74)	70.28 (2.46)	70.41 (2.53)	69.71 (2.97)	71.49 (2.95)	70.17 (2.00)	63.53 (1.12)	74.26 (1.50)	74.13 (1.83)	73.41 (1.77)	71.89 (1.89)	70.58 (2.13)	72.36 (1.68)
TabTransformer	Constant	72.86 (2.58)	70.95 (2.43)	68.27 (2.47)	68.08 (2.39)	65.52 (2.16)	57.97 (4.81)	71.73 (1.62)	70.07 (1.23)	70.53 (1.34)	68.72 (0.92)	65.63 (2.87)	58.93 (2.86)	73.81 (1.90)	73.84 (1.90)	72.83 (2.17)	71.05 (3.31)	67.19 (2.17)	62.90 (3.73)
	KNN	72.89 (2.63)	72.95 (2.54)	72.22 (2.64)	72.82 (2.58)	70.99 (1.64)	67.32 (3.60)	72.89 (1.22)	72.29 (1.56)	72.63 (1.25)	73.15 (0.99)	70.39 (1.47)	66.77 (2.39)	71.87 (2.55)	72.60 (2.20)	71.69 (2.88)	71.19 (2.00)	72.29 (2.02)	68.08 (1.73)
	MICE	73.46 (2.50)	72.65 (2.48)	72.86 (2.51)	72.51 (1.68)	70.01 (2.77)	61.48 (3.87)	73.01 (2.58)	73.12 (2.67)	72.88 (2.00)	73.63 (2.10)	69.60 (2.98)	63.09 (1.58)	73.87 (2.11)	73.87 (1.97)	73.89 (2.10)	72.63 (2.58)	69.92 (1.22)	67.01 (1.92)
XGBoost	Constant	75.11 (0.75)	74.42 (0.66)	72.76 (0.64)	69.96 (1.09)	62.98 (2.51)	58.21 (3.03)	73.21 (0.29)	71.90 (0.95)	70.56 (0.69)	70.14 (0.85)	64.01 (2.25)	58.42 (2.12)	72.78 (0.98)	72.50 (0.89)	72.03 (1.16)	72.49 (0.47)	64.52 (0.95)	58.13 (2.67)
	KNN	75.11 (0.75)	75.23 (0.91)	74.30 (1.19)	74.43 (1.58)	69.25 (2.04)	67.21 (0.76)	74.57 (1.06)	74.45 (1.38)	73.73 (1.01)	73.83 (1.72)	69.65 (0.46)	66.26 (2.01)	74.71 (0.62)	74.82 (0.54)	74.17 (0.80)	73.39 (1.19)	71.95 (1.57)	65.34 (2.05)
	MICE	75.11 (0.75)	74.95 (0.89)	73.84 (0.67)	73.15 (1.91)	65.35 (2.00)	60.79 (2.77)	74.43 (1.14)	73.33 (0.97)	73.19 (0.89)	71.07 (1.66)	66.22 (2.36)	61.38 (1.95)	75.48 (1.07)	75.38 (0.85)	74.49 (0.89)	71.95 (1.10)	68.43 (1.40)	62.52 (4.20)
	Intrinsic	75.11 (0.75)	70.31 (0.73)	70.53 (1.04)	63.30 (2.82)	62.01 (0.54)	54.29 (1.79)	72.67 (1.26)	72.05 (1.41)	70.98 (1.60)	67.69 (1.53)	64.61 (3.06)	57.72 (2.81)	74.30 (1.30)	72.59 (1.07)	72.79 (0.98)	71.65 (1.16)	65.78 (2.46)	64.53 (2.54)

Model	Strategy	Train missing percentage: 25%						Train missing percentage: 50%						Train missing percentage: 75%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
		0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	**77.86 (0.89)**	77.39 (0.74)	**77.65 (0.93)**	**77.76 (1.72)**	69.59 (0.92)	64.77 (1.12)	75.41 (0.80)	75.31 (1.19)	75.44 (1.38)	72.07 (1.44)	65.85 (1.21)	62.30 (1.42)	73.04 (1.15)	71.97 (1.11)	72.04 (1.52)	68.67 (1.93)	68.52 (0.69)	59.30 (3.16)
NAIM w/o regularization	Intrinsic	74.26 (1.41)	74.02 (1.85)	72.92 (0.97)	70.87 (1.35)	65.61 (1.07)	57.92 (2.33)	74.04 (1.24)	75.23 (1.41)	73.37 (0.81)	70.30 (1.78)	65.41 (1.18)	56.71 (1.80)	73.30 (1.69)	70.69 (2.50)	73.54 (1.59)	66.78 (3.09)	62.82 (2.57)	60.50 (1.99)
	Constant	75.69 (1.30)	75.99 (1.24)	75.33 (1.87)	73.65 (2.15)	71.81 (1.91)	65.45 (1.91)	74.80 (1.02)	74.49 (1.19)	73.60 (0.90)	72.28 (1.06)	69.74 (1.12)	70.12 (1.31)	75.02 (1.31)	75.24 (1.40)	73.33 (1.61)	72.18 (1.44)	68.40 (2.16)	64.04 (1.92)
	KNN	75.56 (1.30)	75.94 (1.55)	75.43 (1.45)	75.39 (1.31)	73.30 (1.38)	66.86 (1.18)	75.37 (1.93)	75.30 (2.15)	75.11 (1.45)	73.40 (2.00)	69.18 (2.03)	66.02 (1.93)	75.63 (0.97)	74.99 (1.47)	75.50 (0.88)	74.61 (1.21)	70.10 (1.31)	60.46 (2.37)
	MICE	75.26 (1.82)	75.41 (1.73)	74.01 (2.38)	74.20 (2.60)	74.58 (0.72)	68.47 (2.90)	75.60 (1.20)	76.14 (1.52)	75.01 (1.78)	75.21 (1.34)	73.24 (1.52)	69.84 (1.45)	74.09 (1.12)	73.89 (1.89)	74.15 (0.78)	72.75 (1.19)	69.26 (1.24)	67.65 (1.24)
Adaboost	Constant	66.60 (2.27)	65.85 (2.45)	65.99 (2.35)	68.64 (1.31)	67.87 (0.90)	64.37 (1.71)	68.83 (1.58)	68.37 (1.91)	68.96 (1.60)	69.17 (1.43)	66.13 (1.37)	65.46 (1.75)	61.02 (2.08)	61.71 (1.65)	62.08 (2.26)	60.17 (1.95)	60.91 (2.65)	62.55 (2.38)
	KNN	65.95 (1.49)	65.20 (1.57)	66.16 (1.61)	68.11 (1.98)	66.49 (1.40)	65.36 (3.49)	68.26 (1.07)	68.41 (1.24)	67.78 (1.01)	67.23 (0.92)	64.61 (0.50)	63.84 (1.92)	60.33 (0.91)	60.98 (1.27)	60.27 (0.81)	60.54 (1.52)	58.42 (1.83)	60.24 (3.69)
	MICE	70.11 (0.91)	70.07 (1.10)	68.67 (1.28)	68.67 (1.42)	64.86 (2.56)	62.93 (2.52)	70.89 (1.45)	68.07 (1.66)	68.39 (1.91)	68.42 (0.77)	66.95 (1.31)	64.47 (2.51)	64.52 (2.65)	62.07 (3.33)	60.71 (2.68)	64.31 (2.58)	58.84 (2.45)	62.41 (2.73)
Decision Tree	Constant	54.62 (1.01)	55.50 (0.99)	53.72 (0.87)	54.03 (1.69)	55.57 (1.04)	58.67 (1.48)	54.59 (1.00)	54.95 (1.02)	55.92 (1.25)	55.82 (1.67)	56.22 (2.37)	54.09 (2.37)	48.97 (0.64)	49.54 (0.32)	49.45 (0.88)	49.66 (0.67)	51.41 (1.17)	53.56 (1.41)
	KNN	57.57 (0.79)	57.42 (0.51)	55.57 (0.75)	54.43 (1.43)	56.77 (0.81)	53.47 (1.30)	54.09 (1.39)	54.03 (1.59)	53.52 (0.95)	53.13 (0.99)	51.43 (1.06)	52.93 (1.00)	52.35 (0.89)	52.13 (0.85)	51.49 (0.65)	51.85 (1.11)	52.12 (1.40)	50.14 (1.27)
	MICE	57.61 (1.00)	57.77 (0.90)	56.34 (0.90)	58.17 (0.99)	56.19 (1.20)	52.72 (1.24)	56.65 (1.39)	55.87 (1.75)	55.42 (1.61)	55.23 (1.81)	52.69 (0.66)	55.09 (1.27)	55.04 (1.19)	53.62 (0.72)	55.56 (1.07)	54.25 (1.37)	52.81 (1.08)	50.10 (2.11)
	Intrinsic	58.60 (0.71)	58.26 (0.58)	56.79 (1.37)	56.50 (1.38)	55.14 (1.74)	55.29 (1.52)	53.35 (1.35)	52.99 (1.33)	54.74 (0.98)	56.78 (1.56)	54.66 (1.01)	53.71 (2.16)	47.98 (3.43)	47.72 (3.64)	49.27 (3.30)	50.91 (0.97)	49.62 (1.86)	57.04 (1.60)
FTTransformer	Constant	77.16 (1.00)	**77.51 (1.18)**	75.92 (1.13)	75.63 (1.55)	75.39 (0.81)	68.13 (1.92)	76.28 (1.12)	75.95 (1.23)	**75.90 (1.10)**	74.28 (1.43)	72.37 (1.08)	70.91 (0.80)	55.26 (4.62)	55.61 (4.95)	54.50 (3.90)	54.51 (3.91)	53.03 (2.52)	55.07 (4.44)
	KNN	76.97 (1.16)	77.17 (1.29)	75.73 (1.52)	76.58 (1.72)	75.22 (1.06)	66.61 (2.80)	75.42 (1.52)	75.06 (1.52)	75.07 (1.61)	74.01 (1.79)	71.27 (1.09)	66.64 (1.07)	**75.90 (0.97)**	**76.14 (1.10)**	**76.17 (1.18)**	**74.85 (0.85)**	70.08 (0.72)	61.77 (1.92)
	MICE	76.50 (1.29)	76.69 (1.31)	75.89 (1.36)	77.34 (1.49)	**75.54 (0.96)**	**69.75 (2.84)**	76.11 (1.56)	75.83 (1.51)	75.41 (1.54)	74.53 (1.95)	72.90 (1.43)	70.83 (0.35)	69.92 (4.83)	70.39 (4.89)	68.98 (4.63)	68.29 (4.38)	64.95 (3.80)	62.92 (3.34)
GRAPE	Intrinsic	76.03 (1.32)	70.18 (5.04)	75.84 (1.90)	73.61 (1.35)	70.56 (2.00)	64.23 (1.91)	**76.58 (1.16)**	75.74 (1.26)	74.97 (1.48)	72.89 (2.15)	65.55 (5.71)	61.00 (4.31)	75.47 (1.54)	72.52 (3.66)	73.30 (1.04)	71.90 (1.69)	65.95 (2.18)	59.77 (4.12)
HistGradientBoost	Constant	71.28 (2.03)	71.40 (2.08)	72.05 (1.91)	70.00 (1.72)	65.23 (1.53)	63.27 (2.12)	69.04 (1.23)	68.27 (1.73)	69.07 (1.02)	68.57 (0.88)	66.01 (2.46)	61.62 (3.14)	69.67 (0.83)	69.54 (1.99)	68.37 (1.27)	67.47 (1.86)	61.33 (2.54)	58.93 (1.10)
	KNN	69.97 (1.78)	69.12 (1.56)	70.10 (1.69)	66.47 (1.48)	66.58 (1.68)	61.70 (2.86)	69.43 (1.06)	68.42 (1.57)	69.42 (0.81)	66.26 (0.73)	65.44 (2.12)	62.31 (2.12)	71.66 (1.28)	71.68 (1.27)	71.63 (1.21)	69.97 (2.11)	63.44 (3.00)	59.01 (2.89)
	MICE	73.64 (0.93)	72.35 (1.36)	73.93 (0.61)	71.40 (1.52)	68.97 (1.37)	64.39 (2.67)	70.88 (2.31)	71.10 (2.27)	70.00 (2.12)	69.47 (0.84)	69.19 (1.47)	62.22 (2.55)	69.93 (2.24)	70.11 (2.17)	69.75 (1.45)	67.43 (2.72)	66.08 (3.68)	61.32 (1.69)
	Intrinsic	70.81 (1.02)	70.62 (1.11)	72.14 (1.65)	69.97 (0.96)	67.25 (1.26)	62.32 (2.36)	73.02 (1.02)	72.91 (1.14)	71.67 (1.15)	72.85 (0.99)	69.39 (2.45)	61.86 (2.52)	67.36 (4.80)	67.43 (4.31)	66.14 (2.66)	65.83 (2.20)	64.86 (2.26)	63.98 (2.16)
MLP	Constant	57.65 (2.58)	57.57 (3.30)	57.13 (1.48)	56.75 (2.73)	59.90 (2.50)	57.74 (1.53)	64.44 (1.42)	63.63 (1.41)	63.74 (0.26)	61.55 (1.22)	59.49 (1.83)	57.67 (1.90)	63.05 (4.85)	63.16 (4.57)	62.67 (4.42)	62.26 (4.34)	62.04 (2.03)	63.53 (2.97)
	KNN	54.81 (1.77)	54.73 (1.83)	55.34 (0.76)	57.76 (1.15)	58.38 (2.45)	55.03 (3.02)	57.41 (2.68)	57.02 (2.45)	53.67 (2.16)	57.90 (1.36)	57.07 (1.39)	50.85 (2.02)	51.29 (3.17)	50.93 (2.75)	50.39 (3.35)	47.35 (4.62)	48.73 (1.74)	48.55 (2.38)
	MICE	58.19 (2.33)	58.82 (2.39)	58.87 (1.40)	58.08 (1.64)	58.33 (2.50)	58.71 (1.24)	58.05 (1.54)	57.57 (1.47)	58.32 (1.56)	55.49 (2.66)	57.67 (1.55)	59.90 (1.44)	57.04 (4.30)	54.88 (4.32)	56.92 (3.10)	54.08 (4.08)	52.11 (3.04)	58.44 (1.81)
Random Forest	Constant	74.18 (1.23)	73.53 (1.05)	73.99 (1.27)	73.28 (1.69)	70.36 (1.37)	67.00 (2.02)	75.65 (1.16)	75.94 (1.40)	75.76 (1.23)	73.77 (1.19)	71.40 (2.44)	66.77 (1.75)	70.13 (1.99)	70.32 (1.93)	68.63 (2.17)	69.80 (0.97)	66.81 (1.93)	65.63 (1.15)
	KNN	74.78 (1.24)	75.33 (1.48)	74.61 (1.57)	74.02 (1.31)	71.70 (1.45)	68.46 (3.17)	74.72 (1.25)	73.95 (0.92)	74.73 (1.29)	**75.83 (1.14)**	71.50 (1.96)	65.36 (2.21)	71.71 (3.04)	71.48 (2.86)	70.12 (3.10)	70.26 (2.33)	66.96 (1.44)	60.67 (3.41)
	MICE	76.71 (0.79)	76.49 (0.97)	77.40 (0.58)	76.58 (0.79)	71.95 (1.30)	66.26 (1.37)	76.20 (1.12)	**76.80 (1.52)**	75.24 (1.33)	73.62 (0.71)	72.88 (0.80)	66.23 (1.35)	73.66 (0.92)	71.91 (1.19)	73.52 (1.61)	71.16 (0.71)	66.08 (2.26)	62.61 (2.37)
	Intrinsic	75.36 (1.30)	75.04 (1.59)	75.33 (1.59)	73.51 (1.27)	73.48 (0.65)	69.05 (1.43)	73.98 (0.84)	74.57 (1.69)	73.56 (0.29)	74.76 (1.19)	72.66 (1.49)	65.38 (1.44)	69.13 (4.01)	70.29 (3.38)	68.87 (3.63)	68.71 (2.48)	67.08 (2.85)	65.17 (0.70)
SVM	Constant	74.78 (1.19)	75.31 (1.24)	74.31 (1.42)	73.96 (1.72)	74.13 (0.80)	67.64 (0.80)	73.64 (1.31)	74.21 (1.06)	74.17 (1.32)	73.26 (1.92)	71.31 (1.52)	71.04 (0.81)	73.36 (1.54)	72.79 (1.48)	71.54 (1.19)	71.85 (1.34)	70.07 (0.79)	65.90 (1.67)
	KNN	74.32 (1.35)	74.61 (1.49)	74.06 (1.51)	73.97 (1.97)	72.70 (1.34)	67.93 (2.36)	73.85 (1.49)	74.85 (1.11)	73.94 (1.43)	73.81 (1.88)	70.88 (1.48)	67.04 (1.58)	73.01 (1.08)	72.38 (0.99)	70.56 (1.93)	69.72 (1.67)	66.10 (0.76)	61.49 (1.50)
	MICE	73.95 (1.24)	74.89 (1.25)	73.45 (1.38)	74.60 (1.57)	73.80 (1.23)	69.30 (1.87)	73.24 (1.75)	73.69 (1.18)	72.90 (2.22)	72.69 (1.75)	72.78 (2.21)	**71.96 (0.73)**	71.30 (2.38)	67.73 (4.51)	67.42 (2.38)	70.79 (1.93)	67.02 (1.58)	67.62 (1.83)
TabNet	Constant	70.87 (1.48)	71.15 (1.41)	70.44 (0.78)	69.03 (2.51)	67.36 (1.84)	60.54 (2.18)	66.25 (1.36)	67.05 (1.72)	66.76 (1.37)	66.10 (0.97)	64.67 (2.10)	60.98 (2.79)	67.18 (2.14)	66.38 (1.50)	63.04 (2.33)	59.72 (2.63)	61.58 (2.18)	60.67 (1.78)
	KNN	71.80 (1.75)	71.76 (1.62)	70.23 (1.82)	71.37 (2.20)	67.69 (3.26)	57.44 (3.76)	70.28 (1.08)	69.11 (1.38)	69.28 (1.73)	69.47 (1.42)	64.64 (2.04)	63.81 (2.31)	66.62 (1.68)	65.46 (1.65)	66.50 (2.50)	66.57 (1.44)	59.90 (1.40)	55.59 (2.98)
	MICE	70.98 (1.81)	71.49 (2.16)	70.75 (1.96)	71.35 (1.42)	69.00 (2.48)	66.67 (2.51)	69.05 (2.35)	69.50 (2.55)	67.11 (2.89)	68.08 (1.93)	70.57 (2.41)	69.80 (0.69)	63.49 (2.89)	61.41 (3.62)	62.89 (3.64)	64.99 (1.36)	64.00 (1.62)	64.97 (2.03)
TabTransformer	Constant	73.35 (1.64)	73.59 (1.82)	71.27 (2.25)	69.91 (2.00)	66.69 (1.65)	61.18 (2.20)	73.63 (2.25)	73.72 (1.95)	74.09 (1.96)	70.74 (1.79)	69.50 (1.32)	63.80 (0.96)	75.10 (1.33)	74.90 (1.59)	73.88 (1.49)	73.07 (2.50)	**70.37 (0.67)**	66.18 (1.30)
	KNN	70.61 (1.57)	71.16 (2.20)	70.64 (2.26)	70.65 (1.94)	67.65 (1.40)	62.12 (1.36)	73.18 (1.49)	72.11 (1.23)	73.23 (1.48)	73.32 (1.23)	66.87 (1.02)	64.09 (1.78)	69.69 (1.94)	69.98 (2.09)	69.50 (2.10)	70.17 (1.98)	65.39 (2.17)	58.56 (1.76)
	MICE	72.69 (2.39)	72.55 (2.23)	71.86 (1.83)	71.00 (2.43)	69.24 (1.59)	65.50 (1.44)	74.46 (1.90)	73.65 (1.45)	73.80 (1.65)	71.24 (1.83)	70.09 (1.98)	67.01 (0.71)	72.64 (1.72)	71.53 (2.49)	70.92 (1.12)	71.52 (2.15)	67.81 (1.43)	66.24 (1.39)
XGBoost	Constant	74.72 (0.64)	74.22 (0.82)	74.23 (1.18)	72.31 (0.91)	67.34 (2.11)	68.91 (1.57)	73.18 (1.58)	73.25 (2.08)	73.24 (1.42)	74.03 (1.60)	73.51 (2.20)	69.09 (2.67)	70.33 (1.23)	70.31 (1.25)	70.02 (1.77)	70.50 (1.58)	68.32 (1.99)	66.25 (0.31)
	KNN	74.83 (0.82)	74.81 (0.65)	74.19 (0.80)	71.14 (1.65)	68.70 (2.08)	64.55 (2.07)	74.04 (1.03)	74.11 (1.56)	73.30 (1.02)	71.22 (0.86)	68.42 (1.47)	63.22 (1.89)	74.20 (0.99)	74.86 (1.42)	73.43 (1.16)	71.72 (2.00)	66.66 (2.05)	62.08 (1.74)
	MICE	75.03 (0.57)	74.56 (0.67)	74.43 (0.21)	71.93 (0.95)	68.41 (0.25)	63.87 (2.77)	74.86 (1.42)	74.74 (2.39)	74.45 (1.84)	75.08 (0.56)	69.45 (0.75)	67.66 (1.44)	72.91 (0.84)	72.16 (1.75)	71.62 (1.14)	71.03 (1.69)	62.56 (1.78)	66.38 (1.98)
	Intrinsic	73.39 (1.95)	73.23 (1.39)	73.61 (1.52)	72.38 (1.75)	68.64 (2.63)	62.73 (1.84)	74.73 (0.68)	75.43 (0.53)	74.34 (1.24)	74.42 (1.03)	**74.35 (2.16)**	66.43 (1.19)	68.52 (0.82)	68.26 (0.96)	67.06 (0.81)	68.06 (1.48)	66.94 (1.72)	**68.48 (1.19)**

