A New Random Forest Method for Longitudinal Data Classification Using a Lexicographic Bi-Objective Approach

Caio Ribeiro
School of Computing
University of Kent
Canterbury, UK 30332–0250
Email: C.E.Ribeiro@kent.ac.uk

Alex Freitas
School of Computing
University of Kent
Canterbury, UK 30332–0250
Email: A.A.Freitas@kent.ac.uk

***Abstract*—Standard supervised machine learning methods often ignore the temporal information represented in longitudinal data, but that information can lead to more precise predictions in classification tasks. Data preprocessing techniques and classification algorithms can be adapted to cope directly with longitudinal data inputs, making use of temporal information such as the time-index of features and previous measurements of the class variable. In this article, we propose two changes to the classification task of predicting age-related diseases in a real-world dataset created from the English Longitudinal Study of Ageing. First, we explore the addition of previous measurements of the class variable, and estimating the missing data in those added features using intermediate classifiers. Second, we propose a new split-feature selection procedure for a random forest's decision trees, which considers the candidate features' time-indexes, in addition to the information gain ratio. Our experiments compared the proposed approaches to baseline approaches, in 3 prediction scenarios, varying the "time gap" for the prediction – how many years in advance the class (occurrence of an age-related disease) is predicted. The experiments were performed on 10 datasets varying the class variable, and showed that the proposed approaches increased the random forest's predictive accuracy.**

I. Introduction

Longitudinal datasets contain information about the same cohort of individuals followed through a long period of time, with the same set of variables being measured repeatedly. Such datasets consist 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. This work focuses on the longitudinal analysis of human ageing data [1], [2]; which is a very relevant research area because old age is one of the greatest risk factors for many diseases.

Supervised machine learning (ML) methods can be adapted to directly cope with longitudinal data and use the time-related information of the data. However, few existing supervised ML methods directly cope with longitudinal datasets.

In this article, we focus on the ML task of classification, which aims to predict the value of a nominal class variable for an instance, based on its features' values. These algorithms use training data (instances with known class values) to create a model for predicting the class of test data (previously unseen instances). For our experiments, we used real-world longitudinal datasets created from the English Longitudinal Study of Ageing (ELSA) [3], with the diagnosis of 10 age-related diseases as the class variables, and a set of biomedical features as predictors. We propose adaptations to the ML process that use the time-related information of longitudinal data to increase predictive accuracy, both in the data preprocessing stage and during the execution of the classification algorithm.

This study has two main contributions. First, we add to the dataset class labels measured in waves prior to the target wave, which we named past-class features, as predictive features. We discuss how adding these past-class features changes the classification problem, and propose a method of replacing missing values in these special features using intermediate classification models.

The second contribution is an adaptation to the random forest (RF) [4] ensemble learning algorithm – more precisely, a new lexicographic bi-objective split-feature selection procedure that considers both the information gain ratio and the time index of the candidate features when selecting the split feature of each node in the RF's decision trees. In essence, the lexicographic approach gives priority to select features with a higher information gain ratio, but when more than one candidate features have approximately the same highest gain ratio, the most recent feature among those is selected. This is based on the heuristic that more recent values of biomedical features tend to be more useful for predicting future occurrences of diseases than older values of the same features.

To evaluate these two adaptations, we performed experiments using 3 different prediction scenarios where the input features come from different time-points (waves) of the dataset, simulating predictions of the class label 8 years after, 4 years after or in the same wave as the last input wave. We report results based on 4 different predictive performance metrics: Sensitivity, Specificity, Accuracy and the Geometric Mean between Sensitivity and Specificity. Both proposed methods performed better than the baseline method, in general.

This article is organised as follows. Section 2 describes the datasets used in our experiments and the experimental setup. Section 3 describes and evaluates the first contribution, handling class labels from past waves as predictive features. Section 4 describes and evaluates the second contribution, a lexicographic bi-objective approach for selecting the split feature in the decision trees of a random forest. Finally, Section 5 summarises our findings and proposes future studies.

II. Experimental Setup

A. Dataset Description

For our experiments, we created datasets using data from the English Longitudinal Study of Ageing (ELSA) [3]. The study's core participants (50+ years old UK residents) are interviewed repeatedly, over the years prior to their retirement and beyond. Each wave of the ELSA is two years apart, and every two waves biomedical data is collected by a nurse or health professional, which we refer to as ELSA-nurse data.

After preprocessing (including a conceptual feature selection), the longitudinal dataset has 140 biomedical features (40 unordered nominal and 100 numeric features), and 7096 instances, with 4 ELSA-nurse waves being considered (ELSA waves 2, 4, 6 and 8), each 4 years apart from the next.

Several methods can be used to estimate missing values, but no method is the best for all types of data and applications [5], [6]. Hence, we use internal cross-validation on the training set to select the best out of 5 missing value replacement methods for each of the biomedical features in the dataset. In essence, the 5 used methods try to replace a feature's missing value by: (a) the mean or mode of the feature, for numerical and nominal features, respectively; (b) the mean or mode of the feature among subjects with the same age as the current subject; (c) the value of the feature for the current subject in the previous wave (time point); (d) the mean or mode of the feature values for the current subject in the previous and next waves; (e) a value computed by using the K-Nearest Neighbours algorithm. For more details about these 5 methods and the internal cross-validation for evaluating them, see [7]. For each feature, an internal cross-validation is performed on the subset of training instances where the feature's value is known, and then the method with the highest accuracy (among those 5 methods) is chosen to replace all missing values of that feature.

