Artificial Intelligence Review manuscript No.
(will be inserted by the editor)

A Data-Driven Missing Value Imputation Approach for Longitudinal Datasets

Caio Ribeiro · Alex A. Freitas

Received: date / Accepted: date

Abstract Longitudinal datasets of human ageing studies usually have a high volume of missing data, and one way to handle missing values in a dataset is to replace them with estimations. However, there are many methods to estimate missing values, and no single method is the best for all datasets. In this article, we propose a data-driven missing value imputation approach that performs a feature-wise selection of the best imputation method, using known information in the dataset to rank the five methods we selected, based on their estimation error rates. We evaluated the proposed approach in two sets of experiments: a classifier-independent scenario, where we compared the applicabilities and error rates of each imputation method; and a classifier-dependent scenario, where we compared the predictive accuracy of Random Forest classifiers generated with datasets prepared using each imputation method and a baseline approach of doing no imputation (letting the classification algorithm handle the missing values internally). Based on our results from both sets of experiments, we concluded that the proposed data-driven missing value imputation approach generally resulted in models with more accurate estimations for missing data and better performing classifiers, in longitudinal datasets of human ageing. We also observed that imputation methods devised specifically for longitudinal data had very accurate estimations. This reinforces the idea that using the temporal information intrinsic to longitudinal data is a worthwhile endeavour for machine learning applications, and that can be achieved through the proposed data-driven approach.

Keywords Longitudinal datasets · Missing Value Imputation · Supervised Machine Learning · Class Imbalance

1 Introduction

Longitudinal studies take repeated measures of a set variables, from the same group of subjects, over time several points. The longitudinal datasets derived from

Caio Ribeiro and Alex A. Freitas
School of Computing, University of Kent. Canterbury, United Kingdom.
cer28@kent.ac.uk, A.A.Freitas@kent.ac.uk

these studies are prone to high amounts of missing data, mainly due to attrition (for example, subjects dropping out) (Engels and Diehr, 2003), which can have a significant effect on the analysis of the data.

When a longitudinal dataset with missing data is used in a machine learning (ML) application, the missing values can be handled in different ways. Instances or features with missing values can be removed from the dataset (with the drawback of losing data), ignored (i.e., the ML algorithm has to handle them during its execution), or replaced by an estimated value (missing value imputation). Naturally, when replacing a missing value for an imputed one, it is desirable to make estimations close to what the real value would be.

There are several methods to estimate the value to impute in place of a missing value in the dataset, usually based on information from its known values, and different methods may perform better (achieve estimations closer to the real value) for different situations (Diggle, 2002; Hu et al., 2017; Mallinckrodt, 2013). The performance (i.e., error rate) of a missing value imputation (MVI) method depends on several factors, such as: a) the data distribution (Santos et al., 2017); b) how the missing data appears in the dataset (missing completely at random, missing at random, or missing not at random) (Diggle, 2002; Mallinckrodt, 2013); c) the proportion of instances with missing values; d) the availability of information that can be used to make better imputation.

Typically, statisticians decide how to handle missing data based on assumptions about how the missing values were introduced into the dataset, and based on data distribution and characteristics. However, these assumptions cannot be proven if we do not have all the information about the data collection process, which is the case for most classification problems. That challenge is even greater for longitudinal data, where there are additional ways we can extract information from the dataset to make the estimations.

Instead of making a decision directly, we can choose the best approach based on the data itself. A data-driven approach would avoid additional human bias. Thus, letting the data dictate what is the best approach to estimate its missing values is a viable choice, for ML applications. In this article, we expand on that notion by having this data-driven choice be made separately for each feature in the dataset, as the characteristics of different variables might lead to different MVI strategy choices.

As the main contribution of this work, we propose a novel data-driven approach to select the best method for imputing the missing values in a longitudinal dataset, out of five selected candidate MVI methods. Our aim is to ensure that the MVI method that best fits the characteristics of each feature the dataset is used for that feature. In our approach, the MVI methods are ranked for each feature (variable) in the dataset, and used from best to worst ranked until no missing values remain or no methods can be used.

Our proposed data-driven MVI approach was evaluated using 10 longitudinal datasets created for the ML task of classification. Such datasets are composed of instances (the subjects to be classified) and features, which are variables describing each subject, usually with repeated measures for each time point (called wave) in the dataset. Classification algorithms aim to predict the value of a nominal class variable for an instance, based on the values of its features. These algorithms use training data (a set of instances with known class values) to create a model for predicting the class of previously unseen instances (test data).

The longitudinal datasets were created using data from the English Longitudinal Study of Ageing (ELSA) (Banks et al., 2019). The ELSA study interviews its core participants (who are at least 50 years old) repeatedly, over the years prior to their retirement and beyond, collecting data on health, social, wellbeing and economic circumstances. In our datasets, which focus on biomedical data, the overall proportion of missing values is 38.5%. This is a strong motivation to investigate the effectiveness of several missing value imputation methods.

To evaluate the performance of our proposed data-driven MVI approach, we performed experiments using our ELSA datasets as a benchmark to compare its effectiveness against five missing value imputation methods. These six methods were compared in two scenarios: a scenario independent from any classifier (expanded from the experiments reported in (Ribeiro and Freitas, 2019)), and another scenario where a Random Forest (RF) classifier was trained with datasets with estimated missing values.

In the classifier-independent comparison, the six MVI methods are compared based on their applicability (ratio of missing values for which the method can be used) and estimated average error rate. To calculate this error rate, we use each method to estimate known-data in the dataset and compare the estimated and real values. As expected, the proposed data-driven MVI approach performed overall the best in this comparison, with small error rates and 100% applicability (i.e., all missing values replaced). We also observed that the MVI methods devised specifically for longitudinal data yielded very precise estimations, although they had low applicability.

In the classifier-dependent scenario, we evaluate the impact that the strategy to handle missing values had on the predictive accuracy of a RF classifier. We compared the six methods from the scenario above, adding a baseline approach of doing no imputation, and letting the classifier handle the missing values during its execution. We report on the Sensitivity, Specificity and Accuracy of the models generated for each of the 10 datasets, and apply a non-parametric statistical test to determine whether the classifiers' performances significantly changed. The proposed Data-Driven approach and the K-Nearest-Neighbour approach had the best results in these experiments.

Overall, the contributions of this article can be summarised as follows. We propose the aforementioned data-driven approach that automatically selects the best MVI method (out of a set of predefined candidate methods) for each feature in the dataset. In this paper the data-driven approach was used to select the best out of 5 MVI methods, but the basic idea of the data-driven approach can be used with any set of MVI methods chosen by the user. We concluded that the proposed data-driven MVI approach was the best performing MVI method in our experiments, based on two evaluation scenarios (depending on whether or not a classifier is used to evaluate the results) and several performance criteria: applicability and error rates, when not using a classifier; and a classifier's predictive performance, when a classifier is applied to the data whose missing values were imputed by the MVI methods.

In addition, our experiments reinforced the notion that no MVI method outperforms all others in all occasions, and showed that it is worthwhile to use information in the known data to select the best MVI method for each feature. Furthermore, our results with the MVI methods devised for longitudinal data highlight a need for developing techniques that handle the unique characteristics of this type of data, which is prominent in long-term health studies.

This article is organised as follows. In Section 2 we describe the dataset creation process for our ELSA datasets. Section 3 describes background and related works on missing value imputation, as well as the missing value imputation methods used in our experiments. Section 4 presents our proposed data-driven MVI approach. Section 5 describes our methodology for evaluating the proposed approach, and the experimental results are presented in Sections 6 and 7, for the classifier-independent and classifier-dependent scenarios, respectively. Finally, Section 8 presents our conclusions and future work suggestions.

2 Longitudinal Dataset Creation for the Classification Task

The English Longitudinal Study of Ageing (ELSA) is currently one of the most prominent populational studies of ageing (Banks et al., 2019). The study has, in each of its waves, thousands of respondents from inhabitants of United Kingdom households, which are visited and interviewed every two years (the time interval between two consecutive waves). The study is intended for 50 years of age or older respondents, because it aims to follow the participants for years prior to their retirement and beyond (Banks et al., 2016).

A series of questionnaires are used to collect biomedical data every 2 waves (i.e., roughly every 4 years) in ELSA, when a professional nurse visits the respondents in their home and performs a face-to-face interview and a series of tests. The results of these nurse visits are recorded in separate files in the ELSA database. Currently, the datasets from the nurse-data questionnaires for waves 2 (2004), 4 (2008), 6 (2012) and 8 (2016) are published, so we used data from these files to create the features for our datasets. However, the ELSA data was not collected for machine learning purposes, and so we had to create features and class variables suitable for the classification task of machine learning, as described in the following subsections.

A total of 10 datasets were created with the raw data files from the ELSA nurse-data questionnaires, one for each of the 10 age-related diseases we are interested in predicting. The class variable in each dataset refers to the presence or absence of a positive diagnose for an age-related disease for each instance (ELSA respondent) in wave 8. Although the 10 datasets have different class variables (representing different age-related diseases), all 10 datasets have the same set of features, as explained in more detail later.

2.1 Preparing a Base Dataset

First of all, we discuss the creation of a base dataset – which has been used for creating the 10 nurse-data datasets used in our experiments – and the steps we took to convert the raw ELSA data into datasets suitable for machine learning. These steps included filtering the features and instances of the datasets, representing the different types of missing values (as discussed in Section 2.2) as a single missing value symbol ('?'), and creating the class variables using data from wave 8. This data preparation process is similar to the one applied by Pomsuwan and Freitas (2017), with a few differences in the feature selection process.

In order to have class values for all instances, we only utilised data from respondents that participated in the ELSA's 8th wave (the wave of the class variables). In cases where a respondent did not participate in any of the other waves with nurse data (waves 2, 4 or 6) in the dataset, the values for that wave's features were set as missing for that respondent. For instance, if a respondent was added to the study in wave 6, and participated in wave 8, we kept their record, filling in the values for waves 2 and 4 features with the missing value symbol "?".

After fusing all nurse-data datasets (waves 2, 4, 6 and 8), and removing the participants who did not take part in wave 8, each dataset has 7097 instances.