Table C.5: Average AUC performance and standard error (in brackets) of the experiments on the SeismicBumps dataset. To facilitate the analysis we highlighted the best performance in each column in bold.

		Train missing percentage: 0%						Train missing percentage: 5%						Train missing percentage: 10%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
Model	**Strategy**	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	98.50 (0.21)	98.37 (0.17)	98.18 (0.22)	97.47 (0.17)	95.13 (0.34)	88.86 (0.54)	98.40 (0.13)	98.25 (0.14)	98.14 (0.12)	**97.53 (0.14)**	94.96 (0.22)	89.43 (0.40)	98.44 (0.15)	98.33 (0.18)	98.06 (0.13)	97.45 (0.10)	95.37 (0.24)	89.15 (0.18)
NAIM w/o regularization	Intrinsic	98.13 (0.15)	97.05 (0.20)	95.65 (0.43)	91.38 (1.04)	80.25 (1.93)	66.44 (0.92)	98.20 (0.15)	97.69 (0.09)	97.15 (0.14)	93.50 (0.50)	84.69 (1.09)	69.37 (0.76)	98.14 (0.12)	97.80 (0.12)	97.19 (0.18)	95.25 (0.28)	88.08 (0.90)	74.29 (0.77)
	Constant	98.03 (0.22)	96.09 (0.19)	94.44 (0.39)	89.25 (0.63)	79.71 (1.11)	71.44 (1.85)	97.81 (0.15)	97.24 (0.13)	96.39 (0.21)	93.30 (0.12)	84.54 (0.97)	74.13 (0.61)	97.72 (0.18)	97.04 (0.15)	96.33 (0.29)	94.10 (0.27)	86.85 (0.72)	75.04 (1.39)
	KNN	98.19 (0.17)	97.78 (0.18)	97.19 (0.16)	95.72 (0.12)	92.64 (0.55)	88.43 (0.30)	97.75 (0.12)	97.54 (0.14)	97.06 (0.24)	95.35 (0.26)	92.52 (0.52)	84.63 (0.55)	97.78 (0.32)	97.50 (0.29)	97.04 (0.34)	95.37 (0.36)	91.04 (0.33)	82.39 (0.65)
	MICE	98.08 (0.17)	96.58 (0.25)	95.26 (0.40)	92.42 (0.42)	84.93 (1.15)	75.60 (1.91)	97.87 (0.14)	97.32 (0.19)	97.10 (0.24)	95.30 (0.26)	91.02 (0.52)	84.91 (0.72)	97.77 (0.18)	97.30 (0.17)	96.77 (0.25)	94.98 (0.41)	91.41 (0.55)	83.75 (0.77)
Adaboost	Constant	97.70 (0.14)	94.87 (0.53)	92.20 (1.34)	85.34 (2.12)	75.68 (1.94)	68.82 (1.76)	97.65 (0.22)	97.39 (0.23)	97.11 (0.21)	96.01 (0.09)	92.03 (0.32)	83.63 (0.73)	97.56 (0.27)	97.38 (0.29)	96.97 (0.24)	96.28 (0.17)	92.81 (0.45)	83.40 (0.29)
	KNN	97.70 (0.14)	97.18 (0.20)	96.40 (0.21)	94.72 (0.17)	91.69 (0.40)	86.81 (0.23)	97.58 (0.31)	97.19 (0.27)	96.75 (0.35)	94.84 (0.58)	91.77 (0.29)	83.24 (0.93)	97.66 (0.18)	97.29 (0.24)	96.83 (0.27)	95.50 (0.38)	91.01 (0.49)	82.62 (1.02)
	MICE	97.70 (0.14)	96.00 (0.25)	94.55 (0.55)	89.99 (1.15)	80.77 (1.74)	72.20 (2.17)	97.71 (0.21)	97.12 (0.35)	96.81 (0.23)	95.08 (0.34)	90.82 (0.69)	84.57 (1.07)	97.55 (0.13)	97.04 (0.11)	96.87 (0.14)	95.56 (0.14)	92.14 (0.44)	85.33 (0.48)
Decision Tree	Constant	90.50 (0.46)	85.87 (0.84)	82.00 (1.27)	71.43 (1.96)	61.36 (2.28)	53.65 (1.60)	90.32 (0.64)	89.21 (0.63)	87.89 (0.83)	82.98 (0.78)	76.73 (1.72)	66.23 (3.50)	88.80 (0.58)	88.51 (0.49)	86.64 (0.59)	84.56 (0.64)	79.02 (0.77)	70.16 (0.70)
	KNN	90.50 (0.46)	89.60 (0.39)	88.87 (0.35)	86.55 (0.21)	82.21 (0.57)	78.30 (0.38)	90.54 (0.52)	89.45 (0.45)	88.57 (0.42)	85.61 (0.63)	80.95 (0.26)	72.50 (0.51)	88.97 (0.31)	87.81 (0.52)	87.47 (0.46)	83.44 (0.42)	79.24 (0.47)	70.55 (0.64)
	MICE	90.50 (0.46)	88.11 (0.63)	84.59 (0.68)	78.72 (1.12)	69.04 (0.58)	60.07 (1.39)	89.85 (0.38)	88.40 (0.46)	87.03 (0.36)	82.60 (0.83)	75.35 (1.89)	69.60 (3.56)	89.64 (0.55)	87.70 (0.69)	86.18 (0.43)	83.31 (0.45)	78.03 (0.60)	70.33 (1.48)
	Intrinsic	90.50 (0.46)	89.62 (0.41)	89.08 (0.36)	86.14 (0.38)	78.58 (0.68)	67.70 (0.89)	90.57 (0.32)	88.77 (0.60)	87.21 (0.41)	81.40 (0.59)	72.08 (1.45)	61.61 (3.50)	90.47 (0.18)	89.47 (0.26)	87.70 (0.44)	83.91 (0.34)	75.83 (1.20)	64.04 (2.15)
FTTransformer	Constant	87.98 (8.95)	87.47 (8.83)	87.16 (8.76)	85.22 (8.31)	81.17 (7.43)	74.11 (5.80)	59.47 (8.93)	59.41 (8.87)	59.37 (8.84)	59.25 (8.72)	58.59 (8.10)	57.33 (6.91)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)
	KNN	78.44 (10.95)	78.22 (10.86)	78.06 (10.80)	77.49 (10.58)	76.19 (10.08)	73.03 (8.87)	87.79 (8.91)	87.45 (8.83)	87.33 (8.80)	86.40 (8.58)	84.62 (8.16)	79.40 (6.93)	78.31 (10.89)	78.15 (10.84)	77.99 (10.77)	77.24 (10.49)	75.26 (9.74)	70.23 (7.80)
	MICE	78.58 (11.00)	78.03 (10.79)	77.83 (10.72)	76.64 (10.26)	74.22 (9.34)	70.07 (7.75)	59.58 (9.03)	59.49 (8.94)	59.45 (8.91)	59.16 (8.64)	58.31 (7.84)	57.43 (7.00)	78.33 (10.90)	78.17 (10.84)	78.03 (10.79)	77.53 (10.59)	76.18 (10.08)	72.40 (8.62)
GRAPE	Intrinsic	90.72 (3.66)	89.82 (2.70)	76.58 (7.60)	86.72 (5.70)	87.88 (2.83)	58.74 (5.07)	91.31 (2.22)	91.22 (2.67)	74.25 (4.60)	73.59 (5.73)	77.03 (6.37)	66.18 (5.35)	92.39 (2.58)	94.97 (1.11)	80.03 (7.40)	69.30 (8.71)	78.89 (6.58)	74.75 (7.92)
HistGradientBoost	Constant	98.66 (0.10)	97.95 (0.08)	97.30 (0.09)	94.10 (0.34)	85.01 (0.97)	73.39 (0.87)	98.63 (0.12)	98.43 (0.14)	98.27 (0.14)	97.47 (0.25)	94.14 (0.30)	87.31 (0.27)	98.72 (0.10)	98.58 (0.10)	98.25 (0.13)	97.73 (0.17)	95.03 (0.38)	87.42 (0.30)
	KNN	98.66 (0.10)	98.42 (0.09)	98.10 (0.11)	97.13 (0.13)	94.52 (0.18)	89.89 (0.24)	98.51 (0.13)	98.33 (0.13)	98.05 (0.18)	96.73 (0.22)	94.12 (0.33)	86.76 (0.54)	98.54 (0.11)	98.42 (0.13)	98.04 (0.16)	97.02 (0.25)	93.38 (0.28)	84.96 (0.66)
	MICE	98.66 (0.10)	98.09 (0.10)	97.68 (0.11)	95.64 (0.25)	89.00 (0.61)	78.67 (0.77)	98.64 (0.09)	98.39 (0.15)	98.26 (0.15)	97.18 (0.28)	93.65 (0.27)	87.93 (0.61)	98.59 (0.08)	98.45 (0.09)	98.07 (0.13)	97.35 (0.19)	94.41 (0.40)	87.59 (0.25)
	Intrinsic	98.66 (0.10)	**98.55 (0.11)**	**98.33 (0.08)**	**97.55 (0.19)**	94.91 (0.09)	87.14 (0.53)	**98.72 (0.13)**	**98.50 (0.13)**	**98.31 (0.16)**	97.36 (0.25)	94.25 (0.47)	87.77 (0.35)	**98.79 (0.10)**	**98.65 (0.09)**	**98.35 (0.12)**	**97.78 (0.18)**	95.39 (0.25)	88.18 (0.34)
MLP	Constant	97.63 (0.18)	97.18 (0.15)	96.72 (0.26)	94.89 (0.20)	89.16 (0.73)	79.11 (0.46)	97.60 (0.22)	97.30 (0.26)	96.90 (0.16)	95.72 (0.11)	90.95 (0.59)	82.79 (0.35)	97.36 (0.18)	97.02 (0.17)	96.63 (0.17)	94.95 (0.17)	91.34 (0.53)	81.11 (0.89)
	KNN	97.63 (0.18)	97.35 (0.14)	97.06 (0.34)	96.07 (0.21)	93.59 (0.35)	88.68 (0.32)	97.49 (0.19)	97.21 (0.17)	96.77 (0.17)	95.35 (0.27)	92.60 (0.29)	84.78 (0.53)	97.33 (0.16)	96.90 (0.25)	96.45 (0.30)	95.16 (0.36)	91.79 (0.48)	83.67 (0.51)
	MICE	97.63 (0.18)	97.14 (0.12)	96.82 (0.30)	95.44 (0.16)	91.56 (0.29)	85.28 (0.17)	97.20 (0.14)	97.02 (0.18)	96.71 (0.15)	96.03 (0.19)	92.91 (0.32)	87.61 (0.34)	97.10 (0.15)	96.77 (0.13)	96.63 (0.22)	95.65 (0.23)	93.11 (0.41)	86.32 (0.38)
Random Forest	Constant	98.54 (0.19)	98.01 (0.17)	97.50 (0.24)	94.44 (0.17)	85.90 (0.58)	75.46 (1.32)	98.29 (0.19)	98.01 (0.14)	97.94 (0.15)	97.06 (0.34)	93.46 (0.49)	86.73 (0.31)	98.17 (0.20)	98.00 (0.17)	97.54 (0.31)	97.03 (0.22)	94.14 (0.42)	87.31 (0.31)
	KNN	98.54 (0.19)	98.39 (0.14)	98.14 (0.21)	97.23 (0.21)	95.08 (0.26)	**90.64 (0.30)**	98.42 (0.21)	98.18 (0.22)	98.14 (0.22)	97.02 (0.26)	94.49 (0.32)	87.45 (0.41)	98.35 (0.24)	98.21 (0.20)	97.84 (0.34)	96.77 (0.37)	93.60 (0.35)	85.23 (0.48)
	MICE	98.54 (0.19)	98.23 (0.18)	97.91 (0.21)	96.20 (0.24)	91.00 (0.23)	82.29 (0.72)	98.27 (0.17)	98.02 (0.19)	98.03 (0.17)	96.94 (0.40)	93.34 (0.34)	88.23 (0.54)	98.04 (0.21)	98.00 (0.18)	97.55 (0.24)	96.82 (0.30)	93.92 (0.33)	87.31 (0.22)
	Intrinsic	98.54 (0.19)	98.41 (0.15)	98.19 (0.18)	97.49 (0.18)	**95.24 (0.20)**	88.66 (0.45)	98.36 (0.19)	98.20 (0.16)	98.08 (0.18)	97.51 (0.29)	**95.20 (0.32)**	**89.98 (0.17)**	98.34 (0.20)	98.23 (0.14)	97.82 (0.25)	97.33 (0.26)	**95.51 (0.34)**	**89.34 (0.44)**
SVM	Constant	97.14 (0.15)	96.92 (0.17)	96.79 (0.13)	95.83 (0.14)	93.54 (0.17)	87.04 (0.34)	97.01 (0.11)	96.80 (0.17)	96.63 (0.13)	95.75 (0.15)	93.46 (0.44)	88.84 (0.17)	96.97 (0.14)	96.80 (0.19)	96.53 (0.13)	95.80 (0.14)	93.55 (0.29)	86.89 (0.34)
	KNN	97.14 (0.15)	96.83 (0.22)	96.69 (0.17)	95.44 (0.17)	93.29 (0.21)	88.86 (0.21)	96.99 (0.14)	96.64 (0.23)	96.43 (0.20)	95.13 (0.20)	92.94 (0.38)	86.82 (0.45)	97.00 (0.15)	96.79 (0.22)	96.36 (0.21)	95.45 (0.13)	92.25 (0.38)	84.57 (0.46)
	MICE	97.14 (0.15)	96.89 (0.17)	96.75 (0.15)	95.76 (0.14)	93.40 (0.13)	87.62 (0.29)	97.03 (0.12)	96.79 (0.18)	96.62 (0.15)	95.62 (0.18)	93.33 (0.38)	88.87 (0.40)	97.03 (0.13)	96.89 (0.17)	96.56 (0.14)	95.76 (0.15)	93.57 (0.22)	87.40 (0.20)
TabNet	Constant	97.21 (0.23)	96.77 (0.24)	96.34 (0.23)	93.95 (0.38)	87.86 (0.80)	78.59 (0.51)	97.21 (0.26)	96.96 (0.27)	96.38 (0.21)	95.26 (0.47)	91.21 (0.66)	84.40 (0.52)	97.43 (0.14)	97.23 (0.13)	96.73 (0.30)	95.85 (0.15)	92.59 (0.65)	84.16 (0.55)
	KNN	97.21 (0.23)	96.97 (0.16)	96.66 (0.21)	95.28 (0.22)	93.12 (0.19)	88.56 (0.14)	97.28 (0.25)	96.88 (0.24)	96.61 (0.13)	94.91 (0.47)	92.94 (0.43)	86.00 (0.25)	97.17 (0.12)	97.01 (0.15)	96.31 (0.11)	94.98 (0.22)	91.39 (0.66)	83.19 (0.41)
	MICE	97.21 (0.23)	96.86 (0.22)	96.74 (0.13)	95.35 (0.11)	91.78 (0.37)	84.99 (0.35)	97.18 (0.12)	96.95 (0.14)	96.62 (0.08)	95.46 (0.23)	92.69 (0.44)	87.99 (0.38)	96.84 (0.34)	96.66 (0.29)	96.30 (0.39)	95.58 (0.38)	93.40 (0.40)	87.19 (0.28)
TabTransformer	Constant	98.02 (0.17)	97.31 (0.21)	96.72 (0.23)	94.53 (0.12)	89.24 (0.44)	80.18 (0.33)	97.79 (0.21)	97.32 (0.22)	96.91 (0.21)	95.46 (0.32)	90.77 (0.49)	83.70 (0.42)	97.71 (0.18)	97.22 (0.12)	96.69 (0.31)	95.52 (0.10)	91.28 (0.64)	82.90 (0.34)
	KNN	98.09 (0.19)	97.83 (0.15)	97.33 (0.25)	96.11 (0.18)	93.25 (0.38)	88.29 (0.32)	97.65 (0.17)	97.28 (0.24)	96.81 (0.19)	95.38 (0.21)	92.38 (0.26)	85.34 (0.54)	97.70 (0.17)	97.33 (0.19)	96.68 (0.30)	95.29 (0.12)	91.77 (0.35)	83.30 (0.64)
	MICE	98.04 (0.20)	97.40 (0.21)	96.93 (0.20)	95.12 (0.28)	91.13 (0.21)	84.32 (0.66)	97.87 (0.15)	97.42 (0.21)	97.10 (0.13)	96.05 (0.23)	93.08 (0.43)	88.09 (0.52)	97.50 (0.21)	97.06 (0.18)	96.93 (0.28)	96.04 (0.05)	93.40 (0.35)	87.24 (0.19)
XGBoost	Constant	**98.73 (0.14)**	97.30 (0.14)	96.22 (0.11)	91.03 (0.76)	81.41 (1.34)	72.36 (1.21)	98.58 (0.13)	98.37 (0.15)	98.19 (0.13)	97.13 (0.24)	93.53 (0.23)	85.81 (0.15)	98.55 (0.16)	98.45 (0.17)	97.94 (0.17)	97.29 (0.17)	94.14 (0.44)	86.32 (0.28)
	KNN	**98.73 (0.14)**	98.38 (0.07)	97.99 (0.23)	96.91 (0.13)	93.90 (0.33)	89.44 (0.25)	98.58 (0.13)	98.35 (0.13)	97.98 (0.16)	96.63 (0.19)	93.71 (0.34)	86.13 (0.39)	98.54 (0.13)	98.35 (0.12)	97.94 (0.16)	96.77 (0.29)	92.58 (0.42)	84.44 (0.67)
	MICE	**98.73 (0.14)**	97.71 (0.18)	97.15 (0.13)	94.40 (0.30)	87.28 (0.62)	77.91 (0.95)	98.56 (0.13)	98.24 (0.12)	98.04 (0.13)	96.79 (0.32)	92.84 (0.26)	87.26 (0.39)	98.44 (0.12)	98.17 (0.12)	97.81 (0.15)	96.85 (0.26)	93.58 (0.39)	86.49 (0.17)
	Intrinsic	**98.73 (0.14)**	95.78 (0.28)	92.84 (0.38)	80.42 (1.00)	67.74 (1.26)	62.52 (0.92)	98.64 (0.13)	98.43 (0.15)	98.18 (0.10)	97.14 (0.17)	93.54 (0.45)	84.72 (0.52)	98.65 (0.09)	98.44 (0.10)	98.13 (0.12)	97.44 (0.22)	94.15 (0.17)	85.55 (0.93)