Initial experiments indicated that using this strategy increased the average accuracy of the classification algorithms, when compared to leaving the missing values in the features to be handled by the algorithm during its execution.

The dataset was created for the classification task. Each class variable corresponds to a binary class label on a specific time-point (wave of the ELSA study). The class labels in our datasets refer to the diagnosis of 10 age-related diseases. They are computed from features in the ELSA core interview, related to the diagnosis of these target diseases. These features started being measured in the study's third wave, and have been measured in all waves since. Thus, if a subject I has participated in the ELSA core interview for a given wave t ($3 \leq t \leq 8$), we obtained their class label in that wave, $class_{I,t}$. For subjects who did not take part in the ELSA core interview at a given wave t, their $class_{I,t}$ is marked as a missing value. Note that in the created datasets all instances represent ELSA-nurse participants at the final wave 8, so there are no missing class labels for the final wave. For more details on the target variables' creation, please see [8].

It is important to highlight that, for our specific application the predictive biomedical features were selected from the nurse-data part of the ELSA study, which is collected every two waves; but the class variables are created from the core data in ELSA, for every wave. This is because the biomedical data from the ELSA-nurse requires a physical visit from a health professional to the households of the participants, where several tests are conducted including blood samples, mental health assessments, and motor skill tests. For the ELSA-core interviews, used to create our target variables, most data collection happens over phone calls made to the participant households. Therefore, collecting data for the predictive features is more difficult or expensive than for the different measurements of the class variables, for this ELSA dataset.

B. Experimental Methodology

The baseline scenario for our classification task is to use the 140 biomedical features as predictive features, and the class values at the final wave 8 as the target variable, for each of the 10 age-related diseases separately. However, as we have class information from waves prior to the target wave, we investigate an interesting variation of this classification problem that considers previous class values as predictive features. This new variation corresponds to the scenario where we need to classify patients whose medical history includes information about whether or not they had a given disease in several past waves (time points). Although this past class information makes the problem easier, an intelligent system is still needed to predict future class values, as simply expecting that the known past class value will remain unchanged would not be an adequate use of the available information.

Figure 1 displays characteristics of our 10 datasets, including the Imbalance Ratio (IR) of the target (class) variables. The IR is simply the ratio of the number of majority class instances (patients who were not diagnosed with the age-related disease) over the number of minority class instances (patients who were diagnosed with the disease).

All 10 datasets used in our experiments suffer from class imbalance (see Figure 1), as usual in health data. To address this, in all experiments the training sets were balanced using the "balanced random forest" approach [9], which undersamples majority class instances when sampling instances for learning each decision tree in the RF, to a ratio of 1:1.

In the next two Sections, we report on two sets of experiments. First, in Section III, we compare two approaches to handle missing values in the class variables from past waves – which are used as additional predictive features as mentioned earlier. One is our proposed approach, which consists of using intermediate classification models to handle those missing values (as explained in detail later), and the other is the

For all 10 datasets:
- 7096 Instances;
- 140 Biomedical Features from ELSA-nurse waves 2, 4, 6 and 8;
- Class labels available for ELSA core waves 3, 4, 5, 6, 7 and 8;

Class Label	N Missing Values
Wave 3	2309 (33%)
Wave 4	845 (12%)
Wave 5	769 (11%)
Wave 6	125 (2%)
Wave 7	237 (3%)
Wave 8	0

Dataset (Disease)	Abbrev.	Imbalance Ratio
Arthritis	Arth.	1.35
High Blood Pressure	HBP	1.49
Cataracts	Catar.	2.06
Diabetes	Diab.	6.5
Osteoporosis	Osteo.	9.85
Stroke	Stroke	15.86
Heart Attack	H. Att.	16.7
Angina	Angina	26.51
Dementia	Dem.	59.96
Parkinson's Disease	Park. D.	160.3

Fig. 1. Description of our ELSA datasets.

baseline approach of letting the classification algorithm use its own internal strategy to cope with those missing values. Later, in Section IV, the RF algorithm is modified to better handle longitudinal data. This modification is the second and main contribution of this article, which alters how the split function of the decision trees in the RF selects a feature out of the sampled candidate features, in each node.

The results presented in Sections III and IV show that both approaches improved the learned RF models, according to two global metrics of predictive accuracy (Accuracy and GMean).

III. Handling Class Labels from Past Waves as Predictive Features

Class variables from all waves prior to the target wave 8, referring to past diagnoses of the current target disease, were used as predictive features. These variables are named 'past-class features', and they have on average 12% missing values (see Figure 1). When training classifiers, one can either let the classification algorithm handle the past-class features' missing values or replace them beforehand. In this Section we compare the approach of not replacing the past-class features' missing values, named NoRep (No Replacement) with our proposed approach of replacing past-class features' missing values using intermediate classification models, named RepCM (Replacement by Classification Models).

The RFs were trained and tested using the Weka toolkit's source code[1], in a 10-fold cross-validation, with the parameters $ntrees = 100$ (number of decision trees) and $mtry = \lfloor sqrt(nfeatures) \rfloor = 12$ (number of features randomly sampled to be used as candidate features at each tree node).

A. The NoRep and RepCM Approaches

For the NoRep approach, the past-class features' missing values are handled by the classification algorithm. The RF implementation used in our experiments uses the C4.5 algorithm's [10] technique to cope with missing values when building its decision trees, as follows. Initially, each instance is assigned a 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, in order to compute that feature's information gain (or any other feature evaluation measure), 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 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.

