Reliable and fair machine learning for risk assessment

Pereira Barata, A.P.

Chapter 2

Imputation versus Missing-Indicator

Missingness is a ubiquitous problem inherent to real-world data. When learning a classifier, missing data is detrimental to the classification performance of the final model. Approaches to deal with missingness can be partitioned into methods that either (a) impute or (b) encode missingness.

Depending on the missing mechanism, some missing data-handling techniques are best suited than others in combination with different learners. Under a non-MCAR mechanism —typical of real-world data— a straightforward approach is to apply the missing-indicator method. However, a non-MCAR missing mechanism is not always guaranteed and testing for it does not ensure a reliable result. In this chapter, we experimentally demonstrate that —under MCAR— the negative impact in downstream classification performance derived from the inadequate application of the missing-indicator can be made identical to that of the application of imputation, particularly by deploying a decision tree-based learning algorithm via gradient boosting.

Therefore, a solution to the problem of missing data is to deploy the missing-indicator method in conjunction with a decision tree-based learner, particularly via gradient boosting, therewith addressing RQ1: given data with missing values, which (a) missing data-handling technique and (b) learning algorithm should be jointly selected such that, regardless of the missing mechanism, the detriment to the downstream task performance is minimal when compared to the non-missing (unavailable) case?

The current chapter corresponds to the following publication:
Pereira Barata, A., Takes, F. W., van den Herik, H. J., and Veenman, C. J. (2019). Imputation methods outperform missing-indicator for data missing completely at random. In *2019 International Conference on Data Mining Workshops (ICDMW)*, pages 407–414. IEEE

2.1 Minimising the Impact of Missing Data

Big data analytics encompasses a multitude of challenges in relation to different data aspects or characteristics. Other than volume, variety, and velocity, the concept of veracity (i.e., data quality) plays a key role when addressing real-world problems. As discussed in [Olson, 2003], the quality of data is intrinsically related to the intended use of the data itself. Moreover, to satisfy this notion of usability, data must be trusted and timely, as well as both accurate and complete. In this work we shall be focusing on the latter mentioned aspect of data (completeness), or rather its conceptual counterpart: data missingness.

The phenomenon of missing data is defined as the absence of observational values within a dataset. It is a widespread obstacle which presents itself in many fields of research where data are analysed, such as econometrics [Dardanoni et al., 2011], psychology [Schlomer et al., 2010], and epidemiology [Pedersen et al., 2017]. Regardless of the underlying reasons for the occurrence of missingness throughout different domains, missing data presents a challenge towards the completion of any data-related task.

The task to be performed after imputation of the dataset is referred to as the downstream task (e.g., regression or classification). Choosing how to handle this issue will influence the outcome of the downstream task. In other words, poor application of missing data-handling techniques leads to underwhelming performance and biased results [Choi et al., 2019]. Thus, depending on the problem to be addressed, it is important to carefully select the most appropriate strategy to overcome missingness and minimise the impact of incomplete data on the final outcome of the downstream task [Little et al., 2014]. Also, the type of data that is missing influences the selection of imputation approaches [Feng et al., 2011].

We know from [Garciarena and Santana, 2017] that the effectiveness of an imputation method in classification is tightly associated to the family of classifiers to be used and the missing mechanism affecting the data. In this context, by *family* we mean a set of classification algorithms of which the decision functions are conceptually similar; i.e., algorithms of which the mappings of the input space into a specific category are alike. For example, a tree-based algorithm recursively splits the original input space into segments through a set of relation operator-based rules, whereas a k-nearest neighbours approach checks the mode of the closest k objects according to some distance metric (often euclidean): we consider the two methods to pertain to different families. By missing mechanism, we are referring to the distribution of missing values in the data. It is common practice to categorise these mechanisms as MCAR, MAR, and MNAR. In the real world, it is only possible to distinguish between MCAR and not-MCAR mechanisms.

Generally, imputation methods rely on statistical concepts (e.g., mean and median) or machine learning approaches (i.e., predictions over missing values). Another commonly used approach to data imputation is the missing-indicator method [Huberman and Langholz, 1999], where a new placeholder value or attribute for missingness is generated to indicate the missing value. Although past studies have been conducted to illustrate how these methods affect the bias of results [Knol et al., 2010], there exist several gaps in the literature, of which we mention three.

First, often the missingness characteristics are not fully reported [Malla et al., 2018]. Second, results can be related to one specific domain rather than to a more general level [García-Laencina et al., 2015]: this leads to conclusions that are either ambiguous or not generalisable with respect to the distinct field task of the reported work. Third, authors tend to disagree on which metrics are best at quantifying imputation effects [Van Buuren, 2018].

