NEW LONGITUDINAL CLASSIFICATION APPROACHES AND APPLICATIONS TO AGE-RELATED DISEASE DATA

A THESIS SUBMITTED TO
THE UNIVERSITY OF KENT
IN THE SUBJECT OF COMPUTER SCIENCE
FOR THE DEGREE
OF PHD.

By
Caio Eduardo Ribeiro
December 2021

Abstract

Traditional supervised machine learning techniques need to be adapted when applied to longitudinal datasets, due to their specific characteristics such as a large amount of missing data and the dependency between repeated measurements of the same variables. These adaptations range from data preprocessing techniques that maintain and use information from the underlying temporal data structure of longitudinal datasets, to algorithm adaptations that consider the temporal aspect of the data when making predictions.

In this thesis we focus on the classification task of supervised learning, in the context of longitudinal biomedical and health data from ageing studies. More specifically, we address the problem of predicting the diagnosis of age-related diseases, given several years of observations about each instance (individual).

In order to evaluate our proposed approaches for longitudinal supervised learning (described below), we created 30 longitudinal classification datasets. These datasets are comprised of data from the English and Irish longitudinal studies of ageing, which collect biomedical and self-reported health information on thousands of participants, over multiple waves carried out throughout the years.

Regarding supervised learning algorithms, we focus on decision tree-based algorithms, namely Random Forests (which learn an ensemble of decision trees) and a decision tree algorithm (which learns a single decision tree). These algorithms were chosen because they represent a good trade-off between predictive accuracy and interpretability, which is particularly relevant for our health application. Random Forests are known to achieve high predictive accuracy in general, and are partially interpretable (via feature importance measures), whilst decision trees are directly interpretable, although usually less accurate than Random Forests.

This thesis' main contributions are three new approaches for coping with longitudinal data in supervised learning (particularly classification). The first two

main contributions involve data preparation, namely missing value replacement and the construction of features representing temporal information in the data. These contributions are independent from the choice of classification algorithm to be applied to the longitudinal data, so they are widely applicable to longitudinal studies. The third main contribution involves algorithm adaptation, adapting decision tree-based algorithms to consider the temporal information in the data.

More precisely, the first main contribution of this thesis is the proposal of a data-driven missing value replacement approach to estimate the missing values in longitudinal datasets. The proposed approach performs a feature-wise ranking of an input set of missing value replacement methods, using known data as ground-truth to estimate the error rates of each method. Then it uses that ranking to choose the best missing value replacement method for each feature. Experiments have shown that this approach improved predictive accuracy in general, by comparison with several baseline methods for handling missing values.

The second main contribution consists of several types of constructed temporal features, which are are calculated (in a data preprocessing phase) using the repeated measurements of the original longitudinal features. These constructed features represent different types of temporal patterns that can occur in longitudinal datasets. The constructed features are then added to the original dataset, and used together with the original features when running any chosen classification algorithm. Experiments have shown that the constructed features benefited from datasets with more temporal data available, and that the added features overall increased the predictive accuracy of the Random Forest classifiers.

The third main contribution of this work is an algorithm adaptation approach for decision tree-based algorithms (more precisely, Random Forests and decision tree algorithms) applied to longitudinal data inputs. We adapted the node split function of such algorithms to consider two criteria, using a lexicographic optimisation approach. This approach first tries to select the best split feature at each tree node based on the features' information gain ratio, as the primary criterion. If, however, two or more features have about the same information gain ratio, as a tie-breaking (secondary) criterion, the algorithm prefers to select a more recent feature, since these are assumed to be more relevant for classification than older features. Experiments have shown that this lexicographic split approach led to increased predictive accuracy in general for the Random Forest classifier.

Acknowledgements

I would like to thank my PhD supervisor, Dr. Alex Freitas, for being an amazing source of support and insight throughout this process. Alex is an outstanding example of professionalism, and a lot of my growth is due to his guidance.

My family and friends in Brazil have also been a great source of support. It was easier to come study in a foreign country knowing that I had so many people in my corner, always ready to spend some time with me online and receive me back home with the same joy and warmth they always offered.

My father and my mother are the main reasons I was able to achieve this dream, and their unwavering love and encouragement kept me going in the toughest times. Uncle Murilo and aunt Soraya are two of the greatest party companions one could ask for, and their visits were definitely highlights of my time here. My two brothers helped me grow into a smarter person with the many philosophical discussions (and banter). There are too many people I would like to cite here, and I never forget how lucky I am that this is the case.

I am especially thankful to Belinha, who managed to be a home away from home even though we are technically one country apart. To Marcelo, who I grew up with but thankfully never too much. To Luan, for listening to my long rants and sending me equally long rants that are always thoughtful and entertaining. All my friends are awesome people whom I love and will be happy to keep in my life for the long run.

I have been lucky to also find amazing friends here in the UK. Fabio, Carol and the other Brazilians showed me how meaningful it is to have a community away from home. Jamie, Matt and the other friends in Cornwallis and Woolf showed me that nationalities do not matter when humans want to connect. Calvin and I developed a weird but great symbiosis system that helped us both grow into mostly adult people. You are all great, and I am excited to spend more time in

the UK!

In addition to the research I have done, my experiences teaching in the Computer Science course have been extremely rewarding. I am thankful for the colleagues and professors I worked with over the years, and to the many friendly students I had the opportunity to meet. I would also like to thank the School of Computing for providing the tools and infrastructure required for my research, but most importantly promoting a healthy and welcoming work environment where I made friends and developed my skills over the years.

The English Longitudinal Study of Ageing was developed by a team of researchers based at University College London, NatCen Social Research, the Institute for Fiscal Studies, the University of Manchester and the University of East Anglia. The data were collected by NatCen Social Research. The funding is currently provided by the National Institute on Aging in the US, and a consortium of UK government departments coordinated by the National Institute for Health Research. Funding has also been received by the Economic and Social Research Council. The Irish Longitudinal Study of Ageing was funded by Irish Life, Atlantic Philanthropies, and the Department of Health & Children of Ireland. The study is an inter-institutional initiative led by the Trinity College Dublin. I thank the designers and participants of both studies, that provided the data used in this thesis project.

Contents

List of Tables

List of Figures

Glossary

AD Alzheimer's Disease. 38

ADL Activities of Daily Living. 39

ADNI Alzheimer's Disease Neuroimaging Initiative (study). 38

AGG Aggregated Values (representation of longitudinal data). 41

ANOVA Analysis of Variance. 27

BRF Balanced Random Forest. 82

CHS-CS Cardiovascular Health Study Cognition Study. 39

CLHLS Chinese Longitudinal Healthy Longevity Survey. 39

FS Feature Selection. 18

HBP High Blood Pressure. 58

IADL Instrumental Activities of Daily Living. 161

IG Information Gain. 29

IGR Information Gain Ratio. 30

IR Class Imbalance Ratio. 59

KNN K-nearest Neighbours. 52

LOCF Last Observation Carried Forward. 20

MIMO Multiple-time-points Input and Multiple-time-points Output. 37

MISO Multiple-time-points Input and Single-time-point Output. 37

ML Machine Learning. 3

MVR Missing Value Replacement. 7

RF Random Forest (classifier). 8

RFs Random Forests (algorithm). 5

SepW Separate Waves (representation of longitudinal data). 47

SHIP Study of Health in Pomerania. 39

SIMO Single-time-point Input and Multiple-time-points Output. 37

SISO Single-time-point Input and Single-time-point Output. 37

SVM Support Vector Machines (algorithm). 44

UBB Undersampling Before Bootstrapping. 82

UDLI Union Disregarding Longitudinal Information (representation of longitudinal data). 43

UK United Kingdom. 6

UKLI Union Keeping Longitudinal Information (representation of longitudinal data). 42

Chapter 1

Introduction

Supervised machine learning (ML) techniques use training data to create a model able to make predictions about previously unseen data. In addition to the model's predictions, the models themselves can represent knowledge about the problem being studied. By applying these techniques to analyse real-world datasets, we can partly automate the knowledge discovery process and find patterns that can be used to reach meaningful conclusions about the domain problem.

Supervised ML is a traditional field with a large number of techniques, and it is applied to many research areas. However, ML applications to longitudinal data are under-explored, even though this type of data is becoming more prominent recently (Ribeiro et al. 2017). Because of its temporal nature, longitudinal data incurs challenges and opportunities that make it worthwhile exploring creating novel ML techniques and adapting existing ones, specifically for it.

1.1 Longitudinal Studies of Ageing

The study of human ageing is highly interdisciplinary, with research from various areas of knowledge such as biology, medicine, social sciences and economy, being conducted towards understanding the biological process of ageing and its impacts both on an individual and on a societal scale (Foos and Clark 2016). Currently, this area has been getting increased attention from the scientific community and governmental agencies, especially on countries that have lower birth rates and greater life expectation, where the populational ageing phenomenon is more accentuated.

It is estimated that the global proportion of individuals over 65 years of age will surpass 16% of the population by 2050 (raising from the current 11%) (United Nations and Social Affairs 2019). This ratio will be more pronounced in some countries (e.g., 25% in Europe and North America). The populational ageing phenomenon impacts the entire structure of society, including social security issues, as the ratio of working versus retired people declining can have severe social and economic implications (Lutz, Sanderson and Scherbov 2008). Naturally, this also puts a strain on health systems (Cheng et al. 2020). Thus, understanding the human ageing process is of interest to society as a whole, as it can guide the creation of public policies aimed at the older segment of the society, in addition to helping diminishing and treating cognitive losses and many age-related diseases.

Several countries have been running longitudinal populational studies of ageing, where they collect data on various aspects of the lives of older individuals, including physical and mental health, demographics, and socioeconomic aspects. The data generated by these studies is analysed to determine, for example, the reaction of a patient to a drug, the evolution of their ageing process, or their risk of developing a disease. Typically, population studies generate longitudinal datasets with a large number of variables (hundreds or thousands) describing each participant (instance), with relatively minor changes to the observed variables and participant cohort happening between waves (Kaiser 2013).

Analysing longitudinal data may offer insights, for example, on cause and effect patterns, on how an event affects a variable's values, or how a pattern evolves with time. Due to the high number of independent variables in these studies, ML applications are often needed for performing holistic analyses (i.e., considering hundreds or thousands of independent variables simultaneously).

1.2 Supervised ML Applied to Longitudinal Ageing Data

Longitudinal datasets are a special case of temporal datasets (i.e., datasets that store time-related variations of feature values), where the same set of instances (e.g., patients) is followed through a number of points in time, denominated waves. Longitudinal datasets from human ageing studies typically span several years, with

longer intervals of time between waves, and measure a large amount of features.

Applying supervised ML techniques to longitudinal datasets from human ageing studies could lead to new insights on how the ageing process is affected by variables from several different dimensions, and how these variables relate to each other. The longitudinal data analysis techniques most frequently used in the literature for these studies are based on classical statistics such as Structural Equation Modelling (Mueller and Hancock 2018), which are usually parametric (making strong assumptions about the data distribution) and often detect only linear correlations in data. By contrast, as mentioned earlier, the use of more flexible non-parametric, non-linear supervised machine learning (ML) techniques have been much less explored in the context of longitudinal data (Fabris, de Magalhães and Freitas 2017).

Intuitively, analysing longitudinal data requires using all information available simultaneously. This is corroborated in the literature by Hielscher et al. (2014) who argue that using repeated measures brings better results than considering, for example, only the most recent measurement of each variable. However, traditional ML techniques do not consider the temporal information in longitudinal data, creating a need for adaptations either on the data representation or on the ML algorithms themselves, in order to use information from the underlying structure of the temporal data to improve predictive performance.

1.3 Objectives and Contributions

For this research, we focus on the problem of predicting the diagnosis of age-related diseases (a binary classification problem), using longitudinal data (mainly biomedical and self-reported health data) collected over several years. Our aim is to create models capable of receiving data about unseen instances and making a prediction regarding whether the individual represented by that instance will develop any of the target age-related diseases in the target (last) wave. We focused on decision-tree based algorithms, namely Random Forests (RFs) (Breiman 2001) and C4.5 decision trees (glsDT) (Quinlan 1993). Hence, we are able to generate insights regarding what variables are more relevant for making those predictions, or how the variables are related to each other, promoting further research on this area.

The models were trained and evaluated using data from 30 real-world longitudinal datasets created and preprocessed for this research. The datasets were created from three data sources (10 from each source), namely the nurse visit and core questionnaires of the English Longitudinal Study of Ageing (ELSA), and the questionnaire from the Irish Longitudinal Study of Ageing (TILDA). The ELSA is a prominent ageing study based in the United Kingdom (Banks et al. 2019) that follows thousands of participants from UK households over several years. Its methodology is the basis for the TILDA, a more recent study that follows participants from Irish households (Kenny et al. 2010). Both studies focus on individuals aged 50 and over, and collect data about various aspects of their lives, including biomedical variables such as blood and mobility test results performed by professionals who visit the participants on given waves.

As mentioned earlier, longitudinal data has particular characteristics that may make existing ML techniques less effective. These are: a large number of features (variables), a high volume of missing data, the correlations between consecutive measures of a variable, and temporal patterns in the data (Diggle et al. 2013). In this thesis, we propose contributions in the process of learning knowledge from longitudinal data using supervised ML, directly coping with the characteristics of longitudinal data. We highlight the following specific objectives:

- *To propose a missing value replacement approach to cope with longitudinal data.*
- *To propose a preprocessing approach that adds constructed features as a representation of temporal patterns into a longitudinal dataset.*
- *To propose an adaptation to tree-based classification algorithms that focuses on longitudinal data inputs, which makes the classifier consider the temporal nature of the data.*
- *To evaluate the predictive performance of the methods proposed in the first three objectives in real-world data from longitudinal studies of ageing.*

The proposed contributions focus on these specific objectives, as described in the following Sections.

1.3.1 A data-driven missing value replacement approach

The first main contribution of this thesis is a data preparation approach to cope with the high volume of missing data that is characteristic of longitudinal studies of ageing. The proposed approach uses a set of missing value replacement (MVR) methods to estimate each missing value in the dataset. First, it performs a feature-wise ranking of the MVR methods using their average estimation errors for that feature, calculated in an internal cross-validation that uses the known values of the feature as ground truth. Then, the approach applies the MVR methods for each feature using the ranking as a priority list, going through the methods until it finds one that is applicable, thus replacing every missing value in the dataset.