For the proposed RepCM approach, for each past-class feature in the input waves, starting at the earliest wave (wave 3), we train classifiers making that past-class feature as the target variable. These intermediate classification models are learned using a subset of the training data with all applicable features and instances (i.e., removing features from future waves and instances with missing values for the current target variable), then applied to predict the that past-class feature's missing values, in the original dataset.

Recall that our ELSA-nurse dataset is longitudinal, having 4 waves with Nurse-data features (waves 2, 4, 6, 8) and 6 waves with class variables (3, 4, 5, 6, 7, 8). In our classification problems, the class variable to be predicted is always set at wave 8, the last wave. Hence, in order to investigate different longitudinal prediction scenarios, we used different sets of waves (i.e. different time points) as sources of directly observed predictive features, by varying the "time gap" between the last observed predictive features in the input data and the target class variable at wave 8. More precisely, we investigate three longitudinal prediction scenarios, as shown in Figure 2.

In the first scenario (top of Figure 2) there are two types of predictive features: (a) the standard Nurse-data features, directly available from input data in waves 2 and 4; and (b) the 'observed' past-class features, directly available from input data in waves 3 and 4. Hence, since there is a gap of 4 waves between the last wave of observed data (wave 4) and the wave of the target class variable, and since there is a gap of two years between every two consecutive waves, this scenario involves predicting the target class (disease) 8 years in advance. In the second scenario (middle of Figure 2), the last wave of observed data is wave 6, so this involves the prediction of the target class 4 years in advance. Finally, in the third scenario (bottom of Figure 2), the last wave of observed predictive features is wave 8. For all scenarios, the missing values in the observed past-class features are predicted using intermediate models when applying the RepCM approach, and left unchanged in the NoRep approach.

B. Experimental Results

The RF classifiers were compared based on the 4 metrics: Sensitivity (True Positive Rate), Specificity (True Negative Rate), Accuracy (percentage of correct classifications) and GMean (Geometric mean between Sensitivity and Specificity).

[1]Open-source, available at: https://www.cs.waikato.ac.nz/ml/weka/

These metrics were chosen mainly based on [11, 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 metrics are to be maximised, to simplify the results' analyses. We also use GMean, as a global performance metric that assigns equal importance to the correct prediction of both classes – unlike Accuracy, which assigns much greater importance to the correct prediction of majority-class instances (which are easier to be predicted in general).

Fig. 2. The tested prediction scenarios.

Tables I, II and III show the results of experiments comparing the NoRep and RepCM approaches, with the different ranges of input features. For Table I, the original input features belonged to waves 2 and 4, and the past-class features (which are handled in different ways by the NoRep and RepCM approaches) are from waves 3 and 4. For Table II, the original input features are from waves 2, 4 and 6, and the past-class features are from waves 3, 4, 5 and 6. Finally, Table III uses the entire dataset, so all original features and past-class features are included in the dataset.

In Tables I, II and III, in the dataset column, the '(IR)' stands for the class imbalance ratio of the dataset. In addition, in these three tables, in each pair of adjacent columns with the results of the NoRep and RepCM approaches for a given measure, for each table row (i.e. for each dataset) the highest value of the measure among those two approaches is highlighted in boldface. Finally, the last row of the tables show the number of wins of each approach, for each of the measures.

TABLE I
RANDOM FOREST RESULTS COMPARING NOREP AND REPCM FOR INPUT WAVES 2-4.

Dataset (IR)	Sensitivity		Specificity		Accuracy		GMean	
	NoRep	RepCM	NoRep	RepCM	NoRep	RepCM	NoRep	RepCM
Arth. (1.35)	0.877	**0.898**	**0.697**	0.685	0.800	**0.807**	0.782	**0.784**
HBP (1.49)	0.803	**0.842**	**0.779**	0.742	0.793	**0.802**	**0.791**	0.790
Catar. (2.06)	0.644	**0.723**	**0.727**	0.639	0.671	**0.695**	**0.684**	0.680
Diab. (6.5)	0.851	**0.878**	**0.826**	0.792	0.848	**0.866**	**0.838**	0.834
Osteo. (9.85)	0.729	**0.755**	**0.703**	0.680	0.727	**0.748**	0.716	**0.717**
Stroke (15.86)	0.730	**0.792**	**0.694**	0.651	0.728	**0.784**	0.711	**0.718**
H. Att. (16.7)	0.818	**0.877**	**0.706**	0.668	0.812	**0.866**	0.760	**0.766**
Angina (26.51)	0.761	**0.845**	**0.690**	0.636	0.758	**0.838**	0.725	**0.733**
Dem. (56.96)	0.744	**0.782**	**0.703**	0.676	0.743	**0.779**	0.723	**0.727**
Park. D. (160.3)	0.604	**0.652**	**0.667**	0.591	0.604	**0.652**	**0.634**	0.621
N of Wins	0	**10**	**10**	0	0	**10**	4	**6**

TABLE II
RANDOM FOREST RESULTS COMPARING NOREP AND REPCM FOR INPUT WAVES 2-6.