Stating that one method of imputation outperforms another method is dependant on the type of performance analysis conducted and the missingness assumptions of mechanisms at play [Santos et al., 2019b]. In general, the purpose of imputation is not making a dataset complete, but rather make it possible to handle data for a specific task. Regardless, research in imputation usually reports performance as a function of error between the artificially removed values and the predicted imputation [Amiri and Jensen, 2016]. This is not a viable metric to compare different imputation methods and the missing-indicator method.

A more realistic approach to measuring the impact of imputation methods is to assess the performance on the post-imputation (downstream) task. Little research has been done in measuring how of the missing-indicator approach to handling missing data performs in classification problems compared to imputation methods [Ding and Simonoff, 2010].

In this work, we compare several imputation methods and the missing-indicator method, and measure their differences on the most relevant measure: classification performance. We establish whether the missing-indicator should or should not be used given the specific case of classification problems with numerical data, under the MCAR mechanism of missingness.

The structure of this chapter is as follows: Section 2.2 will deal with the basic concepts which distinguish different types of missingness and how to synthetically generate missing data. In Section 2.3, we briefly describe the added value of the presented research with respect to past work. Section 2.4 consists of the description of the materials and methods used, while in Section 2.5 we refer to our experimental setup and results. Finally in Section 2.6 our conclusions are given and future research directions are suggested.

2.2 Missing Data

Here we introduce the overall concept of missingness. We illustrate and define each missing mechanism in Section 2.2.1. In Section 2.2.2, we further present how to generate missing values under the specific MCAR assumption.

2.2.1 Mechanisms of Missingness

In research on imputation, missing mechanisms are commonly defined according to the distribution of missing values [Little and Rubin, 2019]. In this manner, data can be missing under three different assumptions: MCAR, MAR, and MNAR. Under MCAR, the probability that a data value is missing is the same for all data points. MAR occurs when the events that lead to missingness are completely at random but only within a subset of some other observed variable within that dataset. Lastly, when neither of the previous two missing mechanisms are at play, but rather the missingness is directly related to the actual value that is missing and/or some other variable value, then the missing mechanism is referred to as MNAR. Table 2.1 illustrates the aforementioned mechanisms. In this example, each row represents an instance in a dataset.

The first column (*Class*) represents some class label, and the remaining columns represent the same feature under different missing mechanisms: *Complete* signifies the observations without any missing values, whereas the three remaining columns illustrate how the different missing mechanisms would affect the set of observations. Each number denotes an observed value.

Following the notation of previous literature [Little and Rubin, 2019], we formally define the different missing mechanisms as follows. Let X be an n by p matrix serving as some dataset with $i = 1, ..., n$ instances and $j = 1, ..., p$ features where $x_{i,j}$ is an individual element of X. Each element $x_{i,j}$ may represent either an observation or a missing value, depending on the characteristics of thee dataset. We can divide X into two disjoint objects, $X = (X^{obs}, X^{miss})$, where X^{obs} and X^{miss} represent the observed and missing values of X. Let M be a matrix of the same shape as X with $m_{i,j} \in M$ where $m_{i,j} = 0$ and $m_{i,j} = 1$ indicate the presence or absence of observation $x_{i,j} \in X$, respectively. Then, the missing mechanism is MCAR if:

$$Pr(M = 1|X^{obs}, X^{miss}) = Pr(M = 1); \tag{2.1}$$

MAR if:

$$Pr(M = 1|X^{obs}, X^{miss}) = Pr(M = 1|X^{obs}); \tag{2.2}$$

and MNAR if:

$$Pr(M = 1|X^{\text{obs}}, X^{\text{miss}}) = Pr(M = 1|X^{\text{obs}}, X^{\text{miss}}). \tag{2.3}$$

Table 2.1: Three different missing mechanisms

Class	**Feature**			
	Complete	MCAR	MAR	MNAR
0	18.91	–	–	–
0	13.42	13.42	–	–
0	4.05	–	4.05	4.05
0	4.06	4.06	4.06	4.06
0	18.24	18.24	–	–
0	3.01	–	–	3.01
0	11.37	11.37	11.37	–
0	14.25	–	14.25	–
0	2.74	2.74	2.74	2.74
0	5.24	5.24	–	5.24
0	10.21	–	–	–
1	11.02	11.02	11.02	–
1	7.06	–	7.06	7.06
1	14.29	14.29	14.29	–
1	2.16	–	2.16	2.16
1	5.26	5.26	5.26	5.26
1	0.37	–	0.37	0.37
1	8.24	8.24	8.24	8.24
1	10.36	–	10.36	–
1	6.43	6.43	6.43	6.43
1	1.31	–	1.31	1.31

While it is possible to define these concepts, accurately determining which of these assumptions permeates a dataset is no easy task: the information required to discriminate between MAR and MNAR is, rather unsurprisingly, missing itself. However, such is not the case for MCAR that can be tested for statistically [Little, 1988] albeit false positives and false negatives may still occur.