2.2 Feature Selection and Creation

After the previously described base dataset creation, the next step was to filter out features that were irrelevant to our classification task, from the initial set of 1041 features (from the 4 nurse-data datasets in ELSA's waves 2, 4, 6 and 8). Most of these features refer to metadata about the tests performed by the nurse when visiting the ELSA respondent in their household. For example, there are features recording reasons for unreliable or missing measurements, features recording the specific time of the nurse visit, and multiple measurements of the same variable in the same wave. The latter type of features (e.g., multiple recordings of blood pressure per wave) were merged into a single averaged measurement per wave, to reduce the dimensionality of the dataset. The large number of metadata features is also due to many variables with multiple answers, e.g.: there are four different features used to store reasons for not obtaining a height measurement of a patient, as up to four different reasons could apply simultaneously to a patient. Such metadata features clearly have no relevance for the classification task of machine learning, and so they were all removed.

After this feature selection and creation process, each created dataset has a unique identifier for each instance, as well as 140 features (including the gender and age of the participant in wave 8) and the target (class) variable. The 140 features are divided into 44 "conceptual features", where a conceptual feature may have several measurements of the same basic variable taken over the 4 waves in the dataset. The features in our datasets are briefly described in Appendix A. For each feature, we indicate the waves in the study it appears in, and the data type of its values.

The ELSA nurse-data database has multiple representations for the participant's responses that are not one of the expected values, including, e.g., a code for "not applicable" and another code for "refusal to answer". All such codes were unified and coded as "?", which is the standard missing value symbol for the Weka tool – the machine learning tool used in our experiments.

2.3 Creating Class Labels

For each dataset that we created from the ELSA nurse-data dataset, the binary class variable represents the presence or absence of a positive diagnose for each

ELSA respondent in wave 8, for one of 10 age-related diseases or conditions. This information is not represented directly by any of the variables in the ELSA files, so we combined information about the diagnosis of each of these diseases or conditions, present in several variables of the ELSA core data questionnaire, to create our class labels.

The class labels represent diagnoses for Angina, Arthritis, Cataract, Dementia, Diabetes, High blood pressure, Heart attack, Osteoporosis, Parkinson's Disease, and Stroke. In wave 8 of the ELSA study, each respondent was asked questions regarding the diagnosis of these diseases and conditions, and using the answers for these questions we infer a class label for that respondent, in that wave. All of these questions have binary answers (yes or no), and we label an instance as "0", meaning no diagnosis or "1", meaning the disease was diagnosed for that respondent, on wave 8, based on whether any of the questions regarding the diagnosis of that class was answered with a "yes" by the individual.

As an example, for the class Heart Attack, two questions are asked in the ELSA core questionnaire regarding its diagnosis, represented by two variables: Hedacmi (Whether the respondent confirms a heart attack diagnosis from a previous wave) and Hediami (Whether the respondent newly reported a heart attack diagnosis). Thus, the rule for creating the class label Heart Attack for each instance I in wave 8 is as follows:

IF Hedacmi, for instance I, in wave 8 = "Yes" (1)
OR Hediami, for instance I, in wave 8 = "Yes"(1)
THEN HeartAttack_8 for instance I = "Yes" (1)
OTHERWISE HeartAttack_8 for instance I = "No" (0)

Recall that all instances included in the dataset represent subjects who participated in the latest wave 8, which means that no instance in the created datasets has a missing class label. The nurse-data datasets were then created by distributing the 10 class variables across the 10 datasets, so that each dataset has a different class variable (age-related disease or condition) to be predicted. However, as mentioned earlier, all 10 datasets have the same instances and the same predictive features. This approach for dataset creation was also used by Pomsuwan and Freitas (2017).

3 Background and Related Works

3.1 Missing Value Imputation

One of the challenges of analysing longitudinal data is that studies that follow a set of individuals for a long period can encounter several issues with obtaining data across all waves. A participant may not be reached for one or more waves of a study for several reasons, or the data collection process might not be completely executed (for example, only part of an interview is done) in a given wave. Therefore, it is common that longitudinal studies face a high number of missing values in their datasets, and there are several strategies that can be used to address this issue, some of them taking advantage of the longitudinal nature of the data. For the ELSA datasets used in this article, 38.5% of the values across all features and waves are missing, which makes the approach to simply drop instances (or features) with missing values inadvisable. Hence, we follow the alternative approach of replacing every missing value with some estimated value.

There are many ways to estimate missing values (some particular to longitudinal datasets), and selecting the best imputation method is challenging. That is because, both in theory and in practice, no method for calculating imputation values is the optimal choice for all types of features and datasets (Diggle, 2002; Hu et al., 2017; Mallinckrodt, 2013). The relative performance of a method depends on several factors, such as: a) the data distribution (Santos et al., 2017); b) how the missing values occur in the dataset (missing completely at random, missing at random, or missing not at random) (Diggle, 2002; Mallinckrodt, 2013); c) the proportion of instances (records) with missing values; d) the availability of information that can be used to make better imputation.

3.2 The Chosen Missing Value Imputation Methods

Our experiments use five missing value imputation methods (Gad and Abdelkhalek, 2017; Mallinckrodt, 2013; Albridge et al., 1988), described in Subsections 3.2.1 to 3.2.5; as well as a proposed Data-Driven approach combining these five methods, to be described in Section 4. Most missing value replacement methods, including the proposed Data-Driven approach, were implemented by extending the program code from the Weka[1] toolkit, an open-source machine learning toolkit. The methods devised specifically for longitudinal data, Prev and PrevNext, were completely implemented by us, as currently the Weka toolkit does not handle longitudinal data directly.

In the following, $F_{i,t}$ denotes the value of feature F_i at wave t, and I denotes the instance where the missing value is being imputed. Furthermore, we specify how each method copes with training and testing datasets. This distinction is important in the aforementioned classifier-dependent scenario, although it is irrelevant for the classifier-independent scenario (see the Introduction for a discussion of these scenarios).

3.2.1 Global Mean/Mode

One standard statistical approach is to replace the missing values in feature $F_{i,t}$ by the mean or mode (for numeric or nominal features, respectively) of $F_{i,t}$ over all instances with known values for it in the training set. For this method, the estimated mean/modes are calculated from training instances and used to replace the missing values of $F_{i,t}$ in each instance I, in both the training and test sets.

This method has the advantage of simplicity, but it has important limitations. Unconditional mean/mode imputation frequently underestimates the variability represented in the real data, skewing the values towards a more even distribution, which can lead to false interpretations (Little and Rubin, 2019, Chapter 4). The more variability a feature's values have in reality, the more bias this method adds to the data.

3.2.2 Age-Based Mean/Mode

As an extension of the global mean/mode method, the age-based method uses the age feature to group instances in a way they are intuitively more likely to be

[1] Available at: https://www.cs.waikato.ac.nz/ml/weka/

similar. Naturally, the age of an individual impacts their overall health, so it is expected that, in general, ELSA participants with the same age would have more similar feature values than ELSA participants with different ages. As mentioned earlier, unconditional mean/mode imputation often misrepresents the variability of the feature's values, thus adding the age value of the respondent as a condition for guiding the imputation process is are likely to be a more effective approach, as long as the features' values are correlated with age.

The method works as follows. For each instance I with a missing value on a feature $F_{i,t}$, the method defines a set A of measurements of F_i, taken from instances with the same age value that I had on wave t, in any wave. Thus, the F_i values in A all correspond to measurements of the same feature with the missing values, from individuals who, at the time of that measurement, had the same age as the current instance I at the time t. Then, the missing value is replaced by the mean/mode (for numeric or nominal features, respectively) of the values in A. Note that this method assumes that the age value of an instance is always known, in every wave. This is the case in our datasets, where there are no missing values for the age variable.

For example, if a respondent was 60 years old on wave 4, and their corresponding instance had a missing value for feature $F_{i,4}$, this method would replace that missing value by the mean/mode of all values of F_i related to respondents who were 60 years old, at any time of measurement (at any wave), regardless of the wave where that measurement was obtained. For instances in the test set, as their age value is still known, the method is applied normally, using only values from training instances to create the set A. A similar approach has been used in Zhao et al. (2018), which replaced missing values with the median from individuals with the same age and sex.

3.2.3 Previous Observation Carried Forward (Prev)

In a longitudinal dataset, a feature typically has repeated measurements throughout different waves, and it is common to replace a missing value in a certain wave by its most recent known value from previous waves. This method is known as Last Observation Carried Forward, and is often used on studies using longitudinal datasets (Engels and Diehr, 2003; Zhu, 2014; Minhas et al., 2015; Gad and Abdelkhalek, 2017). We chose to include methods devised specifically for longitudinal data, such as this, in our study to investigate the impact of using temporal information in estimating missing data. However, as in our ELSA nurse datasets there is a gap of 4 years between each pair of adjacent waves, we decided to consider only values from the previous wave as viable for imputation.

Therefore, for the Prev (Previous Observation Carried Forward) method, if the value of feature $F_{i,t}$ is missing for instance I, the method inputs the value of $F_{i,t}$ for I in the previous wave, $F_{i,t-1}$, if known. If $F_{i,t-1}$ is unknown for I, the Prev method is not applicable. Because it uses information from the current instance with a missing value, it is unavoidable to use information from the feature values of test set instances when applying the Prev method to them. Note, however, that the class values of test set instances are never used in this method. Note also that, because this method requires a feature to have been measured in the previous wave of the dataset, it is inapplicable for the first measurement of a feature F_i, which includes all features in the first wave of the created datasets (wave 2 of the ELSA nurse data).

3.2.4 Previous and Next Observations Combined (PrevNext)

As an extension of the Prev method, we also included a method that combines information from both the previous measurement of a feature and its next measurement, increasing the amount of information used in the estimation of the missing values.

In the PrevNext (Previous and Next Observations Combined) method, when the value of feature $F_{i,t}$ is missing for instance I, if both the values of $F_{i,t+1}$ and $F_{i,t-1}$ are known for instance I, the missing value is replaced by: a) for numeric features, the mean of $F_{i,t+1}$ and $F_{i,t-1}$ for instance I; b) for nominal features, the method only replaces the missing value if both values of $F_{i,t+1}$ and $F_{i,t-1}$ are the same (in this case, repeat that value for $F_{i,t}$).

As with the Prev method, because of the 4-year time gap between waves in the nurse-data datasets, only values from the nearest waves are considered viable for imputation. This avoids imputations based on values too far into the future or the past, which are likely inaccurate. For test set instances, the PrevNext method works the same way, as it uses only information about features of the current instance I – without using any class information. This method requires known values for F_i for the current instance I, in both the previous and the next waves of the dataset (for nominal features, these values also need to be the same). Because of these restrictions, the PrevNext method is inapplicable in many cases, including all features in the first and last waves of the created datasets (waves 2 and 8 in the ELSA nurse data).