Dataset (IR)	Sensitivity		Specificity		Accuracy		GMean	
	NoRep	RepCM	NoRep	RepCM	NoRep	RepCM	NoRep	RepCM
Arth. (1.35)	0.939	**0.942**	**0.833**	0.819	0.894	**0.889**	**0.884**	0.878
HBP (1.49)	0.888	**0.901**	**0.844**	0.815	**0.870**	0.866	**0.866**	0.857
Catar. (2.06)	0.730	**0.799**	**0.731**	0.669	0.730	**0.756**	0.730	**0.731**
Diab. (6.5)	0.925	**0.946**	**0.873**	0.857	0.918	**0.934**	0.899	**0.900**
Osteo. (9.85)	0.859	**0.873**	**0.734**	0.713	0.848	**0.859**	**0.794**	0.789
Stroke (15.86)	0.826	**0.892**	**0.729**	0.672	0.821	**0.879**	**0.775**	**0.775**
H. Att. (16.7)	0.932	**0.960**	**0.783**	0.761	0.924	**0.948**	**0.854**	**0.854**
Angina (26.51)	0.829	**0.886**	**0.721**	0.678	0.825	**0.878**	0.773	**0.775**
Dem. (56.96)	0.742	**0.768**	**0.709**	0.703	0.741	**0.767**	0.725	**0.735**
Park. D. (160.3)	0.703	**0.729**	**0.682**	0.636	0.703	**0.728**	**0.693**	0.681
N of Wins	0	**10**	**10**	0	1	**9**	**5**	**5**

TABLE III
RANDOM FOREST RESULTS COMPARING NOREP AND REPCM FOR INPUT WAVES 2-8.

Dataset (IR)	Sensitivity		Specificity		Accuracy		Gmean	
	NoRep	RepCM	NoRep	RepCM	NoRep	RepCM	NoRep	RepCM
Arth. (1.35)	0.950	**0.951**	0.881	**0.883**	0.921	**0.923**	0.915	**0.917**
HBP (1.49)	0.916	**0.924**	**0.900**	0.892	0.910	**0.911**	**0.908**	**0.908**
Catar. (2.06)	0.825	**0.875**	**0.771**	0.750	0.807	**0.834**	0.797	**0.810**
Diab. (6.5)	0.954	**0.971**	**0.905**	0.904	0.948	**0.962**	0.929	**0.937**
Osteo. (9.85)	0.923	**0.939**	0.803	**0.817**	0.912	**0.928**	0.861	**0.876**
Stroke (15.86)	0.894	**0.951**	**0.777**	0.767	0.887	**0.941**	0.833	**0.854**
H. Att. (16.7)	0.974	**0.983**	**0.850**	0.845	0.967	**0.975**	0.910	**0.912**
Angina (26.51)	0.876	**0.902**	**0.779**	0.767	0.872	**0.897**	0.826	**0.832**
Dem. (56.96)	0.763	**0.782**	**0.764**	0.730	0.763	**0.780**	**0.763**	0.755
Park. D. (160.3)	0.794	**0.831**	**0.758**	0.712	0.794	**0.830**	**0.776**	0.769
N of wins	0	**10**	**8**	2	0	**10**	2.5	**7.5**

The use of these three prediction scenarios allows us to observe how predicting a class label measured in a wave years after the last input wave affects the classifiers' predictive accuracy. There is a noticeable increase in all performance metrics when comparing the scenarios in Tables I, II and III, with the latter having classifiers with the best performance for all 10 datasets. That was expected, as wave 8's biomedical features were measured at about the same time of the target variable (disease diagnosis), and the input waves 2-8 scenario also has past-class values up to wave 7 (2 years before the final wave) included in their sets of predictive features.

For all three Tables (varying the set of input waves), the RepCM approach outperforms the NoRep approach in terms of the Sensitivity, Accuracy and GMean metrics, whilst NoRep's Specificity values were greater, reflecting the usual trade-off between Sensitivity and Specificity. Note that the changes

to the datasets, when applying each approach, are only the inputted missing values for the (up to 5) past-class features – whose originally missing values were replaced by values predicted by intermediate classifiers when using the RepCM approach, and left in the dataset (to be handled by the RF algorithm) when using the NoRep approach.

Because of the imbalance in our datasets favouring the majority class (the positive class), the Sensitivity (or True Positive Rate) has much greater impact in the Accuracy values than the Specificity (True Negative Rate), so it is expected that the RepCM approach would have the best Accuracy values in all cases (with only a single tie for the least imbalanced dataset, Arthritis, in the 2-6 input scenario), as its Sensitivity results were superior in every model. However, the GMean metric considers both Sensitivity and Specificity with an equal weight, and the GMean values were overall better for the RepCM approach in Tables 1 and 3, where RepCM obtained 6 and 7.5 wins (out of 10 datasets) over NoRep, respectively. In Table 2, RepCM and NoRep had a tie in terms of GMean values, each with 5 wins. Therefore we believe it is fair to say that overall the proposed RepCM is the best choice out of these two approaches.

Then, we performed the Wilcoxon signed rank test to compare the average rankings (across the 10 datasets) of the NoRep and RepCM approaches, considering the usual base significance level $\alpha = 0.05$. We also applied the Bonferroni Correction for multiple hypothesis testing for each set of results involving the 3 prediction scenarios, for each performance measure. Hence, with this correction, the Wilcoxon test's result is considered statistically significant if its p-value is smaller than the adjusted $\alpha = 0.05/3 = 0.0167$.

The p-values of the test's results are shown in Table IV. For the Sensitivity and Specificity measures, the differences between NoRep and RepCM were significant for all three prediction scenarios, with the results being in favour of RepCM for Sensitivity and in favour of NoRep for Specificity. For the Accuracy measure, the results were significantly different in favour of RepCM in two prediction scenarios, with no significant difference in the other scenario. For the GMean measure, the results were significantly different in favour of RepCM in one prediction scenario, with no significant difference in the other two scenarios. In summary, RepCM obtained significantly better results than NoRep in 6 cases (spread across 3 measures), whilst the converse was true in only 3 cases (all for the Specificity measure).