For our experiments, we chose methods from basic statistics, a ML method, and methods devised specifically for longitudinal data. We performed two series of experiments to evaluate our proposed data-driven MVR approach. The first is a classifier-independent comparison that calculates the average estimation error and applicability (percentage of missing values replaced) for each method. The second is a classifier-dependent comparison of models created from datasets that used each of the MVR methods in our set, as well as a baseline of not performing imputation and letting the classification algorithm cope with the missing values during training.

1.3.2 Constructed temporal features for longitudinal datasets

The second main contribution is the proposal of a series of Constructed Temporal Features (CTFs), created from the original features in the dataset and added to it to increase predictive performance. The CTFs are created to directly represent possible temporal patterns in the data, such as monotonic increase/decrease over time, or how an instance's value compares to values from individuals with the same age. We perform experiments with six different types of CTFs, and three of them are novel contributions of this work.

Our proposal was to add all six types of CTFs to the longitudinal datasets in a data preprocessing step, prior to training the models. We performed experiments with adding each type of CTF individually, as well as adding all of them together. In the experiments, we compared three feature sets, namely a baseline of using only the original features, a feature set comprised only of constructed features,

and the proposed approach of combining original and constructed features in a single dataset.

1.3.3 A lexicographic bi-objective split for decision tree-based classification algorithms

The third and final main contribution of this thesis is an adaptation to decision tree-based classification algorithms for coping with longitudinal data. The adaptation adds a bias in favour of feature values measured in the more recent waves (time-points), following an intuitive notion that such recent feature values are more closely related to the target (class) variables, which are measured in the final wave of the dataset. This is done by changing the data-split function in decision tree-based classification algorithms in such a way that makes them bi-objective, so that in addition to considering the information gain ratio (or other entropy metric) as a measure of a feature's quality, it adds the time-index (wave id) of the feature as a tie-breaking criterion. That is, when the gain ratios of two candidate features are very similar, the data-split function chooses the feature that was most recently measured.

In order to determine when two candidate features have "equivalent" information gain ratios, we added a threshold parameter, defined by the user, so that if the difference between two gain ratios is smaller than the chosen threshold, the features are considered tied. However, to avoid the issues caused by having an added parameter, we also propose an automated data-driven threshold selection, which chooses a threshold value based on an internal cross-validation process with the training instances. Thus, the new parameter can be chosen automatically based on the available data.

The proposed adaptation is tested with Random Forest (RF) classifiers (Breiman 2001), which are ensembles of decision trees, and with decision trees learned by the well-known C4.5 algorithm (Quinlan 1993). We performed experiments evaluating the lexicographic split approach with and without the added aforementioned CTFs, first comparing the proposed automated threshold selection with fixed threshold values (representing user choices), and then comparing the standard split criterion with the proposed lexicographic split approach.

1.3.4 Evaluating the classification models on real-world data

Another contribution of our work is the fact that all our models are evaluated using four predictive accuracy measures, using the 30 datasets created from real-world data created for this thesis. The samples from the source studies are representative of the populations of the United Kingdom (ELSA) and Ireland (TILDA), and the chosen problem of predicting the diagnosis of age-related diseases is particularly relevant given the populational ageing phenomenon.

Therefore, in Chapter 6 we analysed the best classification models generated during our research, as a contribution to the study of human ageing. For this analysis, we created Random Forests and C4.5 decision trees for the datasets that had the best results in our evaluation, for each data source. For each of the selected models, we discuss how the most important predictive features are associated with the target variables in the literature, citing peer-reviewed medicine research.

1.4 Thesis Structure

Chapter 2 contains a background review on supervised ML, with a focus on the classification task, and on decision trees and Random Forests (the types of classification algorithm most relevant to this research). This review contains all the information the reader requires to comprehend our work, and various references to sources for deeper study.

Continuing the review of the literature, we surveyed supervised ML publications that used data from longitudinal studies. In Chapter 3, we include a review of related works, within the context of how longitudinal data is represented and coped with by supervised ML algorithms. In this Chapter we also propose a new taxonomy of approaches to cope with the temporal information associated with longitudinal data. This contribution can help researchers starting their own projects to organise the existing literature, and identify the most important works for their research.

In Chapter 4 we report the creation and preprocessing of 30 longitudinal datasets created from the English and Irish Longitudinal Studies of Ageing (ELSA

and TILDA, respectively), with the diagnosis of age-related diseases as their binary target variables. The main contribution in this Chapter is a novel data-driven missing value replacement approach that performs a feature-wise selection of the best strategy to cope with the missing data, based on an internal cross-validation process.

After preprocessing the datasets in the previous Chapter, we move on to enriching them with Constructed Temporal Features (CTFs) in Chapter 5. We propose 3 novel types of CTFs and experiment with adding 6 types of CTF, individually and simultaneously, to the longitudinal datasets.

Our final main contribution is an algorithm-adaptation approach presented in Chapter 6: a lexicographic bi-objective split approach for decision tree-based classifiers. The proposed modification adds the time-index (wave id) of a feature as a tie-breaking criterion when selecting a data-split feature in a decision tree, adding a bias in favour of more recent feature values. In this Chapter we also interpret the best models created in our research, as a human ageing study contribution.

Finally, in Chapter 7 we summarise the contributions of our work, and present our conclusions and suggestions for future research.

1.5 Publications Derived from this Research

The following papers were published or accepted for publication during the course of this research, based on our results. For each publication, we briefly mention its main contributions. The only publication that was not peer-reviewed in this list is the book chapter, which was an invited book chapter.

Journal publication

C. Ribeiro and A. A. Freitas. (2021). "A data-driven missing value imputation approach for longitudinal datasets". *Artificial Intelligence Review*, 30 pages. DOI https://doi.org/10.1007/s10462-021-09963-5. In this article we define a novel missing value replacement approach that employs a series of methods, including some devised specifically for longitudinal data, using the known information in the dataset to estimate the best option for replacing each missing data point.

Conference publications

C. Ribeiro and A. A. Freitas. (2020)."A New Random Forest Method for Longitudinal Data Classification Using a Lexicographic Bi-Objective Approach". *Proceedings of the 2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 806-813. IEEE. This paper defines an algorithm adaptation for tree-based classifiers, tested on the Random Forests algorithm, that changes the split feature selection function to consider temporal information in longitudinal data inputs. It also has a contribution about propagating predicted class values from earlier waves which is not mentioned in this thesis, as it was considered out of scope.

C. Ribeiro and A. A. Freitas. (2021). "Constructed Temporal Features for Longitudinal Classification of Human Ageing Data". *Proceedings of the 9th IEEE International Conference on Healthcare Informatics (ICHI'21)*, 7 pages. IEEE. In this paper we define six constructed features that explicitly represent temporal information in longitudinal data, to be added in a preprocessing step for any longitudinal machine learning application.

Short-papers published on workshop proceedings

C. Ribeiro and A. A. Freitas. (2019). "Comparing the effectiveness of six missing value imputation methods for longitudinal classification datasets". *Proceedings of the 3rd Workshop on AI for Aging, Rehabilitation and Independent Assisted Living (ARIAL) - held as part of the IJCAI-19 international conference.* 5 pages. This short-paper defines a data-driven comparison of different missing value replacement approaches that uses known values in the dataset as ground-truth. This was later expanded in the 2021 article mentioned earlier.

C. Ribeiro and A. A. Freitas. (2019). "A Mini-Survey of Supervised Machine Learning Approaches for Coping with Ageing-Related Longitudinal Datasets". *Proceedings of the 3rd Workshop on AI for Aging, Rehabilitation and Independent Assisted Living (ARIAL) held as part of the IJCAI-19 international conference.* 5 pages. This short-paper defines four methods of representing longitudinal datasets in machine learning applications, and how these representations affect the project.

Book chapter

F. Pereira, T. Oliveira, A. Duarte, J. Henriques, S. Paredes, T. Rocha, P. Carvalho, C. Ribeiro and A. A. Freitas. "Chapter 26 Machine learning in the context of better healthcare in aging" – *To appear in: "Aging: From Fundamental Biology to Societal Impact"*. Elsevier. This book chapter summarises some of the contributions in the papers mentioned earlier, namely the lexicographic split and the data-driven missing value replacement, in the context of healthcare applications that employ machine learning techniques.

Chapter 2

Background

In this Chapter we discuss the concepts and methods of supervised machine learning (ML) related to the work carried out for this thesis. This Chapter is organised as follows. In Section 2.1, we define the classification and regression tasks of supervised ML. Section 2.2 contains a more in-depth discussion of some of the specific challenges around classification problems, followed by a discussion about evaluating and comparing different classifiers in Section 2.3. Sections 2.4 and 2.5 review the two types of supervised machine learning algorithms that are relevant to our work, namely decision trees and random forests, respectively. Finally, Section 2.6 has our representation of longitudinal datasets as inputs for machine learning algorithm, used throughout this thesis.

2.1 Supervised Machine Learning

The task of creating algorithms able to learn a model from a set of training instances (examples) and then using the model to predict a target variable's value in a separate set of instances, called testing instances (unobserved during training) is named supervised machine learning (ML).

In supervised ML, a target variable can either have continuous values, which characterises a regression problem, or nominal (or categorical) values, characterising a classification problem. Models created by supervised learning processes can be used in many areas with complex problems, such as biology, finance and physics. In addition to the predictions of target variable values they provide, the models themselves can often be analysed for insight regarding the collected data

and relationships between variables (Jiang, Gradus and Rosellini 2020).

Supervised ML problems require that the datasets be composed of instances (records of cases, or objects) and features (characteristics of each object). The features must be the same for all instances, and typically they take either nominal (categorical values that can be ordered or unordered, which changes how the feature can be interpreted and treated) or numeric (often continuous) values (Quinlan 1993).

2.1.1 Regression

In regression problems, the algorithm creates a model from a set of instances and aims to correctly predict the value of a continuous target variable for testing instances. There are many applications for regression algorithms, such as weather forecasting, stock prediction, measuring the effects of drugs, etc. In this text, we will focus on linear regression methods, which produce a linear model as described by Equation 2.1, since they are more relevant to our work.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_d x_d \tag{2.1}$$

Where y is the target (dependent) variable. $X_1, ..., X_d$ are the features (independent variables) and d is the number of features in the dataset. $\beta_1, ..., \beta_d$ are the weights (coefficients) applied to the corresponding features, and β_0 is a constant.

The main assumption made by a linear regression model is that the features used to predict the dependent (target) variable's value are linearly independent. That is rarely the case in practice, and it is also difficult to prove that a set of features is the best to predict a given target variable. When learning a linear model, the goal is to find a set of features that are as independent from each other as possible and, at the same time, as related to the target variable as possible.

The most common estimate of the predictive accuracy (or error) of a regression model is through its root mean squared error (RMSE), depicted in Equation 2.2. The RMSE is measured in the same scale as the target variable, which can help interpretation, but this can make the results misleading when comparing

regression models for different problems.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}} \tag{2.2}$$

where y_i is the actual value for the ith observation (instance), $\hat{y}_i$ is the predicted value, and n is the number of instances.

A well-known regression analysis method, which is mentioned in the Chapter 3 (where we discuss the related works), is the Lasso (least absolute shrinkage and selection operator), which learns from the data a linear model in the form of Equation 2.1, by solving the optimisation problem in Equation 2.3 (Tibshirani 1996). The Lasso works under the assumption that, among a large set of features used as predictors to determine a target variable's value, only a small subset of them are actually important for that prediction. The method adjusts the feature's weights (the β coefficients in Equation 2.3), ultimately making many of them have no prediction power (zero weight). This is achieved by using the constraint in the last part of Equation 2.3, which forces the sum of the absolute values of the regression coefficients to be less than or equal to a threshold t. The Lasso performs a feature selection that achieves, in general, good predictive accuracy if the linear regression model is appropriate for the data, while simultaneously making the model more interpretable, since fewer variables are considered.

$$\min_{\beta_0,\beta} \left\{ \frac{1}{n} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_i x_i)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{d} |\beta_j| \leq t \tag{2.3}$$

The fused Lasso is a variation of this method devised to account for spatial or temporal information in the data (Tibshirani et al. 2005). By assuming that the features in a dataset can be ordered in a meaningful way, the fused Lasso adjusts the weights of the features and of their temporal (or spatial) successors, maintaining the relation between those features. In other words, if a given feature is penalised by the fused Lasso weight adjustment, the features that are temporally (or spatially) related to it (such as a repeated measure for the same variable in a longitudinal dataset) are adjusted accordingly.

2.1.2 Classification

Let the target variable of a supervised ML problem take a known set of nominal values, called labels. The classification task consists of, given a training set of labelled instances, creating a model (called classifier) for correctly predicting the label of an unknown-class instance – i.e., an instance in the testing set.

The most common configuration of a classification problem is the binary classification, where there are two possible outcomes, usually called positive or negative, and each instance is assigned one class label. Other configurations, which have different challenges and specific heuristics and algorithms proposed for them, are multi-class problems, with more than two labels, and multi-label classification problems where each instance is assigned multiple classes (Tsoumakas, Katakis and Vlahavas 2009). Classifiers can be applied in a wide variety of pattern recognition problems in different areas, such as face detection, handwriting recognition, predicting protein function, etc (Bishop 2006).

In summary, classification models must make a decision between distinct labels for instances in the testing set, using the information from the feature (independent variable) values of these instances. There are many aspects that affect the performance (predictive accuracy) of a classifier, some related to the data collection and preparation process, and others related to the chosen classification algorithm, for learning the model. This thesis focuses on the binary classification task of supervised machine learning, so we will go more in depth about the intricacies of classification problems, including how to evaluate and compare classifiers, in the following Sections.

2.2 Challenges for the Classification Task

In this Section we discuss various aspects of supervised learning that make the classification task challenging. Each of these is the focus of entire research projects, but all of them need to be addressed in some way for any classification project. In Chapter 4 we discuss in detail how each of these challenges is addressed in our project.

2.2.1 Data collection

For real-world data, the process of creating a dataset for use in supervised ML starts at the process of collecting data. This collection can be from automated storage from computer programs (e.g., business transactions in a company), measurements taken by specific tools (e.g., temperature and pressure measurements in a meteorology station) , or human input (e.g., data from interviews, exam results or questionnaires). The focus on this work is on human input data from longitudinal studies of ageing, mainly taken from questionnaires filled by health professionals or by the participants of the studies.