3.2.5 K-Nearest Neighbours (KNN)

This method uses the K-Nearest Neighbours (KNN) algorithm, which is a well-known supervised machine learning algorithm to estimate missing values in a more sophisticated way than previously described imputation methods. The KNN algorithm determines the K training instances most similar to the one with a missing value to be replaced (instance I), and calculates the mean/mode of $F_{i,t}$ in that set of nearest neighbours, using that mean/mode as an estimation of the missing value. K, the number of neighbours, is a user-defined parameter. Note that the previously described age-based method can be seen as a particular case of the KNN method where the similarity between instances is measured using only the age feature; whereas in the general KNN method any set of features can be used to define the similarity (or distance) measure between instances.

Importantly, any distance-based algorithm such as KNN can be affected by the so-called 'curse of dimensionality' (Kouiroukidis and Evangelidis, 2011), where instances appear to be more similar as the number of features (dimensions) used for the distance calculation increases, making the task of determining an instance's neighbours considerably harder. To avoid this issue, we made the KNN algorithm only consider as features (for distance calculations) the subject's age, gender, and the values of F_i in every wave other than t (the wave with the missing value to be replaced) where the F_i value is not missing. Even though the age and gender values are available for all instances in the dataset, if a feature has been measured in only one wave, or the value of F_i was missing in all of the waves other than t for the current instance, we considered this method could not be applied.

Our initial experiments with the KNN algorithm used only the values of F_i at waves other than t to calculate distances, but it was common to have several instances with the same distance to the current instance, especially for nominal features. This is an issue as the furthest neighbour within the set of K nearest neighbours could be randomly chosen out of several instances with the same distance to the current instance I. This would lead to an undesirable stochastic effect in the choice of K nearest neighbours. To reduce this issue, we added the age and gender features into all the distance calculations, which reduced the occurrence of this issue to under 1% of the instances, for $K = 7$.

The KNN algorithm is used as an imputation method as follows: replace the missing value of a feature $F_{i,t}$, by the prediction of the KNN algorithm with $K = 7$ (this method will be referred to as 7NN from here on). The prediction is given by the mean or the mode value of $F_{i,t}$ of instance I's K nearest neighbours, for numeric and nominal features respectively. For nominal features, if two or more values are tied with the highest frequency among the K nearest neighbours, one of those values is randomly chosen as the mode. We evaluated different K values (1,3,5,7,9) in preliminary experiments, and observed little difference in the average error values, with $K = 7$ producing the best results overall. Naturally, 7NN chooses the nearest neighbours exclusively from training instances, as it cannot have access to test set instances for that choice.

3.2.6 A Conceptual Comparison Between the Five Imputation Methods

When selecting which methods to compare in our experiments (reported later in Sections 6 and 7), we aimed to have representations of different types of methods for missing value imputation. We started with one of the most simplistic approaches, the Global mean/mode method, representing methods from basic statistics that are often used as a baseline method. However, the assumption that the mean/mode value over all known values can accurately replace every missing value is over optimistic, and it may mask characteristics of the data by adding noise (i.e., making the data seem more evenly distributed than it is in reality). Then, we chose to adapt this method to make it somewhat more sophisticated and related to our specific problem, adding to our experiments the Age-based mean/mode method, which we hoped would provide more accurate estimates for the ELSA dataset.

In addition, we also selected for our experiments two methods devised specifically for longitudinal data, the Prev and PrevNext methods. These methods use longitudinal information from known values of feature F_i at other time points (waves) in the current instance I to make their estimations, so each estimated value is arguably more related to the current instance, in comparison to the Global and Age-based mean/mode methods. One important disadvantage of the Prev and PrevNext methods is that they require a known value of F_i in the previous or both the previous and next waves to t (wave with the current missing value to be replaced), which may not be available.

Finally, there are approaches for estimating missing values that are more sophisticated and more computationally demanding. To represent those, we selected the 7NN method, which is a supervised machine-learning algorithm that outputs estimated values computed from instances considered most similar to the current instance with a missing value, which are intuitively likely to have a similar value for the current feature. The 7NN method requires $O(n^2)$ distance calculations to compute its distance matrix (where n is the number of instances in the dataset) when performing cross-validation, and has the added challenge of the curse of dimensionality, where using too many features to calculate the distance between neighbours hinders the effectiveness of the method (Beyer et al., 1999). Our implementation of 7NN greatly reduces the number of features used in the distance calculation, by only considering measurements of the current feature F_i at waves other than the current wave t, as well as the age and gender features, to calculate the distance. Thus, it can be considered a longitudinal missing value imputation method (akin to the Prev and PrevNext methods), as it uses time-related information to find the nearest neighbours.

3.3 Related Works

Several studies have compared multiple missing value imputation methods. However, these studies are usually more focused on specific aspects of handling missing data, usually working with only one type of feature (categorical, numeric or binary), having fewer features in the dataset, and using artificial datasets or datasets with artificially included missing data. There are two main approaches to evaluate the performance of missing value imputation methods. The first is to evaluate how missing value imputation changes the accuracy of a classification/regression model (Minhas et al., 2015; Hu et al., 2017). The results of such evaluations are dependent on the algorithm used to create the model. The second approach is to replace known values in a synthetic (Zhu, 2014; Gad and Abdelkhalek, 2017) or real (Engels and Diehr, 2003; Belger et al., 2016) dataset, and compare the estimated values to the ground-truth, calculating the error rate. We use both approaches: firstly, we use real data from the ELSA and estimate every known value in the dataset using the six imputation methods, then we employed the Random Forests (RF) classification algorithm to evaluate the models generated by datasets prepared with each method. To the best of our knowledge, our study is the first one that combines the classifier-dependent and classifier-independent evaluation of MVI methods, making for a more complete analysis of the performance of our proposed data-driven approach.

Table 1 contains characteristics that describe the cited related works, for comparison with our own study (described in the last row of this table). Regarding the proportion (%) of missing values in the datasets, as mentioned earlier, it is common for longitudinal datasets to have a high proportion of missing values, and that is observed in all studies that mentioned the ratio of missing values. One important characteristic that sets our approach apart from the related works is the number of features in our datasets. The cited studies performed experiments using datasets with very few (Belger et al., 2016; Gad and Abdelkhalek, 2017; Zhu, 2014) or between 16 and 48 features (Engels and Diehr, 2003). The ELSA dataset used in our experiments has 45 longitudinal features, with their repeated measures across time totalling 138 features with missing values, for each dataset.

Table 1 Number of waves (time points), features and percentage of missing values in the related works about comparing missing value imputation methods for longitudinal datasets. The names in the rows refer to the first authors in the references used in the comparison, respectively: (Engels and Diehr, 2003; Minhas et al., 2015; Belger et al., 2016; Gad and Abdelkhalek, 2017; Zhu, 2014; Hu et al., 2017).

Reference	**Waves**	**Number of Features**			**Missing Values**	**Type of Datasets**
		Numeric	**Nominal**	**Total**		
Engels	10	40	0	40	21.8%	Real
Minhas	6	16	0	16	30%	Real
Belger	4	1	0	1	10-40%	Artificial
Gad	6	1	1	2	45.6%	Artificial and Real
Zhu	5	2	0	2	4-22%	Artificial
Hu	2	?	?	48	?	Real
This study	4	99	39	138	38.5%	Real

The missing value imputation methods compared in each of the aforementioned studies are shown in Table 2[2], for comparison with our work. As shown in this table, the mean imputation and previous observation (usually LOCF) methods are the most common approaches for estimating missing values; and among the more sophisticated methods, the Linear Regression and KNN algorithms are often used.

As discussed in Section 3.2, our study contains methods representing different strategies for estimating missing values. These include statistics-based methods (Global mean/mode, Feature-based input using Age as the feature), methods devised for longitudinal data (Prev, PrevNext), and more sophisticated methods, based on machine learning (KNN) and our proposed Data-Driven approach, combining these 5 imputation methods. This selection of methods was made to include representative methods from very different approaches to missing value estimation in our experiments. In addition, among the studies mentioned in Table 2, this current study is the only one to include the proposed Data-Driven approach, which automatically selects the best imputation method for each feature specifically, among a set of candidate methods.

Among the cited related works, (Engels and Diehr, 2003) has the most similar approach for evaluating imputation methods. However, in (Engels and Diehr, 2003) the imputation methods were evaluated on just 4 longitudinal features, whereas in this work the methods are evaluated in 45 longitudinal features, representing a wider diversity of feature types and distributions. In addition, our work includes 39 nominal features, which are treated differently from the numeric features by our imputation methods. In their conclusion, Engels and Diehr mention that a method able to select the best-fitting imputation method for each feature in a dataset would likely provide better estimations. We have proposed such a method in our Data-Driven approach, discussed in Section 4.

To summarise, most of the related works discussed in this section have used relatively small number of features to evaluate Missing Value Imputation (MVI) methods, and most works evaluated MVI methods on numerical features only. By contrast, in this current work we use a larger number of features, including

[2] In Table 2, the studies were categorised by the types of methods employed to handle the missing values, so similar methods, such as our Prev (previous observation carried forward) and the LOCF, were considered part of the same category.

Table 2 Missing value imputation methods used in the related works. The names in the columns refer to the first authors in the references used in the comparison, respectively: (Engels and Diehr, 2003; Minhas et al., 2015; Belger et al., 2016; Gad and Abdelkhalek, 2017; Zhu, 2014; Hu et al., 2017).

Method/Reference	**Engels**	**Minhas**	**Belger**	**Gad**	**Zhu**	**Hu**	**This study**
Case deletion		X	X	X	X		
Random value	X			X			
Mean input	X	X	X	X	X	X	X
Class-based input	X						
Feature-based input							X
Previous observations	X	X		X	X		X
Posterior observations	X						
Previous and posterior observations	X						X
Multiple imputation				X	X	X	
Monte Carlo Markov Chains			X				
Expectation maximisation				X			
Linear Regression	X			X			
K-nearest neighbours		X		X			X
Data-driven method selection							X

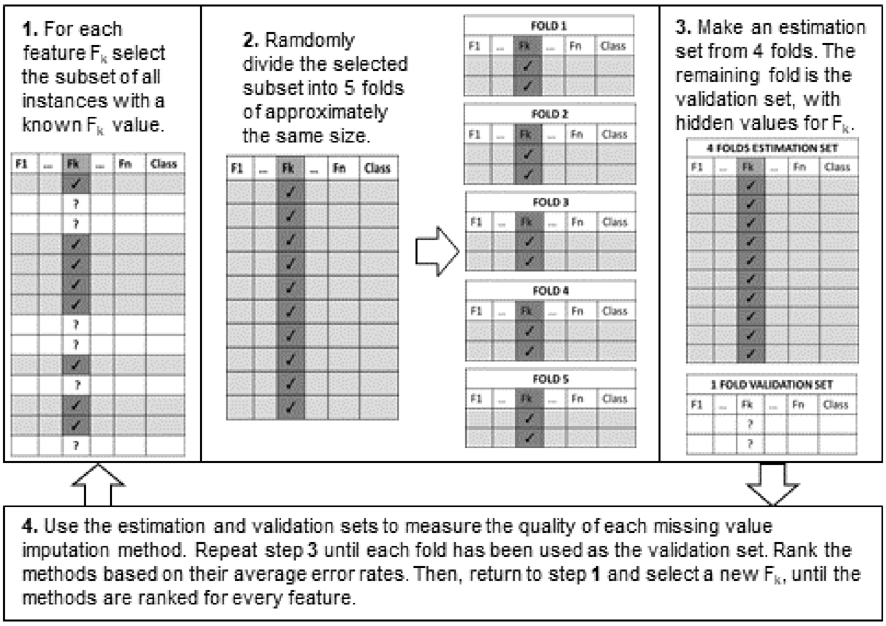
both numerical and nominal features. In addition, each study evaluated several MVI methods, but each method was separately applied to the data. That is, no related work has proposed to apply multiple MVI methods to each feature and automatically select the best method for each feature in a data-driven fashion, as proposed in this work.