TABLE IV
P-VALUES OF THE WILCOXON SIGNED-RANK TESTS COMPARING THE NOREP AND REPCM APPROACHES. VALUES MARKED WITH AN "*" ARE OF TESTS WHERE THE NULL HYPOTHESIS WAS REJECTED WITH AN ADJUSTED SIGNIFICANCE LEVEL OF $\alpha = 0.0167$.

Comparing NoRep/RepCM	Sensitivity	Specificity	Accuracy	GMean
Input 2-4	0.0009*	0.0009*	0.0009*	0.2869
Input 2-6	0.0009*	0.0009*	0.0210	0.7796
Input 2-8	0.0009*	0.0164*	0.0009*	0.0163*

IV. THE LEXICOGRAPHIC APPROACH FOR SELECTING SPLITTING FEATURES

Although there are many supervised machine learning (ML) algorithms for classification and regression, few of them can directly cope with longitudinal datasets. Longitudinal features contain time-related information that is usually disregarded by these standard ML algorithms, and could be harnessed to improve predictive performance. In this section we propose and evaluate an adaptation to the random forest algorithm that explores temporal information when selecting the feature to be used in a decision-tree node split.

The adaptation consists of considering not only the features' information gain ratios but also their time points (wave ids) when choosing the split feature inside a decision tree's node, making the decision bi-objective. The rationale for this bi-objective feature evaluation is that we intend to add a bias favouring more recent information. Intuitively, the further in the past a feature value was measured, the less it is related to the class label, so when two features have seemingly equivalent gain ratios, the best decision would be to select the most recent of the two.

More precisely, when choosing the feature to be used in a node's split, the decision trees in our adapted random forest algorithm will consider the maximising the gain ratio as the primary objective and maximising the time-index of the features (wave ids) as the secondary objective. The rationale for prioritising gain ratio over the time index is that this is clearly the most important criterion for improving predictive accuracy, whilst preferring more recent feature values as a tie-breaking criterion is a heuristic for improving accuracy.

This approach of optimising objectives in priority order is sometimes called the lexicographic approach [12], and it has been used in decision tree algorithms for conventional (non-longitudinal) classification before [13]. However, to the best of our knowledge, a lexicographic approach such as the one proposed in this article has never been used for longitudinal classification before. However, a similar strategy of using time-related information in the split decision was used in [14], where the authors combine entropy gain and a time-related distance measure in their split criteria, for an application in time series datasets.

In the standard algorithm, when a decision tree of the RF is selecting a feature to be used as the split feature in a node, it randomly samples a set S of features from the dataset ($|S| = mtry$, as mentioned earlier), and orders these features based on their Information gain ratio $g(f_{i,j})$ (feature i measured at time j), selecting the one with the greater gain value.

For the lexicographic split-feature selection approach, we consider a threshold th as an additional parameter, and consider two features equivalent when the difference between their gain ratios is lower than this threshold. All features equivalent to the initial best feature are compared based on their time-indexes (wave id), and the most recent feature is selected. This process is described in Algorithm 1. Note that, although we are considering the gain ratio function $g(f_{i,j})$ as the primary

metric for selecting the split feature, it could be replaced by other metrics such as the information gain.

Algorithm 1 Lexicographic Split Feature Selection function, applied at each node of a decision tree. Receives a set of features S and a tie-threshold th (set by the user), and returns the selected $bestfeature$, based on gain ratio and the feature's time-index.

function $LexicographicSplitFeatureSelection(S, th)$
  $S.DescendingOrder(gainratio)$
  $bestf \leftarrow S[0]$
  $CandidateFeatures.add(bestf)$
  $pos \leftarrow 1$
  while $|g(bestf) - g(S[pos])| < th$ AND $pos < S.length$ **do**
    $CandidateFeatures.add(S[pos])$
    $pos++$
  end while
  $CandidateFeatures.DescendentOrder(\textit{time-index})$
  $bestf \leftarrow CandidateFeatures[0]$
  return $bestf$
end function

For example, consider a set S consisting of a feature $f_{1,1}$ with a gain ratio of $g(f_{1,1}) = 0.7$, and a feature $f_{2,2}$ with a gain ratio of $g(f_{2,2}) = 0.67$. In the standard decision tree algorithm, $f_{1,1}$ would be selected for the split as it has the greater gain value. In the lexicographic approach, that depends on the value of th. If $th = 0.05$, we have $|g(f_{1,1}) - g(f_{2,2})| < th$, so the features' gain ratios are considered equivalent and $f_{2,2}$ is selected instead, because it was measured at time-point 2 instead of 1. However, if $th = 0.01$, we have $|g(f_{1,1}) - g(f_{2,2})| > th$, so the features' gain ratios are not considered equivalent, and the selection proceeds normally, selecting $f_{1,1}$ based on its higher gain ratio. In the unlikely scenario of an exact tie for both the gain ratio value and the time-point criterion, a random selection is performed (the algorithm's default tie break).

The disadvantage of the lexicographic approach is the additional parameter to be selected by the user, the tie-definition threshold th. To address that disadvantage, we implemented a data-driven approach that selects a value for th using an internal cross-validation. More precisely, this data-driven selection of the th value performs an internal 5-fold cross-validation using the training set only. It creates RF classifiers using five possible threshold values (0.0, 0.005, 0.01, 0.015, 0.02), and chooses the value that yields the model with the best average Accuracy over the 5 folds in the internal cross-validation. The threshold value th is selected using this data-driven approach for each fold in the external cross-validation process.