A classification model needs to be trained with an adequate number of training instances to be robust enough to be considered reliable when predicting the target variable's value for unknown instances. Naturally, more complex problems often require more data to build a reliable predictive model. In addition, the quality of the instances also influences in the quality of the model. The data collection process is designed to provide enough data about the problem domain to allow for in-depth studies about the variables in the database.

For real-world datasets, it is common to have imprecise values (noise) for some instances, in all features including the target variable, due to errors in measurement or input (Bolón-Canedo, Sánchez-Maroño and Alonso-Betanzos 2015). The values of the target variable are particularly important, as the entire classifier is built for predicting them. If the 'ground truth' is inaccurate for a large number of the training instances, the model's predictive performance is hindered, and the classifier might be rendered useless. Therefore, minimising noisy data, particularly in the target variable, is an important part of creating the dataset for machine learning.

The data collection process generates databases that are used as sources for the creation of ML datasets (some projects require multiple data sources to create a single dataset). These datasets should go through a cleaning (removing duplicates, insertion errors, outliers, etc.) and, when possible, enhancing (missing values replacement, transformations in the data, etc.) process, to improve the classification model's reliability (García, Luengo and Herrera 2015).

2.2.2 Feature selection

In the data collection process, as acquiring and storing data becomes cheaper, the database designers tend to measure a large number of variables related to the domain problem, which typically results in high dimensional databases. This adds unwanted complexity to the problem and may hinder the performance of the ML algorithms (Verleysen and François 2005). The ideal dataset for ML is composed only of features highly related to the target variable, and these features should be as independent of each other as possible.

Feature selection (FS) is the task of finding the optimal subset of features for an input dataset, given a feature subset evaluation function to be optimised (typically based on a feature subset's predictive power). In this Section, we discuss FS as a data preprocessing task, rather than the "embedded" FS approach, where features are selected during the construction of the classification model.

In the context of ML, the mains gains of performing this task come from the added interpretability and improved prediction performance of the model (Bolón-Canedo, Sánchez-Maroño and Alonso-Betanzos 2015). A well performed FS can significantly improve the predictive power of a model, if the right features are kept and irrelevant or uninformative features (i.e., too noisy features, or features that are too correlated with others) are correctly identified and removed. Feature selection is also used to reduce chances of overfitting, since having fewer irrelevant features in the dataset reduces the chances of the model incorporating unnecessary information.

The FS task can be performed manually, with the help of descriptive statistics and domain knowledge, but there are also many different automated FS techniques proposed in the literature. Algorithms for FS can be categorised as filters, wrappers, or embedded techniques (Chandrashekar and Sahin 2014). Filters apply a quality function independent from the ML algorithm, and are typically faster but make strong assumptions about the data. Wrappers use the performance of models generated with different subsets to compare them, which is computationally expensive but makes no assumption about the independence of predictive features. Embedded techniques reduce the feature set as the model is created, as part of the ML algorithm that generates the model.

Feature selection techniques can be adapted to better handle the specific aspects of longitudinal datasets (Tsagris, Lagani and Tsamardinos 2018). For example, we expect correlation on longitudinal data between features representing the same variable, measured in different time-points, so the FS technique needs to take that into account. Moreover, the inclusion of the time-axis into the dataset often leads to a high-dimensionality scenario (Adhikari et al. 2019; Ribeiro et al. 2017), which can be prohibitive for some FS techniques, but also increases the need for an initial reduction of the feature set.

Finding an optimal subset is a combinatorial task, so FS is a NP-Hard problem (Amaldi and Kann 1998). The FS task can be the focus of an entire research, as it is a challenging task that can generate new insights on the data, and further understanding of complex problems.

2.2.3 Missing data

It is common in ML datasets to have some unknown values of features for some instances, due to various different reasons. Instances with missing values add uncertainty to the data, and can hinder the prediction capabilities of the supervised ML method. Many ML algorithms have some built-in form of dealing with missing data, but we can also employ some data preparation techniques to estimate values for (or remove) missing values in a dataset. Note that, sometimes, the presence of a missing value can provide useful information, e.g. the result of a medical test (feature) may be missing because the patient does not need or cannot do the exam.

A simple way to handle missing values in datasets is to delete instances or features that have too many missing values, reducing the dataset size. However, this approach also throws away known values in the deleted instances or features. For a deletion strategy, as a case by case analysis is often not possible, one would need to establish a threshold on the maximum acceptable frequency of missing values for either an instance or feature, for example, deleting features with more than 20% of missing values.

Another common strategy to handle missing values is to replace them with an estimated value based on other information present in the dataset. On a non-temporal dataset, a missing value is typically replaced by the mean or mode of

the known feature values, for numerical and nominal features, respectively. On a longitudinal dataset, there is the added possibility of using information from other time-points, such as the last observation carried forward (LOCF) method, which replaces a missing value by the last known value for that feature, from the same instance, in past time-points of the dataset (Minhas et al. 2015). It is also possible to apply a supervised ML algorithm to estimate missing values, by setting a feature with missing values as the target variable, using the other features as predictive features, and using the predictions as estimations. However, this approach is very computationally expensive when the dataset contains many features with missing values (Rahman and Davis 2013).

Choosing a strategy for estimating missing data in a dataset is a challenge 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 et al. 2013; 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 et al. 2013; Mallinckrodt 2013); c) the proportion of instances with missing values; d) the availability of information that can be used to make better imputation. Therefore, when a dataset has a large volume of missing data, one should dedicate some time and effort in selecting the best way to handle this issue.

2.2.4 Overfitting

A major issue faced by classifiers is the possibility of overfitting to the training data, which hinders the model's ability to generalise, increasing its prediction errors when classifying previously unseen testing instances (Bramer 2007). Overfitting also adds complexity to the model, due to irrelevant features being added as predictive features, which makes the model harder to interpret, even when it does not harm the model's performance (Fawagreh, Gaber and Elyan 2014).

Overfitting is easily identified when the model performs well on its training data, but significantly worse on testing data. An overfitted model might lead to the misclassification of instances that are not similar enough to any instance in the training set.

This can happen for different reasons, many of those are related to the data collection task, as discussed earlier. The model's construction can be affected by noisy data in the training set, those being either irrelevant features, instances that are outliers or a wrong value in the target variable. Furthermore, the training set is a sample of the real population relevant to the problem, and that sample might not be representative of all possible cases or scenarios.

However, in the context of classification problems, a common cause for overfitting is class imbalance, where the class distribution is highly uneven. Imbalanced classes can introduce bias in favour of the majority class, because of the ways models are constructed to maximise quality measures such as accuracy (i.e., a guess for the majority class has a higher chance of being correct, so the model is more likely to take it). In real-world datasets, it is common that the class which is the most important to predict is the one with fewer available instances in the training set, such as diseased patients, fraudulent clients, defective machines, etc. Therefore, it is important to handle the class imbalance, either in the design of the algorithm or in a data preprocessing stage (Kaur, Pannu and Malhi 2019).

There are several strategies to reduce overfitting, implemented in classification algorithms either during the creation of the model or as a posterior adjustment, such as pre-pruning and post-pruning in decision trees, discussed later on. Importantly, there is also the concern of underfitting a model to the training data, where the opposite problem happens: the model becomes so generalised that it does not reflect the underlying data relationships represented in the training set.

2.2.5 Class imbalance

Many binary classification problems involve datasets where the number of instances in the minority class represents only a small portion of the available data, but correctly classifying an instance as a part of the minority class is the most important aspect of the problem. There are many methods devised specifically for this type of problem, sometimes referred to as rare-event mining (Haixiang et al. 2017), and the most common method for the classification task of machine learning is to manipulate the training data in an effort to reduce the class imbalance. Class imbalance handling methods can be summarised as follows. By artificially changing the proportions or misclassification cost of instances of each class in the

training set, we skew the classification algorithm into generating models that put more priority to the minority class, making it more likely to classify previously unseen instances as minority class instances. Naturally, the test set should not be changed, as it needs to represent the real observed proportions in the data.

The proportion of instances of each class that will be represented in the balanced training dataset has to be chosen apriori, and is typically 1:1 (for each instance of the minority class, one instance of the majority class remains in the dataset). There are two base strategies for changing the class ratio in a training dataset: reducing majority class instances (undersampling) or increasing the number of minority class instances (oversampling) (Kaur, Pannu and Malhi 2019).

Undersampling is a class balancing approach that consists of (usually randomly) removing instances of the majority class from the training dataset, to reduce bias in favour of the majority class. The model is trained with the balanced dataset and then validated in a test set with the original (real) distribution of instances from each class, hopefully performing better due to the reduced bias. The trade-off for undersampling approaches is that, by removing instances from the majority class, the classifier creates models without using all the information available in the training data. Instances ignored by the undersampling process potentially represent relevant information for the creation of the model.

On the other hand, oversampling methods increase the number of minority class instances, reducing the class imbalance of the dataset without removing majority class examples. This can be achieved either by re-sampling instances of the minority class (i.e., creating copies of them in the training set, which is the same as increasing their individual weight) or by artificially creating new minority class instances, using existing information. The trade-off for oversampling approaches is that the added minority instances can negatively impact performance. Copied minority class instances may overfit the model, reducing the classifier's ability to generalise for unseen instances, and added synthetic instances might not accurately reflect the reality of the problem being studied.

Some studies compared different approaches for handling class imbalance in tree-based classifiers, and concluded that both undersampling and oversampling can be effective, in general (Drummond, Holte et al. 2003; López et al. 2013; Yap et al. 2014). One strategy to mitigate the downsides, and combine the advantages, of applying undersampling and oversampling methods is to combine them into

a hybrid approach (Effendy, Baizal et al. 2014). Hybridising two or more class imbalance handling methods requires a decision of how each method will affect the class imbalance of the training set, with the classical approach being combining an oversampling method to raise minority class instances to 50% of the initial dataset's size, and an undersampling method to reduce majority instances down to 50% of the initial dataset's size. This results in a dataset with the original number of instances, but with a class ratio of 1:1.

The datasets used in our experiments have a severe class imbalance ratio (IR). We chose to investigate variations of the undersampling strategy, as described in Section 4.7.1 to mitigate this issue. We chose undersampling instead of oversampling as the former is more computationally efficient. In addition to that, we also wanted to avoid creating artificial instances in biomedical and health data datasets. Then, we perform experiments in our data preprocessing Chapter, comparing two undersampling approaches for Random Forest (RF) classifiers.

2.3 Evaluating and Comparing Classifiers

When choosing strategies in a classification task, we need to be able to compare different classification models. There are three main considerations when choosing between different options: how efficient the algorithm is for training a model and classifying new instances, how interpretable the model is and, often most importantly, the accuracy of the predictions made by the model.

Efficiency can be an important consideration for applications with large volumes of data to be analysed, or high frequency of use of the model. If a model needs to constantly make predictions, for example in an on-line learning problem, the time it takes to classify previously unseen instances becomes more important. Similarly if the training time increases exponentially with the dataset size, an algorithm may not be recommendable for some problems, even though its predictions are more accurate.

Regarding model interpretability, some algorithms generate "black box" models that, although accurate, are very difficult to be interpreted by the user. This may reduce trust in the model, and users that do not understand how a prediction is made might disregard them entirely (Freitas 2014; Caruana et al. 2015). As mentioned earlier, analysing the models themselves can bring insights into the

complex problems where ML is applied, such as highlighting important features for a classification problem. Therefore, classification algorithms that generate interpretable models are generally preferable (Rudin 2019), especially for recommendation applications where the user's trust in the model is an important factor.

Finally, the quality of a classifier is determined by how accurate its predictions are for unseen instances. The point of keeping the testing instances from the classifier, to then compare its predictions of test data to the ground truth, is that we hope that the model will be able to match its testing performance in the future, when unlabelled data needs to be labelled. Naturally, this depends on how representative the sample is, and how much the problem changes in the future samples.

2.3.1 Measuring predictive performance

In order to evaluate the predictive performance of a classifier, we look into its predictions on the testing set, for each class label separately (i.e., making a class label the positive class). Considering a binary classification problem, a prediction can have one of four outcomes:

- **True positive (TP)**: The model correctly classified an instance as positive.
- **False positive (FP)**: The model wrongly classified an instance as positive.
- **True negative (TN)**: The model correctly classified an instance as negative.
- **False negative (FN)**: The model wrongly classified an instance as negative.

Naturally, the goal of a predictive model is to increase the rates of TP (True Positive classifications) and TN (True Negative classifications), which signify correct predictions, as much as possible. We can analyse the TP and TN rates individually, as local performance metrics, but it is also important to consider both types of prediction at once, with global performance metrics that consider both classes simultaneously. Several predictive performance measures are based on the above rates, and in this thesis we use the following four evaluation metrics (Japkowicz and Shah 2011):

- **Sensitivity (or Recall)**: a local metric of the true positive rate (given by Equation 2.4, where # denotes "the number of"). For problems where false negatives are the least desirable outcome, such as clinical diagnosis applications, the ML algorithm needs to maximise mainly Sensitivity.
- **Specificity**: a local metric that represents the true negative rate (given by Equation 2.5). It is a complementary measure to Sensitivity.
- **Accuracy**: the fraction of correct predictions made by the model over all predictions (given by Equation 2.6). This is a widely used global performance metric, however in highly imbalanced datasets the majority class has a much bigger impact on Accuracy, which can mask bad results for the minority class in a model.
- **GMean**: The geometric mean between Sensitivity and Specificity (given by Equation 2.7). This is another global performance metric, but it gives the exact same weight to both classes regardless of the class distribution in the data.

$$Sensitivity = \frac{\#TP}{\#TP + \#FN} \tag{2.4}$$

$$Specificity = \frac{\#TN}{\#FP + \#TN} \tag{2.5}$$

$$Accuracy = \frac{\#TP + \#TN}{\#TP + \#TN + \#FP + \#FN} \tag{2.6}$$

$$GMean = \sqrt{Sensitivity * Specificity} \tag{2.7}$$

To better illustrate the aforementioned issue with the Accuracy metric consider, for example, a case where 90% of the testing instances belong to the positive

class, and 10% to the negative class. In this case, a classifier that simply guesses the positive class for every testing instance (regardless of its features' values) would have a Accuracy of 90%, even though it has no predictive power. Its GMean value would be 0, however, correctly reflecting this issue. Note that because GMean is calculated as a square root of a multiplication, it changes the scale of the values such as a GMean value close to 0.5, for example. has a different meaning than it would for Accuracy. This is desirable because of the effect is has on examples such as the one mentioned here, which keeps overfitted models from appearing successful, but it is important to remember this change when interpreting GMean results.