		Train missing percentage: 25%						Train missing percentage: 50%						Train missing percentage: 75%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:					
Model	**Strategy**	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%	0%	5%	10%	25%	50%	75%
NAIM	Intrinsic	98.37 (0.12)	98.24 (0.14)	98.05 (0.08)	97.36 (0.13)	95.11 (0.44)	89.03 (0.49)	97.93 (0.22)	97.77 (0.25)	97.73 (0.27)	97.13 (0.28)	95.06 (0.21)	89.16 (0.52)	97.38 (0.24)	97.25 (0.25)	**97.14 (0.27)**	**96.70 (0.29)**	**95.38 (0.25)**	90.34 (0.31)
NAIM w/o regularization	Intrinsic	98.21 (0.12)	97.95 (0.12)	97.60 (0.15)	96.08 (0.23)	90.92 (0.63)	77.07 (1.43)	97.78 (0.17)	97.70 (0.18)	97.50 (0.13)	96.54 (0.29)	93.88 (0.49)	83.60 (0.38)	97.45 (0.16)	**97.31 (0.16)**	97.11 (0.19)	96.43 (0.23)	94.54 (0.40)	88.77 (0.56)
	Constant	97.63 (0.13)	97.30 (0.12)	97.05 (0.15)	95.81 (0.21)	91.90 (0.42)	81.48 (1.41)	96.89 (0.33)	96.69 (0.42)	96.36 (0.50)	95.49 (0.57)	93.22 (0.51)	86.71 (0.89)	95.83 (0.42)	95.08 (0.59)	94.57 (0.56)	93.69 (0.22)	92.85 (0.18)	88.45 (0.45)
	KNN	97.66 (0.20)	97.31 (0.28)	96.75 (0.25)	95.10 (0.25)	90.56 (0.48)	79.42 (0.49)	96.59 (0.22)	96.36 (0.22)	95.96 (0.29)	94.82 (0.30)	90.22 (0.43)	77.77 (0.90)	94.66 (0.78)	94.24 (0.80)	94.09 (0.68)	92.70 (0.72)	88.24 (0.69)	78.26 (0.39)
	MICE	96.93 (0.30)	96.23 (0.39)	95.72 (0.39)	94.49 (0.47)	91.35 (0.64)	85.03 (0.61)	95.04 (0.87)	94.16 (0.47)	93.58 (0.51)	93.34 (0.25)	91.74 (0.38)	85.39 (0.62)	95.02 (0.83)	93.37 (1.08)	92.81 (1.25)	92.06 (1.14)	91.27 (0.32)	87.17 (0.68)
Adaboost	Constant	97.69 (0.13)	97.57 (0.10)	97.33 (0.14)	96.25 (0.23)	94.29 (0.17)	86.22 (0.39)	97.02 (0.06)	96.98 (0.03)	96.73 (0.12)	96.26 (0.08)	94.35 (0.34)	88.49 (0.31)	96.05 (0.25)	95.93 (0.26)	95.70 (0.29)	95.54 (0.26)	93.72 (0.28)	89.13 (0.51)
	KNN	97.38 (0.17)	97.12 (0.22)	96.68 (0.27)	95.44 (0.27)	91.78 (0.56)	80.59 (0.65)	96.20 (0.18)	96.08 (0.19)	95.84 (0.21)	94.64 (0.27)	90.60 (0.26)	78.61 (0.54)	94.63 (0.14)	94.25 (0.13)	94.20 (0.09)	92.88 (0.23)	89.07 (0.34)	79.95 (0.42)
	MICE	97.54 (0.13)	95.97 (0.45)	95.61 (0.40)	95.10 (0.27)	92.96 (0.18)	86.12 (0.42)	96.46 (0.21)	94.01 (0.68)	93.31 (0.93)	93.24 (0.37)	92.58 (0.39)	86.08 (0.40)	96.07 (0.25)	91.67 (1.02)	89.77 (0.87)	89.86 (0.70)	90.56 (0.35)	87.45 (0.59)
Decision Tree	Constant	86.82 (0.53)	85.97 (0.34)	85.46 (0.55)	84.44 (0.58)	80.05 (0.76)	71.28 (0.62)	79.03 (2.12)	79.73 (1.63)	79.97 (1.31)	80.71 (1.08)	79.77 (0.43)	73.98 (0.48)	60.23 (5.64)	62.37 (4.91)	63.56 (4.26)	66.54 (2.90)	71.56 (0.87)	72.87 (0.73)
	KNN	87.49 (0.49)	86.43 (0.44)	86.10 (0.63)	84.43 (0.44)	78.89 (0.29)	67.71 (0.75)	83.76 (0.68)	83.37 (0.69)	82.14 (0.69)	79.90 (0.72)	74.57 (0.83)	64.57 (0.38)	75.57 (0.80)	75.23 (0.69)	74.79 (0.73)	71.16 (0.79)	67.31 (0.40)	62.08 (0.40)
	MICE	88.45 (0.31)	84.87 (0.86)	83.88 (0.43)	82.24 (0.62)	79.14 (0.52)	71.39 (0.77)	84.88 (0.81)	81.05 (0.64)	80.27 (0.55)	77.94 (0.60)	76.85 (0.59)	73.60 (0.74)	72.76 (0.76)	69.22 (0.74)	67.29 (0.83)	65.85 (0.76)	68.12 (1.33)	70.89 (0.86)
	Intrinsic	87.39 (0.37)	86.86 (0.60)	86.35 (0.61)	85.91 (0.51)	81.34 (0.55)	71.25 (0.34)	81.20 (3.20)	81.48 (2.38)	82.50 (1.90)	82.36 (1.09)	79.88 (0.45)	73.36 (0.47)	69.63 (4.53)	70.03 (4.16)	70.54 (3.65)	72.79 (2.00)	76.39 (0.76)	75.45 (1.08)
FTTransformer	Constant	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)
	KNN	68.75 (10.82)	68.67 (10.78)	68.57 (10.72)	68.29 (10.56)	67.04 (9.84)	62.57 (7.26)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)
	MICE	78.16 (10.84)	76.03 (10.06)	76.46 (10.20)	77.17 (10.46)	76.07 (10.04)	72.37 (8.61)	86.98 (8.72)	79.98 (7.24)	81.80 (7.58)	84.57 (8.15)	83.82 (7.98)	79.51 (6.96)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)	50.00 (0.00)
GRAPE	Intrinsic	85.72 (4.28)	85.59 (3.59)	80.23 (4.49)	86.30 (3.87)	71.46 (4.28)	56.06 (2.19)	92.18 (2.61)	83.33 (6.30)	87.35 (4.84)	80.24 (9.12)	79.89 (8.30)	71.40 (7.70)	94.47 (0.56)	94.91 (0.67)	95.62 (0.36)	95.32 (0.40)	88.87 (4.46)	88.42 (0.43)
HistGradientBoost	Constant	98.54 (0.12)	98.36 (0.12)	98.23 (0.14)	97.52 (0.10)	95.62 (0.06)	89.24 (0.32)	98.16 (0.11)	98.11 (0.12)	97.80 (0.11)	97.23 (0.16)	95.36 (0.30)	89.38 (0.49)	97.19 (0.16)	97.07 (0.17)	96.77 (0.28)	96.39 (0.14)	94.35 (0.33)	89.92 (0.62)
	KNN	98.36 (0.15)	98.17 (0.18)	97.93 (0.18)	96.98 (0.16)	93.67 (0.30)	82.67 (0.59)	97.75 (0.08)	97.57 (0.11)	97.25 (0.11)	96.25 (0.15)	92.54 (0.31)	81.00 (0.38)	96.14 (0.14)	95.88 (0.10)	95.62 (0.17)	94.37 (0.19)	90.13 (0.48)	81.28 (0.47)
	MICE	98.16 (0.14)	97.68 (0.17)	97.40 (0.16)	96.75 (0.17)	93.93 (0.09)	87.09 (0.48)	97.61 (0.05)	97.04 (0.07)	96.62 (0.08)	96.08 (0.14)	94.31 (0.34)	87.77 (0.47)	96.06 (0.22)	95.38 (0.38)	94.79 (0.22)	94.50 (0.24)	92.57 (0.28)	88.50 (0.55)
	Intrinsic	**98.65 (0.15)**	**98.54 (0.14)**	**98.33 (0.17)**	**97.73 (0.10)**	**95.71 (0.16)**	89.17 (0.38)	**98.29 (0.15)**	**98.21 (0.14)**	**98.00 (0.13)**	**97.38 (0.17)**	**95.61 (0.23)**	89.57 (0.45)	**97.46 (0.12)**	97.28 (0.11)	97.08 (0.18)	96.57 (0.23)	94.75 (0.35)	90.43 (0.63)
MLP	Constant	97.11 (0.17)	96.85 (0.17)	96.48 (0.16)	95.40 (0.24)	92.33 (0.22)	83.73 (0.66)	96.32 (0.31)	96.05 (0.26)	95.72 (0.33)	94.68 (0.40)	91.28 (0.49)	84.55 (0.82)	93.67 (0.47)	93.43 (0.55)	93.04 (0.39)	92.45 (0.40)	90.03 (0.48)	84.62 (0.57)
	KNN	96.44 (0.18)	95.95 (0.15)	95.89 (0.25)	93.85 (0.25)	90.21 (0.59)	78.91 (0.76)	94.34 (0.41)	94.04 (0.38)	93.68 (0.39)	92.34 (0.20)	86.44 (0.44)	74.02 (0.64)	92.40 (0.13)	91.88 (0.11)	91.31 (0.30)	89.32 (0.15)	82.97 (0.42)	73.43 (0.54)
	MICE	96.08 (0.17)	87.98 (0.70)	90.65 (0.60)	94.12 (0.26)	92.44 (0.24)	86.01 (0.38)	93.31 (0.94)	76.13 (1.41)	81.23 (1.27)	88.83 (0.48)	89.88 (0.69)	84.52 (0.30)	89.75 (0.62)	70.86 (1.10)	68.65 (1.95)	76.01 (1.23)	84.18 (0.77)	82.28 (0.42)
Random Forest	Constant	97.83 (0.21)	97.65 (0.25)	97.38 (0.24)	96.52 (0.35)	95.08 (0.25)	88.42 (0.43)	97.14 (0.14)	97.03 (0.13)	96.81 (0.13)	96.23 (0.17)	94.41 (0.40)	89.01 (0.60)	95.56 (0.23)	95.62 (0.28)	95.20 (0.20)	95.29 (0.26)	93.55 (0.51)	88.96 (0.71)
	KNN	97.99 (0.22)	97.76 (0.23)	97.49 (0.27)	96.27 (0.26)	93.49 (0.27)	83.43 (0.50)	96.73 (0.21)	96.50 (0.23)	96.22 (0.21)	95.00 (0.30)	91.70 (0.44)	81.48 (0.49)	94.48 (0.32)	94.18 (0.30)	94.10 (0.41)	92.91 (0.24)	88.61 (0.54)	79.78 (0.55)
	MICE	97.39 (0.25)	96.96 (0.28)	96.67 (0.35)	95.88 (0.21)	93.55 (0.25)	86.83 (0.52)	96.45 (0.12)	95.97 (0.10)	95.66 (0.18)	95.11 (0.18)	92.90 (0.30)	87.48 (0.34)	94.69 (0.28)	93.90 (0.30)	93.53 (0.25)	93.14 (0.30)	91.34 (0.46)	86.89 (0.71)
	Intrinsic	98.13 (0.22)	97.94 (0.25)	97.76 (0.23)	96.93 (0.30)	95.37 (0.24)	**89.80 (0.39)**	97.73 (0.20)	97.56 (0.17)	97.28 (0.19)	96.84 (0.22)	94.75 (0.24)	89.37 (0.43)	96.80 (0.16)	96.59 (0.18)	96.43 (0.26)	96.07 (0.29)	94.23 (0.46)	89.83 (0.70)
SVM	Constant	96.64 (0.06)	96.38 (0.13)	96.32 (0.10)	95.38 (0.12)	93.66 (0.31)	87.61 (0.30)	95.52 (0.18)	95.38 (0.21)	95.22 (0.19)	94.54 (0.25)	92.62 (0.34)	87.56 (0.30)	92.79 (0.42)	92.54 (0.50)	92.39 (0.50)	92.01 (0.44)	91.32 (0.55)	87.01 (0.76)
	KNN	96.81 (0.04)	96.64 (0.11)	96.33 (0.14)	95.14 (0.30)	92.23 (0.31)	82.39 (0.18)	95.27 (0.23)	95.08 (0.25)	94.79 (0.17)	93.83 (0.21)	89.96 (0.41)	80.69 (0.38)	93.09 (0.30)	92.74 (0.36)	92.73 (0.37)	91.41 (0.36)	87.80 (0.35)	79.25 (0.20)
	MICE	96.76 (0.08)	93.80 (0.40)	94.71 (0.30)	95.24 (0.09)	93.39 (0.25)	87.51 (0.48)	96.17 (0.23)	90.12 (0.56)	91.25 (0.43)	93.45 (0.28)	92.58 (0.32)	87.44 (0.35)	93.23 (0.69)	84.20 (0.92)	82.33 (1.92)	87.40 (1.04)	89.84 (0.51)	86.76 (0.80)
TabNet	Constant	96.14 (0.37)	95.66 (0.45)	95.76 (0.44)	94.67 (0.66)	92.06 (0.64)	84.37 (0.96)	96.07 (0.24)	95.99 (0.21)	95.53 (0.30)	94.74 (0.28)	92.24 (0.38)	85.76 (0.37)	92.01 (0.75)	91.49 (0.72)	90.97 (0.86)	89.97 (0.87)	86.99 (0.52)	80.79 (1.09)
	KNN	96.66 (0.21)	96.22 (0.24)	96.03 (0.17)	94.73 (0.22)	91.14 (0.34)	80.71 (0.18)	94.83 (0.48)	94.46 (0.49)	93.85 (0.58)	92.48 (0.50)	87.66 (0.52)	77.08 (0.77)	92.14 (0.52)	91.32 (0.54)	91.08 (0.68)	88.95 (0.82)	81.97 (0.96)	72.57 (1.29)
	MICE	96.74 (0.18)	93.18 (0.55)	94.00 (0.43)	94.84 (0.28)	93.00 (0.20)	86.79 (0.36)	95.41 (0.72)	89.28 (1.12)	90.59 (1.07)	92.76 (0.53)	91.42 (0.76)	86.33 (0.60)	91.80 (1.67)	83.49 (1.74)	81.43 (1.55)	85.20 (2.01)	87.65 (1.30)	83.91 (1.08)
TabTransformer	Constant	97.12 (0.20)	96.83 (0.25)	96.24 (0.21)	95.10 (0.10)	91.70 (0.32)	83.89 (0.30)	96.23 (0.30)	96.10 (0.34)	95.77 (0.41)	94.36 (0.34)	91.00 (0.45)	84.26 (0.24)	95.73 (0.24)	95.36 (0.39)	94.94 (0.35)	93.88 (0.29)	91.49 (0.58)	85.25 (0.59)
	KNN	97.14 (0.32)	96.83 (0.32)	96.23 (0.30)	94.58 (0.31)	90.32 (0.45)	79.77 (0.30)	94.90 (0.20)	94.70 (0.28)	93.69 (0.38)	92.34 (0.53)	87.46 (0.63)	74.06 (0.38)	90.90 (0.64)	90.30 (0.59)	90.44 (0.68)	87.94 (0.42)	82.42 (0.60)	72.63 (0.66)
	MICE	96.37 (0.17)	91.77 (0.68)	93.02 (0.53)	94.78 (0.07)	92.91 (0.35)	86.95 (0.28)	93.79 (0.75)	83.48 (1.33)	86.42 (0.93)	91.38 (0.31)	91.64 (0.55)	85.16 (0.46)	93.72 (0.60)	80.57 (1.07)	78.47 (1.80)	84.65 (1.22)	88.71 (0.70)	85.49 (0.93)
XGBoost	Constant	98.44 (0.13)	98.21 (0.15)	98.07 (0.13)	97.23 (0.08)	95.45 (0.17)	88.41 (0.41)	97.93 (0.11)	97.81 (0.13)	97.54 (0.16)	96.96 (0.10)	95.20 (0.28)	**89.69 (0.37)**	95.29 (0.34)	95.23 (0.35)	95.02 (0.31)	95.23 (0.22)	94.23 (0.41)	89.91 (0.62)
	KNN	98.25 (0.18)	98.03 (0.21)	97.85 (0.17)	96.59 (0.20)	93.26 (0.29)	81.75 (0.61)	97.42 (0.16)	97.23 (0.16)	96.88 (0.18)	95.78 (0.18)	91.98 (0.30)	79.78 (0.69)	95.61 (0.19)	95.33 (0.20)	94.91 (0.19)	93.42 (0.34)	88.82 (0.64)	80.32 (0.64)
	MICE	97.98 (0.12)	97.48 (0.15)	97.12 (0.21)	96.27 (0.15)	93.76 (0.08)	86.67 (0.40)	97.26 (0.11)	96.73 (0.13)	96.05 (0.19)	95.52 (0.16)	93.85 (0.29)	87.14 (0.43)	95.42 (0.33)	94.27 (0.50)	93.44 (0.38)	93.10 (0.37)	91.44 (0.57)	87.95 (0.44)
	Intrinsic	98.54 (0.14)	98.39 (0.09)	98.16 (0.13)	97.45 (0.15)	95.50 (0.10)	87.95 (0.67)	98.12 (0.18)	98.06 (0.17)	97.84 (0.13)	97.26 (0.19)	95.51 (0.30)	89.62 (0.36)	96.24 (0.13)	96.18 (0.18)	95.83 (0.14)	95.82 (0.23)	94.61 (0.35)	**90.60 (0.57)**

Table C.6: Average AUC performance and standard error (in brackets) of the experiments on the Spambase dataset. To facilitate the analysis we highlighted the best performance in each column in bold.