We compared the RF with the lexicographic approach against the standard RF (without the lexicographic approach) in experiments using the ELSA-nurse dataset in the three prediction scenarios discussed in Section 3 (summarised in Figure 2), applying the RepCM method (as it was deemed the best choice in the previous Section's experiments). Tables V, VI and VII show the results of this comparison, for each prediction scenario. Note that we performed additional experiments fixing the threshold th value as each of the values used by the data-driven approach, but the data-driven selection of the best th value was shown to be more reliable in these experiments as well, so to save space here we are only comparing using the lexicographic approach to not using it.

TABLE V
RESULTS OF RANDOM FOREST WITH AND WITHOUT THE LEXICOGRAPHIC APPROACH USING INPUT WAVES 2-4.

Dataset (IR)	Sensitivity		Specificity		Accuracy		GMean	
	No Lexic	With Lexic	No Lexic	With Lexic	No Lexic	With Lexic	No Lexic	With Lexic
Arth. (1.35)	0.898	**0.912**	**0.685**	**0.678**	0.807	**0.813**	0.784	**0.787**
HBP (1.49)	0.842	**0.854**	**0.742**	**0.733**	0.802	**0.806**	0.790	**0.791**
Catar. (2.06)	0.723	**0.77**	**0.639**	0.597	0.695	**0.713**	**0.680**	0.678
Diab. (6.5)	0.878	**0.907**	**0.792**	0.776	0.866	**0.889**	0.834	**0.839**
Osteo. (9.85)	0.755	**0.806**	**0.680**	0.662	0.748	**0.793**	0.717	**0.731**
Stroke (15.86)	0.792	**0.842**	**0.651**	0.627	0.784	**0.83**	0.718	**0.727**
H. Att. (16.7)	0.877	**0.901**	**0.668**	0.651	0.866	**0.887**	**0.766**	**0.766**
Angina (26.51)	0.845	**0.851**	**0.636**	**0.636**	0.838	**0.843**	0.733	**0.735**
Dem. (56.96)	**0.782**	0.763	0.676	**0.689**	**0.779**	0.762	**0.727**	0.725
Park. D. (160.3)	**0.652**	0.643	**0.591**	0.561	**0.652**	0.642	**0.621**	0.6
N of Wins	2	**8**	**8.5**	1.5	2	**8**	3.5	**6.5**

TABLE VI
RESULTS OF RANDOM FOREST WITH AND WITHOUT THE LEXICOGRAPHIC APPROACH USING INPUT WAVES 2-6.

Dataset (IR)	Sensitivity		Specificity		Accuracy		GMean	
	No Lexic	With Lexic	No Lexic	With Lexic	No Lexic	With Lexic	No Lexic	With Lexic
Arth. (1.35)	0.942	**0.942**	0.819	**0.83**	0.889	**0.894**	0.878	**0.884**
HBP (1.49)	0.901	**0.903**	0.815	**0.828**	0.866	**0.873**	0.857	**0.865**
Catar. (2.06)	0.799	**0.845**	**0.669**	0.632	0.756	**0.775**	**0.731**	**0.731**
Diab. (6.5)	0.946	**0.969**	**0.857**	0.812	0.934	**0.948**	**0.900**	0.887
Osteo. (9.85)	0.873	**0.919**	**0.713**	0.7	0.859	**0.898**	0.789	**0.802**
Stroke (15.86)	0.892	**0.931**	**0.672**	0.634	0.879	**0.914**	**0.775**	0.769
H. Att. (16.7)	0.960	**0.971**	0.761	**0.766**	0.948	**0.959**	0.854	**0.862**
Angina (26.51)	0.886	**0.891**	0.678	**0.69**	0.878	**0.884**	0.775	**0.784**
Dem. (56.96)	0.768	**0.769**	**0.703**	**0.703**	**0.767**	**0.767**	**0.735**	**0.735**
Park. D. (160.3)	0.729	**0.737**	**0.636**	0.53	0.728	**0.735**	**0.681**	0.625
N of Wins	0	**10**	**5.5**	4.5	0.5	**9.5**	4	**6**

TABLE VII
RESULTS OF RANDOM FOREST WITH AND WITHOUT THE LEXICOGRAPHIC APPROACH USING INPUT WAVES 2-8.

Dataset (IR)	Sensitivity		Specificity		Accuracy		GMean	
	No Lexic	With Lexic	No Lexic	With Lexic	No Lexic	With Lexic	No Lexic	With Lexic
Arth. (1.35)	**0.952**	**0.952**	0.888	**0.892**	0.925	**0.926**	0.919	**0.921**
HBP (1.49)	**0.924**	**0.924**	**0.894**	**0.894**	**0.912**	**0.912**	**0.909**	**0.909**
Catar. (2.06)	0.876	**0.885**	**0.751**	0.748	0.835	**0.94**	0.811	**0.813**
Diab. (6.5)	0.972	**0.973**	0.903	**0.909**	0.962	**0.964**	0.937	**0.94**
Osteo. (9.85)	0.941	**0.942**	**0.813**	0.807	**0.929**	**0.929**	**0.875**	0.872
Stroke (15.86)	0.952	**0.954**	0.770	**0.779**	0.941	**0.943**	0.856	**0.862**
H. Att. (16.7)	**0.984**	0.983	0.845	**0.85**	**0.976**	**0.976**	0.912	**0.914**
Angina (26.51)	0.902	**0.905**	0.764	**0.791**	0.897	**0.901**	0.830	**0.846**
Dem. (56.96)	0.781	**0.787**	**0.743**	0.736	0.781	**0.786**	**0.762**	0.761
Park. D. (160.3)	0.836	**0.838**	0.727	**0.742**	0.835	**0.837**	0.780	**0.789**
N of Wins	2	**8**	3.5	**6.5**	1.5	**8.5**	2.5	**7.5**

For these experiments, there is a clear trend towards the lexicographic approach having higher predictive accuracy in general. More precisely, as can be observed in the 'number of wins' rows at the bottom of Tables V, VI and VII, the lexicographic approach obtained higher Sensitivity, Accuracy and GMean values in the large majority of the cases, whilst obtaining in general lower Specificity in most cases.