We know from [Van Buuren, 2018] that specific imputation methods that perform well under some condition might not be applicable under another condition. As such, it is not only important to determine what mechanism shapes the missingness within the data that will be used to address a particular problem, but it is also imperative to report it.

2.2.2 Synthesising Missing Values

Imputation studies rely heavily on generating synthetic missingness [Bertsimas et al., 2017]. The removal of observations can be labelled as either univariate or multivariate, depending on the number of features selected to have their observations deleted by some percentage. The synthesis of missingness varies depending on the target missing mechanism to be implemented, as different conditions have to be met to satisfy specific occurrences of missing values.

Under MCAR univariate missingness generation, selecting the feature of which values will be removed can be performed either randomly [Rieger et al., 2010] or under some other condition imposed by the researcher [Twala, 2009]. In the multivariate case, past work mainly distinguishes between a local vs global [Garciarena and Santana, 2017] approach to value removal; the former ensures that every feature has the same proportion of missing values, while the latter considers the entire dataset for value deletion which does not ensure such a stratified missingness condition. The generation of synthetic missing data will be further described in the context of our methods.

2.3 Related Work

The concept of missing data in literature is addressed from different perspectives depending on the purpose of the research itself. While some studies approach missingness as a preprocessing step in their actual endeavour, other work focuses solely on the techniques used to do so. This dichotomy highlights different view points, depending whether or not missingness of data is the object of interest in a study. We specifically elaborate on past work that relates to how missingness is reported and handled when solving real-world problems; i.e., the application of the missing-indicator method in the context of specific domains. Moreover, we highlight the performance measurement methods applied in the context of missing data-handling techniques according to the current literature.

Following the first topic of interest, authors in [Malla et al., 2018] conducted an overview study of how missingness is addressed in the context of propensity score estimation; 167 articles were analysed in their research. Nearly 68% of these articles based their findings on assumptions that would only hold if data were MCAR. However, only one of these studies presented evidence for such a strong assumption. The remainder offered no explanation towards the reason data was missing nor which missing mechanism was at play. This observation led to biased results and skewed reported conclusions which posed a serious issue, especially given that the contexts of these studies were medical trials.

In other medicine-related domains scrutinised by us, contradicting evidence is reported with respect to the application of the missing-indicator method. Specifically, while authors in [Groenwold et al., 2012] state that *"the missing-indicator method is a valid method to handle missing baseline covariate data, irrespective of the mechanism of missingness"*, the work presented in [Van der Heijden et al., 2006] concluded that *"in multivariable diagnostic research complete case analysis and the use of the missing-indicator method should be avoided, even when data are MCAR"*. This discrepancy in the current literature is indicative of a real substantial problem that can only be addressed through further exposition of missing data as a subject of interest, by generating research that aims to offer general guidelines that practitioners may follow on how to handle missing data. In this sense, our contribution addresses whether a particularly common method of handling missing data – the missing-indicator method – should be used under the testable MCAR mechanism scenario.

Focusing on a different scope, studies such as [Amiri and Jensen, 2016] report on imputation techniques and their performance. In this work, the authors developed a novel imputation method to be applied on missing numerical data. They compared their method against frequently used imputation methods and reported the comparative performances yielded. The performance was measured as a function of the error between the imputed values and the original values. Despite being an intuitive approach, using this error as a performance measure for imputation is not adequate. This is thoroughly elaborated in [Van Buuren, 2018], where the distinction between predictive methods and imputation is established. The author ultimately asserts that imputation is not prediction and states that *"we cannot evaluate imputation methods by their ability to re-create the true data"*. In short, using regression error to measure performance leads to biased conclusions.

Studies such as [Garciarena and Santana, 2017] address this imputation performance issue by using classifier performance (downstream task) as a proxy for imputation adequacy. In such a study, different imputation techniques (both single and multiple) were used, in association with several classification algorithms across distinct datasets. However, the missing-indicator method was not encompassed within the experimental setup provided.

Taking these facts into consideration, we propose to comprehensively measure the impact of the missing-indicator method. We do so by using a downstream task as a viable proxy for missing data-handling performance under MCAR. Since MCAR is the only missing mechanism that can be tested against in real-world problems, we specify it as the base for our work; in this manner we ensure that the conditions under which our controlled experiments are performed can be statistically diagnosed in real-world scenarios.