Appendix D

NAIM Statistical analysis

Table D.1 presents the percentages of experiments in which NAIM is significantly better than the competitors. Additionally, the averages across rows and columns are provided at the right and bottom margins, respectively, to identify an average trend. The analysis of the row and column averages reveals that the win rate of NAIM significantly exceeds the loss rate, with a considerable minimum advantage of 7.8% over the competitors. This observation, reinforced by the data in the bottom right-hand corner, which indicates that NAIM, on average, wins against its competitors in 58.7% of the cases, while it only loses in 1.6% of the cases, underlines NAIM's ability to provide competitive performance on tabular data without the need for any imputation strategy.

		Datasets											
		ADULT		BankMarketing		OnlineShoppers		SeismicBumps		Spambase		Mean	
Model	**Strategy**	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss
Adaboost	Constant	19.4	11.1	13.9	0.0	69.4	0.0	0.0	0.0	38.9	0.0	28.3	2.2
	KNN	36.1	2.8	19.4	0.0	100.0	0.0	0.0	0.0	75.0	0.0	46.1	0.6
	MICE	55.6	0.0	8.3	2.8	97.2	0.0	13.9	0.0	86.1	0.0	52.2	0.6
Decision Tree	Constant	100.0	0.0	100.0	0.0	100.0	0.0	97.2	0.0	100.0	0.0	99.4	0.0
	KNN	100.0	0.0	100.0	0.0	100.0	0.0	100.0	0.0	100.0	0.0	100.0	0.0
	MICE	100.0	0.0	100.0	0.0	100.0	0.0	97.2	0.0	100.0	0.0	99.4	0.0
	Intrinsic	100.0	0.0	100.0	0.0	97.2	0.0	100.0	0.0	100.0	0.0	99.4	0.0
FTTransformer	Constant	100.0	0.0	19.4	0.0	52.8	0.0	0.0	0.0	100.0	0.0	54.4	0.0
	KNN	100.0	0.0	38.9	0.0	94.4	0.0	0.0	0.0	100.0	0.0	66.7	0.0
	MICE	100.0	0.0	19.4	0.0	80.6	0.0	0.0	0.0	100.0	0.0	60.0	0.0
GRAPE	Intrinsic	100.0	0.0	88.9	0.0	94.4	0.0	0.0	0.0	100.0	0.0	76.7	0.0
HistGradientBoost	Constant	100.0	0.0	2.8	19.4	88.9	0.0	5.6	0.0	19.4	0.0	43.3	3.9
	KNN	100.0	0.0	8.3	13.9	100.0	0.0	0.0	0.0	36.1	2.8	48.9	3.3
	MICE	100.0	0.0	8.3	8.3	100.0	0.0	0.0	0.0	38.9	0.0	49.4	1.7
	Intrinsic	8.3	80.6	0.0	13.9	19.4	0.0	2.8	0.0	5.6	2.8	7.2	19.4
MLP	Constant	100.0	0.0	100.0	0.0	100.0	0.0	36.1	0.0	88.9	0.0	85.0	0.0
	KNN	100.0	0.0	100.0	0.0	100.0	0.0	27.8	0.0	83.3	0.0	82.2	0.0
	MICE	100.0	0.0	100.0	0.0	100.0	0.0	36.1	0.0	97.2	0.0	86.7	0.0
Random Forest	Constant	100.0	0.0	16.7	0.0	50.0	0.0	2.8	0.0	44.4	0.0	42.8	0.0
	KNN	100.0	0.0	16.7	0.0	88.9	0.0	0.0	0.0	44.4	2.8	50.0	0.6
	MICE	100.0	0.0	11.1	0.0	75.0	0.0	0.0	0.0	52.8	0.0	47.8	0.0
	Intrinsic	66.7	0.0	19.4	0.0	13.9	0.0	2.8	0.0	2.8	2.8	21.1	0.6
SVM	Constant	100.0	0.0	72.2	0.0	88.9	0.0	0.0	0.0	77.8	0.0	67.8	0.0
	KNN	100.0	0.0	52.8	0.0	100.0	0.0	0.0	0.0	86.1	0.0	67.8	0.0
	MICE	100.0	0.0	63.9	0.0	88.9	0.0	0.0	0.0	72.2	0.0	65.0	0.0
TabNet	Constant	100.0	0.0	13.9	5.6	72.2	0.0	0.0	0.0	100.0	0.0	57.2	1.1
	KNN	100.0	0.0	25.0	5.6	100.0	0.0	0.0	0.0	97.2	0.0	64.4	1.1
	MICE	100.0	0.0	38.9	0.0	77.8	0.0	0.0	0.0	100.0	0.0	63.3	0.0
TabTransformer	Constant	100.0	0.0	13.9	0.0	30.6	0.0	0.0	0.0	94.4	0.0	47.8	0.0
	KNN	100.0	0.0	13.9	0.0	80.6	0.0	0.0	0.0	83.3	0.0	55.6	0.0
	MICE	100.0	0.0	38.9	0.0	80.6	0.0	0.0	0.0	88.9	0.0	61.7	0.0
XGBoost	Constant	100.0	0.0	0.0	11.1	22.2	0.0	0.0	0.0	41.7	0.0	32.8	2.2
	KNN	100.0	0.0	16.7	5.6	88.9	0.0	0.0	0.0	38.9	0.0	48.9	1.1
	MICE	100.0	0.0	22.2	8.3	83.3	0.0	0.0	0.0	63.9	0.0	53.9	1.7
	Intrinsic	16.7	63.9	13.9	8.3	25.0	0.0	22.2	0.0	33.3	0.0	22.2	14.4
Mean		88.7	4.5	39.4	2.9	78.9	0.0	15.6	0.0	71.2	0.3	58.7	1.6

Table D.1: Percentages of wins and losses of NAIM correct predictions compared with the competitors using the Wilcoxon signed-rank test.

Appendix E

NAIM Computational analysis

To provide a quantitative comparison of the computational complexity of the evaluated DL approaches, we computed the number of floating-point operations (FLOPs) and the total number of trainable parameters for each model. It is important to note that ML models were excluded from this analysis, as their training and inference procedures are inherently less computationally demanding and not directly comparable with DL architectures. Similarly, the computational contribution of the imputation methods was not considered, since imputers mainly affect the preprocessing and training stages, whereas their cost during inference is negligible and varies significantly across implementations. The results are summarized in Table E.1, which reports the average number of operations and parameters across the five datasets. The reported FLOPs correspond to the total number of operations required for a single forward pass, while the number of parameters reflects the count of all trainable weights within the model.

	Adult		**BankMarketing**		**OnlineShoppers**		**SeismicBumps**		**Spambase**	
Model	**FLOPs**	**# Params**	**FLOPs**	**# Params**	**FLOPs**	**# Params**	**FLOPs**	**# Params**	**FLOPs**	**# Params**
NAIM	1.03×10^6	7.99×10^4	1.48×10^6	7.98×10^4	1.26×10^6	7.98×10^4	1.33×10^6	7.95×10^4	4.29×10^6	8.04×10^4
FTTransformer	2.96×10^6	1.78×10^5	4.33×10^6	1.77×10^5	3.63×10^6	1.77×10^5	3.86×10^6	1.76×10^5	1.53×10^7	1.78×10^5
GRAPE	3.78×10^4	3.73×10^3	5.73×10^4	3.93×10^3	4.73×10^4	3.83×10^3	5.06×10^4	3.86×10^3	2.28×10^5	5.15×10^3
MLP	2.12×10^4	1.06×10^4	1.28×10^4	6.40×10^3	1.52×10^4	7.60×10^3	4.80×10^3	2.40×10^3	1.18×10^4	5.90×10^3
TabNet	1.00×10^4	6.67×10^3	1.16×10^4	7.02×10^3	1.08×10^4	6.84×10^3	1.10×10^4	6.85×10^3	2.61×10^4	9.41×10^3
TabTransformer	1.53×10^6	2.20×10^5	1.96×10^6	2.44×10^5	1.33×10^6	2.11×10^5	7.32×10^5	1.87×10^5	2.30×10^3	1.77×10^5

Table E.1: Computational complexity analysis of the evaluated deep learning models in terms of FLOPs and number of trainable parameters across datasets.

As expected, transformer-based models (NAIM, FTTransformer, and TabTransformer) exhibit higher computational demands compared to simpler architectures such as MLP and TabNet, reflecting the additional cost introduced by the self-attention mechanism and feature embedding layers. Nevertheless, NAIM maintains a balanced trade-off between expressivity and efficiency, achieving competitive FLOPs and parameter counts despite its specialized

mechanisms for handling missing data. Overall, this analysis confirms that NAIMs ability to robustly manage incomplete tabular data does not come at a prohibitive computational cost. In fact, it remains within the same order of magnitude as comparable transformer-based approaches while significantly outperforming them in predictive performance, as discussed in the main text.