To analyse how much the lexicographic approach impacted the resulting random forests, we counted how often a tie of

gain ratio values happened and how often it resulted in a change of the selected feature for a node when learning each tree in the forest. About 25% of the nodes had at least one feature tied (i.e., gain ratio within threshold value) with the feature with the highest gain ratio, and about 50% of those ties resulted in a different feature being selected. Thus, around 12.5% of all tree nodes in the decision trees generated in our experiments had a different feature being selected as the split feature due to the lexicographic split approach. This is a considerable ratio, as our RF models are an ensemble of 100 unprunned decision trees.

We also counted how many times each th value was selected in the data-driven approach, over the 10-fold cross-validation procedure, for each of the 10 datasets (i.e. 100 threshold choices in total). The thresholds $th = 0, 0.005, 0.01, 0.015$ and 0.02 were selected $21, 21, 14, 25$ and 19 times, respectively. This confirms that that there is no single threshold which is the best for all cases, further justifying the use of the data-driven approach for tuning the threshold to the current training data. Note that $th = 0$ is not equivalent to not applying the lexicographic approach, as features would be considered equivalent with this threshold if they have the exact same gain ratio value, and a different feature could be selected if one of them is from a different time-point (wave).

Once again, we performed the Wilcoxon signed rank test, applying the Bonferroni correction, to compare the no lexic and lexic approaches, with adjusted $\alpha = 0.05/3 = 0.0167$.

The p-values of the results are shown in Table VIII. For the Sensitivity measure, the differences between lexic and no lexic approaches were significant for all three prediction scenarios, with the results being in favour of lexic. For the Specificity measure, the results were significantly different in favour of No lexic in only one prediction scenario, with no significant difference in the other two scenarios. For the Accuracy measure, the results were significantly different in favour of lexic in two prediction scenarios, with no significant difference in the other scenario. For the GMean measure, the results were not significantly different in any prediction scenario. In summary, lexic obtained significantly better results than no lexic in 5 cases (spread across the Sensitivity and Accuracy measures), whilst the converse was true in only one case (for the Specificity measure).

TABLE VIII
P-VALUES OF THE WILCOXON SIGNED-RANK TESTS COMPARING THE NO LEXIC AND LEXIC APPROACHES. VALUES MARKED WITH AN "*" ARE OF TESTS WHERE THE NULL HYPOTHESIS WAS REJECTED WITH AN ADJUSTED SIGNIFICANCE LEVEL OF $\alpha = 0.0167$.

Comparing NoLexic/Lexic	Sensitivity	Specificity	Accuracy	GMean
Input 2-4	0.0162*	0.0097*	0.0233	0.2026
Input 2-6	0.0019*	0.0863	0.0045*	0.4165
Input 2-8	0.0144*	0.1015	0.0108*	0.0377

A. Feature Importance Analysis

The experiments reported in this article focused on longitudinal human ageing datasets, involving the prediction of age-related diseases. Therefore, it is worthwhile to discuss trends in the learned classifiers, as their findings might help further research in this area. Note that in this study we did not optimise the hyper-parameters of the random forest algorithm for each of the 10 datasets (with different target diseases) – we focused instead on evaluating the predictive performance of the proposed approaches using random forests with standard hyper-parameter settings. Hence, the findings reported in this section are not optimised for each dataset, they are simply general trends we found when analysing the best models learned in our experiments, i.e. the random forests learned using the RepCM and Lexicographic approaches, using the full input data, from waves 2-8.

Random forest models are not as interpretable as individual decision trees [15], but they provide some interpretability by ranking the features in decreasing order of importance as computed by a feature importance measure [16], and then identifying the top-ranked features. This approach shows general trends for features to be selected across the nodes of the trees in the random forest. We used the feature importance measure available in Weka's implementation of the RF algorithm, which is based on the average decrease of class impurity of each feature over all tree nodes across all decision trees in the forest. We analysed the 10 features with the highest average class-impurity decrease for each of our 10 datasets, totalling 100 features.

Considering different measurements of the same feature across waves (time points) as part of the same "conceptual feature", we noted that 10 conceptual features appeared 4 or more times in that list. We show these features in Table IX. The first two columns of this table show a feature's code in the ELSA study and its description. The last column shows the class variable (disease) for which a feature was among the 10 features with the highest importance, and between brackets a count of how many measurements of that feature (across waves, or time points) were in the best 10 for the corresponding class – except for the sex feature, whose value does not vary with time in the dataset. Five of these features are related to a subject's blood sample analysis. Among the other features, one is the subject's sex, one is the outcome of a mobility test, and three are related to the subject's recent history of medical interventions.