The area under the ROC curve is another quality measure frequently used to evaluate classifiers. The true positive rate (Sensitivity) is plotted in the Y-axis against the false positive rate ($1 - \text{Specificity}$) in the X-axis, generating the Receiver Operating Characteristics (ROC) curve, as a threshold for the probability of the positive class is varied. The ROC curve can be used to visually demonstrate the ability of a predictive model to distinguish between two classes. The biggest the area under the ROC curve, the better the model (Centor 1985; Flach 2016).

2.3.2 Statistical tests for comparing classifiers

In addition to using predictive performance metrics such as Accuracy and GMean, it is useful to perform statistical tests to determine whether the difference in performance between classifiers is statistically significant or not.

Statistical tests are used to increase our confidence that observed results did not happen due to chance. They often output a measure of probability called *p-value*, which is compared against a significance level α, chosen by the user, to determine whether the null hypothesis of the test can be rejected ($p\text{-value} < \alpha$) or not. In most statistical comparison tests the null hypothesis is that the samples being compared are from populations with the same distribution. In our context, this would mean that the classifiers had equivalent performances. Thus, if the null hypothesis of a test is rejected, we can claim with a confidence level of $1 - \alpha$ that the classifiers' performances are not equivalent (Demšar 2006).

There are various tests in statistics that can be used for this type of comparison, and their adequacy for a given situation depends mainly on the data

distribution, sample size and method, and dependence relationships between variables (Japkowicz and Shah 2011). Over the course of this thesis, we perform experiments comparing different techniques using multiple (up to 30) longitudinal datasets. For all our comparisons, we chose non-parametric rank-based tests to avoid having to 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. The tests used in this thesis are the following.

When comparing more than two approaches at once, we use 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.

For Chapter 4, the chosen post-hoc non-parametric test was the Wilcoxon signed-rank text (Wilcoxon 1992). This test is traditionally used to compare only two approaches, but we used it in a pairwise comparison for these experiments because we focused on comparing our proposed approach against each of the others. Note, however, that when performing multiple comparisons simultaneously, it is recommendable to adjust the target α value to avoid getting significant results due to chance. This is because, over multiple consecutive comparisons, a significance level of, say, 95% will eventually yield a false result. So, in these tests, we used Holm's procedure for multiple tests (Holm 1979). In essence, the procedure adjusts the α value for each pairwise comparison, based on the number of tests being done and the rank of each comparison's *p-value* (i.e., the α value gets closer to zero as the number of consecutive comparisons increases, reducing the chances of wrongly assigning significance in a comparison due to chance).

The post-hoc non-parametric test we chose for the multiple comparisons with significant Friedman *p-values* in Chapters 5 and 6 is the Nemenyi test (Nemenyi 1962), which is more suitable for comparing multiple approaches simultaneously. We use the Nemenyi test to perform a pairwise comparison of the different classifiers, and determine which pairs had significantly different results.

2.4 Decision Tree Algorithms

One of the traditional types of classification methods are decision tree algorithms, which produce a graphical classification model in the form of a tree (Quinlan 1993). In a decision tree, each node corresponds to either a feature (internal node) or a class label (leaf node). An example is shown in Figure 2.1. Each edge coming out from an internal node corresponds to a path for a value or range of values of that node's feature. When classifying an instance, we start at the root node and, according to the value of the root node's feature, follow one of the possible paths further down the tree, testing the values of the features in internal nodes to choose the next path, until a leaf node is reached and the class is decided.

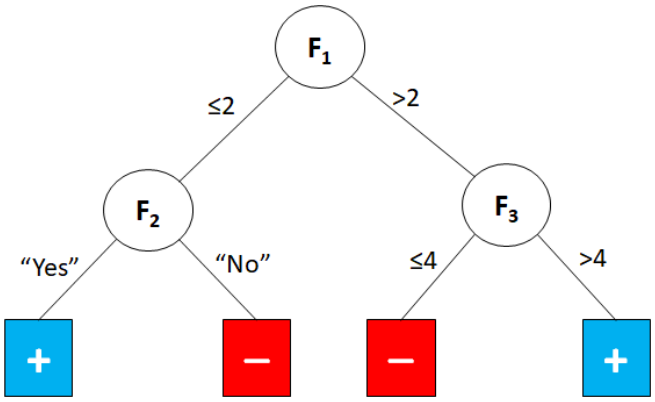

Figure 2.1: Decision tree example.

The decision tree learning strategy belongs to the divide-and-conquer paradigm, which breaks a complex problem into smaller sub-problems recursively, until the sub-problems are small enough to be solved, and then combines the solutions to the sub-problems into a final solution to the initial problem. When building a tree from a training set, the root node contains all training instances, and at each subsequent branching, the training set is divided into subsets of instances which belong to each path of the tree, therefore subsequent nodes are trained with fewer instances. This reduces training time, but also reduces the reliability of the information found at lower levels of a decision tree, because they are based on less evidence (information from fewer instances). Hence, among two or more trees that are consistent with the training set, in general the smallest tree is preferred.

The problem of finding the smallest decision tree consistent with a training

set is NP-Complete (Hyafil and Rivest 1976), which creates a need for heuristic approaches for tree-building. Most decision-tree algorithms are non-backtracking greedy methods. That means that the decisions about which feature to use for an internal node and how to partition the data based on that feature's values is made based on the local training set at the current node, and is not revisited at any point. That decision is usually made on the basis of maximising some local measure of predictive power such as the Information Gain (IG) (Quinlan 1993).

In addition to being fast, decision tree algorithms have the advantage of not making assumptions about the distribution of feature values (Quinlan 1993).

2.4.1 Interpreting a decision tree

A great advantage of decision trees is that they are easy to interpret – as long as the tree is not too large. The embedded feature selection performed by the tree constructing method makes it so that typically a decision tree contains only a subset of the input features. This makes interpreting the model easier, since the chosen features (and cutting points chosen by the algorithm to split the data) are more relevant than those left out by the algorithm (Freitas 2014). In addition, broadly speaking, features at higher nodes are more relevant for classification than features at lower nodes, since the former tend to be used to classify more instances.

Since there is only a single path that leads from the root node to each leaf node of the tree, every tree can be represented by a set of classification rules - one rule for each leaf node (Quinlan 1987a). Classification rules produced by decision trees have the form: *IF (condition) THEN (class)*, where the condition is the set of tests performed in all nodes belonging to a path that leads to the leaf node with a given class. This representations loses the natural hierarchy of the graph representation of the decision trees, so we no longer know which features were more or less relevant to reach the conclusion.

Bologna and Hayashi (2018) compared rules generated by 3 different supervised ML techniques on datasets from 25 binary classification problems, measuring the overall fidelity (degree of matching between the model's and rules classifications) and rule size of the generated set of rules. They concluded that a decision-tree based technique, Boosted Shallow Trees (Vezhnevets and Vezhnevets 2005), generated the set of rules with overall the smallest size and the highest fidelity.

2.4.2 Split evaluation functions

Decision tree algorithms use a split evaluation function to select the best feature at each internal node and to decide how to best split the feature values into paths that will lead to an accurate classification of the instances. These split evaluation functions typically analyse, for each possible branching, what is the class distribution of the resulting subsets of instances that would be created. The chosen branching is the one that divides the dataset into subsets such that, in each subset, as many instances as possible belong to a single class.

A popular split evaluation function is the Information Gain (IG) measure, which is based on entropy, a measure from information theory that represents the amount of uncertainty in the data. For a classification problem, this uncertainty is the presence of more than one class in a dataset, so a subset of data that contains only instances of a single class would have no entropy, whereas a subset with the same ratio of instances of each class would have the highest possible entropy value.

Entropy-based splits have the property of continuously decreasing the entropy of the subsets, meaning each branching will result in subsets with an entropy value that is lesser than (or equal to) the previous one, and the greater this difference, the greater the gain from that split. The IG is a calculation of how much gain a branching on a given feature (on a given split point) will bring, and the algorithm chooses the split with the greater IG at each branching. The recursive branching process of creating a decision tree stops when the entropy of all resulting subsets is zero (Quinlan 1987b).

An important weakness of the IG is that it is biased towards features with more values, as those are more likely to be able to divide the data with small subsets with low entropy (Quinlan 1993). For this reason, that measure was modified into the Information Gain Ratio (IGR), which is the measurement used in the split function for our experiments. The IGR avoids the aforementioned bias of the IG by dividing it by a correction factor that takes into account how many data subsets will be created by partitioning the data in the current node based on the values of the selected feature. However, this correction makes IGR favour the creation of unbalanced trees, where the depths of some branches can be much greater than others (Harris 2002).

2.4.3 Decision tree pruning

The decision tree construction process recursively partitions the dataset until each subset in a partition contains instances of a single class, or until it is not possible to make any improvements on the partition. Naturally, this results in complex trees that are prone to overfitting the training data. This especially hinders the tree's generalisation capabilities when there is noise in the training data, because the tree will incorporate noisy data in its structure. As mentioned earlier, nodes at lower levels of a decision tree are usually created with smaller subsets of the training set, reducing their reliability.

Tree pruning techniques were devised to reduce the size of decision trees, improving their generalisation capability by reducing overfitting. Decision trees may be pruned during (pre-pruning) or after (post-pruning) their creation process.

The pre-pruning approach aims to stop a branching process during the creation of the tree, based on some condition. Some classic pre-pruning techniques are to stop a branching if the size of the instance subset that will be created for a path is smaller than a threshold value, or the tree has reached a maximum depth (Bramer 2007). When a branching is stopped, the branch is replaced by a leaf node, which will predict, for new instances, the most frequent class at that node.

The post-pruning approach involves analysing a tree after its creation and replacing subtrees by leaf nodes, or smaller subtrees, in a way that roughly maintains the classification accuracy of the tree for the training set (typically, a threshold is defined to decide how much error can be introduced by pruning). This strategy is slower than pre-pruning, but more reliable, since it performs the pruning using information from the complete tree, so that the impact of a change can be better evaluated (Quinlan 1993).

It is worthwhile to note that any pruning naturally incurs in a reduction of the fitting of the decision tree to the training data. Schaffer (1993) argues that pruning strategies introduce a bias favouring smaller trees into the algorithm, and that choosing an inappropriate strategy may actually make its predictive performance worse.

The goal of pruning is to make the tree better at predicting the class of new instances, so using a validation set (part of the training set not used for building the tree) is one approach to evaluate the success of a pruning operation. However, this approach reduces the number of training instances available to build the

decision tree. One pruning technique using this approach is the Reduced-error pruning, proposed by Quinlan (1987b).

2.5 Random Forests

2.5.1 Principles of ensemble learning

According to Condorcet's jury theorem (Condorcet 1785), under certain assumptions, a jury composed of independent voters deciding on a problem tends to reach the correct decision (with a probability tending to 1) as the number of individuals in the jury grows to infinity. Ensemble learning borrows from that concept, combining different predictive models to solve a problem. Ensembles of classifiers have been shown to outperform single-classifier models in several empirical studies (Yan and Goebel 2004; Brazdil et al. 2008; Hosni et al. 2019).

An ensemble's performance is dependent on two main aspects: the accuracy of the base learners combined to form it, and how diverse their predictions are (Zhou 2012). The diversity aspect influences how much gain we can get from combining classifiers, meaning an ensemble of correlated classifiers shows less improvement on overall accuracy than one of independent classifiers. The task of generating classifiers that make predictions that are as different as possible, for the same problem and from the same training data, is challenging, especially considering that we also want each classifier to be as accurate as possible. This trade-off between the accuracy of each classifier and the diversity of the ensemble needs to be addressed by the ensemble method.

After generating the set of base classifiers there is also the task of combining their results to get the best performance. The simplest combination methods for a nominal output is to use a voting system, where the majority vote decides the result for the ensemble (averaging is the equivalent of this strategy for a numeric output). There are also more advanced methods of combining results, such as weighted and probability-based votes (Sagi and Rokach 2018).

2.5.2 The Random Forest algorithm

Decision forests are ensembles of different decision trees created from a dataset. The main challenge for learning a decision forest is how to obtain the right variability of the decision trees that compose it, which will increase the generalisation capability of the model. One of the most used methods of learning a decision forest is the Random Forests (RFs) algorithm, introduced by Breiman (2001). In RFs, the ensemble diversity is obtained in two ways:

- Training each decision tree with a different data subset, obtained by randomly sampling instances with replacement from the full dataset. Each decision tree of a random forest will have a dataset with about 63% of the instances in the original dataset represented at least once. The remaining instances, named out-of-bag (OOB) instances, can be used as a test set to evaluate that decision tree.

- At each internal node, the tree randomly samples a subset of the features of the dataset, so that the split evaluation function selects the best feature among those.

By adding that controlled variability into its trees, the RF is able to achieve a good generalisation capability without the need to apply tree pruning methods, which makes the computational complexity of the RF significantly smaller than ensembles of pruned decision trees. After the decision trees are trained, the ensemble can be used to classify a new instance by first classifying that instance using each decision tree, and then choosing the class voted by the majority of the decision trees in the RF.

The RF algorithm has two main parameters (Touw et al. 2012): *ntrees*, which is the number of decision trees in the forest, and *mtry*, which is the number of features randomly sampled in each node, to be evaluated using the split evaluation function. Two often used ways to set a value for *mtry* are using the square root of the number of features in the dataset d, or using $mtry = \lfloor log_2(d) \rfloor + 1$.

The original RF algorithm utilises the Gini Index heuristic, shown in Equation 2.8 as its split evaluation function. In the classification task, the Gini Index measures the class impurity in a set of instances (i.e., how much the set deviates from a perfect distribution). For a decision tree branching, the algorithm chooses

the split which decreases the class impurity of the dataset the most. As with the IG measure, the Gini Index has a bias in favour of continuous features and nominal features with many values. In addition to that, it was also found to be susceptible to be skewed toward the majority class (Flach 2003).

$$\text{Gini}(t) = 1 - \sum_{i=1}^{N} P(C_i|t)^2 \tag{2.8}$$

Where t represents the branching on a given feature, N is the number of class labels in the dataset, and $P(C_i|t)$ is the probability of the instance belonging to class i given that the branching t is carried out.