2.4 Method

Here we provide the resources and methodology used. We begin with Section 2.4.1 by describing the application of the missing-indicator in the context of our work. In Section 2.4.2, we summarise the datasets we use in our experiments. Section 2.4.3 reports how missing data was synthesised. Lastly in Section 2.4.4, imputation and classifier methods are denoted.

2.4.1 Missing-Indicator

The missing-indicator method should not be regarded as an imputation method in itself. Rather, it can and should be viewed as an addition to any imputation being performed. In other words, regardless of what approach is used to fill in missing values – mean, median, regression-based imputations, etc. – the missing-indicator will always be applicable. The underlying concept of the missing-indicator method focuses on the encoding of missingness itself. In practice, this encoding can be regarded as the addition of a binary indicator variable. Concretely, our implementation of the missing-indicator method is as follows: every missing value is replaced with a placeholder value; then a second column is created for every feature with missing values. This new column holds values of either 0 or 1 representing the absence or presence of a missing value in the original feature, respectively (Fig. 2.1). This approach is derived from past literature, where every value $x_{i,j} \in X$ is replaced by the product of itself multiplied by $(1 - m_{i,j} \in M)$ [Bennett, 2001]. Our choice of a placeholder value of zero reflects also the consensus in methodological approaches applied by practitioners [Zhang, 2016].

Feature 1	Feature 2
18.91	14.25
13.42	—
—	2.5
4.06	—

Feature 1	Feature 2	Indicator 1	Indicator 2
18.91	14.25	0	0
13.42	0	0	1
0	2.5	1	0
4.06	0	0	1

Figure 2.1: **Imputation through missing-indicator.** The table to the left represents a 4-rows slice of some dataset with missing values in feature column 1 and feature column 2. The table to the right represents the yielded version of the previous table using the missing-indicator method.

2.4.2 Data

A total of 22 datasets were collected from an open-source dataset repository [Alcalá-Fdez et al., 2009], all of which were associated with a classification task.

Every dataset is comprised of a set of numerical features and a class column. These datasets are complete (i.e., no missing values) and vary significantly in sample size, dimensionality, and class balance. The definition of class balance follows.

Definition 2.1 – ***Class balance***

Class balance is the quantification of the difference between the number of samples pertaining to the positive and negative classes in a dataset.

A summary of the datasets can be seen in Table 2.2. The column *Class Balance* represents the ratio between the frequency of the minority class and the majority class: a value close to 0 indicates large class imbalance, while a value of 1 means perfect class balance.

Table 2.2: Summary of dataset characteristics

Dataset	#Rows	#Features	Class Balance
Appendicitis	106	7	0.25
Australian	690	14	0.80
Bands	365	19	0.59
Bupa	345	6	0.73
Coil2000	9822	85	0.06
Haberman	306	3	0.36
Heart	270	13	0.80
Hepatitis	80	19	0.19
Ionosphere	351	33	0.56
Magic	19020	10	0.54
Mammographic	830	5	0.94
Monk-2	432	6	0.90
Phoneme	5404	5	0.42
Pima	768	8	0.54
Ring	7400	20	0.98
Sonar	208	60	0.87
Spambase	4597	57	0.65
Spectfheart	267	44	0.26
Titanic	2201	3	0.48
Twonorm	7400	20	1.00
Wdbc	569	30	0.59
Wisconsin	683	9	0.54

We specify class balance as it plays a role when selecting the appropriate performance metric to measure classifier performance. Past literature states that given an imbalanced classification problem, the area under the precision-recall curve is more informative than the AUC [Saito and Rehmsmeier, 2015], which is equivalent to the average precision measure (AP). Thus, we usedAP to measure classifier performance, as well as specifying the minority class as the class to be modelled for every dataset. Given our aim at producing a comparative study, we address data and its characteristics in this segment rather than as part of our experimental setup.

2.4.3 Value Removal

To compare different imputation methods and their impacts on classifier performance for various missingness rates, artificially removal of values from the complete datasets was required. Following common use in literature, we removed 10%, 20%, 30%, 40%, and 50% of observations to measure how different percentages of missing values affect the impact of particular missing data-handling methods. We cap the missing proportions at 50% since higher values might damage the original dataset too much to extract meaningful results.

We used a multivariate local approach to generate missing values. In this manner, every feature had the same percentage of missing values for a given proportion of missingness. The removal followed a uniform distribution so that every value within a feature vector had the same probability of being removed. In practice, given some dataset subset with n observations, and p features, for a fixed missing proportion $q \in [0, 1]$, each feature vector had $\lceil q \times n \rceil$ observations removed. For clarification, we remark that a class label is not a feature.