4 The Proposed Data-Driven Missing Value Imputation Approach

In addition to the five selected missing value imputation methods discussed in Section 3.2, we propose an approach that selects these methods dynamically, on a feature-wise basis, ranking the methods based on information contained in the dataset itself. This approach, referred to as the Data-Driven approach from here on, can be implemented with any set of missing value imputation methods, in principle. This approach works as follows.

Consider a set of n missing value imputation methods $S = \{M_1, ..., M_n\}$, and a dataset with a set of d features $\{F_1, ..., F_d\}$. For each feature F_i at wave t ($F_{i,t}$) in a dataset, the Data-Driven approach creates a subset of the original dataset, composed of all the instances with known values for $F_{i,t}$ (removing instances where $F_{i,t}$'s value is missing). This subset is hereafter called the known data subset for $F_{i,t}$. Then, each method from S has its average estimation error rate measured in a 5-fold cross-validation performed in that known data subset. That is, the known data subset for the current feature $F_{i,t}$ is randomly partitioned into 5 folds of about the same size, and each imputation method is executed 5 times, each time using a different fold as a held-out "validation" subset, and the other four folds as the "estimation" subset. This process is summarised in Figure 1.

In the validation subset, the known values of $F_{i,t}$ are temporarily hidden from the imputation method being evaluated, and the method uses all instances in the estimation subset to determine the best value to be imputed for each instance



1. For each feature F_k select the subset of all instances with a known F_k value.

2. Ramdomly divide the selected subset into 5 folds of approximately the same size.

3. Make an estimation set from 4 folds. The remaining fold is the validation set, with hidden values for F_k.

4. Use the estimation and validation sets to measure the quality of each missing value imputation method. Repeat step 3 until each fold has been used as the validation set. Rank the methods based on their average error rates. Then, return to step 1 and select a new F_k, until the methods are ranked for every feature.

Fig. 1 Cross-validation approach to evaluate missing value imputation methods.

in the validation subset. The estimated values are then compared with the true, known values of $F_{i,t}$ in the validation set, and an error measure is computed. If $F_{i,t}$ is nominal, the error value, for each instance is 0 or 1, depending on whether or not the estimated value matched the known value. If $F_{i,t}$ is numeric, the error is the absolute value of the difference between the estimated and known values of $F_{i,t}$. The estimation error associated with each imputation method is the average of its errors over the instances for which the method could be applied. If a method cannot be applied to any given instance in the validation set, we assigned the maximum average error value of 1 for that instance. Note that information in the validation sets is not used to calculate estimated values, except for the Prev, PrevNext and 7NN methods, which use information about the feature values in the current instance I, but not its known $F_{i,t}$ value, of course.

The imputation methods are then ranked based on their average error, where the smallest-error method is assigned rank 1 and the largest-error method is assigned the worst rank (5). If two methods have the same average error, they share a rank (e.g., if two methods tie for the first place, both get a rank of 1.5).

For each feature $F_{i,t}$, the Data-Driven approach performs the imputation of missing values (in the data subset where the $F_{i,t}$ value is really unknown) using the imputation methods' rankings obtained for $F_{i,t}$. First, it tries to use the first-ranked method to estimate the missing value. If that method cannot be applied to the current instance, it then tries the second-ranked method, and so on, until it either finds a method that can estimate a value for the current instance or runs out of imputation methods to try. In our experiments, the latter case is not an issue as the Global mean/mode method can be applied to any instance. However, with a different set of imputation methods, the Data-Driven approach may fail to replace a missing value, if none of the methods can be used to replace it.

In summary, the Data-Driven approach uses the data to calculate an approximation of how accurate each of the available methods will be to estimate the missing values of a feature, then ranks these methods and applies the best-suited method (for that feature) to make its estimation. Naturally, this process is costly, especially if one or more of the imputation methods is computationally expensive (such as the 7NN algorithm in our experiments). However, intuitively the Data-Driven approach's estimations are more flexible and sophisticated, and make use of the different advantages provided by each method. As there is no "one-size-fits-all" approach when imputing missing data (i.e., no imputation method is the best for all features), a method that is able to make feature-wise decisions is intuitively more effective. However, this effectiveness depends on the reliability of its computed ranking of missing value imputation methods, and for features with few known values, the Data-Driven approach might be misled into selecting a poor imputation method.

5 Methodology for Evaluating the Proposed Data-Driven Imputation Method

There are two main approaches to evaluate the performance of missing value imputation methods, in the area of supervised machine learning (classification or regression tasks). The first is to use an imputation method to estimate the missing values in a data preprocessing phase, and then evaluate how that imputation changes the performance of the classification/regression model trained with the imputed values. The results of such evaluations, referred to as classifier-dependent evaluations from now on, are dependent on the algorithm used to create the model, but provide a direct measure of the impact of a missing value handling method on the predictive accuracy of a particular classification model.

The second approach, classifier-independent evaluation, is to use an imputation method to replace known values in a synthetic or real dataset, and comparing the estimated values to the ground-truth, calculating estimation quality metrics such as the error rate. This type of evaluation provides a comparison that is unrelated to how the chosen machine learning algorithm handles missing values, but it has the advantage of providing a more generic measure of how accurate the imputation methods are at estimating 'artificial' missing values (as it is not possible to compare estimations of real missing values to a ground-truth).

In our study, we use both approaches: firstly, we use data from our ELSA dataset and estimate every known value in the dataset using six imputation methods, and rank them for each feature in the dataset, based on their average estimation error (classifier-independent evaluation). Then, to implement the classifier-dependent scenario, we employed the Random Forest (RF) classification algorithm (Breiman, 2001) to evaluate models generated by datasets prepared with each imputation method. We also compared the results of this approach against the baseline approach of performing no imputation in a preprocessing step (letting the RF algorithm use its own internal method for handling missing values).

The RF algorithm was chosen for these experiments because it has been shown to be among the best classification algorithms in general in terms of predictive performance (Fernández-Delgado et al., 2014; Zhang et al., 2017), and it is also relatively fast – being much faster in general than other algorithms with high predictive performance, like support vector machines and neural networks.

6 Experiments Comparing MVI Methods on a Classifier-Independent Scenario

We performed a series of experiments to evaluate the estimation accuracy of the six Missing Value Imputation (MVI) methods (the five methods described in Section 3.2 and the proposed Data-Driven missing value imputation approach described in Section 4) in the classifier-independent scenario. The setup used in these experiments can be replicated for comparing any number of missing value imputation methods, even outside of the area of classification. The experiments presented in this Section are an expansion of results discussed in our previous work (Ribeiro and Freitas, 2019), with the addition of the Data-Driven approach we propose in this study as one of the methods being compared.

For each imputation method we compute: *(a)* its applicability, i.e., for which proportion of the missing values in the dataset the method can be applied; and *(b)* its normalised average error rate, for nominal and numeric features separately, and over all features. The error values reported in Table 3 were computed by running a nested cross-validation procedure, with an external 10-fold cross-validation and an internal 5-fold cross-validation procedure, as follows.

For the external cross-validation, the instances with known values of each feature are divided into 10 data subsets (folds), and each fold is used as a validation fold in turn, by hiding the values of the feature in that fold, whilst the other 9 folds are combined into estimation sets, which are used to estimate these hidden values using each of the MVI methods. The estimated values are then compared against the known (previously hidden) values values in the validation sets, in order to compute the estimation error. This external 10-fold cross-validation is enough to measure the error of the individual MVI methods, but a further internal 5-fold cross-validation procedure (applied to each of the 10 estimation sets) is needed to implement and measure the error of the data-driven approach.

For the internal cross-validation, the instances in the current estimation set are divided into 5 folds, and each fold is used as an inner-validation fold in turn, by hiding the values of the feature in that fold, whilst the other 4 folds are used to estimate these hidden values using each of the 5 MVI methods. The estimated values are then compared against the known (previously hidden) values values in the inner-validation sets, in order to compute estimation errors used to rank the MVI methods, as required by the data-driven approach. Note that the ranking of MVI methods is produced from the estimation set only, without accessing the (external) validation set. This was done to avoid providing the data-driven approach with more information than what is used by the other MVI methods. The ranking is then used to select the best applicable MVI method for the current feature, when imputing values in the validation set; and again the estimated values are compared against the known (previously hidden) values values in the validation sets, in order to compute the estimation error. The final error for the data-driven approach is computed by averaging over the errors on the 10 validation sets.

Regarding the applicability of each method, the 7NN method could not be applied to features that did not have repeated measurements in other waves (2 out of the 138 features with missing values), or to instances where all of the other measurements for the current feature (whose missing value is being replaced) had missing values. The Global mean/mode method can be applied to every missing value in the dataset. The Age-based method was not applicable in relatively rare cases where there were no known values of the current feature for any subjects with the same age of the current instance's subject.

The PrevNext and Prev methods, however, could not be applied in many cases, since the Prev method requires the current feature to have a known value in the previous wave, and the PrevNext method requires the current feature to have a known value in both the previous and the next waves in the dataset, which is even less common. By definition, Prev is inapplicable for features in the first wave, and PrevNext is inapplicable for both the first and last wave features (note that, in the datasets used in our experiments, there are only four waves). In addition, in many other cases these two methods are potentially applicable for a feature, but cannot be applied in practice because the current instance does not have the required known values.