In general, the RF method achieves good predictive accuracy values when compared to other state-of-the-art supervised ML techniques. Fernández-Delgado et al. (2014) performed experiments comparing 17 families of classifiers (179 classifiers in total) on 121 datasets, and concluded that the RF family obtained overall the best predictive performance. Another important advantage of the RF is the possibility of estimating the importance of each feature in predicting the class, which adds interpretability to the model, and will be further discussed later.

RFs handle well datasets with a high ratio of features/instances, which are prone to overfitting. Scornet et al. (2015) have shown that RFs are able to adapt to sparse frameworks. The authors claim that since, the RF selects splits mainly among the most informative features, the irrelevant features have little impact on their performance. In addition, ensemble methods allow us to have several classifiers working in the same problem, which also reduces the problem of lacking adequate data, and decreases the risk of obtaining a local minimum (Rokach 2016).

Analogously to other types of ensemble, a RF achieves its greatest predictive accuracy when the trees that compose it are as accurate as possible, and make prediction errors as diverse as possible. It has been shown that when the errors made by the decision trees of a decision forest are less correlated, the accuracy of the entire ensemble is better (Ali and Pazzani 1995).

2.5.3 Feature importance measures

As a RF is composed of many decision trees, interpreting each tree individually to assess the importance of the features becomes too cumbersome. However, there are ways to gauge feature importance in a RF, which improves the method's interpretability and allows RFs to be used in feature selection applications (Li et al. 2020). The most common global feature importance measures (referring to all instances of the dataset) for RFs are the permutation importance measure (PIM) and the Gini importance measure (GIM).

The PIM of a feature is calculated by the average over all decision trees of the difference between the classification accuracy of a decision tree on two different versions of the OOB instances: with the original feature values and with randomly permuted values of that same feature. The GIM is the average reduction in Gini Index over all nodes which use that feature for branching, over all decision trees in the forest (Touw et al. 2012). The GIM is one measure belonging to the general category of impurity decrease measure, which measures the average impurity decrease over all nodes in a decision tree or forest. A related measure is the average of information gain (or gain ratio) over all nodes, which is the feature importance measure used in our feature importance analysis in Section 6.5.

Hapfelmeier et al. (2014) proposed a measure that better reflects the importance of features with missing values. Instead of permuting the values of the feature to compare with the regular values, they randomly allocate the instances to one of the child nodes of the node with the feature being analysed. This circumvents problems with missing values, since the classification of each instance ignores the value of that feature.

It is also possible to calculate a local measure of feature importance, to assess the relevance of a feature to classify a single instance. This allows an analysis of features that are relevant to classify a specific subset of instances in the dataset, albeit they might not be much relevant globally. An example of importance measure that can be calculated either globally or locally is the Intervention in Prediction Measure (IPM), proposed by Epifanio (2017). The IPM of a feature for a given instance is the percentage of times that feature was used in a node in a path from the root node of a tree to the leaf node used to classify that instance in that tree, over all decision trees in the forest. That is, the IPM is the percentage of all splits done in paths followed by that instance that used the analysed feature to classify

that instance.

2.6 Longitudinal Dataset Inputs for Supervised Machine Learning

As mentioned earlier, ML algorithms take datasets of instances and features as input, and often raw data sources from real-world data collected for various reasons require cleaning and preprocessing before becoming viable datasets for learning models. In some cases, such as data that has a time-axis, this preparation process includes transforming the data into a two-dimensional representation (instances and features), which can be achieved in different ways (this is discussed further in Chapter 3). In order to be able to discuss features with a temporal aspect, we define a representation of longitudinal datasets used in this thesis as follows.

Consider a longitudinal study that collects data from a set of instances (participants) over fixed intervals of time. We call each variable observed in this study a conceptual feature, and give it an index i to refer to that variable. Thus, we can say that the study observes a set of conceptual features F_i where $i = 1..d$, and d is the number of variables (dimensions) observed. If a feature is observed multiple times throughout the study, as is the case for most features in a longitudinal study, we also give it a time-index j referring to the wave (time-point) of the study when the measurement took place, where $j = 1..t$ and t is the number of waves in the dataset. We then represent each feature in the longitudinal dataset as $F_{i,j}$, using a combination of both its conceptual feature index i and its time-index j.

Chapter 3

Supervised Machine Learning for Longitudinal Datasets

In this Chapter we discuss different representations of longitudinal data as input for supervised machine learning (ML) algorithms, and review various related studies. As this Chapter's main contribution, we propose a new taxonomy for representations of longitudinal datasets for ML, and use it to analyse the related works of supervised ML applications using real-world longitudinal datasets. The taxonomy proposed in this Chapter was presented in a short paper published in the proceedings of a workshop (Ribeiro and Freitas 2019b).

3.1 Representations of Datasets with Multiple Time-Points

Jie et al. (2017) divide supervised machine learning methods that use data from multiple time-points into four categories, accordingly to the number of input and output time-points used by the learning method: 1) Single-time-point Input and Single-time-point Output (SISO), 2) Single-time-point Input and Multiple-time-points Output (SIMO), 3) Multiple-time-points Input and Single-time-point Output (MISO), and 4) Multiple-time-points Input and Multiple-time-points Output (MIMO). In the terminology used in this thesis, the inputs are features, the outputs are target variables, and time-points are waves. Note that their representation can be used for time-series data as well as for longitudinal datasets.

A SISO dataset consists of a single wave with features and target variables, that is, a non-temporal dataset. A SIMO dataset also only has features from a single wave, but the target variables span multiple waves. The former could be used for prediction problems where it is of interest to investigate how the time passage would affect a target variable using only features from a single time-point.

Longitudinal ageing studies usually fall into the other two categories. A MISO dataset has features in multiple waves but target variables only in a single wave (typically, the last wave of the dataset). A MIMO dataset has both features and target variables available in multiple waves (typically, all waves). An important characteristic of longitudinal datasets is that it is possible that the time-points of features and target variables might not overlap, according to the categorisation of datasets described in this Section.

3.2 Longitudinal Databases Used in ML research

In this Section we briefly review some of the main longitudinal databases that have been used in the literature on longitudinal supervised ML methods. It also mentions the main goal of the ML studies using datasets derived from those databases, regarding the type of variable they try to predict. These target variables are typically age-related diseases.

We characterised the datasets used in each study according to Jie et al. (2017)'s four categories. All the original general longitudinal databases from which specific datasets were created for ML purposes are categorised as MIMO, because they contain time-series of variables that can be used either as features or as class labels in the created ML datasets. If a study chose to, from these MIMO databases, create a multiple input, single output ML dataset for their experiments, we denominated their dataset as MISO.

The Alzheimer's Disease Neuroimaging Initiative (ADNI) study started in 2004, and generated a longitudinal database that has been used by four of the reviewed studies. The ADNI study follows a sample of over 1000 subjects, visiting them twice a year (waves 6 months apart from each other) and collecting several types of biomarkers (potentially used as features by ML algorithms) including blood tests, tests of cerebrospinal fluid, and MRI/PET imaging for Alzheimer's disease (AD) clinical trials and diagnosis. The subjects have a diagnose in each

wave of the study, being classified as cognitively normal (CN), mild cognitive impairment (MCI), or Alzheimer's Disease (AD).

Regarding the four studies that utilised the ADNI database[1], Minhas et al. (2015)'s study focused on subjects that were from the Mild Cognitive Impairment (MCI) class in the first wave, and investigated the difference between those who progressed to AD (subjects classified as AD in the last wave) and those who did not (MISO dataset). Mo et al. (2013)'s and Cui et al. (2019) studies focused instead on differentiating subjects classified as Cognitively Normal (CN) in the last wave from those diagnosed as AD in the last wave (MISO datasets). Huang et al. (2016), and Jie et al. (2017) focused on predicting scores related to AD that are available in the ADNI database, working with regression algorithms to predict the scores' values in each wave of the study (MIMO dataset). Finally, Bhagwat et al. (2018) predicted the trajectories of scores related to AD, namely whether a score would be stable or decreasing in the final wave, making this a classification problem (MISO dataset).

Zhang et al. (2016) created MISO datasets from the database originated by the Chinese Longitudinal Healthy Longevity Survey (CLHLS)[2]. The authors created three datasets, one for each pair of consecutive waves of the survey (2002-2005, 2005-2008, 2008-2011). For each of these datasets, they predicted a target variable, Activities of Daily Living (ADL) in the last wave.

The database from the Cardiovascular Health Study Cognition Study (CHS-CS) [3] also has scores related to AD. The database has thousands of cognitive, metabolic, cardiovascular, cerebrovascular, and neuroimaging variables obtained twice a year, between 1990 and 2012, from people of ages 65 to 108 years old. In their study, Adhikari et al. (2019) calculated the odds of death and dementia for each target wave, and they used features from all waves in the database, creating a MIMO dataset from the CHS-CS database.

Data from the Study of Health in Pomerania (SHIP) [4] were used by Niemann et al. (2015), in order to predict a liver disorder. The SHIP study aims to investigate the prevalence and incidence of common risk factors, sub-clinical

[1] http://adni.loni.usc.edu/

[2] https://sites.duke.edu/centerforaging/programs/chinese-longitudinal-healthy-longevity-survey-clhls/

[3] https://chs-nhlbi.org/

[4] http://www.bioshare.eu/content/study-health-pomerania

disorders and clinical diseases, and how these are associated with each other. The data is collected 5 years apart for each participant, and the study currently has 3 published waves: SHIP-0 (1997-2001), SHIP-1 (2002 -2006), SHIP-2 (2008-2012). Although the data for the several diseases investigated in the study is available in every wave, the dataset in Niemann et al. (2015)'s study is MISO, since they only predicted the target variable in the last wave, for each experiment.

Du et al. (2015)'s study used data from the Phil Bowen Amyotrophic Lateral Sclerosis (ALS) Prediction Prize4Life challenge[5] to predict the progression of the disease in ALS patients, using their current and past statuses. The monthly database originated for the challenge has 12 consecutive waves with about 44 time-varying features (features with one value per wave) and 34 time-invariant features (features with a single value for all waves, such as biological sex), including demographics, medical history, and lab test data. The target variable, an ALS score, is predicted in every wave of the dataset used in the study, and the dataset is categorised as MIMO.

The human viral challenge studies[6] generated three gene-expression datasets that contain gene expression data from human volunteers, who were infected with H3N2 influenza, rhinovirus (HRV) and respiratory syncytial virus (RSV). The waves consisted of one set of daily recordings of gene expressions of the individuals in the study. The class label is available only in the last wave of each dataset, characterising all three datasets as MISO datasets. Radovic et al. (2017)'s study aimed to predict whether the samples belonged to symptomatic or asymptomatic participants of the study.

The database from the English Longitudinal Study of Ageing (ELSA) [7] contains thousands of variables related to several different aspects of the subjects' lives, including health and disability, socio-economic, well-being and biological markers of disease. The goal of the study, whose subjects are visited every two years, is to allow a multidisciplinary analysis of the several aspects that influence human ageing. In their study, Pomsuwan and Freitas (2017) used data from the nurse visits in the ELSA study (a special data subset containing mainly biomedical variables, with 4-year gaps between waves) to predict several age-related diseases.

[5]https://www.synapse.org/#!Synapse:syn2826267/wiki/71167
[6]http://hvivo.com/
[7]https://www.elsa-project.ac.uk/

The authors only predicted class variables from the wave 7 the study, categorising the dataset as MISO.

3.3 A Taxonomy for Representing Longitudinal Data in Machine Learning

As discussed in the previous Section, longitudinal datasets typically fall into the MISO or MIMO categories, in Jie's classification, both of which involve multiple-time-point inputs, i.e., they contain features in multiple waves. This classification does not give information on how the time-related information in the longitudinal features is represented when the dataset is used as an input for ML algorithms, however. Hence, in this Section we propose a new categorisation of approaches for representing such datasets with multiple waves of features for ML applications.

Consider a longitudinal dataset consisting of t consecutive waves $W_{1,\ldots,t}$, each of those comprised by d features and n instances. In such a dataset, there is time-related information associated with each feature, and this information can be represented in different ways, when using a longitudinal dataset as the input for a ML algorithm. Figure 3.1 shows 4 strategies to represent multiple feature waves for a ML algorithm.

The simplest representation of a multiple-wave longitudinal dataset, consisting of a separate input file for each wave (denoted by SepW for "separate waves") as shown in Figure 3.1(a), is used when one wants to apply a ML algorithm to each wave of data separately. This representation has the advantage of simplicity and generality, since any standard (non-longitudinal) ML algorithm can be applied. However, the ML algorithm will be unable to exploit the temporal information contained in the full dataset. If one desires to apply a ML algorithm to all waves at the same time they need to perform some sort of data flattening (i.e., removing the time dimension to represent the data in two dimensions, instead of three), which is a way of representing a multi-wave longitudinal dataset on a single input file.