We specify the notion *subset* because we did not apply missingness to the entire dataset at once. Rather, we first split the dataset into 10 equal segments and removed observations in each segment. These segments were used later on to compute classifier performances; i.e., they comprised our train-test splits. Should we have created our train-test splits a posteriori, then segments of the dataset could have had fallen under a non-homogeneous MCAR assumption, given the randomness of the splitting process. Thus, we ensured that all train and test segments used were under the same missing proportion conditions.

2.4.4 Imputers and Classifiers

Four simple and commonly used imputation methods were selected and implemented to serve as comparison against missing-indicator: mean (Mean), median (Median), linear regression (Linear), and extreme gradient boosting regression (XGBR). These methods were selected for their differing frameworks.

One important factor to take into consideration is which imputation framework to use: single or multiple imputation. We chose to implement the single imputation variant of each algorithm for our comparisons, rather than performing a multiple imputation implementation. The rational behind our decision is given below.

In multiple imputation [Carpenter and Kenward, 2012], several distinct imputed versions of the original missing dataset are generated. In practice, this framework makes use of any single imputation method (such as linear regression imputation) and adds to it a component of randomness (by bootstrapping, for example). This generates different complete variations of the same original missing-valued dataset. The resulting analysis output of each complete dataset is then combined (i.e., pooled). We chose single over multiple imputation because all compared imputation methods could be wrapped within a multiple imputation framework; we are only interested in comparing the imputation methods themselves, not the outcome of single vs multiple imputation.

While for mean and median imputations the missing values depend on the available values in the same column, for the regression-based imputers the values in a column are assumed to depend on the values of the same sample in other columns. Consequently, for a regression-based imputer, to impute values for column j, all other columns should be complete. This observation is based on the ones used in the literature [Bertsimas et al., 2017, Van Buuren, 2018]. In the first case, random values are generated to populate the initial incomplete dataset, whereas the latter uses mean imputation to do so. We refer to these initial imputation states as *warm starts* hereinafter. A warm start is defined as follows [Van Buuren, 2018].

Definition 2.2 – ***Warm start***

A warm start in prediction-based imputation is the initial step in which the dataset is made complete so as to enable learning of a model towards predicting the imputation values for the feature of interest.

In our case, these warm starts are the complete version of a missing-valued dataset, generated through mean imputation. They serve as a starting point for regression-based imputation, allowing for regressors to be trained by using a provisional complete matrix. After a regression model is fit, imputation can be performed, and classification models can be learned from the complete data.

The classification algorithms implemented were a k-nearest neighbours classifier (KNNC), a support vector machine classifier (SVMC) with a radial basis function kernel, and an extreme gradient boosting classifier (XGBC). We chose these algorithms as they cover the spectrum of the current state-of-the-art and their documented application in several domains.

2.5 Experiments

In this segment we address our experimentation. In Section 2.5.1 we describe how the methods mentioned previously were implemented. In Section 2.5.2, we present the resulting outputs. Lastly, we discuss our results in Section 2.5.3.

2.5.1 Experimental Setup

Our implementation was performed in *Python* using peer-reviewed libraries [Pedregosa et al., 2011,Chen and Guestrin, 2016,Oliphant, 2007], which are open-source. All programming objects were initialised using the default set of parameters supplied by each object's corresponding package, safe for random state parameters. For reproducibility purposes, a random seed value of 42 was set where appropriate (i.e., tree-based algorithms and sample selection during train-test splitting). We describe and illustrate how the aforementioned methods were applied within our experimental setup.

For each dataset, we split the entire dataset into 10 segments (folds) in a stratified manner as to ensure class balance across all folds as in the entire dataset. Through 10-fold CV we computed benchmark AP values per classifier. These benchmark values are derived from the original complete datasets and will serve to illustrate the differences across imputation methods.

In each fold, features had their values deleted according to our previously defined value-removal approach. In this manner, five distinct instances of the same fold were generated where each instance has a specific percentage of missing values. Within a dataset, for each missing proportion, we created 10 train-test sets. Each of these train-test sets was comprised of one distinct fold that served as the testing subset, and the joint set of the other remaining folds that served as the training subset. This setup of train-test splits was used to both apply the aforementioned imputations methods, as well as compute classifier performances.

Every configuration of splits per dataset for a specific missing proportion contained the same index instances. In other words, every imputation and classification procedure was always applied to the same subsets of the original dataset across the different algorithms to be compared. After structuring and creating all splits per dataset according to different missing proportions, we began the imputation processes.

For the missing-indicator, imputations required no distinction between train and test subsets: for every subset, missing values are converted to 0 while adding an extra dimension per feature, valued either 0 or 1 as previously described. While this method was applied without differentiating train or test instances, such was not the case for the remaining imputers.