The applicability percentage (over all missing values in the dataset, how many could be replaced using the method) of each method is shown in the last but one row of Table 3. This Table also presents the mean error rate over the nominal features, the mean absolute error over the numeric features and over all features, for each imputation method. The mean error of a method is calculated considering only the instances where it could be applied, so for features where only some of the missing values could be replaced, the average error was calculated only over those values.

Note that every numerical feature in the dataset has had its values normalised before the missing value imputation methods were applied. The normalisation method used was min-max, where each feature value is divided by the difference between the minimal and maximum values observed for that feature in the entire dataset, producing values in the [0..1] range. Therefore, the average error values were also in this range, as nominal features had 0 or 1 error values (for a match and non-match, respectively), and numerical features had the difference between the estimated and real value as the error.

Table 3 Error rates (in [0..1]) of the imputation methods, computed by 5-fold cross-validation, considering only instances where the methods were applicable. For nominal features each value represents the mean error rate (over 39 features) and for numeric features each value is the mean absolute error (over 99 features). The last two rows show the applicability (%) and run time (minutes) of each method. The best result for each row is shown in boldface font.

Feature type (number)	**7NN**	**Age-based**	**Global mean**	**Prev**	**PrevNext**	**Data-Driven**
Nominal (39)	0.049	0.078	0.068	0.055	**0.048**	**0.048**
Numeric (99)	0.083	0.083	0.082	0.078	**0.075**	0.077
All features (138)	0.077	0.082	0.078	0.07	**0.068**	**0.068**
Applicability	81.79%	97.08%	**100.00%**	35.57%	2.95%	**100.00%**
Run Time (mins)	9.77	0.3	0.1	**0.02**	**0.02**	57.5

As shown in Table 3, when considering only numeric features, the PrevNext method has the smallest mean average error, very closely followed by the Data-Driven and Prev methods. When considering either only numeric features or all features together, the PrevNext and Data-Driven method are tied for the smallest mean error. However, these values need to be interpreted together with the applicability of each method.

The Prev method had low error values, but low applicability, meaning it was only able to estimate feature values for 35.57% of the missing values in the dataset. This was even worse for the PrevNext method, which was able to estimate only 2.95% of the missing values. As mentioned earlier, this is due to the fact that these methods require a known value of the current feature in the previous (Prev) or in both the previous and next (PrevNext) measurements (waves) of that feature, and those values may not exist (if there is no previous or next wave) or also be missing for some instances in the dataset.

It is worthwhile to mention that, although it did not obtain the best results, the 7NN method also performed well. The 7NN method has longitudinal characteristics in our specification, as it calculates the distance between instances using measurements of the current feature in different waves. This shows that the more complex strategy adopted by 7NN paid off in its results, when compared to simpler approaches such as the Age-based mean/mode and Global mean/mode methods, and also when compared to the Prev and PrevNext methods, which achieved smaller mean error rates but had greatly reduced applicabilities (35.6% and 2.95%, respectively). Note that, by contrast, 7NN achieved an applicability of 81.78%.

The Data-Driven approach has an applicability of 100%, because it ranks every method and, if the first-ranked method is unable to estimate the current missing value, it tries the next method in the ranking, and so on, until an applicable method is found or all methods have been tried. Hence, the fact that the Global mean/mode method has applicability of 100% guarantees that the Data-Driven approach also has an applicability of 100%. That, allied to the Data-Driven approach's low error values, clearly sets it as the best approach in the classifier-independent comparison. In summary, the Data-Driven approach makes use of the advantages presented by different methods, and is able to reliably choose, in a feature-wise manner, which out of a set of missing value imputation methods is the most effective.

Finally, we measured the time required to run the classifier-independent experiment with each of the methods. The run times reported in the last row of Table 3 are the time taken by the 10-fold cross-validation to be run for the entire dataset, for each method. These run times were measured on a computer with an AMD Ryzen 5 3600x 6-Code Processor, with 3.80GHz and 8GB of RAM memory, running Java 8 on Windows 10. As can be seen in Table 3, as expected, 7NN is more time consuming than the other individual missing value replacement methods, with the former taking about 10 min, whilst each of the other individual methods took less than a minute. The time taken by the data-driven approach was as expected much larger, about 57 min. This can be explained by the fact that, in order to select the best individual missing value replacement method, in each fold of the external cross-validation procedure (i.e. for each pair of estimation and validation sets), the data-driven approach performs an internal 5-five cross-validation procedure on the estimation set where each individual method is run 5 times. So, the total run time of the data-driven approach is intuitively approximately 5 times larger than the sum of the run times taken by the individual methods, which is roughly dominated by the 10 min taken by KNN (since all the other individual methods together take just about 0.5 min).

7 Experiments Comparing MVI Methods on a Classifier-Dependent Scenario

In this Section, we evaluate the effect of using each of the missing value handling methods, discussed in Sections 3.2 and 4, on the predictive accuracy of Random Forest (RF) classifiers, using 10-fold cross-validation.

For all experiments in this Section, each Missing Value Imputation (MVI) method was used in a data preprocessing phase, before training the classifier, using only training set instances to compute replacement values for every missing value in the training and test datasets. The Prev, PrevNext and 7NN methods are exceptions, in the sense that they use feature values (but not class labels) of the current instance in the test set, as mentioned earlier. In addition to the MVI methods compared in the classifier-independent scenario (Section 6), for the experiments in this current Section with the Random Forest (RF) classifier we added a baseline approach of not using any of the MVI methods. Thus, the baseline consists of not changing the missing values in a preprocessing step, and instead let the RF algorithm handle them during its execution.

We used the RF implementation in Weka, which uses the C4.5 algorithm's (Quinlan, 1993) technique to cope with missing values when building its decision trees, as follows. Initially, each instance is assigned an instance weight of 1. When an instance has a missing value for a feature which is a candidate to be selected for the current tree node, for the purpose of computing that feature's information gain (or other feature evaluation measure, depending on the RF implementation), the weight of that instance is distributed across the child nodes, based on the distribution of the known values of that feature in the local training set associated with the current node. To clarify, suppose that a binary feature $f_{j,t}$ has 70% of its known local samples valued as 0 and the remaining 30% valued as 1. The 0 and 1 child nodes of $f_{j,t}$ would receive, for each instance with a missing value of that feature, a fractional instance with weights 0.7 and 0.3, respectively. The same fractional distribution of the instance is performed during the testing phase, when the built tree is used to classify previously unseen test instances.

As mentioned earlier, the other MVI methods compared in this Section are each of the five methods described in Section 3.2 and our Data-Driven approach (Section 4), where the methods are ranked for each feature based on their mean errors, calculated using an internal cross-validation on the training set (i.e., without using the test set). We emphasise that the Data-Driven approach, in this scenario, ranks the methods for the current feature based on an internal cross-validation, iteratively dividing the training set instances into its estimation and validation sets, to avoid using test set instances in its decision-making process.

7.1 Class Imbalance Handling

Our created datasets exhibit the problem of class imbalance, where for each class variable (disease), the number of instances of the positive class (meaning the subject was healthy in the sense of not having a disease) is substantially greater than the number of instances of the negative class (meaning the subject had a disease). Because of this class imbalance problem, before comparing the MVI methods we needed to decide on a strategy to reduce the bias towards the majority class in our datasets. Thus, we performed experiments with two random undersampling methods that bring the ratio of positive to negative instances in the training set down to a 1:1 ratio – i.e., for each instance of the minority class in the training set, only one instance of the majority class is kept. This 1:1 ratio is a default approach adopted by several studies (López et al., 2013; Weiss and Provost, 2003), including a study that used similar datasets to the ones used in our experiments (Pomsuwan and Freitas, 2017).

There are two intuitive ways of applying undersampling to Random Forest classifiers: (a) removing instances from the majority class in a preprocessing step, then performing the bootstrapping for every tree in the forest with the same pool of training instances, or (b) undersampling the majority class when creating each bootstrap sample of instances to be used to learn each tree of the RF, so that undersampling is performed within the RF algorithm.

The first method, which we are calling Undersampling Before Bootstrapping (UBB), is simpler to implement, since it does not require any modification of the standard RF algorithm. When applying the UBB method, the decision trees in the RF are built from bootstrap samples of a training set with balanced class proportions, and the majority class instances that were discarded in the undersampling process are never seen by the RF.

The second method of applying undersampling to RFs is the Balanced Random Forest (BRF) algorithm (Chen et al., 2004). The BRF receives the entire imbalanced training set as input. Then, for each tree in the forest, it draws a bootstrap sample of minority class instances, and randomly draws the same number of instances from the majority class instances, meaning the subset of instances used to generate the tree has the desired ratio (1:1) of instances in each class. The rest of the RF algorithm remains unchanged. In this method, all training instances of the majority class have a chance of being used in the creation of the model, increasing the variability of training instances, a desirable characteristic for the RF algorithm.

Because of this increased variability, intuitively the BRF method would generate classifiers that are less overfitted to a part of the training set than classifiers generated with the UFF. In order to confirm that intuition, we report the results of experiments comparing the UBB and BRF methods in Section 7.2. Next, we move on to comparing the MVI methods using the BFR method (which performed better) in Section 7.3.

In all result Tables reported in this Section, the datasets are ordered based on their class Imbalance Ratio (IR), calculated by dividing the number of instances in the majority class by the number of instances in the minority class. Classifiers trained from datasets with higher IR values usually have decreased performance, due to an added bias for classifying instances in the majority class (to artificially increase the overall accuracy).

The RFs were trained and tested using the Weka toolkit, with the default parameters $ntrees = 100$ (number of decision trees) and $mtry = \lfloor log_2(d) \rfloor + 1 = 8$ (number of features randomly sampled to be used as candidate features at each tree node), where the total number of features is $d = 140$, and $\lfloor x \rfloor$ is the "floor" of x, i.e., the biggest integer which is smaller than or equal to x.

The RF classifiers were evaluated based on the following metrics: Sensitivity (True Positive Rate), Specificity (True Negative Rate), and Accuracy (percentage of correct classifications). These metrics were chosen based on (Malley et al., 2011, Chapter 4), who claim that for imbalanced biomedical data, models should have their results analysed using metrics that consider their ability to predict each class separately (i.e., Sensitivity and Specificity) and at least one "global" measure of performance considering both classes – in our case, we chose Accuracy, which is the complement of the Error measure suggested by the authors. We chose to use Accuracy rather than Error so that all 3 metrics are to be maximised, for consistency in the analysis of the results.