One form of data flattening is the use of aggregation functions (AGG), which computes some summary measure (or other types of constructed features using the original feature values), such as the median, average or mode, over the values

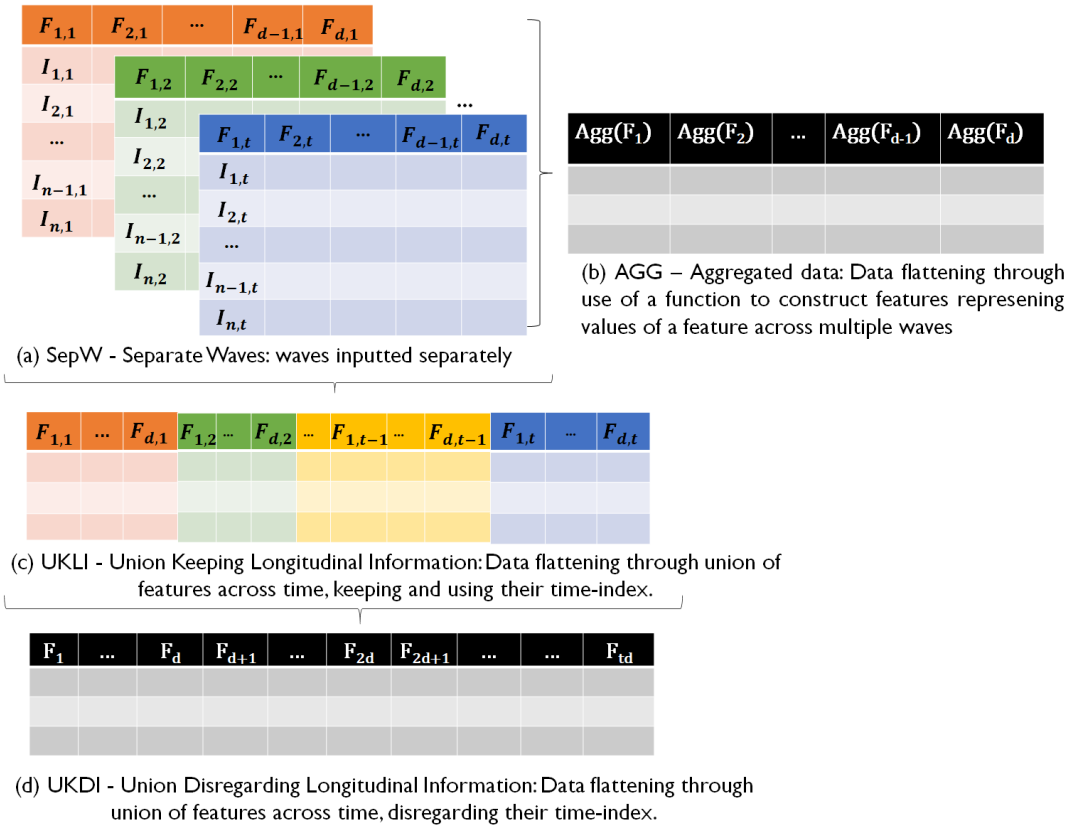

Figure 3.1: Multiple-wave longitudinal data representation scenarios. For each feature, $F_{i,j}$, i represents its feature index and j represents its time-index.

of a given feature in each wave of the dataset. This approach, shown in Figure 3.1(b), has the advantage of using a data storage space of the same size as a single wave of the longitudinal dataset, since one aggregate feature represents all different values of that feature throughout all the waves. However, even though the time-related information is not completely ignored, its details are lost, and the ML algorithm will be unable to identify precise time-related changes that a feature might have exhibited over the waves of the dataset.

In order to keep all the time-related information and still represent the longitudinal dataset on a single input file, one can perform another type of data flattening: a Union of the features Keeping Longitudinal Information across time (UKLI), as shown in Figure 3.1(c). In this approach, the different waves of the dataset are represented through a single dataset made by the union of all features in all waves. Hence, the time-related information associated with the features is

represented by denoting a feature by both its sequential feature id and its time-index (wave number), and this information is considered by the ML algorithm. To clarify, let $F_{i,j}$ denote the value of conceptual feature i (conceptual feature meaning the variable being measured repeatedly over the waves of the longitudinal study) in wave j. In this representation, the ML algorithm should treat the relationship between two values of the same feature in different waves, say $F_{1,1}$ and $F_{1,2}$, as conceptually different from the relationship between two values of different features in the same wave, say $F_{1,1}$ and $F_{2,1}$.

How that differentiation is made by the ML algorithm is up to the researcher, but should it not be done, then the time-related information would be irrelevant, even if the naming of the features represents their waves. In that case, corresponding to scenario Union Disregarding Longitudinal Information (UDLI) in Figure 3.1(d), the longitudinal dataset is merely represented as a merging of its waves, disregarding the time-index of each feature. Hence, two values of the same feature in two different waves are treated in the same way as two values of two different features in the same wave.

In some cases, authors may choose to remove the representation of the time-index (wave number) entirely as it is not used in the analysis. In such cases, to avoid that the merged dataset have multiple features with the same feature index, one could rewrite the feature indexes of all features after the merging of dataset waves, using sequential incrementing, as shown in the column headings of the table in Figure 3.1(d).

It is important to highlight that the differentiation between the UKLI and UDLI representations in this taxonomy do not depend only on how the data is stored, but on how it is used in the learning process. Even if the time-index is kept in the naming of the features, if this does not make any difference in the analysis of the data by the ML algorithm or other tasks of the knowledge discovery process, we consider the representation UDLI, as the time-index of the features is being disregarded.

3.4 ML Approaches for Longitudinal Datasets

Some studies used the standard versions of existing ML algorithms in their experiments with longitudinal datasets. Most of these coped with the temporal

information of the data through its representation of the dataset (data transformation approach). We reviewed some studies with this type of approach, and discuss our findings in Section 3.4.1. The other type of approach for coping with longitudinal data inputs is algorithm adaptation. We revised studies that adapted existing algorithms to employ strategies that coped with the longitudinal nature of the data in Section 3.4.2. For each work, we highlight the strategies used to cope with the longitudinal datasets and the representation of the data based on our proposed Taxonomy.

3.4.1 Data transformation approaches

Zhang et al. (2016) did not detail the dataset preprocessing in their study, stating that the features were selected empirically, based on existing research, and merged data from two consecutive waves into a single dataset, disregarding longitudinal information (UDLI representation, Figure 3.1(d)). The authors employed the standard C4.5 decision tree algorithm (Quinlan 1993) to predict the Activities of Daily Living (ADL) status (healthy or disability) in a given wave of the dataset, using features from that wave and the previous wave. It is important to note that the authors used the value of the class label in the previous wave as a feature, which makes the prediction problem substantially easier. It seems intuitive that an individual that currently has ADL issues (difficulty to perform daily tasks) will likely still have them in the next wave.

The study by Mo et al. (2013) fused both waves of a longitudinal dataset into a single dataset using the UDLI representation, where the class variable actually belonged to the first wave of the dataset, meaning the authors used features from the second wave to predict the class in a past wave. They selected features that were the most relevant for differentiating patients diagnosed with Alzheimer's Disease from those diagnosed as Cognitively Normal.

Minhas et al. (2015)'s study compared the AGG (Figure 3.1(b)) and UDLI representations. The authors used two summary measures, namely the arithmetic mean and the median of each feature throughout the waves. They used features from the first 6 waves of the ADNI study to predict the class label on wave 6, using standard SVM to predict a subject's conversion from the MCI class to the AD class. In the first experiment, they used only the first wave's feature values for

training, without performing missing value imputations and without adding the summarising features to the dataset (UDLI). For the rest of their experiments, the authors compared variants of five different techniques to cope with missing values (see Section 3.5) and the approach of adding one of the two summarising features (mean or median) to the dataset (combining scenarios AGG and UDLI, and the ML algorithm still does not use the time-related information of the features in any of the experiments). The authors concluded that the added summarised features improved accuracy and the AUC (Area Under the ROC Curve), even though a controlled experiment consisting of using only the missing value imputation techniques, without adding the summarised features, was not done. Using only the summarised features (AGG representation) was not tested. Therefore, it is not possible to confirm whether the improvement was caused by adding the aggregated longitudinal data, due to a lack of a controlled experiment.

A different approach based on combining the UDLI and AGG representations was used by Niemann et al. (2015), who grouped instances considering the features observed in each wave, creating features related to clustering information in each wave (such as an instance's distance to a cluster's centroid, the number of members of each class in the k-nearest neighbours of an instance, the cohesion and silhouette index of the instance) and how those changed in relation to previous waves. As mentioned, created features such as these fall into our AGG categorisation. None of the constructed features used by the authors required a comparison between instances from different waves. This was by design, to ensure that if the number of clusters changes, or if a cluster identity changes throughout waves, the features would still be valid. The temporal features derived from the clustering results were added to the original dataset prior to feature selection, and the features from the 3 waves were merged and used for learning ignoring their temporal information. Even though the constructed features consider the waves, meaning some time-related information is still represented in the dataset, the time-index of the original features is still disregarded (UDLI and AGG).

A temporal variation of the minimum Redundancy–Maximum Relevance (mRMR) filter algorithm for feature selection was proposed by Radovic et al. (2017). Putting it simply, for each feature, they calculated an average of the correlations between that feature and the class label across all waves. We considered this an aggregation function, categorising their data representation as AGG. That strategy was better

than using more complex measures, according to the experiments performed by the authors. The study aimed to classify patients from a dataset as symptomatic or asymptomatic using gene expression data.

Pomsuwan and Freitas (2017) joined the waves of the ELSA dataset maintaining the time-related information of the variables (UKLI representation, Figure 3.1(c)). The features were divided into groups; each group containing both variations of the same base feature across time (i.e., across different waves) and constructed features representing the differences (increase and decrease in value) of the same base feature from wave to wave. They transformed the data so that each group would be small enough to be inputted into the exhaustive search version of the Correlation-based Feature Selection (CFS) method (Hall 1999). The study used medical data from three different previous waves from the ELSA study, predicting whether individuals would develop an age-related disease in a future wave. The strategy was tested for 10 different diseases separately (10 binary classification problems), using the C4.5 decision-tree and Naive Bayes algorithms. Their proposed method showed an improvement over the standard CFS greedy forward search applied to all features (without using temporal information for dividing the features into groups), and over not performing feature selection when tested with Naive Bayes, but did not significantly improve the results when tested with the C4.5 decision tree algorithm.

3.4.2 Algorithm adaptation approaches

Adhikari et al. (2019) created a new dataset from the original CHS-CS database. Instead of the biannual features from the CHS-CS database, the constructed dataset had data from individuals of each age in the 65..98 range as waves, totalling 34 waves. For example, the wave for age 70 would have data from all subjects in the CHS-CS study when they were 70, regardless of when that data was collected. Their model predicted the odds of either death or dementia (different models were trained for each type of prediction) of a subject when they reach $t + 10$ years of age, where t is the subject's age at the last wave of the dataset. The problem tackled by the authors was longitudinal classification, though they proposed a regression algorithm producing a linear model. The authors used a Lasso regression model (Tibshirani 1996), and proposed regularizers for it that

considered the time-related information in the variables (UKLI representation). The Lasso regularization encourages overall sparsity in the coefficients of the active features (i.e. features with coefficient greater than 0 in the linear model) in each wave. The fused Lasso (Tibshirani et al. 2005) regularisation encourages contiguity in the coefficients of the active features across waves.

Similarly, Jie et al. (2017) proposed an adaptation to the Lasso algorithm, to predict Mini Mental State Examination (MMSE) and Alzheimer's Disease Assessment Scale-Cognitive Subscale (ADAS-Cog) scores using longitudinal magnetic resonance imaging data. The authors proposed a novel temporally-constrained group Lasso method, named tgLasso, which uses two different weight smoothing techniques. The first is a fused smoothness term, which requires that two weights for the same feature at adjacent waves have a small difference (originated from the fused Lasso). The second is an output smoothness term proposed by the authors, which requires that the model's outputs at two adjacent waves also have a small difference. In their study, in one of the experiments, the four waves of the dataset were used separately (SepW representation, Figure 3.1(a)) for estimating regression variables. In the other experiments, two or more consecutive waves were joined into a single dataset with the time-related information on the variables being considered by the proposed temporal group Lasso (tgLasso) algorithm, which included a smoothing technique that considers the time-index of the features (UKLI representation, Figure 3.1(c)). They tested predicting the score in all waves, one at a time, using only the first wave's features, and gradually incremented the number of feature waves included in the dataset. The experimental results showed that the tgLasso significantly improved regression performance when compared with the regular Lasso and the group Lasso methods.

Another regression algorithm adaptation was proposed by Du et al. (2015). The authors extended a previous longitudinal SVM classification algorithm, LSVC (Chen and DuBois Bowman 2011), by making it a longitudinal regression algorithm. LSVC extends the well-known support vector machine (SVM) to longitudinal data by estimating the SVM hyperplane separating parameters using additional proposed temporal trend parameters, which take into account observational dependence within subjects. They created two types of datasets, the first being a merge of the data from all waves considered in each given experiment (1 to t), keeping the temporal information (UKLI representation), which is used to

calculate temporal trend parameters. The second approach created a new dataset with only the means of the values of each feature from all waves considered in the dataset (AGG representation). Their results showed a better performance for the first type of dataset.

Huang et al. (2016)'s study aims to predict some Alzheimer's Disease longitudinal clinical scores. The authors tested their model to predict the score for each individual in all the waves after the first wave, using features from the current and all past waves of the dataset as input, simultaneously (UKLI representation, as the features' time-indexes are used to create these incremental datasets). The authors presented a Random Forest (RF) regression algorithm adapted for sparse regression. The authors justified their choice to ignore the complex relations of longitudinal clinical scores by stating that the RF model can handle non-linearity in data better than the Lasso regression, and their algorithm outperformed a previous Lasso study that introduces a smoothness function prior to applying their regression model, which supports that claim. The proposed RF algorithm outperformed standard RF and other popular regression methods, namely Lasso regression, Ridge regression, and SVM. The RF regression model that had the best prediction results started at the first wave and used its feature values to predict the score for the second wave, then incorporating this prediction onto the dataset. Hence, they used multiple instances of the UKLI representation, as stated previously, since in each run of the algorithm, the dataset is represented by a merging of the waves currently being used, keeping time-related information in the features. The algorithm then used features and scores from the first and second waves to create a new model, to predict the score for the third wave, repeating this process until the score for the final wave of the dataset was predicted.

Bhagwat et al. (2018) proposed a Longitudinal Siamese Network (LSN) algorithm that combines data from two input time-points, having separate but dependent network branches used for each time-point. Typically, Siamese networks are used for calculating similarities between inputs, thus their motivation for using them to represent change over time in longitudinal data. Their classification problem was predicting the trajectory classes (stable, decline or fast-decline) for mini-mental state exam and Alzheimer Disease Assessment Scale scores in the ADNI datasets. The twin branches of the proposed LSN have a weight-sharing function, which calculates identical weights for the nodes at each layer of both

networks, whilst each branch receives input data from a given time-point (baseline and follow-up), which makes their data representation SepW. The output of the two networks is concatenated, representing the change over time in the input data, and is then used in the classification task.

Cui et al. (2019) combined Convolutional Neural Network (CNN) and Recurrent Neural Networks (RNN) built with cascaded bidirectional gated recurrent units (BGRU), in a framework to predict Alzheimer's Disease diagnosis, given the MRI scan image input data in the ANDI longitudinal dataset. The CNN part of the framework focuses on learning spatial features from the input data (feature extraction task), which are then fed to the RNNs. In the RNN/BGRU part the authors used a SepW representation, feeding data from different time-points to different BGRUs, in the first layer of their neural network. In subsequent layers, the data from a time-point is correlated with the previous and next time-points (using forwards and backwards GRUs, making them bidirectional). The final layer in the BGRU cascade outputs features that capture the temporal variation of the input longitudinal features, which are then used in the classification problem. The creation of these output features is an AGG representation, as they are created from multiple time-point observations and represent the temporal data as a single final value. Thus, the representation of longitudinal data in this work is both SepW (for the training phase) and AGG (for the final classification step).