Both regression-based imputation methods Linear and XGBR required warm starts to be initialised and applied. Since warm starts are yielded by Mean and given that Mean and Median applied similarly (although computing different statistics), we proceed to describe the imputation setup of both these methods.

Given a train-test split within our experimental setup, let X_{train} and X_{test} be the sets corresponding to the train and test portions of the split, respectively. Let $X_{train}^{obs} \in X_{train}$ and $X_{train}^{miss} \in X_{train}$ be disjoint sets representing observed and missing values, correspondingly, of the train portion of the train-test split. Conversely, let $X_{test}^{obs} \in X_{test}$ and $X_{test}^{miss} \in X_{test}$ be disjoint sets representing observed and missing values, respectively, of the test portion of the train-test split. For each feature indexed at $j \in \{1, ..., p\}$, mean and median statistics were computed from the set of observations $x_{:,j} \in X_{train}^{obs}$, for a total of p mean values and p median values per train-test split. We then replaced the missing values in both X_{train}^{miss} and X_{test}^{miss} with the mean or median of the respective feature.

The setup used to deploy both Linear and XGBR was the same. We started by considering any arbitrary train-test split and retrieving the corresponding imputed train-test set produced through Mean. These imputed train and test subsets served as the warm starts required to generate imputations through both regression-based methods. Each of these methods generated imputations based on the regression methods associated to them. For each train-test split, a total of p regression models were learned per method, and per feature.

Let R_j be an object representing the regression model to be used to impute over feature $j \in \{1, ..., p\}$. R_j had to be passed an initial set of values from which to learn. The values were a subset of the training warm start. This subset had the same width as the original corresponding dataset, but only contained the instances for which feature j has observations. R_j would then be trained on the subset in question to learn to model feature j from all other remaining features. Past the regression-training step, imputation was performed by R_j on both test and train segments of the train-test split.

After all imputation methods were applied to all train-test splits derived from each dataset for all missing proportions, classifier performance was computed. AP values were generated through 10-fold CV using the aforementioned train-test splits. The values were then averaged so that each classifier yields one value of AP per dataset per imputation method per missing proportion.

2.5.2 Results

We computed the mean performance over the 22 datasets, fixed on missing proportion, per imputer, for each classifier. Figs. 2.2, 2.3, and 2.4 illustrate the performance per imputers KNNC, SVMC, and XGBC, respectively.

Wilcoxon signed-rank tests [Wilcoxon, 1992] were applied to measure the statistical significance of the differences between missing-indicator and the remaining methods. The resulting p-values associated with KNNC, SVMC, and XGBC are shown in Table 2.3, Table 2.4, and Table 2.5, respectively.

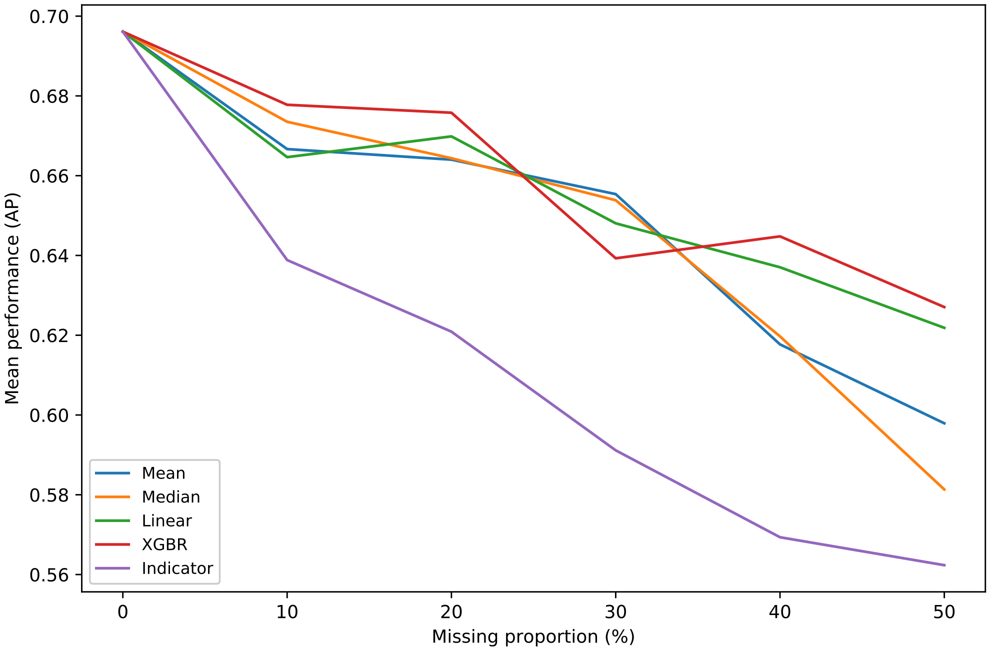

Figure 2.2: **KNNC mean performances across missing proportions.** Mean performances (vertical axis) across different proportions of missingness (horizontal axis) per five missing data-handling methods. Indicator refers to missing-indicator (purple).