7.2 Comparing the UBB and BRF methods for class balancing

In this Section we compare the predictive performance of the UBB and BRF undersampling methods for Random Forests (RF) in our ELSA nurse-data datasets. For each of the 10 classes (age-related diseases), with different Imbalance Ratios (IR), we created RF classifiers using the UBB and BRF undersampling methods.

The IR measure, as mentioned earlier, is calculated by dividing the number of majority class instances (individuals not diagnosed with a disease) by the number of minority class instances (individuals diagnosed with a disease). The IR value is an indication of how imbalanced the class distribution of a dataset is, and our 10 datasets have very different levels of class imbalance, with IR values ranging from 1.35 (Arthritis) to 160.3 (Parkinson's Disease), depending on how rare the age-related disease is.

Tables 4 and 5 show the average Sensitivity (True Positive rate) and Specificity (True Negative rate) of the RF models, over a 10-fold cross-validation. In the last row of the Tables, we report how many times each class-balancing method (UBB and BRF) got a higher value (i.e., was the winner) across the 10 datasets – equal values, with 3 decimal places being considered, mean that each method got 0.5 'win' points.

Table 4 Average Sensitivity values for RF with the UBB and BRF undersampling methods for each dataset/imputation method combination, over a 10-fold cross-validation. The last row contains the number of wins of each method, and the best value in each row is in boldface.

Dataset (IR)	**Baseline**		**Globalmean**		**Agebased**		**Prev**		**PrevNext**		**7NN**		**Data-Driven**	
	UBB	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**
Arthritis (1.35)	0.695	**0.703**	0.664	**0.659**	**0.678**	**0.678**	0.665	**0.674**	0.644	**0.658**	**0.681**	0.674	**0.679**	**0.679**
High BP (1.49)	0.652	**0.653**	0.692	**0.697**	**0.705**	0.694	**0.647**	**0.647**	**0.644**	**0.644**	**0.700**	0.694	0.697	**0.699**
Cataract (2.06)	0.660	**0.662**	0.673	**0.671**	0.669	**0.675**	0.659	**0.677**	0.630	**0.641**	**0.676**	0.668	**0.672**	0.671
Diabetes (6.5)	0.654	**0.661**	**0.765**	0.763	0.752	**0.762**	0.680	**0.687**	0.653	**0.666**	**0.782**	**0.782**	**0.781**	0.773
Osteoporosis (9.85)	0.632	**0.644**	0.688	**0.702**	**0.700**	0.693	0.638	**0.648**	0.630	**0.648**	0.685	**0.706**	0.689	**0.695**
Stroke (15.86)	**0.616**	0.603	0.699	**0.705**	**0.705**	0.704	**0.623**	0.616	**0.574**	0.570	0.697	**0.713**	0.691	**0.703**
Heart Attack (16.7)	0.627	**0.646**	0.699	**0.717**	0.696	**0.700**	0.615	**0.625**	0.624	**0.630**	0.702	**0.718**	0.719	**0.727**
Angina (26.51)	0.611	**0.627**	**0.698**	0.695	0.694	**0.698**	**0.633**	0.610	**0.611**	0.610	0.689	**0.698**	0.702	**0.710**
Dementia (46.96)	0.703	**0.706**	**0.745**	**0.745**	**0.755**	0.752	0.675	**0.679**	0.671	**0.682**	0.764	**0.768**	**0.753**	0.748
Parkinson's (160.3)	0.567	**0.601**	0.656	**0.670**	0.664	**0.681**	0.541	**0.581**	0.537	**0.562**	0.660	**0.677**	0.633	**0.649**
N of "Wins"	1	**9**	2.5	**7.5**	4.5	**5.5**	2.5	**7.5**	2.5	**7.5**	3.5	**6.5**	3.5	**6.5**

Table 4 shows a noticeable trend for higher Sensitivity values when using the BRF method. For every MVI method, BRF outperformed UBB for the majority of the datasets, with the greatest difference being observed for the Baseline approach, where BRF won 9 out of 10 times.

On the other hand, as seen in Table 5, overall higher Specificity values were observed when using the UBB method, although the difference was not as clear. The number of wins for UBB was greater than the number of wins for BRF for 4 out of the 7 MVI methods, with the greatest difference being observed in the

Table 5 Average Specificity values for RF with the UBB and BRF undersampling methods for each dataset/imputation method combination, over a 10-fold cross-validation. The last row contains the number of wins of each method, and the best value in each row is in boldface.

Dataset (IR)	**Baseline**		**Globalmean**		**Agebased**		**Prev**		**PrevNext**		**7NN**		**Data-Driven**	
	UBB	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**
Arthritis (1.35)	**0.548**	0.533	0.511	**0.524**	0.502	**0.507**	**0.546**	**0.546**	0.556	**0.561**	0.502	**0.506**	0.506	**0.518**
High BP (1.49)	0.701	**0.702**	**0.629**	0.618	0.601	**0.604**	**0.666**	0.655	0.680	**0.691**	0.600	**0.605**	0.610	**0.616**
Cataract (2.06)	**0.674**	**0.674**	**0.601**	0.593	**0.595**	0.587	**0.636**	0.621	**0.693**	0.676	**0.595**	0.589	**0.599**	0.580
Diabetes (6.5)	**0.795**	0.793	0.700	**0.710**	**0.657**	0.654	**0.803**	0.802	**0.825**	0.812	0.660	**0.707**	**0.686**	0.680
Osteoporosis (9.85)	**0.715**	0.710	**0.603**	0.583	**0.598**	0.591	**0.670**	0.662	**0.702**	0.674	**0.623**	0.589	0.576	**0.597**
Stroke (15.86)	**0.725**	**0.725**	0.597	**0.599**	**0.573**	0.516	0.695	**0.725**	**0.735**	0.733	0.580	**0.602**	0.560	**0.570**
Heart Attack (16.7)	**0.718**	0.699	0.562	**0.572**	0.581	**0.612**	**0.740**	0.723	0.705	**0.716**	**0.588**	0.574	**0.605**	0.592
Angina (26.51)	**0.741**	0.691	**0.590**	0.541	0.501	**0.564**	0.667	**0.713**	**0.730**	0.706	**0.573**	0.542	0.576	**0.622**
Dementia (46.96)	**0.790**	0.761	**0.670**	0.642	**0.679**	0.655	**0.787**	0.771	0.742	**0.746**	0.638	**0.665**	0.610	**0.667**
Parkinson's (160.3)	0.625	**0.687**	**0.602**	0.449	**0.584**	0.531	**0.690**	0.624	0.633	**0.687**	**0.586**	0.411	**0.522**	0.512
N of "Wins"	**7**	3	**6**	4	**6**	4	**7.5**	2.5	**5**	**5**	**5**	**5**	4	6

Prev method, where the UBB had a higher Specificity value in 7.5 out of the 10 datasets.

The opposing results between Sensitivity and Specificity measures are expected, since these performance metrics evaluate the classifier's abilities to predict different classes, and usually the prediction of one class can be improved, but in detriment of the other class. It is important to note, however, that the BRF method still managed to get better Specificity values in some cases, achieving the same number of wins than UBB for the PrevNext and 7NN methods, and even winning 6 out of 10 times for the Data-Driven approach.

As a global measure of the RF models' performances, their average Accuracy values are reported in Table 6.

Table 6 Average Accuracy values for the UBB and BRF undersampling methods for each dataset/imputation method combination, over a 10-fold cross-validation. The last row contains the number of wins of each method, and the best value in each row is in boldface.

Dataset (IR)	**Baseline**		**Globalmean**		**Agebased**		**Prev**		**PrevNext**		**7-NN**		**Data-Driven**	
	UBB	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**	**UBB**	**BRF**
Arthritis (1.35)	0.672	**0.673**	**0.667**	0.665	**0.664**	0.658	**0.655**	0.650	0.658	**0.663**	**0.660**	0.658	0.662	**0.666**
High Blood Pressure (1.49)	**0.632**	0.630	0.599	**0.602**	0.603	**0.605**	0.614	**0.620**	0.607	**0.616**	**0.605**	0.602	0.606	**0.611**
Cataract (2.06)	0.665	**0.666**	**0.649**	0.646	0.645	**0.646**	0.652	**0.659**	0.651	**0.652**	**0.650**	0.642	**0.648**	0.641
Diabetes (6.5)	0.673	**0.679**	**0.756**	**0.756**	0.739	**0.747**	0.696	**0.703**	0.676	**0.686**	0.766	**0.772**	**0.768**	0.761
Osteoporosis (9.85)	0.639	**0.650**	0.680	**0.691**	**0.691**	0.683	0.641	**0.649**	0.636	**0.650**	0.679	**0.695**	0.679	**0.686**
Stroke (15.86)	0.615	**0.629**	**0.694**	0.689	0.687	**0.692**	**0.634**	0.614	**0.615**	0.614	0.684	**0.692**	0.697	**0.707**
Heart Attack (16.7)	**0.623**	0.610	0.693	**0.698**	**0.697**	0.693	**0.627**	0.622	**0.584**	0.580	0.690	**0.706**	0.683	**0.695**
Angina (26.51)	0.632	**0.648**	0.692	**0.709**	0.689	**0.694**	0.622	**0.630**	0.629	**0.635**	0.695	**0.710**	0.712	**0.720**
Dementia (56.96)	0.704	**0.707**	**0.743**	**0.743**	**0.754**	0.750	0.677	**0.681**	0.672	**0.683**	0.761	**0.766**	**0.750**	0.746
Parkinsons (160.3)	0.568	**0.601**	0.655	**0.668**	0.663	**0.679**	0.542	**0.581**	0.537	**0.563**	0.658	**0.674**	0.632	**0.647**
Number of "Wins"	2	**8**	4	**6**	4	**6**	3	**7**	2	**8**	3	**7**	3	**7**

In the Accuracy analysis, the BRF method again performs better than the UBB method, getting the highest number of wins (between 6 and 8 out of 10 datasets) for all missing value imputation methods.

Therefore, when analysing the performance of the RF models, both on a class-based analysis (measuring Sensitivity and Specificity) and considering the global measure of Accuracy, the BRF method was overall superior to the UBB method. This trend can be explained by the notion that, in the UBB method, the model overfits more on the positive instances (the majority class, of healthy individuals), as it is trained with a dataset having less variability of positive instances. Conversely, when applying the BRF method, the model is trained with a wider variety of positive class instances due to the undersampling happening inside each bootstrapping process (for each tree in the RF), so different decision trees in the RF are likely to learn to detect different aspects of the majority class.

In conclusion, overall the BRF method performed better than the UBB method in our experiments, and is intuitively better due to using more varied training instances of the majority class, so from here on all our experiments will be performed with datasets where the majority-class instances are undersampled using the BRF method.