Appendix F

MARIA Complete Results

AUC		Train missing percentage: Ω						Train missing percentage: 5%						Train missing percentage: 10%						Train missing percentage: 30%						Train missing percentage: 50%						Train missing percentage: 75%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:						Test missing percentage:						Test missing percentage:						Test missing percentage:					
Approach	**Model**	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%
Early ML with imputer	AdaBoost	83.89	83.53	83.10	80.24	76.17	69.71	84.27	83.89	83.43	80.27	76.46	70.63	84.12	83.44	83.27	79.91	76.56	69.73	81.96	81.67	80.98	78.33	75.11	69.47	81.13	80.66	80.38	78.17	75.52	70.25	80.44	79.98	79.67	77.65	74.74	71.19
	DecisionTree	76.40	75.59	74.91	70.88	66.96	60.07	74.04	73.48	73.03	70.33	67.36	61.25	74.41	73.90	73.40	70.43	66.69	60.79	71.53	71.20	70.70	68.35	66.63	61.50	70.01	69.74	69.23	67.54	66.37	61.87	73.05	72.25	71.68	68.97	65.89	62.79
	HistGradientBoost	88.05	87.39	86.80	83.60	79.25	71.36	87.85	87.47	86.87	83.31	78.47	71.18	88.00	87.33	86.64	82.95	78.76	70.89	86.78	86.49	85.83	82.90	79.30	70.62	84.91	84.46	83.98	81.67	78.24	70.82	79.48	79.24	78.78	76.45	74.34	70.42
	RandomForest	88.75	88.17	87.66	84.08	79.67	71.68	88.81	88.28	87.71	83.39	78.69	71.48	88.49	88.15	87.79	83.82	79.73	71.34	87.78	87.62	86.87	83.60	79.44	71.04	87.05	86.62	85.91	82.41	78.59	71.63	87.13	86.37	85.60	82.38	78.28	71.74
	SVM	86.43	86.08	85.70	82.76	78.28	71.80	86.56	86.17	86.01	82.61	78.37	71.85	86.71	86.39	86.09	82.99	78.87	71.97	86.74	86.57	86.04	83.20	79.21	72.07	86.15	85.70	85.21	83.01	78.90	72.87	84.82	84.42	83.90	81.73	78.42	74.01
	XGBoost	88.00	87.32	86.71	82.68	77.94	70.04	88.07	87.56	87.07	83.36	78.33	70.57	87.80	87.19	86.77	83.08	78.53	70.90	86.44	86.16	85.60	82.86	79.19	71.17	85.16	84.86	84.35	81.91	79.33	72.36	85.87	85.47	84.99	82.65	79.65	74.40
Early ML without imputer	DecisionTree	78.33	77.22	76.51	71.92	67.45	60.68	78.12	77.23	76.18	71.71	67.15	60.62	78.08	77.47	77.24	73.24	68.06	60.84	76.35	76.23	75.86	72.71	69.79	64.83	76.19	75.75	75.29	73.11	71.55	66.33	76.32	75.78	75.57	74.25	71.72	68.38
	HistGradientBoost	89.99	89.22	88.44	84.35	79.59	71.70	89.89	89.33	88.70	85.17	80.14	72.19	90.00	89.44	88.90	85.31	80.25	71.13	89.00	88.38	87.64	84.38	80.50	72.31	87.26	86.74	86.19	82.90	79.15	71.80	81.88	81.37	80.94	78.63	76.12	71.71
	RandomForest	90.78	90.26	89.69	86.32	81.63	73.58	**90.65**	90.14	89.49	86.11	81.42	72.98	90.68	90.23	89.75	86.48	81.77	72.85	**90.39**	89.93	89.45	86.45	82.46	74.17	**90.03**	**89.75**	**89.30**	86.63	83.11	75.48	88.25	87.84	87.40	85.05	81.82	76.36
	XGBoost	90.07	89.41	88.46	84.03	78.75	70.43	90.10	89.64	89.07	86.22	82.23	74.02	89.71	89.39	89.02	86.27	82.63	74.28	88.97	88.68	88.39	86.10	83.46	76.07	89.05	88.81	88.42	86.02	83.26	76.98	88.15	87.66	87.36	85.23	82.28	**76.95**
Early DL with imputer	FTTransformer	87.47	87.15	86.65	83.60	79.15	72.01	86.50	86.27	85.86	83.11	78.54	71.60	86.54	86.26	85.92	83.28	78.96	71.88	86.42	86.30	85.43	82.89	79.19	72.39	85.49	85.16	84.44	82.07	78.67	72.95	71.80	71.56	71.19	69.94	68.12	65.41
	MLP	84.98	84.62	84.14	80.82	76.63	69.86	84.96	84.44	84.04	80.81	76.89	69.40	84.22	83.93	83.48	80.58	76.85	69.50	83.88	83.63	83.15	80.45	76.77	69.87	82.89	82.27	81.74	79.53	76.53	70.47	86.27	85.82	85.33	83.33	80.29	74.55
	TabNet	85.62	85.11	84.90	81.50	76.70	69.97	84.76	84.21	83.81	81.07	76.37	69.40	84.41	84.10	83.52	80.32	76.38	69.00	83.83	84.01	83.03	80.49	76.15	69.65	81.42	81.28	81.01	78.85	75.33	70.27	73.50	73.77	73.22	73.03	72.80	70.24
	TABTransformer	84.65	84.32	83.92	80.98	77.43	70.57	84.06	83.72	83.58	80.98	76.96	70.34	83.91	83.56	83.19	80.50	76.26	70.09	83.16	82.94	82.39	80.23	76.91	70.77	82.60	82.18	81.45	79.63	76.38	70.82	86.35	85.94	85.39	83.34	80.35	74.80
Intermediate DL with imputer	FTTransformer	80.43	79.98	79.25	75.57	71.45	63.68	69.22	68.99	68.54	66.57	64.75	60.19	67.73	67.60	67.16	65.41	63.66	59.58	57.28	57.12	56.86	55.90	55.17	53.25	52.86	52.83	52.68	52.12	51.95	50.85	50.00	50.00	50.00	50.00	50.00	50.00
	MLP	86.62	86.16	85.60	82.97	79.35	71.43	87.19	86.80	86.28	83.83	80.43	72.24	86.65	86.15	85.59	83.19	79.95	72.48	78.66	78.52	78.36	77.12	74.97	69.38	76.84	76.64	76.29	75.25	73.66	69.19	67.90	67.75	67.54	66.79	65.43	62.12
	TabNet	75.54	75.07	74.20	71.40	67.48	61.05	79.15	78.69	78.30	75.51	71.77	65.70	76.87	76.54	76.15	73.40	70.03	63.66	64.28	63.48	63.32	62.18	59.52	56.47	64.81	64.04	64.01	63.00	60.77	56.24	54.49	54.74	54.40	53.92	53.61	53.97
	TABTransformer	85.20	84.41	83.69	80.66	76.25	68.57	84.17	83.55	82.98	80.24	76.08	68.24	84.95	84.37	83.91	80.87	76.70	68.84	83.25	82.86	82.33	79.97	76.71	69.11	82.73	82.33	81.89	79.62	76.23	70.78	81.35	81.02	80.52	78.40	75.24	69.84
Intermediate DL	MARIA	**90.94**	**90.55**	**90.20**	**87.78**	**83.97**	75.24	90.49	**90.26**	**89.86**	**87.59**	**84.17**	76.80	**90.76**	**90.45**	**90.03**	**87.94**	**84.38**	76.36	90.28	**89.94**	**89.56**	**87.39**	**84.00**	76.49	89.92	89.63	89.25	**87.07**	**84.00**	76.86	**88.57**	**88.22**	**87.88**	**85.72**	**83.05**	76.61
Late ML with imputer	AdaBoost	83.94	82.90	82.19	78.55	75.09	69.87	83.77	83.15	82.85	80.83	78.48	73.30	83.01	82.46	82.05	80.24	77.74	73.06	82.47	82.00	81.72	79.54	77.22	72.41	82.23	81.81	81.50	79.46	76.74	71.12	83.08	82.48	81.90	79.63	76.62	71.13
	DecisionTree	77.36	76.58	75.74	72.42	69.04	62.32	77.64	77.21	76.53	74.12	71.32	63.76	77.30	76.98	76.61	74.37	71.36	63.97	76.51	76.21	76.11	74.01	71.07	64.48	76.38	76.32	76.02	72.94	70.08	62.83	75.52	74.92	74.27	71.53	69.12	63.08
	HistGradientBoost	83.29	82.42	81.63	77.80	74.38	68.17	82.70	82.11	81.33	78.64	75.33	68.73	82.82	82.21	81.52	78.96	75.88	69.45	82.78	82.18	81.61	79.07	75.81	70.19	81.57	81.15	80.75	78.24	75.30	68.93	80.72	80.17	79.61	77.19	74.04	68.01
	RandomForest	84.92	84.26	83.34	79.77	75.92	68.86	84.63	84.29	83.81	81.66	78.85	72.49	84.48	84.20	83.76	81.88	79.08	72.86	84.15	83.88	83.49	81.52	78.84	73.27	83.49	83.33	82.94	80.93	78.18	72.38	83.18	82.68	82.25	80.05	77.30	71.50
	SVM	88.13	87.67	87.14	84.47	80.92	73.94	88.21	87.71	87.22	84.73	81.00	73.78	88.12	87.60	87.13	84.52	80.68	72.95	88.08	87.55	87.04	84.36	80.29	72.88	87.42	86.84	86.27	83.31	78.96	71.46	85.28	84.15	83.23	79.04	73.58	66.62
	XGBoost	87.65	87.03	86.43	83.26	79.46	72.08	87.36	86.91	86.41	84.08	81.17	74.80	87.19	86.75	86.42	84.05	80.90	74.87	86.60	86.29	85.88	83.62	80.61	74.52	86.20	85.93	85.52	83.10	80.17	74.00	85.34	84.97	84.33	81.89	78.91	72.71
Late ML without imputer	DecisionTree	83.18	82.38	81.92	78.24	75.01	68.07	83.25	82.77	82.26	79.26	76.18	69.68	83.07	82.66	82.17	79.60	76.86	71.26	82.12	81.76	81.53	79.19	75.69	69.46	81.90	81.60	81.47	78.84	75.99	69.75	80.23	79.78	79.30	76.76	73.75	67.96
	HistGradientBoost	84.66	83.90	83.14	79.65	75.97	69.85	84.51	84.04	83.44	80.67	77.54	70.45	84.43	83.95	83.37	80.66	77.72	71.19	84.42	83.95	83.58	80.71	77.50	71.75	83.75	83.26	82.91	80.25	77.32	70.55	81.79	81.08	80.52	77.85	74.60	69.03
	RandomForest	88.70	88.21	87.70	84.84	81.37	75.23	89.00	88.66	88.29	86.22	83.39	**77.60**	88.89	88.56	88.18	86.10	83.43	**77.89**	88.70	88.38	88.11	85.99	83.22	**77.89**	88.36	88.04	87.81	85.85	83.20	**77.16**	87.93	87.53	87.14	84.96	82.05	76.53
	XGBoost	89.05	88.46	88.05	85.28	81.87	**75.66**	88.99	88.61	88.21	85.97	82.97	77.14	88.89	88.57	88.22	86.02	83.00	77.35	88.46	88.17	87.83	85.54	82.39	76.91	88.23	87.92	87.64	85.27	82.27	76.12	86.25	85.91	85.41	82.99	79.82	74.26
Late DL with imputer	FTTransformer	87.15	86.67	86.07	83.56	80.34	73.76	88.13	87.69	87.25	85.02	81.73	75.51	87.92	87.44	87.03	84.87	81.86	75.26	86.76	86.34	85.85	83.66	80.78	74.13	85.08	84.73	84.27	82.00	79.00	72.82	82.61	82.18	81.60	79.10	76.12	67.04
	MLP	87.13	86.66	86.13	83.55	80.03	72.68	87.52	87.13	86.71	84.26	80.89	74.23	87.26	86.85	86.49	84.18	81.17	74.39	87.16	86.82	86.47	84.18	81.35	74.96	86.49	86.09	85.64	83.31	79.99	73.95	83.93	83.49	82.99	80.59	77.01	71.81
	TabNet	87.14	86.71	86.10	83.56	80.10	73.73	87.19	86.73	86.37	84.01	80.60	74.09	86.90	86.52	86.14	83.87	80.87	74.48	85.72	85.37	84.90	82.70	79.79	73.70	85.03	84.73	84.32	82.25	79.10	72.43	80.57	80.20	79.82	77.21	74.48	68.31
	TABTransformer	86.42	85.77	85.18	82.57	78.39	71.01	85.28	84.74	84.26	81.68	77.87	70.41	85.64	85.13	84.62	81.99	78.59	70.96	84.96	84.34	83.81	81.33	78.24	71.47	84.04	83.61	83.19	80.61	77.21	70.76	82.76	82.36	81.91	79.37	75.60	63.81