Table 2.3: KNNC Wilcoxon p-values

	Mean	**Median**	**Linear**	**XGBR**
10%	0.030853	0.001549	0.020271	0.005506
20%	0.000779	0.006082	0.000136	0.001731
30%	0.002401	0.001549	0.002401	0.004981
40%	0.001932	0.001103	0.000259	0.000069
50%	0.094528	0.223429	0.015577	0.001549

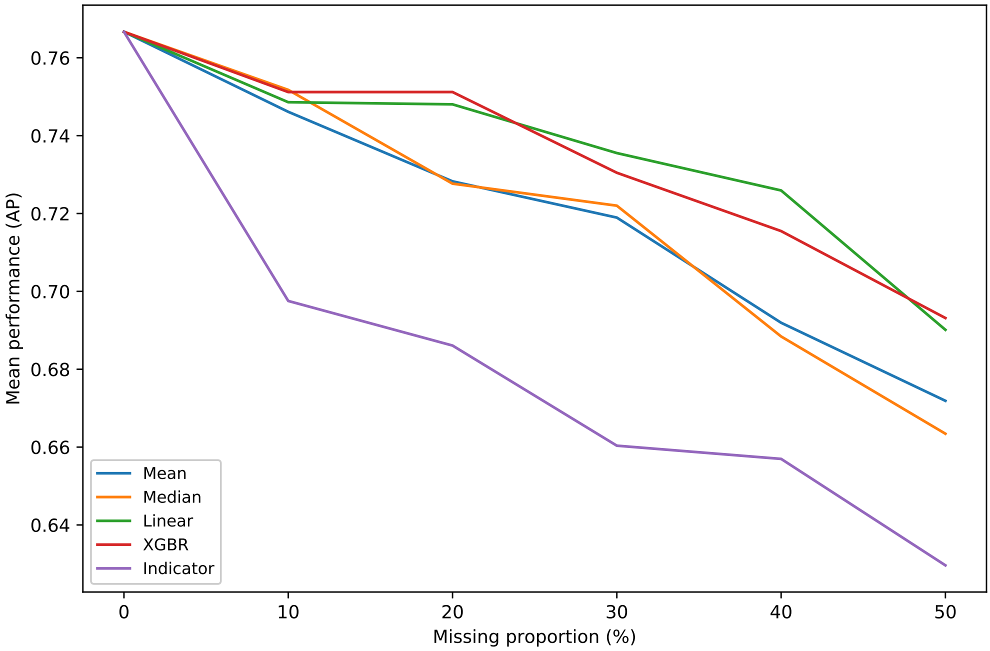

Figure 2.3: **SVMC mean performances across missing proportions.** Mean performances (vertical axis) across different proportions of missingness (horizontal axis) per five missing data-handling methods. Indicator refers to missing-indicator (purple).

Table 2.4: SVMC Wilcoxon p-values

	Mean	**Median**	**Linear**	**XGBR**
10%	0.026155	0.014239	0.004063	0.001237
20%	0.033462	0.030853	0.000136	0.000155
30%	0.001731	0.001932	0.000334	0.000615
40%	0.039249	0.013005	0.000483	0.002673
50%	0.014239	0.088298	0.000483	0.001237

Within each classifier setup fixed at missing proportion, the 22 average AP values computed using missing-indicator were compared against the remaining methods in a pairwise fashion. One p-value was yielded for each comparison of missing-indicator vs per imputation method fixed at a given missing proportion.