Lash and Street (2020) applied inverse classification models to longitudinal data, for recommending behaviours to mitigate risk of cardiovascular disease. Inverse classification problems aim to find recommendations that lead to changed feature values that result in a desired classification. Their application compared the effect of adopting personalised recommendations (such as changes to diet and exercise frequency) on different time-points on the risk mitigation results. The proposed algorithm adaptation is a framework for inverse classification that considers longitudinal data inputs, creating temporal links between different measurements and comparing the temporal effects of implementing recommendations at different time-points. Their framework calculates a risk estimation for each wave of the dataset, and adds these past estimations as predictive features in following waves, instead of using the repeated measurements of the features. Thus, the data representation in this work is a combination of SepW (only measurements of the current wave are used) with AGG (the risk estimation of previous waves is used

as an aggregated measurement).

3.5 Methods for Coping with Missing Values

One of the challenges of analysing longitudinal data is that studies that follow individuals for a long period can encounter several issues with obtaining data across all waves. One of the most relevant of these issues is attrition, which is the complete absence of an instance in a wave of a longitudinal dataset, either due to dropping out or joining in later waves. Therefore, it is common that longitudinal studies face a high number of missing values in their datasets (Saiepour et al. 2019), and this is a critical aspect of the inherit complexity of analysing longitudinal data (Anagnostou et al. 2021). There are several strategies that can be used to address this issue, some of them taking advantage of the longitudinal nature of the data. In this Section, we mention strategies adopted by some of the reviewed works to cope with the missing values in their longitudinal datasets.

Minhas et al. (2015) experimented with five different strategies to cope with missing values in their study in a data preprocessing phase (before applying the ML algorithm). The authors reported that the last observation carried forward (LOCF) strategy had the best results, in comparison to the other four strategies: (a) deleting instances with missing data, (b) not doing any preprocessing on the original dataset, (c) replacing missing values with the mean value of the feature for all instances in the same wave, and (d) replacing it with the value from the nearest neighbour, considering all features in the previous wave. The LOCF technique assumes that a feature is likely to maintain its previous value in follow up waves, so when encountering a missing value for an individual (instance), it tries to replace it with the most recent known value for that feature from the same individual, from previous waves. It is worthwhile to note that, in the preprocessing phase, the dataset was in the UKLI representation, which allowed them to use the LOCF and nearest neighbour (d) strategies, which consider the temporal aspect of the data. After coping with missing values, the authors represented their dataset using the AGG and UDLI strategies for their classification experiments.

Adhikari et al. (2019) also used temporal information to cope with the missing values in their dataset in a data preprocessing phase, applying LOCF when possible. For cases where there was no previous value available for a feature for

a given instance (individual), the authors calculated the global median across all waves, from people of the same age, of the values of that feature. Their justification for that approach was that subjects of the same age were more likely to share similar values for the features, since the features of the CHS-CS dataset are all health-related.

Huang et al. (2016) proposed a two-stage longitudinal score prediction to input the missing score values (some of them used as features and others as target variables) for the subjects of the ADNI study who did not have all the scores present. The authors argued that, even though the most direct way of predicting these scores would be through linear interpretation, the performance of the final predictive model would not be as good due to the subjects whose scores do not change linearly through time. Therefore, they employed, in the first stage, a predictive model using only the first wave's features (which did not have any missing values), to predict scores at each future wave. Then, in the second stage, they performed a regression task where the training set was composed of the baseline features (features from the first wave) and the longitudinal scores, both the observed and predicted ones. This means that the target-wave scores were predicted using both known score values and values that were calculated by the regression model built in the first stage.

3.6 Summary of the Reviewed Studies

Table 3.1 contains a summary of the studies reviewed in this Chapter, ordered by publication date. For each study revised, we state: (a) the problem type (classification, regression, or a focus on replacing missing data or feature selection), (b) the main algorithm proposed or used in the study (a classifier/regression algorithm or a data preprocessing algorithm), (c) the characterisation of the dataset following the taxonomy proposed by Jie et al. (2017), (d) the time indexes of the feature waves, (e) the time indexes of the class label waves, and (f) the longitudinal dataset representation scenario used in the main experiments, according to the representations in the proposed taxonomy.

Regarding columns 'Feature Waves' and 'Class Waves' in Table 3.1, depending on the number of experiments performed, we sometimes used a t variable to represent the time-indexes. In those cases, the range (if separated by two dots,

e.g. 1..3) or set (if separated by commas, e.g. 2,4,6) of values that t can take are provided; i.e., in those cases the studies experimented with all variations of t in the value range or set provided in Table 3.1. By contrast, when no t variable is used, this means that a single experiment is represented in the Table, even though other experiments not relevant to our study might have been performed.

Table 3.1: Summary table of the revised studies

Study	**Problem Type**	**Main Algorithm**[8]	**Dataset**	**Feature waves**	**Class waves**	**Representation on experiments**
Mo et al. (2013)	Classification	Ensemble vs SVM	MISO	1..2	1	UDLI
Minhas et al. (2015)	Missing value imputation	SVM	MISO	1..6	6	AGG + UDLI
Niemann et al. (2015)	Classification	RF, DT, NB, KNN	MISO	1..3	3	UDLI
Adhikari et al. (2019)	Regression	Multinomial Fused Lasso	MIMO	t = 1..24	t+1..t+10	UKLI
Du et al. (2015)	Regression	Longitudinal SVR*	MIMO	t = 9..11	12	AGG UKLI
Zhang et al. (2016)	Classification	C5.0 Decision tree	MISO	t and t+1, t = 1..3	t+1	UDLI
Huang et al. (2016)	Regression	Random forest	MIMO	1..t, t=2..5	t	UKLI
Jie et al. (2017)	Regression	tgLasso*	MIMO	1..t, t = 2..4	t = 2..4	SepW UKLI
Radovic et al. (2017)	Feature selection	Temporal mRMR*	MISO	1..t, t=14,16,21	t	AGG
Pomsuwan and Freitas (2017)	Feature selection	CFS	MISO	2,4,6	7	UKLI
Cui et al. (2019)	Classification	CNN and RNN	MISO	1..t, t=1..6	t	SepW + AGG
Bhagwat et al. (2018)	Classification	LSN*	MISO	1..2	2	SepW
Lash and Street (2020)	Classification	SVM	MIMO	1..t, t=1..3	t=1..3	SepW + AGG

The work presented in the next Chapters of this thesis has novel contributions related to the topics analysed in this review. Chapter 4 has contribution on handling missing data in longitudinal datasets. The two types of approach defined in our taxonomy are addressed in the following Chapters. Chapter 5 has a data transformation proposal for creating constructed features to be added to the original dataset (UKLI+AGG representation). Finally, Chapter 6 covers an algorithm adaptation approach, changing the standard Random Forest classification algorithm for coping directly with longitudinal data inputs. The insight obtained from this related works analysis was fundamental for our decision-making process in all following Chapters.

[8]New algorithms that were proposed in the article are marked with a *.

Chapter 4

Data Preprocessing

In this Chapter we discuss the preprocessing steps taken to create the datasets used in our research, including a description of the data sources, the missing value replacement (Section 4.2) and class balancing (Section 2.2.5) tasks. The main contribution of this Chapter is the proposal of a novel Data-Driven missing value replacement approach for longitudinal datasets. We present the results from a series of classifier-independent and classifier-dependent experiments evaluating the proposed approach and discuss how it can improve both the estimation accuracy of replaced missing values and the predictive performance of the random forests generated with fully imputed datasets. Part of the experiments in this Chapter were published in a conference short-paper (Ribeiro and Freitas 2019a), and the main contribution was published as a journal article (Ribeiro and Freitas 2021b).

Regarding the implementation and experiments performed in this and the subsequent Chapters of this thesis, they were done using the developer version of the Weka data mining toolkit[1], using Java 8 (build 1.8.0_311-b11). This tool was chosen, as opposed to the arguably more popular Python library scikit-learn, for the following reasons.

The only decision tree algorithm currently implemented in scikit-learn is the CART tree, and it cannot receive categorical (nominal) variables in its input (all our datasets have a combination of numeric and nominal features), so we would have to transform all our features into numeric, likely leading to reduced performance. Weka has several DTs implemented, including the C4.5 (Quinlan 1993) DT which has desirable properties such as the way it handles missing values

[1]Weka Version 3.9.1, open-source, available at: https://www.cs.waikato.ac.nz/ml/weka/

(this is discussed in detail later in Section 4.7). In addition to that, we performed initial experiments with Python's scikit-learn implementation and found that the internal code of the classification algorithms was harder to adapt, compared to the Java code in Weka. For these reasons, we chose to focus our efforts on the Weka tool for all experiments reported in this thesis.

4.1 Dataset Creation

4.1.1 The ELSA-core and ELSA-nurse datasets

The English Longitudinal Study of Ageing (ELSA) is currently one of the most prominent populational studies of ageing (Abell et al. 2018; Banks et al. 2019). 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.

The ELSA has, in each of its waves, thousands of respondents from inhabitants of United Kingdom households, which take part of a core interview every two years (the time interval between two consecutive waves), answering questions about various aspects of their lives, including demographic, health, wellbeing and economics. Data from this core questionnaire is used to create the class labels for all ELSA datasets, and to create the ELSA-core datasets. For this project, we used data from the core waves 1-8 (2002-2016) - the 9th wave of the study was published in late 2020, after most of the experiments of this thesis had been completed.

In addition, special questionnaires are used to collect biomedical data every 2 waves (i.e., roughly every 4 years), 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, which we used to create our ELSA-nurse datasets. We used data from all four currently published waves of the ELSA study with data collected by a nurse: waves 2, 4, 6 and 8 (2004-2016).

A total of 20 longitudinal datasets were created with the raw data files from the ELSA-core and ELSA-nurse questionnaires, each with a combination of one of two data sources (core data or data collected by a nurse) and one of 10 age-related diseases used as class (target) variables. The class variable in each dataset refers to the presence (negative class) or absence (positive class) of a diagnose for an

age-related disease, for each instance (ELSA respondent), in wave 8. For all 10 diseases, the positive class is the majority class, with an increased degree of class imbalance for rarer diseases, such as Dementia and Parkinson's Disease. Note that, in order to have class labels for all instances (ELSA respondents), we only utilised data from respondents that participated in the ELSA's 8th wave. In cases where a respondent did not participate in any of the other waves in the dataset, the values for that wave's features were set as missing for that respondent.

The 10 ELSA-nurse datasets share the same set of predictive features, as do the 10 ELSA-core datasets, even though they have different class variables (representing different age-related diseases), as explained in more detail later.

4.1.2 The TILDA datasets

The Irish Longitudinal Study of Ageing (TILDA) is based on the ELSA study, with very similar data collection methodology and economic, social and health data gathered from participants aged 50 and over residing in Ireland (Kenny et al. 2010). The interviews of the participants happen every two years, with separate health assessments (collecting biomedical information) in waves 1 and 3. The TILDA started its data collection in October 2009, and up until late 2020 only its first 4 waves were published, thus we only included the first 4 waves in this study. Currently, TILDA has 5 waves published and is doing its data collection for wave 6.

For this research, we focused on the health data in the TILDA datasets. Thus, the selected features for the TILDA datasets are similar to the ELSA-nurse (biomedical data, collected on health assessments[2]) and ELSA-core (questionnaires answered by the participant) features. As with the ELSA-nurse and ELSA-core datasets, the 10 TILDA datasets share the same predictive features, with 10 binary class variables representing the reported diagnosis of age-related diseases.

[2]TILDA health assessments were on waves 1 and 3 and include the same type of data measured by nurses in the ELSA-nurse questionnaires, such as blood samples, grip strength and mobility assessments

4.1.3 Preparing a base dataset

In this Section we discuss the creation of a base dataset, which has been used for creating the 30 datasets used in our experiments, and the steps we took to convert the raw ELSA and TILDA 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 4.1.4) as a single missing value symbol ('?'), and creating the class variables. The data preparation process is similar to the one applied by Pomsuwan and Freitas (2017), who also created ELSA-nurse datasets, with a few differences in the feature selection process, and the fact that our initial datasets do not include constructed temporal features (discussed in Chapter 5).

Note, however, that Pomsuwan and Freitas (2017) created only 10 ELSA-nurse datasets, and their datasets contained only data up to wave 6 of the ELSA study, which was the most recent wave with available nurse-data files when they created their datasets. By contrast, in this work we not only created 10 more updated ELSA-nurse datasets (including data from wave 8), but also created 10 new ELSA-core and 10 new TILDA datasets, as mentioned earlier.

As the sets of ELSA and TILDA participants are updated at every wave, with new respondents being added and others leaving the studies due to various reasons, each of these databases has partially disjoint sets of instances across their waves. That is, for any given pair of waves, some participants will occur in both waves, whilst other participants will occur in only one of those waves. We considered that every respondent that participated in the interview for the last wave considered in this project should be included in the dataset. For instance, in the ELSA-nurse datasets, 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 "?". The final longitudinal datasets had 7097, 8405 and 5715 instances for ELSA-nurse, ELSA-core and TILDA, respectively.

4.1.4 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.

For the ELSA-nurse datasets, from the initial set of 1041 features (all features

from the 4 nurse-data datasets in ELSA's waves 2, 4, 6 and 8), we removed all redundant features, metadata[3], and features deemed irrelevant for our classification problem (predicting age-related diseases). For the cases of redundant features that represent different measures of the same variable in the same wave (e.g., multiple recordings of blood pressure), we replaced those with a new feature defined as the mean value of the redundant features. After this reduction, the ELSA-nurse datasets have 141 features.

For the ELSA-core datasets, the initial set of over 7000 features was reduced to a set of 352 features, following the same process used for ELSA-nurse data. In this reduced feature set, however, there were several binary variables that stored whether the respondent had experienced a specific event from a set of related events. For example, there is a set of binary questions to check how many of a set of activities from daily life (ADL) the user reported having difficulty. For this type of feature, we reduced the dataset by merging them into a numeric feature reporting how many of the set of features the participant responded positively to, i.e., how many ADL they had difficulty with, etc. After this reduction, the ELSA-core datasets have 171 features.