MCC		Train missing percentage: Ω						Train missing percentage: 5%						Train missing percentage: 10%						Train missing percentage: 30%						Train missing percentage: 50%						Train missing percentage: 75%					
		Test missing percentage:						Test missing percentage:						Test missing percentage:						Test missing percentage:						Test missing percentage:						Test missing percentage:					
Approach	**Model**	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%	Ω	5%	10%	30%	50%	75%
Early ML with imputer	AdaBoost	54.79	52.72	50.17	40.24	33.64	20.94	52.81	51.37	49.36	38.99	31.18	22.48	53.05	50.56	48.38	39.52	32.32	20.22	48.93	46.92	44.84	36.28	28.66	20.09	43.57	42.27	40.46	33.99	27.48	18.45	28.35	28.29	29.33	25.76	22.07	15.53
	DecisionTree	47.80	46.58	44.73	38.54	31.37	19.83	46.20	44.75	42.88	34.76	29.04	18.01	44.65	43.66	41.97	33.92	27.69	19.63	40.69	39.81	39.34	33.36	26.16	17.39	36.17	35.78	33.81	29.31	23.39	14.03	44.41	42.97	40.94	34.06	28.85	18.00
	HistGradientBoost	58.61	56.60	**54.52**	44.52	34.87	22.48	58.19	56.04	54.43	44.89	36.42	22.96	57.67	55.52	53.37	42.55	34.75	23.08	55.49	53.72	51.84	42.97	32.90	22.60	54.55	52.47	51.01	42.03	34.13	22.60	27.61	27.33	27.29	25.40	22.49	22.41
	RandomForest	54.79	52.91	51.61	41.66	33.86	19.23	55.18	53.02	51.44	40.69	33.62	20.06	53.87	51.82	49.53	40.27	33.48	21.44	50.85	48.98	46.88	38.01	28.90	20.07	50.85	48.30	45.97	38.33	30.33	19.72	53.78	51.56	49.84	42.43	34.47	20.96
	SVM	57.10	54.63	52.12	41.24	32.08	18.35	57.37	54.96	52.28	41.02	31.46	18.71	56.82	54.02	51.59	41.36	32.40	19.80	55.32	53.24	50.62	39.64	30.37	19.02	52.66	50.71	48.75	38.46	29.73	20.19	48.26	47.03	44.79	36.95	29.75	16.61
	XGBoost	57.46	55.76	53.61	43.94	33.32	20.51	57.35	55.65	53.78	43.54	35.00	21.61	57.11	55.04	52.97	41.62	34.66	21.35	55.94	53.75	52.03	42.75	32.09	22.77	52.75	50.56	49.15	41.18	33.33	21.90	52.79	51.41	49.40	41.52	32.67	21.92
Early ML without imputer	DecisionTree	50.50	48.47	45.60	36.30	30.37	19.03	47.92	46.89	44.33	36.91	29.49	17.91	49.06	48.16	46.02	37.68	29.18	19.46	47.25	45.69	43.15	37.75	29.75	21.54	43.04	42.85	41.62	36.97	30.84	20.99	48.37	47.12	45.26	38.39	31.58	20.63
	HistGradientBoost	57.42	55.17	51.54	42.20	33.53	**22.91**	58.58	57.29	54.86	46.57	35.53	22.72	58.26	56.28	54.31	47.40	37.13	23.32	55.16	53.87	52.76	45.62	36.21	24.98	55.90	54.47	51.92	43.48	34.62	23.17	26.38	25.68	25.19	24.03	21.81	24.69
	RandomForest	56.06	53.66	50.92	41.57	32.56	21.50	55.68	54.34	51.71	43.96	35.53	21.96	55.88	53.80	51.42	44.15	37.11	23.81	53.71	52.77	51.04	45.27	34.95	24.30	52.98	51.57	50.21	44.08	35.23	23.12	55.63	54.60	53.38	45.07	38.30	**25.91**
	XGBoost	58.14	55.48	52.37	42.73	33.17	21.74	58.07	56.55	53.45	44.36	34.08	22.29	57.90	56.14	53.53	44.57	36.10	20.53	56.53	54.51	52.67	45.97	35.88	23.75	57.44	**56.15**	53.04	45.25	35.98	23.65	52.90	51.54	49.49	41.91	33.93	24.68
Early DL with imputer	FTTransformer	52.35	50.42	48.16	38.19	30.47	18.55	51.48	49.63	48.08	38.17	31.07	18.41	52.10	50.03	48.29	38.16	31.14	19.27	50.93	48.56	46.44	37.76	29.58	19.99	47.92	45.86	43.84	34.36	26.92	18.87	8.45	8.03	7.63	6.28	4.55	2.49
	MLP	51.19	50.05	48.00	38.34	30.08	17.24	50.92	49.07	47.31	38.09	29.17	17.29	48.83	47.38	45.56	36.15	30.07	18.58	44.89	43.24	41.20	32.92	24.76	17.47	42.87	41.90	39.75	32.47	26.55	18.63	48.48	47.08	44.76	37.61	29.30	20.56
	TabNet	45.17	44.02	42.48	35.59	27.02	15.10	45.09	43.80	41.23	33.52	26.21	14.82	45.96	44.79	43.77	34.69	26.64	17.38	41.29	39.85	38.59	31.54	25.03	16.09	37.14	36.78	35.41	30.88	23.44	13.36	24.38	23.16	22.22	20.05	14.00	11.16
	TABTransformer	54.07	51.80	49.87	40.22	31.33	18.50	53.09	51.60	49.93	40.00	33.10	20.86	52.12	50.29	48.27	37.22	28.19	20.53	49.64	47.69	46.17	36.11	27.57	18.15	46.16	43.63	41.69	35.98	27.23	16.94	49.23	47.71	45.91	38.42	31.07	21.18
Intermediate DL with imputer	FTTransformer	0.65	0.59	0.52	0.39	0.29	-0.15	0.26	0.31	0.27	0.37	0.20	0.15	4.67	4.48	4.30	3.72	2.98	1.42	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	MLP	49.77	47.85	45.59	36.87	29.72	19.77	50.15	48.32	46.01	37.72	30.80	19.33	49.03	46.85	44.70	36.56	28.91	19.02	48.41	47.19	45.14	37.79	30.11	20.60	48.80	46.83	44.66	36.41	29.34	20.19	17.37	16.91	16.28	12.75	9.74	6.39
	TabNet	1.95	1.96	1.55	2.03	1.68	1.82	2.04	2.06	3.04	1.86	1.08	0.53	2.08	1.92	2.11	1.31	1.57	-0.42	2.03	1.24	0.73	1.67	0.77	0.49	1.48	2.57	2.65	1.91	0.31	-0.58	1.32	1.32	1.67	1.29	0.11	-0.52
	TABTransformer	50.26	47.79	45.50	36.52	27.94	16.28	47.98	46.59	44.57	35.27	26.92	17.47	49.71	48.04	45.27	36.84	28.00	18.46	47.64	45.57	43.57	37.22	28.89	19.30	46.87	45.03	43.53	35.70	28.67	17.87	41.20	40.39	39.30	32.19	27.08	18.60
Intermediate DL	MARIA	**59.07**	**56.87**	54.36	**46.24**	**36.77**	22.36	**59.59**	**57.80**	**56.00**	46.62	37.02	22.95	**59.75**	**58.00**	**55.49**	46.56	37.68	25.88	**57.64**	**55.91**	**54.40**	**46.91**	37.39	25.20	**57.57**	55.78	**53.42**	**46.66**	37.59	24.65	**58.06**	**56.55**	**54.28**	**46.88**	**38.66**	25.10
Late ML with imputer	AdaBoost	46.49	44.68	42.75	34.73	27.14	18.65	43.86	42.56	41.65	34.83	28.54	20.28	44.82	43.60	42.01	35.41	29.46	19.50	43.16	41.75	40.45	34.40	28.48	18.23	42.87	42.05	40.97	33.95	28.11	18.78	41.69	40.41	38.67	31.77	26.49	17.40
	DecisionTree	30.13	30.30	28.81	25.00	18.51	12.43	27.96	28.31	27.87	24.68	22.64	20.06	29.76	29.22	27.65	24.42	23.17	21.65	28.12	27.74	26.87	23.08	19.08	17.15	29.35	29.37	28.32	24.91	23.62	21.69	27.68	27.02	26.15	23.30	20.97	17.80
	HistGradientBoost	35.13	33.65	31.33	24.44	18.23	7.13	34.42	33.62	32.14	26.64	22.75	18.06	33.66	32.30	31.44	26.47	23.84	20.72	34.12	32.89	31.69	27.82	23.00	18.72	33.22	32.31	30.01	26.46	21.10	19.42	33.41	32.47	30.58	26.17	21.96	18.97
	RandomForest	43.16	41.24	38.55	30.62	22.72	10.44	43.12	42.67	41.97	36.97	32.15	25.73	42.16	41.56	40.63	38.22	32.74	26.42	41.97	41.63	40.91	36.37	30.38	24.98	42.12	41.48	40.15	36.54	29.79	22.22	39.48	38.84	37.46	32.71	27.88	21.42
	SVM	46.09	44.05	42.06	33.39	25.67	12.25	45.51	43.56	41.69	33.58	26.30	14.02	45.17	43.00	41.00	33.55	26.89	15.66	40.12	39.01	37.29	30.67	23.94	15.29	32.38	31.43	30.79	26.87	22.91	15.65	15.92	15.16	14.61	12.23	10.34	6.54
	XGBoost	44.84	43.09	41.45	32.96	25.63	13.87	46.16	44.67	42.82	35.00	27.17	15.52	46.07	44.50	43.18	35.50	27.99	18.07	46.38	45.19	43.20	35.82	28.75	18.78	46.24	45.09	42.83	34.99	27.18	18.43	44.52	42.70	40.80	34.12	27.52	18.15
Late ML without imputer	DecisionTree	38.39	37.38	35.48	29.32	22.65	11.68	40.89	40.48	40.32	36.93	32.36	27.28	40.57	39.73	38.60	33.98	30.04	24.47	39.11	38.12	38.12	35.12	31.34	25.27	37.20	36.69	35.59	32.32	28.67	24.90	36.02	35.56	35.40	31.61	28.04	22.58
	HistGradientBoost	38.64	36.57	35.02	27.50	20.40	6.46	37.95	36.98	36.00	30.40	26.26	19.50	38.06	37.22	36.56	31.23	27.33	21.58	36.90	35.70	34.40	29.11	25.14	21.49	37.85	37.24	35.09	29.11	24.06	21.29	34.14	33.20	31.29	27.04	22.60	19.77
	RandomForest	51.15	49.50	47.27	40.30	30.21	15.15	54.03	53.07	51.68	**46.93**	**41.32**	**31.64**	52.99	52.42	51.18	**47.62**	**42.38**	**32.16**	52.00	51.24	50.26	46.82	**41.70**	**31.15**	51.21	50.59	49.49	45.74	**39.99**	**30.35**	46.93	45.92	44.85	40.80	34.37	24.86
	XGBoost	46.82	44.75	42.62	36.67	29.29	14.74	46.42	44.85	43.20	36.81	29.43	19.28	46.84	45.66	44.37	37.53	30.87	20.87	46.08	44.54	42.96	36.45	29.96	19.22	46.77	45.47	43.32	37.49	28.62	20.15	44.90	43.13	40.90	34.22	27.74	18.86
Late DL with imputer	FTTransformer	47.62	45.40	43.05	34.46	27.28	14.05	47.50	45.63	43.60	34.70	26.93	15.08	47.70	45.75	43.73	34.90	27.78	16.39	46.71	45.37	43.33	35.85	28.51	17.52	47.12	45.78	43.67	36.13	28.95	18.60	22.96	22.64	21.89	17.71	13.92	8.78
	MLP	43.24	41.57	39.71	32.88	24.55	11.78	43.35	41.85	40.09	33.48	26.84	17.30	43.36	42.11	40.04	34.20	28.14	18.10	42.70	41.32	39.57	34.43	27.69	18.63	43.67	42.32	41.01	34.81	27.94	19.78	45.80	44.39	42.55	34.98	27.72	19.19
	TabNet	40.23	38.92	36.91	30.29	24.87	13.91	41.15	40.03	38.39	31.90	25.73	16.02	39.00	37.95	36.11	30.50	24.65	16.15	36.24	35.42	34.14	29.00	23.65	16.74	32.59	31.27	30.37	25.07	20.40	13.96	14.21	14.13	13.25	12.11	9.04	6.14
	TABTransformer	41.30	39.59	37.70	30.24	22.42	8.15	41.31	40.11	38.27	31.44	24.84	13.25	42.70	41.13	39.17	32.48	26.21	15.21	40.21	39.38	38.16	32.91	26.69	17.27	40.52	39.62	38.30	32.22	26.45	17.94	35.85	35.10	34.30	29.31	24.88	17.84