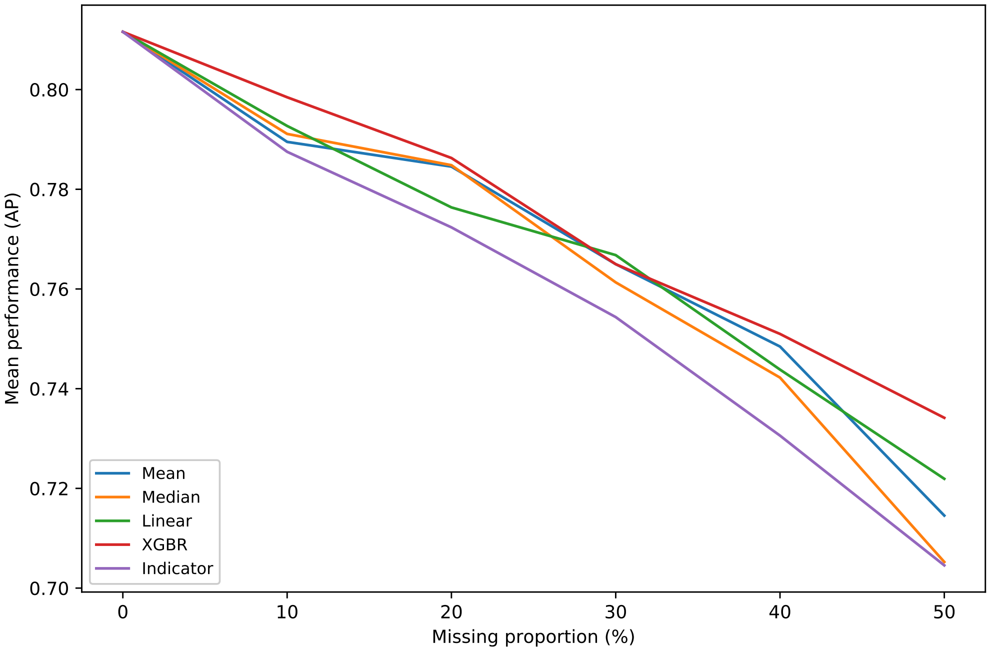

Figure 2.4: **XGBC mean performances across missing proportions.** Mean performances (vertical axis) across different proportions of missingness (horizontal axis) per five missing data-handling methods. Indicator refers to missing-indicator (purple).

Table 2.5: XGBC Wilcoxon p-values

	Mean	**Median**	**Linear**	**XGBR**
10%	0.807624	0.883846	0.987049	0.066608
20%	0.236019	0.445498	0.236019	0.082403
30%	0.445498	0.858282	0.020271	0.188557
40%	0.101106	0.426376	0.407742	0.139625
50%	0.262686	0.987049	0.066608	0.003302

2.5.3 Discussion

By definition —Eq. 2.1— the distribution of missing values under MCAR in a dataset is completely independent of any aspect of the dataset itself. By using the missing-indicator method, a new variable is introduced per missing-valued feature. The usage of the missing-indicator method generates a binary variable of which the distribution is also independent of any aspect of the data.

In other words, under the MCAR scenario this missing data-handling approach is adding a noise variable to the data which will make the classification task more difficult to solve.

Classifiers such as XGBC, which incorporate feature selection mechanisms, are less prone to be influenced by the added noise through the incorporation of missing-indicator variables (Fig. 2.4). However, should a classification algorithm be not so robust as assumed, then the overall classification performance is expected to drop. This is illustrated in Fig. 2.2 (KNNC) and Fig. 2.3 (SVMC), where an accentuated difference in average performance with missing-indicator vs all other imputers can be seen. The distinction is further denoted statistically by analysing Tables 2.3, 2.4, and 2.5. In the first two tables, nearly all p-values are under 0.01 (i.e., significant difference between missing-indicator and imputation methods). Table 2.5 suggests little disparity in performance between imputation and missing-indicator methods.

2.6 Chapter Conclusion

Handling missing data is a general problem encountered in most machine learning tasks. Different methods exist in the literature to address this problem, of which imputation and missing-indicator are the most predominant. Depending on underlying missing mechanism, the learner used, and the choice of missing data-handling method, the downstream task performance may vary.

When dealing with real-world data, a non-MCAR scenario is traditionally assumed and a viable option is to use the missing-indicator method. However, even with a negative MCAR test output, a false negative may still be possible which may jeopardise the performance of the downstream task. Accounting for this liability, it is necessary to attenuate the performance decrease derived from using the missing-indicator method under MCAR.

In this work, we have extensively assessed the performance of the missing-indicator approach under the testable MCAR missing mechanism towards a downstream classification task. We compared it to common imputation methods based on both statistical and machine-learning approaches. We computed classifier performances from three distinct algorithms applied to 22 datasets, each instanced with different proportions of missing values.

The comparative impact on classifier performance of each imputer was illustrated, and statistical significance tests were applied to further validate our findings. We observed that, as expected, the missing-indicator method systematically underperforms relative to all imputation methods. Yet, the negative impact of the missing-indicator method (compared to imputation methods) can be made negligible via adequate learner selection.

In conclusion, our research shows that the missing-indicator method is a viable option when handling real-world data, even if the missing mechanism is not correctly assessed, so long as a decision tree-base learner is used,concretely via gradient boosting. As a closing remark for upcoming research, we state that using real missing-valued datasets rather than ones with synthetically generated missingness might provide more realistic and robust results.