7.3 Comparing the Missing Value Imputation Methods

Once we have chosen the BRF undersampling as the method for handling the class imbalance in our datasets, based on the results reported in the previous Section, in this Section we analyse which of the Missing Value Imputation (MVI) methods is the most adequate for our datasets. The Sensitivity and Specificity results obtained by each MVI method (with BRF undersampling) are presented in Tables 7 and 8, respectively. For this analysis, we ranked all 7 methods from the best (rank 1) to the worst (rank 7) based on each of the measures, using three decimal places, and having tied methods share the same average rank – e.g., if two methods are joint first, each is assigned a rank of 1.5.

Table 7 Average Sensitivity (True Positive Rate) values for the Random Forest algorithm, over a 10-fold cross-validation, using different missing value handling methods. The last row contains the average rank of each method, and the best value in each row is in boldface.

Dataset (IR)	**Baseline**	**Globalmean**	**Agebased**	**Prev**	**PrevNext**	**7-NN**	**Data-Driven**
Arthritis (1.35)	**0.702**	0.66	0.678	0.674	0.658	0.674	0.679
High Blood Pressure (1.49)	0.653	0.697	0.695	0.647	0.644	0.694	**0.699**
Cataract (2.06)	0.662	0.671	0.675	**0.677**	0.64	0.667	0.67
Diabetes (6.5)	0.661	0.763	0.762	0.687	0.666	**0.781**	0.773
Osteoporosis (9.85)	0.644	0.702	0.693	0.648	0.648	**0.706**	0.695
Stroke (15.86)	0.603	0.705	0.704	0.616	0.57	**0.713**	0.703
Heart Attack (16.7)	0.645	0.717	0.7	0.625	0.63	0.718	**0.727**
Angina (26.51)	0.627	0.695	0.698	0.61	0.61	0.698	**0.71**
Dementia (56.96)	0.706	0.745	0.752	0.679	0.682	**0.768**	0.748
Parkinson's (160.3)	0.601	0.67	**0.681**	0.581	0.562	0.677	0.648
Average Rank	5.2	3.2	2.85	5.35	6.5	**2.4**	2.5

Table 8 Average Specificity (True Negative Rate) values for the Random Forest algorithm, over a 10-fold cross-validation, using different missing value handling methods. The last row contains the average rank of each method, and the best value in each row is in boldface.

Dataset (IR)	**Baseline**	**Globalmean**	**Agebased**	**Prev**	**PrevNext**	**7-NN**	**Data-Driven**
Arthritis (1.35)	0.533	0.524	0.507	0.546	**0.56**	0.505	0.518
High Blood Pressure (1.49)	**0.702**	0.618	0.604	0.655	0.691	0.605	0.616
Cataract (2.06)	0.676	0.593	0.587	0.623	**0.677**	0.589	0.58
Diabetes (6.5)	0.794	0.71	0.655	0.804	**0.813**	0.708	0.684
Osteoporosis (9.85)	**0.709**	0.581	0.59	0.662	0.673	0.589	0.598
Stroke (15.86)	0.724	0.599	0.52	0.722	**0.734**	0.601	0.568
Heart Attack (16.7)	0.698	0.571	0.608	**0.721**	0.716	0.574	0.591
Angina (26.51)	0.69	0.539	0.55	**0.721**	0.709	0.547	0.62
Dementia (56.96)	0.764	0.635	0.642	**0.764**	0.73	0.669	0.655
Parkinson's (160.3)	**0.652**	0.409	0.5	0.606	0.636	0.394	0.485
Average Rank	2.05	5.5	5.7	2.25	**1.7**	5.6	5.2

Regarding the Sensitivity values (Table 7), the 7NN method obtained the lowest average rank (2.4) and 4 best values across the 10 datasets, closely followed by the Data-Driven method with the second lowest average rank (2.5) and 3 best values.

As expected, the trade-off for this is observed in the Specificity (Table 8) values, which show lower values for the 7NN and Data-Driven methods. The Prev, PrevNext and Baseline methods have the best Specificity results, with PrevNext achieving the lowest average rank (1.7) and the highest number of best Specificity values (4 out of 10). It is worth noting, however, that the success of the Prev and PrevNext methods is largely due to the success of the baseline approach for handling missing values embedded in the RF algorithm. This is because Prev and PrevNext have a low rate of applicability, as discussed in Section 6, and for the missing values where these methods cannot be applied, the RF's baseline approach is used.

For the analysis of global predictive performance, we present the models' Accuracy results in Table 9.

Table 9 Average Accuracy values for the Random Forest algorithm, over a 10-fold cross-validation, using different missing value handling methods. The last row contains the average rank of each method, and the best value in each row is in boldface.

Dataset (IR)	**Baseline**	**Globalmean**	**Agebased**	**Prev**	**PrevNext**	**7-NN**	**Data-Driven**
Arthritis (1.35)	**0.63**	0.602	0.605	0.62	0.616	0.602	0.611
High Blood Pressure (1.49)	**0.673**	0.665	0.658	0.65	0.663	0.658	0.666
Cataract (2.06)	**0.666**	0.646	0.646	0.659	0.652	0.642	0.641
Diabetes (6.5)	0.679	0.756	0.747	0.703	0.686	**0.772**	0.761
Osteoporosis (9.85)	0.65	0.691	0.683	0.649	0.65	**0.695**	0.686
Stroke (15.86)	0.61	0.698	0.693	0.622	0.58	**0.706**	0.695
Heart Attack (16.7)	0.648	0.709	0.694	0.63	0.635	0.71	**0.72**
Angina (26.51)	0.629	0.689	0.692	0.614	0.614	0.692	**0.707**
Dementia (56.96)	0.707	0.743	0.75	0.681	0.683	**0.766**	0.746
Parkinson's (160.3)	0.601	0.668	**0.679**	0.581	0.563	0.674	0.647
Average Rank	4.15	3.6	3.6	5.4	5.45	**2.8**	3

Accuracy values reflect how well a model predicts both positive and negative class instances. Note, however, that the proportion to which each class contributes to the accuracy value is dependent on the proportion of instances of each class in the dataset. As the Accuracy values are calculated by dividing the sum of true positive and true negative predictions by the total number of predictions, and the positive class represents the majority of instances, the number of true positive predictions has a bigger impact on the accuracy value than the number of true negative predictions.

In the Accuracy results the 7NN method has the smallest average rank (2.8), followed by the proposed Data-Driven approach (average rank of 3), repeating the trend observed when analysing Sensitivity (Table 7). When considering Accuracy, the Baseline models (learned from datasets without any missing value imputation) were outperformed by both the 7NN and Data-Driven methods for 7 out of the 10 datasets. These results corroborate the conclusions of our classifier-independent comparison of the missing value imputation methods (Section 6), where the 7NN and Data-Driven methods had overall the best results considering both their applicability and average estimation error rates.

To further investigate the difference between the RF models' performances, we compared their results using two non-parametric statistical significance tests, as follows. First, we performed the Friedman's test, a rank-based non-parametric version of ANOVA with repeated measures (Friedman, 1940). The Friedman's test can be used to compare the performance of several classification models simultaneously, and infer whether their results are statistically equivalent or not. In the latter case, a second, non-parametric, post-hoc statistical test would be required to determine whether or not different pairs of models have equivalent performance.

We chose non-parametric rank-based tests as, due to the small sample size (10 datasets), we could not assume a normal distribution of the data (Higgins, 2004, Chapter 4), which is necessary for several other statistical tests that could be used to compare classifier results.

We applied the Friedman's test to the Accuracy results for the 7 missing value-handling techniques, with the usual significance level of $\alpha = 0.05$. The test resulted in a *p-value*$=$ 0.022774, which meant there was enough evidence to reject the null hypothesis (that the models' performances were equivalent).

Therefore, we applied a post-hoc non-parametric test to compare the best ranked method (7NN) to each of the other imputation methods, adjusting the α values using Holm's procedure for multiple tests (Holm, 1979). The test's results showed that 7NN can be considered significantly superior to the Prev (*p-value*$=$ 0.0015 compared against adjusted $\alpha = 0.0083$) and PrevNext methods (*p-value*$=$ 0.0019 compared against adjusted $\alpha = 0.01$), and equivalent to all other methods.

8 Conclusions

In this article we proposed a Data-Driven missing value imputation approach, and performed two sets of experiments comparing it to different strategies to handle missing values in 10 longitudinal datasets. The datasets were created for the task of classification, using real data from the English Longitudinal Study of Ageing, with 10 age-related diseases as target variables and 140 features (mostly biomarkers, with numeric, nominal and binary values), and a proportion of 38.5% missing values.

The proposed Data-Driven approach performs a feature-wise ranking of a set of missing value imputation methods, based on a calculation of their average estimation error for the known values of each feature. Then, it applies these methods to replace the missing values in each feature, starting from the best ranked one (lowest estimation error) up until no missing values remain, or no more methods can be used. We chose five missing value imputation methods for our experiments in this study, but any set of methods can be used by the Data-Driven approach. The chosen methods were Global Mean/mode and Age-based Mean/Mode, Previous Observation Carried Forward, Previous and Next Observations Combined, and K-Nearest Neighbours Mean/Mode. This set of methods includes representatives of standard statistics methods, methods devised specifically for longitudinal data, and a more sophisticated method that uses a machine learning algorithm.

The proposed approach has been evaluated in two sets of experiments. First, we compared the five imputation methods to the Data-Driven approach on a classifier-independent evaluation where we calculated their applicabilities and average error rates. Then, we trained Random Forest classifiers with each of these methods and a baseline of doing no imputation, in a classifier-dependent approach where we investigated how the chosen method to handle missing data affected the performance of the resulting classifiers.

For the classifier-independent evaluation, the experimental results showed that each of the six tested imputation methods was the most accurate for some features in the datasets, which corroborates the notion that no single imputation method is the best for every feature. The most sophisticated method, the proposed Data-Driven approach, was considered the best-performing method overall, due to its 100% applicability rate and low mean error values.

The two methods devised specifically for longitudinal data (Prev and PrevNext) had very low applicability (35.57% and 2.95%, respectively). However, they had the small average error rates both considering the numeric features and considering all features in the dataset, with PrevNext having the smallest average error over all methods. That, together with the good performance of the KNN method (which also had longitudinal characteristics), shows the value of considering the often ignored temporal aspect of the data when handling missing values in a longitudinal dataset.