V. CONCLUSIONS

We proposed two novel adaptations to learn classifiers for longitudinal datasets. The proposed adaptations use time-related information available in longitudinal data, often ignored by standard classifiers. We performed experiments comparing these adaptations to baseline approaches using 10 real-world longitudinal classification problems. The experiments used 4 predictive accuracy measures and 3 prediction scenarios (varying the input data's time points). Both adaptations increased the predictive accuracy of random forest classifiers.

For the first adaptation, we investigated two ways of adding past-class features as predictive features in the longitudinal dataset: adding them without replacing their missing values

TABLE IX
FEATURES THAT WERE AMONG THE 10 HIGHEST IMPORTANCE ONES THE MOST OFTEN, OVER THE 10 DATASETS.

ELSA code	Description	Predicted disease (count)
clotb	Blood: Whether has clotting disorder	Arthritis (1), Cataract (1), Dementia (1), Osteoporosis (2), Stroke (1)
hdl	Blood: High-density lipoprotein (HDL) level (mmol/l)	Angina (1), High BP (1), Osteoporosis (1), Parkinson's D. (1), Stroke (1)
hgb	Blood: Haemoglobin level (g/dl)	Arthritis (1), Dementia (1), High BP (1), Parkinson's D. (2)
igf1	Blood: Insulin-like growth factor (IGF-1) level (nmol/l)	Arthritis (1), Heart Attack (2), Parkinson's D. (1), Stroke (1)
trig	Blood: Triglyceride level (mmol/l)	Angina (1), Dementia (1), High BP (1), Heart Attack (1)
sex	Sex of the participant	Angina, Cataract, Dementia, Diabetes, High BP, Stroke
mmssre	Outcome of side-by-side stand test (seconds the patient stands on their own)	Angina (1), Diabetes (2), Stroke (1)
chestin	Lung function: Whether had any respiratory infection in last 3 weeks	Angina (1), Cataract (1) High BP (1), Heart Attack (2), Osteoporosis (2)
eyesurg	Whether have a detached retina or had eye or ear surgery in the past 3 months	High BP (1), Osteoporosis (1), Stroke (2)
hasurg	Whether had abdominal or chest surgery in the past 3 months	Angina (1), Arthritis (1), Cataract (1), High BP (1),

(NoRep), and replacing the past-class features' missing values using intermediate classifiers (RepCM). The experiments confirmed that learning intermediate classification models increased the overall predictive accuracy of the final classifiers.

We highlight that the RepCM approach can be applied regardless of the chosen classification algorithm, as it is part of the data preprocessing stage, and is recommended for longitudinal datasets with substantial changes in their instance sets over the course of the study – i.e., high dropout rates or new instances added frequently, which is common in health data and tends to increase the number of missing values.

The second adaptation consists of a bi-objective lexicographic criterion for selecting a node splitting feature in the random forest's decision trees. Hence, when multiple features have about the same information gain ratio, the time-index information is used to favour more recent measurements, as intuitively those are more valuable for increasing predictive accuracy. The proposed lexicographic approach improved the predictive accuracy of our random forest classifiers, when compared to the baseline split criterion based only on the information gain ratio. The lexicographic approach led to the choice of a different feature in a substantial number of nodes (about 12.5%), and performed overall better than the baseline in all 3 tested prediction scenarios. This shows that the added bias in favour of more recent features was a worthwhile change to the random forest algorithm, for our longitudinal datasets.

As future work, we plan to apply both approaches proposed in this article to other datasets and prediction scenarios. Furthermore, we plan to investigate different adaptations to data preprocessing and classification algorithms that can be applied specifically to longitudinal datasets.

ACKNOWLEDGMENT

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

REFERENCES

[1] A. Kaiser, "A review of longitudinal datasets on ageing," *Journal of Population Ageing*, vol. 6, no. 1-2, pp. 5–27, 2013.

[2] C. Ribeiro and A. A. Freitas, "A mini-survey of supervised machine learning approaches for coping with ageing-related longitudinal datasets," in *3rd Workshop on AI for Aging, Rehabilitation and Independent Assisted Living (ARIAL), held as part of IJCAI-2019*, 2019.

[3] 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." 2019.

[4] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[5] P. Diggle, P. J. Diggle, P. Heagerty, P. J. Heagerty, K.-Y. Liang, S. Zeger *et al.*, *Analysis of longitudinal data*. Oxford University Press, 2002.

[6] 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*, vol. 68, pp. 112–120, 2017.

[7] 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.

[8] 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)*. IEEE, 2017, pp. 739–746.

[9] C. Chen, L. Breiman *et al.*, "Using random forest to learn imbalanced data," *Univ. of California., Berkeley*, vol. 110, no. 1-12, p. 24, 2004.

[10] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

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

[12] A. A. Freitas, "A critical review of multi-objective optimization in data mining: a position paper," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 2, pp. 77–86, 2004.

[13] M. P. Basgalupp *et al.*, "Legal-tree: a lexicographic multi-objective genetic algorithm for decision tree induction," in *Proceedings of the 2009 ACM symposium on Applied Computing*, 2009, pp. 1085–1090.

[14] H. Deng, G. Runger, E. Tuv, and M. Vladimir, "A time series forest for classification and feature extraction," *Information Sciences*, vol. 239, pp. 142–153, 2013.

[15] A. A. Freitas, "Comprehensible classification models: a position paper," *ACM SIGKDD explorations newsletter*, vol. 15, no. 1, pp. 1–10, 2014.

[16] W. G. Touw, J. R. Bayjanov, L. Overmars *et al.*, "Data mining in the life sciences with random forest: a walk in the park or lost in the jungle?" *Briefings in bioinformatics*, vol. 14, no. 3, pp. 315–326, 2013.