Table F.1: Average AUC and MCC performance of the experiments across the respective tasks in the "missing modalities" setting. To facilitate the analysis we highlighted the best performance in each column in bold.

AUC		Train missing percentage: Ω				Train missing percentage: 30%				Train missing percentage: 50%				Train missing percentage: 75%			
		Test missing percentage:				Test missing percentage:				Test missing percentage:				Test missing percentage:			
Approach	**Model**	Ω	30%	50%	75%	Ω	30%	50%	75%	Ω	30%	50%	75%	Ω	30%	50%	75%
Early ML with imputer	AdaBoost	83.89	77.53	82.18	78.04	76.65	76.10	73.36	68.80	84.44	78.21	81.80	78.00	81.66	72.82	80.07	76.00
	DecisionTree	76.40	65.40	74.30	67.49	65.10	65.63	60.00	58.39	74.89	62.44	73.13	67.49	68.93	58.40	68.47	65.15
	HistGradientBoost	88.05	78.35	86.40	80.06	79.34	78.84	75.44	69.79	87.90	77.65	85.73	79.81	85.73	74.23	83.66	78.90
	RandomForest	88.75	80.64	86.95	81.84	80.97	80.64	77.47	70.50	88.13	78.84	86.35	81.36	85.85	75.24	84.33	78.91
	SVM	86.43	75.79	84.62	79.75	76.49	75.93	72.47	65.13	86.66	74.99	84.51	79.54	85.57	74.34	83.10	78.37
	XGBoost	88.00	78.30	86.12	79.19	78.65	78.14	73.92	68.52	87.15	76.14	85.15	78.99	84.42	71.01	82.86	78.53
Early ML without imputer	DecisionTree	78.33	64.65	75.27	65.65	66.07	66.74	64.49	55.37	76.77	62.93	75.63	67.68	71.45	59.06	70.62	69.75
	HistGradientBoost	89.99	80.59	87.27	78.82	81.67	80.54	75.29	67.26	89.93	80.70	88.07	82.05	**89.10**	78.54	**87.56**	82.62
	RandomForest	90.78	**82.27**	**89.22**	**83.63**	**82.96**	**82.35**	**78.85**	**72.10**	90.17	**81.20**	**89.04**	**84.85**	88.09	**78.87**	87.12	83.94
	XGBoost	90.07	79.44	86.55	77.30	80.16	79.14	75.04	68.34	89.22	78.17	87.83	82.61	86.89	74.42	86.76	**84.16**
Early DL with imputer	FTTransformer	87.10	79.70	85.50	80.29	79.19	78.76	74.88	69.07	86.34	77.20	84.49	79.37	83.52	71.44	80.79	76.96
	MLP	85.08	75.44	83.14	77.86	72.99	73.90	71.12	65.51	83.94	72.07	82.60	77.55	82.80	72.24	80.88	76.93
	TabNet	85.11	75.86	84.11	78.72	76.03	75.67	74.13	68.04	84.93	75.74	82.95	77.78	80.52	68.61	79.22	73.15
	TABTransformer	84.87	73.84	83.03	77.86	72.35	72.23	69.82	64.22	84.24	73.18	82.53	78.17	82.43	71.14	80.54	75.68
Intermediate DL with imputer	FTTransformer	76.82	69.54	74.18	69.98	72.40	70.00	65.30	60.30	68.47	64.18	66.80	64.03	50.00	50.00	50.00	50.00
	MLP	86.61	78.94	84.96	81.16	80.63	79.60	75.42	69.67	82.58	67.02	81.44	78.70	75.61	50.00	75.61	73.68
	TabNet	75.58	69.67	73.57	70.89	69.88	68.47	64.16	59.89	73.29	63.37	72.14	70.10	67.17	62.60	67.72	63.89
	TABTransformer	85.32	75.91	83.66	79.54	73.95	72.88	69.73	63.43	83.90	73.93	82.22	78.21	81.46	67.70	79.43	76.11
Intermediate DL	MARIA	**90.80**	80.60	87.81	82.75	82.31	81.29	74.02	65.10	**90.24**	80.06	87.80	82.13	88.21	74.30	85.58	81.07
Late ML with imputer	AdaBoost	83.94	78.32	81.73	76.69	77.61	77.39	73.46	67.79	83.50	76.31	81.90	78.41	81.86	73.34	80.38	77.72
	DecisionTree	77.36	66.37	75.27	70.16	67.78	67.27	64.59	57.76	76.02	65.82	74.34	70.11	72.44	60.34	71.09	67.16
	HistGradientBoost	83.29	73.04	80.87	75.24	73.52	73.26	69.43	64.36	82.30	71.83	80.34	75.68	80.74	69.54	78.87	75.05
	RandomForest	84.92	71.79	82.80	78.52	73.02	71.86	68.06	63.66	84.12	70.63	82.52	78.76	81.98	65.25	80.92	78.38
	SVM	88.13	78.98	86.20	81.69	80.33	79.16	76.01	70.70	87.82	78.98	85.96	82.00	86.78	78.21	85.44	81.93
	XGBoost	87.65	77.55	85.37	79.86	78.79	78.32	74.07	67.20	87.32	77.01	85.18	80.59	85.06	72.99	83.55	79.69
Late ML without imputer	DecisionTree	83.18	70.14	79.75	69.13	71.52	70.37	65.09	59.79	80.85	68.42	78.47	70.73	77.18	65.68	75.43	73.72
	HistGradientBoost	84.66	73.00	80.95	73.02	74.27	73.65	66.82	62.13	84.25	73.65	81.71	74.83	82.48	71.05	79.93	75.12
	RandomForest	88.70	77.01	86.52	80.82	79.62	78.78	74.39	69.19	88.79	79.40	87.22	82.78	86.87	76.65	85.64	83.03
	XGBoost	89.05	78.21	86.13	76.76	80.63	80.26	75.00	68.33	88.34	77.70	87.20	81.87	86.68	75.02	85.97	83.27
Late DL with imputer	FTTransformer	87.78	79.09	85.87	81.40	80.98	80.06	76.09	71.17	86.64	76.37	84.79	80.89	85.20	73.03	83.96	80.11
	MLP	87.13	79.12	85.09	81.02	81.01	80.29	76.56	71.65	86.76	78.90	84.89	81.23	85.31	77.10	83.98	80.23
	TabNet	87.12	78.08	85.03	81.09	78.46	77.40	73.67	69.37	86.43	76.42	84.90	80.81	83.60	72.95	82.20	79.62
	TABTransformer	85.46	75.05	83.28	79.44	77.69	77.03	72.83	67.32	85.03	75.37	83.20	79.77	82.54	70.90	80.95	77.77

MCC		Train missing percentage: Ω				Train missing percentage: 30%				Train missing percentage: 50%				Train missing percentage: 75%			
		Test missing percentage:				Test missing percentage:				Test missing percentage:				Test missing percentage:			
Fusion Strategy	**Model**	Ω	30%	50%	75%	Ω	30%	50%	75%	Ω	30%	50%	75%	Ω	30%	50%	75%
Early ML with imputer	AdaBoost	54.79	42.40	51.87	**46.94**	44.12	44.11	40.29	30.76	53.53	45.20	50.99	42.47	51.08	42.22	46.91	37.64
	DecisionTree	47.80	38.64	47.23	40.77	38.40	36.56	36.00	32.32	43.48	32.36	43.23	37.08	33.69	24.19	33.26	28.50
	HistGradientBoost	58.61	**47.79**	53.01	44.53	47.86	45.95	40.78	**32.98**	57.44	46.68	54.51	44.51	54.05	41.73	50.22	40.28
	RandomForest	54.79	38.09	51.59	44.54	39.81	39.01	38.88	32.20	53.59	38.16	50.69	42.98	44.33	25.86	44.49	37.26
	SVM	57.10	41.61	53.19	44.17	43.28	41.26	39.00	29.67	54.38	37.35	51.54	42.02	52.83	34.05	50.11	37.68
	XGBoost	57.46	44.66	53.51	45.20	46.22	**46.28**	**41.62**	32.04	56.68	45.40	52.90	**45.54**	53.47	40.92	51.37	40.75
Early ML without imputer	DecisionTree	50.50	41.29	43.22	28.99	41.48	39.92	34.16	20.07	45.78	37.01	44.76	37.41	35.60	26.52	36.33	33.04
	HistGradientBoost	57.42	43.88	49.58	29.49	47.14	45.64	41.46	22.76	57.88	44.28	55.38	44.82	56.04	**44.07**	**54.26**	45.10
	RandomForest	56.06	38.42	50.28	33.00	42.12	40.37	31.33	20.58	52.59	34.93	51.20	41.21	46.08	25.57	45.29	42.13
	XGBoost	58.14	46.00	47.47	27.78	**48.00**	45.58	40.06	18.42	56.21	41.52	54.83	42.64	53.35	40.86	53.05	**45.43**
Early DL with imputer	FTTransformer	54.17	38.05	50.58	43.36	37.30	36.66	31.88	28.88	52.11	32.33	50.00	42.52	47.66	30.32	46.48	38.45
	MLP	51.21	34.08	48.72	42.34	36.02	35.18	32.03	26.29	48.59	28.66	47.22	40.75	44.42	25.81	44.60	36.96
	TabNet	46.36	30.30	43.46	37.48	31.89	30.90	28.81	21.53	45.93	29.30	43.93	35.34	36.05	23.08	36.13	27.17
	TABTransformer	54.49	42.26	50.37	42.80	39.70	39.38	32.66	26.18	50.57	34.41	48.52	40.81	46.82	28.53	44.49	37.93
Intermediate DL with imputer	FTTransformer	1.51	3.78	0.95	0.54	1.06	1.06	0.34	0.36	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	MLP	49.13	30.24	49.06	43.20	30.27	30.27	29.74	25.54	49.06	29.98	47.34	40.56	44.52	26.83	43.10	37.52
	TabNet	3.96	5.72	3.85	1.51	0.90	2.22	1.12	1.13	4.68	3.10	3.34	1.10	1.18	2.51	1.65	3.08
	TABTransformer	49.81	32.64	48.40	42.15	34.27	34.99	31.17	22.42	47.44	31.33	45.46	38.93	41.22	24.66	39.51	35.37
Intermediate DL	MARIA	**59.95**	47.06	**53.88**	42.60	47.77	44.40	38.09	25.97	**59.57**	**48.00**	**55.39**	40.96	**56.42**	41.22	52.40	42.27
Late ML with imputer	AdaBoost	46.42	30.37	45.47	41.85	31.14	30.32	28.16	25.78	45.78	32.32	44.54	38.01	37.39	29.56	37.58	32.85
	DecisionTree	30.16	27.84	27.18	21.15	28.41	27.06	22.40	16.46	27.99	25.08	25.78	20.66	18.89	17.24	19.61	16.24
	HistGradientBoost	35.13	22.53	32.12	26.85	25.70	26.32	22.32	13.38	35.17	27.19	30.50	22.35	30.55	19.08	26.95	22.00
	RandomForest	43.04	32.86	41.00	33.13	33.82	33.07	29.92	19.00	41.99	30.20	39.18	30.88	34.38	22.08	30.81	24.97
	SVM	46.09	27.90	44.58	35.87	27.80	27.90	26.28	17.45	45.15	27.70	43.31	33.60	42.64	26.28	39.93	31.09
	XGBoost	44.84	27.55	42.74	37.98	30.74	29.78	24.76	23.25	46.09	29.40	43.86	33.62	42.39	28.90	40.64	33.60
Late ML without imputer	DecisionTree	38.39	35.36	31.03	21.85	34.26	33.29	25.21	16.56	33.85	32.32	31.72	21.83	23.22	20.32	23.73	23.37
	HistGradientBoost	38.64	27.10	29.57	16.51	30.44	24.62	18.54	9.88	37.66	28.90	35.78	25.60	32.54	25.40	32.38	25.31
	RandomForest	51.15	38.32	44.81	23.88	41.10	39.10	28.09	9.27	48.43	33.86	44.40	30.03	40.88	26.74	39.77	36.04
	XGBoost	46.82	30.09	37.94	20.43	30.09	27.89	22.04	10.45	47.55	28.77	45.78	33.15	44.14	29.60	43.39	37.15
Late DL with imputer	FTTransformer	47.53	29.40	45.98	37.91	29.13	29.13	27.19	19.10	47.32	28.52	44.35	35.38	43.20	27.42	41.10	33.14
	MLP	43.73	27.41	42.31	36.72	26.24	26.24	24.08	21.16	43.05	25.42	41.41	33.73	40.70	24.28	38.85	32.08
	TabNet	39.52	23.30	39.00	32.32	23.50	24.38	22.82	18.08	37.61	19.68	36.40	29.02	31.43	18.09	29.39	24.57
	TABTransformer	42.25	25.60	40.67	35.87	25.24	25.40	24.04	19.02	40.01	22.42	37.77	31.07	37.42	23.32	35.89	30.19