For the classifier-dependent scenario, first we performed a series of experiments with two undersampling approaches, to decide on the one that best fitted our datasets, and chose the Balanced Random Forest approach. Then, we compared the effects of employing seven different approaches to handle missing values in our longitudinal datasets. These strategies were using the five selected methods and the proposed Data-Driven approach to replace as many missing values as possible in the dataset, as well as the baseline approach of not replacing the missing values – letting the classification algorithm handle them internally. We analysed the performances of RF models trained with datasets created with each approach using three metrics: Sensitivity, Specificity and Accuracy.

The 7NN and Data-Driven methods achieved the best results for this set of experiments, with one of these two methods obtaining the highest Sensitivity and Accuracy values in 7 out of 10 datasets, and both methods obtained the two best (lowest) average ranks for both metrics. Although the 7NN approach slightly outperformed the Data-Driven approach in this classifier-dependent evaluation, the latter has the advantage of guaranteeing that every missing value will be replaced, whereas the 7NN method was applicable to only 81.79% of the missing values in the datasets.

Future work may involve additional experiments with the proposed Data-Driven approach, both with different longitudinal datasets and different sets of missing value imputation methods. As a general recommendation, we highlight the inclusion of methods devised for longitudinal data in machine learning applications for this type of dataset, given the good results these methods obtained in our experiments.

Acknowledgements The English Longitudinal Study of Ageing was developed by a team of researchers based at the University College London, NatCen Social Research, and the Institute for Fiscal Studies. The data were collected by NatCen Social Research. The data are linked to the UK Data Archive and freely available through the UK data services (https://discover.ukdataservice.ac.uk).

References

K. M. Albridge, J. Standish, and J. F. Fries. Hierarchical time-oriented approaches to missing data inference. *Computers and Biomedical Research*, 21(4):349–366, 1988.

J. Banks, E. Breeze, C. Lessof, and J. Nazroo. *The dynamics of ageing: Evidence from the English Longitudinal Study of Ageing 2002-15 (Wave 7)*. Institute for Fiscal Studies, London, 2016. URL http://www.elsa-project.ac.uk/publicationDetails/id/8696.

J. Banks, G. Batty, K. Coughlin, K. Deepchand, M. Marmot, J. Nazroo, Z. Oldfield, N. Steel, M. A. Steptoe, Wood, and P. Zaninotto. English longitudinal study of ageing: Waves 0–8, 1998–2017.[data collection], 2019.

M. Belger, J. Haro, C. Reed, M. Happich, K. Kahle-Wrobleski, J. Argimon, G. Bruno, R. Dodel, R. Jones, B. Vellas, et al. How to deal with missing longitudinal data in cost of illness analysis in alzheimer's disease—suggestions from the geras observational study. *BMC Medical Research Methodology*, 16(1): 83, 2016.

K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft. When is "nearest neighbor" meaningful? In *International conference on database theory*, pages 217–235. Springer, 1999.

L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

C. Chen, A. Liaw, L. Breiman, et al. Using random forest to learn imbalanced data. *University of California, Berkeley*, 110(1-12):24, 2004.

P. Diggle. *Analysis of longitudinal data*. Oxford University Press, 2002.

J. M. Engels and P. Diehr. Imputation of missing longitudinal data: a comparison of methods. *Journal of clinical epidemiology*, 56(10):968–976, 2003.

M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim. Do we need hundreds of classifiers to solve real world classification problems? *The journal of machine learning research*, 15(1):3133–3181, 2014.

M. Friedman. A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics*, 11(1):86–92, 1940.

A. M. Gad and R. H. M. Abdelkhalek. Imputation methods for longitudinal data: A comparative study. *International Journal of Statistical Distributions and Applications*, 3(4):72, 2017.

J. J. Higgins. *Introduction to modern nonparametric statistics*. Brooks/Cole, Pacific Grove, CA, 1st edition, 2004.

S. Holm. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, pages 65–70, 1979.

Z. Hu, G. B. Melton, E. G. Arsoniadis, Y. Wang, M. R. Kwaan, and G. J. Simon. Strategies for handling missing clinical data for automated surgical site infection detection from the electronic health record. *Journal of Biomedical Informatics*, 68:112–120, 2017.

N. Kouiroukidis and G. Evangelidis. The effects of dimensionality curse in high dimensional knn search. In *2011 15th Panhellenic Conference on Informatics*, pages 41–45. IEEE, 2011.

R. J. Little and D. B. Rubin. *Statistical analysis with missing data*, volume 793. John Wiley & Sons, 2019.

V. López, A. Fernández, S. García, V. Palade, and F. Herrera. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*, 250:113–141, 2013.

J. D. Malley, K. G. Malley, and S. Pajevic. *Statistical learning for biomedical data*. Cambridge University Press, 2011.

C. H. Mallinckrodt. *Preventing and treating missing data in longitudinal clinical trials: a practical guide*. Cambridge University Press, 2013.

S. Minhas, A. Khanum, F. Riaz, A. Alvi, S. A. Khan, A. D. N. Initiative, et al. Early alzheimer's disease prediction in machine learning setup: Empirical analysis with missing value computation. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 424–432. Springer, 2015.

T. Pomsuwan and A. A. Freitas. Feature selection for the classification of longitudinal human ageing data. In *IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 739–746. IEEE, 2017.

J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993. ISBN 1-55860-238-0.

C. Ribeiro and A. A. Freitas. Comparing the effectiveness of six missing value imputation methods for longitudinal classification datasets. In *3rd Workshop on AI for Aging, Rehabilitation and Independent Assisted Living (ARIAL), held as part of IJCAI-2019*, 2019.

M. S. Santos, J. P. Soares, P. Henriques Abreu, H. Araújo, and J. Santos. Influence of data distribution in missing data imputation. In A. ten Teije, C. Popow, J. H. Holmes, and L. Sacchi, editors, *Artificial Intelligence in Medicine*, pages 285–294, Cham, 2017. Springer International Publishing. ISBN 978-3-319-59758-4.

G. M. Weiss and F. Provost. Learning when training data are costly: The effect of class distribution on tree induction. *Journal of artificial intelligence research*, 19:315–354, 2003.

C. Zhang, C. Liu, X. Zhang, and G. Almpanidis. An up-to-date comparison of state-of-the-art classification algorithms. *Expert Systems with Applications*, 82: 128–150, 2017.

J. Zhao, Q. Feng, P. Wu, R. Lupu, R. A. Wilke, Q. S. Wells, J. Denny, and W.-Q. Wei. Learning from longitudinal data in electronic health record and genetic data to improve cardiovascular event prediction. *bioRxiv*, 2018. doi: 10.1101/366682. URL `https://www.biorxiv.org/content/early/2018/07/11/366682`.

X. Zhu. Comparison of four methods for handing missing data in longitudinal data analysis through a simulation study. *Open Journal of Statistics*, 4(11):933, 2014.

Appendix A: Feature Description Table

Table A1: Description of the features in the ELSA nurse-data dataset.

Conceptual Feature	**Description**	**Present in waves**			**Type**
		2	**4**	**6**	
sex	Gender of the participant	Not applicable			Binary
sysval	Mean systolic blood pressure	X	X	X	Numeric
diaval	Mean diastolic blood pressure	X	X	X	Numeric
pulval	Pulse pressure	X	X	X	Numeric
mapval	Mean arterial pressure	X	X	X	Numeric
mmgsd_avg	Mean grip strenght with dominant hand	X	X	X	Numeric

Table A1 continued from previous page

mmgsn_avg	Mean grip strenght with non-dominant hand	X	X	X	Numeric
clotb	Blood sample: whether has clotting disorder	X	X	X	Binary
cfib	Blood fibrinogen level (g/l)	X	X	X	Numeric
chol	Blood total cholesterol level (mmol/l)	X	X	X	Numeric
hdl	Blood High-density lipoprotein (HDL) level (mmol/l)	X	X	X	Numeric
trig	Blood triglyceride level (mmol/l)	X	X	X	Numeric
ldl	Blood LDL cholesterol level (mmol/l)	X	X	X	Numeric
fglu	Blood glucose level while fasting (mmol/L)	X	X	X	Numeric
rtin	Blood ferritin level (ng/ml)	X	X	X	Numeric
hscrp	Blood C-reactive protein (CRP) level (mg/l)	X	X	X	Numeric
hgb	Blood haemoglobin level (g/dl)	X	X	X	Numeric
hba1c	Blood glycated haemoglobin level (mmol/mol)	X	X	X	Numeric
htval	Height (cm)	X	X	X	Numeric
wtval	Weight (Kg)	X	X	X	Numeric
bmiobe	Body mass index grouped according to World Health Organization definitions	X	X	X	Nominal
wstval	Mean waist (cm)	X	X	X	Numeric
hipval	Mean hip (cm)	X	X		Numeric
whval	Mean waist/hip ratio	X	X		Numeric
hasurg	Whether had abdominal or chest surgery in the past 3 months	X	X	X	Binary
eyesurg	Whether have a detached retina or had eye or ear surgery in the past 3 months	X	X	X	Binary
hastro	Whether been admitted to hospital with a heart complaint in the past month	X	X	X	Binary
chestin	Lung function: Whether had any respiratory infection in last 3 weeks	X	X	X	Binary
htfvc	LUNG: Highest technically satisfactory value for Forced Vital Capacity	X	X	X	Numeric
htfev	LUNG: Highest technically satisfactory value for Forced Expiratory Volume	X	X	X	Numeric
htpf	LUNG: Highest technically satisfactory value for Peak Flow	X	X	X	Numeric
mmssre	Outcome of side-by-side stand	X	X	X	Nominal
mmstre	Outcome of semi-tandem stand	X	X	X	Nominal
mmftre2	Outcome of full tandem stand according to age	X	X	X	Nominal

Table A1 continued from previous page

mmlore	Leg raise (eyes open): Outcome	X	X	X	Nominal
mmlsre	Leg raise (eyes shut): Outcome	X	X	X	Nominal
mmcrre	Single chair rise outcome	X	X	X	Nominal
mmrroc	Outcome of multiple chair rises, split by age	X	X	X	Nominal
igf1	Blood insulin-like growth factor (IGF-1) level (nmol/l)		X	X	Numeric
wbc	White blood cell count (x 10ˆ9 cells/litre)		X	X	Numeric
mch	Blood mean corpuscular haemoglobin level (pg/cell)		X	X	Numeric
apoe	Blood apolipoprotein E (apoE) level (mmol/l)	X			Numeric
dheas	Blood dehydroepiandrosterone (DHEAS) level (umol/l)		X		Numeric
vitd	Vitamin D level (unit)			X	Numeric
indager	Respondent age			X	Numeric