Table F.2: Average AUC and MCC performance of the experiments across the respective tasks in the "all missing" setting. To facilitate the analysis we highlighted the best performance in each column in bold.

Appendix G

MARIA Statistical Analysis

To assess the statistical significance of the differences between MARIA and competing methods, we performed a Wilcoxon signed-rank test based on binary correctness vectors (correct vs. incorrect predictions) on the concatenated test sets of the five folds, i.e., the entire dataset. Each test was conducted independently for every experimental setting, defined by a specific combination of missingness levels in training and test sets. For each competitor, we computed the rate of experiments in which MARIA performed significantly better ($p < 0.05$) or significantly worse ($p < 0.05$). The results for the "missing modalities" and "all missing" settings are summarized in Tables G.1 and G.2, respectively.

Across both scenarios, MARIA consistently outperforms models relying on late fusion strategies, regardless of whether they rely on ML or DL backbones, and whether they use imputation or not. This is particularly noteworthy given that late fusion architectures are, by design, naturally equipped to handle missing modalities. However, our results show that such models often fail to fully exploit the complementary information available across modalities, leading to inferior performance compared to MARIA.

When compared against other intermediate fusion methods, MARIA achieves statistically significant wins in more than 60% of the experiments, and loses at most in 5% of cases, in the most challenging all missing setting. These results underscore the models robustness at all levels of incompleteness, effectively handling both missing features and entirely missing modalities.

Early fusion approaches present a more complex scenario. MARIA consistently outperforms both early fusion models that rely on imputation, whether based on ML or DL backbones, although the margin is narrower compared to what is observed against late and intermediate fusion strategies. Notably, early fusion methods that do not employ imputation emerge as some of the most competitive baselines: in the challenging all missing setting, MARIA still achieves statistically significant wins in over 30% of the experiments and loses

in fewer than 10%, yet the gap is considerably smaller than with other competitor groups. This result highlights the robustness of deterministic, imputation-free early fusion strategies under certain conditions, while at the same time reinforcing MARIA's superior flexibility in handling both partially and fully missing data, thanks to its attention-based architecture, which dynamically adjusts to the available information at both the feature and modality levels.

Overall, these results confirm that MARIA maintains a clear statistical advantage across all fusion strategies and backbone types, with the most notable gains observed against late fusion and imputation-dependent methods. Its ability to simultaneously manage missing features and entire modalities without relying on external imputers is a key factor behind its consistent performance across a wide range of experimental settings.

		ADNI Diagnosis				ADNI Prognosis								AIforCovid				Mean	
		Binary		Multiclass		12m		24m		36m		48m		Death		Prognosis			
Approach	Model	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss
Early ML with imputer	AdaBoost	28.00	0.00	100.00	0.00	100.00	0.00	92.00	0.00	64.00	0.00	96.00	0.00	76.00	0.00	80.00	4.00	79.50	0.50
	DecisionTree	92.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	92.00	0.00	100.00	0.00	100.00	0.00	96.00	0.00	97.50	0.00
	HistGradientBoost	24.00	0.00	100.00	0.00	24.00	0.00	32.00	0.00	32.00	0.00	48.00	0.00	4.00	0.00	84.00	4.00	43.50	0.50
	RandomForest	12.00	0.00	100.00	0.00	8.00	12.00	20.00	0.00	4.00	12.00	20.00	4.00	12.00	0.00	28.00	4.00	25.50	4.00
	SVM	4.00	0.00	100.00	0.00	24.00	4.00	32.00	0.00	28.00	8.00	36.00	0.00	0.00	0.00	84.00	0.00	38.50	1.50
	XGBoost	12.00	0.00	100.00	0.00	0.00	8.00	16.00	0.00	8.00	0.00	44.00	0.00	4.00	0.00	64.00	4.00	31.00	1.50
	Mean	28.67	0.00	100.00	0.00	42.67	4.00	48.67	0.00	38.00	3.33	57.33	0.67	32.67	0.00	72.67	2.67	52.58	1.33
Early ML without imputer	DecisionTree	68.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	96.00	4.00	95.50	0.50
	HistGradientBoost	32.00	0.00	84.00	0.00	28.00	0.00	44.00	0.00	32.00	0.00	44.00	0.00	20.00	0.00	68.00	0.00	44.00	0.00
	RandomForest	4.00	0.00	76.00	0.00	8.00	12.00	8.00	0.00	0.00	4.00	8.00	4.00	16.00	4.00	0.00	4.00	15.00	3.50
	XGBoost	12.00	0.00	20.00	0.00	0.00	8.00	20.00	0.00	4.00	4.00	24.00	0.00	0.00	0.00	0.00	4.00	10.00	2.00
	Mean	29.00	0.00	70.00	0.00	34.00	5.00	43.00	0.00	34.00	2.00	44.00	1.00	34.00	1.00	41.00	3.00	41.12	1.50
Early DL with imputer	FTTransformer	24.00	4.00	100.00	0.00	40.00	0.00	28.00	0.00	32.00	0.00	48.00	0.00	0.00	0.00	40.00	4.00	39.00	1.00
	MLP	28.00	0.00	100.00	0.00	80.00	0.00	92.00	0.00	72.00	0.00	68.00	0.00	0.00	0.00	76.00	0.00	64.50	0.00
	TabNet	84.00	0.00	100.00	0.00	4.00	0.00	84.00	0.00	76.00	0.00	100.00	0.00	0.00	0.00	96.00	0.00	68.00	0.00
	TABTransformer	24.00	0.00	100.00	0.00	48.00	0.00	92.00	0.00	72.00	0.00	60.00	0.00	4.00	0.00	76.00	4.00	59.50	0.50
	Mean	40.00	1.00	100.00	0.00	43.00	0.00	74.00	0.00	63.00	0.00	69.00	0.00	1.00	0.00	72.00	2.00	57.75	0.38
Intermediate DL with imputer	FTTransformer	96.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	0.00	0.00	96.00	0.00	86.50	0.00
	MLP	0.00	4.00	100.00	0.00	20.00	0.00	52.00	0.00	28.00	0.00	64.00	0.00	0.00	0.00	84.00	0.00	43.50	0.50
	TabNet	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	96.00	0.00	100.00	0.00	0.00	0.00	96.00	0.00	86.50	0.00
	TABTransformer	16.00	0.00	100.00	0.00	48.00	0.00	92.00	0.00	76.00	0.00	96.00	0.00	0.00	0.00	76.00	0.00	63.00	0.00
	Mean	53.00	1.00	100.00	0.00	67.00	0.00	86.00	0.00	75.00	0.00	90.00	0.00	0.00	0.00	88.00	0.00	69.88	0.12
Late ML with imputer	AdaBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	DecisionTree	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	HistGradientBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	RandomForest	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	SVM	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	XGBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	Mean	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
Late ML without imputer	DecisionTree	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	HistGradientBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	RandomForest	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	XGBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	Mean	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
Late DL with imputer	FTTransformer	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	MLP	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	TabNet	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	TABTransformer	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	Mean	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00

Table G.1: Percentages of wins and losses of MARIA correct predictions compared with the competitors in the "missing modalities" scenario using the Wilcoxon signed-rank test.

		ADNI Diagnosis				ADNI Prognosis								AIforCovid				Mean	
		Binary		Multiclass		12m		24m		36m		48m		Death		Prognosis			
Approach	Model	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss	% Win	% Loss
Early ML with imputer	AdaBoost	33.33	0.00	100.00	0.00	100.00	0.00	77.78	0.00	37.50	0.00	44.44	0.00	31.25	0.00	18.75	12.50	55.38	1.56
	DecisionTree	88.89	0.00	100.00	0.00	100.00	0.00	88.89	0.00	87.50	0.00	88.89	0.00	100.00	0.00	81.25	0.00	91.93	0.00
	HistGradientBoost	22.22	0.00	100.00	0.00	11.11	11.11	22.22	11.11	0.00	12.50	22.22	0.00	0.00	25.00	12.50	18.75	23.78	9.81
	RandomForest	11.11	0.00	100.00	0.00	22.22	11.11	11.11	11.11	12.50	31.25	0.00	11.11	0.00	25.00	0.00	37.50	19.62	15.88
	SVM	11.11	0.00	100.00	0.00	22.22	0.00	0.00	11.11	0.00	12.50	0.00	11.11	0.00	25.00	43.75	0.00	22.13	7.46
	XGBoost	33.33	0.00	100.00	0.00	0.00	11.11	22.22	11.11	0.00	18.75	11.11	11.11	0.00	18.75	31.25	12.50	24.74	10.42
	Mean	33.33	0.00	100.00	0.00	42.59	5.55	37.04	7.41	22.92	12.50	27.78	5.55	21.88	15.62	31.25	13.54	39.60	7.52
Early ML without imputer	DecisionTree	77.78	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	87.50	0.00	68.75	0.00	91.75	0.00
	HistGradientBoost	11.11	0.00	33.33	0.00	22.22	0.00	11.11	11.11	12.50	6.25	22.22	0.00	0.00	18.75	0.00	43.75	14.06	9.98
	RandomForest	0.00	0.00	33.33	0.00	11.11	22.22	11.11	11.11	0.00	12.50	11.11	11.11	0.00	25.00	0.00	62.50	8.33	18.05
	XGBoost	11.11	0.00	22.22	11.11	22.22	22.22	33.33	22.22	25.00	6.25	22.22	0.00	0.00	12.50	0.00	18.75	17.01	11.63
	Mean	25.00	0.00	47.22	2.78	38.89	11.11	38.89	11.11	34.38	6.25	38.89	2.78	21.88	14.06	17.19	31.25	32.79	9.92
Early DL with imputer	FTTransformer	11.11	0.00	100.00	0.00	33.33	0.00	22.22	11.11	12.50	25.00	22.22	11.11	0.00	25.00	0.00	12.50	25.17	10.59
	MLP	44.44	0.00	100.00	0.00	66.67	0.00	55.56	0.00	50.00	0.00	44.44	0.00	0.00	25.00	37.50	12.50	49.83	4.69
	TabNet	88.89	0.00	100.00	0.00	33.33	0.00	66.67	0.00	56.25	0.00	55.56	0.00	0.00	18.75	56.25	12.50	57.12	3.91
	TABTransformer	22.22	0.00	100.00	0.00	66.67	11.11	55.56	0.00	56.25	0.00	44.44	0.00	0.00	25.00	43.75	6.25	48.61	5.29
	Mean	41.66	0.00	100.00	0.00	50.00	2.78	50.00	2.78	43.75	6.25	41.66	2.78	0.00	23.44	34.38	10.94	45.18	6.12
Intermediate DL with imputer	FTTransformer	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	0.00	25.00	93.75	0.00	86.72	3.12
	MLP	0.00	0.00	100.00	0.00	22.22	0.00	33.33	11.11	31.25	12.50	44.44	0.00	0.00	25.00	43.75	12.50	34.37	7.64
	TabNet	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	0.00	25.00	87.50	0.00	85.94	3.12
	TABTransformer	11.11	0.00	100.00	0.00	77.78	0.00	66.67	0.00	56.25	0.00	44.44	0.00	0.00	25.00	56.25	6.25	51.56	3.91
	Mean	52.78	0.00	100.00	0.00	75.00	0.00	75.00	2.78	71.88	3.12	72.22	0.00	0.00	25.00	70.31	4.69	64.65	4.45
Late ML with imputer	AdaBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	DecisionTree	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	HistGradientBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	RandomForest	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	SVM	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	XGBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	Mean	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
Late ML without imputer	DecisionTree	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	HistGradientBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	RandomForest	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	XGBoost	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	Mean	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
Late DL with imputer	FTTransformer	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	MLP	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	TabNet	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	TABTransformer	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00
	Mean	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00	100.00	0.00

Table G.2: Percentages of wins and losses of MARIA correct predictions compared with the competitors in the "all missing" scenario using the Wilcoxon signed-rank test.