For the TILDA datasets, the initial set of over 4000 features was reduced to a set of 81 features, also following the same process used for ELSA-nurse data. Note that the reduction in this dataset was the greatest yet, because we only included in the TILDA datasets the features that were directly related to what would be 'nurse-data' information, i.e., biomedical features. As mentioned earlier, the TILDA dataset has both core and nurse features in its main dataset, hence the high amount of discarded features.

After this feature selection and creation process, each created dataset has a unique identifier for each instance (respondent), as well as the predictive features (including the sex and age of the participant in the final wave), and the class (target) variable. The predictive features can be divided into "conceptual features", where a conceptual feature may have several measurements of the same basic variable taken over the waves of the study. For instance "cholesterol" is a conceptual feature, and the level of cholesterol at each wave constitutes a specific

[3]The ELSA-nurse, ELSA-core and TILDA databases have several features that describe information about the interview itself, or about other features, such as the reason a test was not conducted. Those were all removed from our dataset because we did not consider them as potentially predictive of the class label.

base feature, whose identification includes the wave number. The features in the ELSA-nurse, ELSA-core and TILDA datasets are described in Appendix A. For each feature, we indicate the waves in the study it appears in, and the data type of its values.

All data sources used in this study have multiple representations for responses that are not one of the expected values, including, e.g., a code for "not applicable" and another for "refusal to answer". These values were coded as "?" (the standard missing value symbol for Weka files), defining the missing values of the dataset.

4.1.5 Creating class labels

For all created datasets, the binary class variable represents the presence or absence of a positive diagnose for each ELSA or TILDA respondent, for an age-related disease or condition, at the wave of the class.

This type of information is not represented directly by any of the variables in the ELSA dataset. Thus, for the ELSA-nurse and ELSA-core datasets we combined information about the diagnosis of each of the target diseases, present in several variables of the ELSA-core questionnaire, to create our class labels.

The ELSA class labels represent diagnosis for Angina, Arthritis, Cataract, Dementia, Diabetes, High blood pressure (HBP), Heart attack, Osteoporosis, Parkinsons Disease, and Stroke. Starting at the third wave of the ELSA-core questionnaire, each respondent was asked, in every wave, questions regarding the diagnosis of these diseases and conditions, and using the answers for these questions we infer a class label for that respondent, for a given 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, in that wave, 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, for each wave t in the range: $3 \leq t \leq 8$, is as follows.

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

With these rules, we were able to infer a class label for every respondent that was participating in the ELSA core study in the waves from 3 to 8. Naturally, all instances included in the dataset represent subjects who participated in the latest wave 8, and the final class label added to the base datasets is the one created from the data in wave 8. This means no instance in the base datasets has a missing class label in wave 8 (if a respondent did not participate in any of the other waves, a missing value was assigned for their class label in that wave). 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 (ELSA respondents who participated in wave 8 of the study) and the same predictive features. This approach for dataset creation was also used by Pomsuwan and Freitas (2017).

It is important to highlight that the ELSA and TILDA participants themselves are reporting the diagnosis of the target diseases in the interviews, and there is no clinical data available corroborating their answers. Thus, even though we take the data available as ground-truth, it is likely that some patients were undiagnosed or did not report their diagnosis (false negatives), and that some patients wrongly reported their positive diagnosis (false positives).

Table 4.1 shows the names of the class variables in terms of age-related diseases, the names of the original ELSA variables used to create the class variables in this work, and the class imbalance ratios for the ELSA-nurse and ELSA-core class variables. The class imbalance ratio (IR) is calculated by dividing the number of majority class instances by the number of minority class instances.

For the TILDA dataset, there were features in the last wave that directly informed whether the participant had been diagnosed with the target disease. These features were used as the class variables for the final wave (wave 4) in our TILDA datasets, and we did not add class variables for other waves. Table 4.2 shows the names of the class variables in terms of age-related diseases, the names of the original TILDA variables used as class variables in this work, and the class

Table 4.1: ELSA-nurse and ELSA-core class variables and their class imbalance ratios.

Class variable	**Elsa-core variables used to create the class label**	**ELSA-nurse Class Imbalance Ratio**	**ELSA-core Class Imbalance Ratio**
Heart Attack	Hedacmi - Whether confirms heart attack diagnosis Hediami - Heart attack diagnosis newly reported	16.70	19.06
Angina	Hedacan - Whether confirms angina diagnosis Hedasan - Whether still has angina Hediaan - Angina diagnosis newly reported Hediman - Angina diagnosis newly reported (merged)	26.51	29.49
Stroke	Hedacst - Whether confirms stroke diagnosis Hediast - Stroke diagnosis newly reported	15.86	18.35
Diabetes	Hedacdi - Whether confirms diabetes or high blood sugar diagnosis Heacd - Whether ever been told has diabetes by doctor Hediadi - Diabetes or high blood sugar diagnosis newly reported	6.50	7.80
High Blood Pressure	Hedacbp - Whether confirms high blood pressure diagnosis Hedasbp - Whether still has high blood pressure Hediabp - High blood pressure diagnosis newly reported Hedimbp - High blood pressure diagnosis newly reported (merged)	1.49	2.58
Dementia	Hedbdde - Whether confirms dementia diagnosis Hedbsde - Whether still has dementia Hedibde - Dementia diagnosis newly reported	56.96	52.20
Cataract	Heopcca - Whether confirms cataract diagnosis Heopsca - Whether still has cataract Heoptca - Cataract diagnosis newly reported	2.06	3.38
Arthritis	Hedbdar - Whether confirms arthritis diagnosis Hedbsar - Whether still has arthritis Hedibar - Arthritis diagnosis newly reported	1.35	2.52
Osteoporosis	Hedbdos - Whether confirms osteoporosis diagnosis Hedbsos - Whether still has osteoporosis Hedibos - Osteoporosis diagnosis newly reported	9.85	11.84
Parkinsons	Hedbdpd - Whether confirms Parkinsons Disease diagnosis Hedbspd - Whether still has Parkinsons Disease Hedibpd - Parkinsons Disease diagnosis newly reported	160.30	112.07

imbalance ratios for the TILDA class variables.

4.2 Missing Value Replacement

As discussed in Chapter 3, datasets from longitudinal studies are prone to high amounts of missing values, mainly due to participants dropping out of the study, or joining it on a later wave (attrition). For the ELSA-nurse, Elsa-core and TILDA datasets used in this thesis, 38.5%, 19.1% and 9.5% (respectively) of the values across all features and waves are missing, which makes the approach to simply drop instances (or features) with missing values inadvisable.

Another option would be to leave the missing values in the dataset, for the classification algorithm to cope with. However, we intend to use features constructed out of temporal data in our project (discussed later, in Chapter 5), and incomplete datasets would affect the algorithm's ability to construct these features. In this

Table 4.2: TILDA class variables and their class imbalance ratios.

Class Variable (Age-related disease)	**TILDA variable used as class variable**	**Class Imbalance Ratio**
High blood pressure or hypertension	PH201_01	2.38
Arthritis (including osteoarthritis, or rheumatism)	PH301_03	2.92
Osteoporosis, sometimes called thin or brittle bones	PH301_04	9.53
Cataracts	PH105_1	10.83
Diabetes or high blood sugar	PH201_05	13.44
Cancer or a malignant tumour (including leukaemia or lymphoma but excluding minor skin cancers)	PH301_05	17.02
Angina	PH201_02	20.70
A heart attack (including myocardial infarction or coronary thrombosis)	PH201_03	25.24
Ministroke or Transient Ischemic Attack (TIA)	PH201_07	50.74
A stroke (cerebral vascular disease)	PH201_06	79.62

context we chose imputation (replacing missing values by estimations) as the main approach to handle missing values for this study, meaning every missing value was replaced by an estimated value in a data preprocessing step, before applying the classification algorithm.

As discussed earlier, there are many ways to estimate missing values (some particular to longitudinal datasets), and selecting the best imputation method is challenging. As our approach for handling the missing data in our datasets, in this Chapter we propose a novel Data-Driven approach to select and apply the most effective imputation method for each feature. We report the results of experiments using our ELSA-core, ELSA-nurse and TILDA datasets as a benchmark to compare the effectiveness of the Data-Driven approach against five missing value replacement methods. These methods were compared in two scenarios: a scenario independent from any classifier, and another scenario where a Random Forest (RF) classifier was trained with datasets with estimated missing values.

4.3 The Chosen Missing Value Replacement Methods

Our experiments use five missing value replacement methods (Gad and Abdelkhalek 2017; Mallinckrodt 2013; Albridge, Standish and Fries 1988), described in Subsections 4.3.1 to 4.3.5; as well as a proposed Data-Driven approach combining these five methods, to be described in Section 4.4. 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, as preprocessing steps done in a classification (supervised learning) setting cannot use class labels from the test set, whilst class labels in the training set can be of course used.

4.3.1 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.

4.3.2 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 similar. Naturally, the age of an individual impacts their overall health, so it is expected that, in general, ELSA and TILDA participants with the same age would have more similar feature values than 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 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. (2019), which replaced missing values with the median from individuals with the same age and sex.

4.3.3 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), (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 datasets there is a gap of 2 (ELSA-core and TILDA) to 4 (ELSA-nurse) years between each pair of adjacent waves, we decided

to consider only values from the previous wave as viable for imputation, to avoid using data too far in the past.

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 dataset.

4.3.4 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 2 to 4-year time gap between waves in our 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 dataset.

4.3.5 K-Nearest Neighbours

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 MVR 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, sex, 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 sex 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 sex 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 a MVR 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.

4.3.6 A conceptual comparison between the five MVR methods

When selecting which methods to compare in our experiments (reported later in Sections 4.6 and 4.7), we aimed to have representations of different types of methods for missing value replacement. 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 ageing datasets.

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 of 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 sex features, to calculate the distance. Thus, it can be considered a longitudinal missing value replacement method (akin to the Prev and PrevNext methods), as it uses time-related information to find the nearest neighbours.

4.4 The Proposed Data-Driven Missing Value Replacement Approach

In addition to the five selected missing value replacement methods discussed in Section 4.3, we propose an approach that selects these methods dynamically, feature-wise, ranking the methods based on information contained in the dataset itself. This strategy, referred to as the Data-Driven approach from here on, can be implemented with any set of missing value replacement methods, in principle. The procedure for applying this method is as follows.

Consider a set of n missing value replacement 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 method 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 MVR 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 4.1.

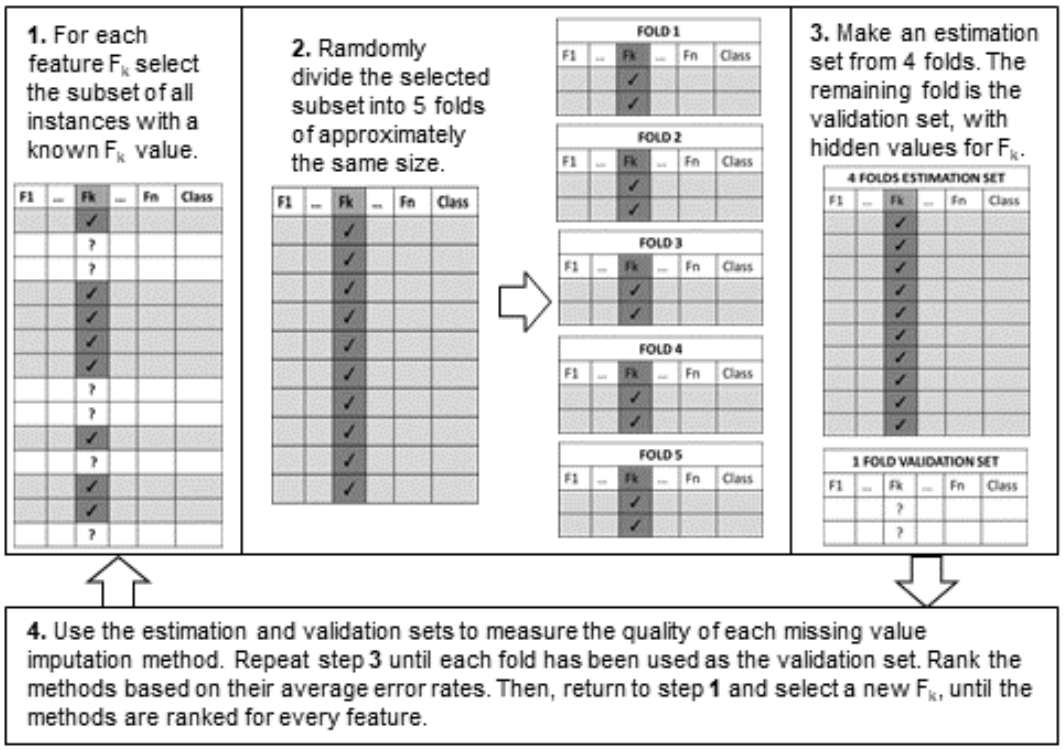

Figure 4.1: Cross-validation approach to evaluate missing value replacement methods.

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

The 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 methods' rankings obtained for $F_{i,t}$. First, it tries to use the first-ranked method to estimate the missing value. If the 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 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. With a different set of 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 MVR 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 MVR 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 replacement methods, and for features with few known

values, the Data-Driven approach might be misled into selecting a poor method.

4.5 Methodology for Evaluating the Proposed Missing Value Replacement Approach

There are two main approaches to evaluate the performance of missing value replacement methods, in the area of supervised machine learning (classification or regression tasks). The first is to use a MVR 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 in this thesis, 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 error rate and bias. The advantage of this type of evaluation is that it provides a comparison that is unrelated to how the chosen machine learning algorithm handles missing values, providing a more generic measure of how accurate the MVR 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 each of our datasets and estimate every known value in the dataset using six MVR methods, and rank them for each feature in the dataset, based on their average estimation error (classifier-independent evaluation). Then, we employed the Random Forest (RF) classification algorithm to evaluate models generated by datasets prepared with each method, and a baseline approach of performing no missing value replacement in a preprocessing step (letting the RF algorithm use its own method for handling missing values).

4.5.1 Related work on classifier-independent comparisons of MVR methods

Several studies discuss handling missing values on longitudinal datasets, and evaluate the performance of different MVR methods. In this Section, we compare some of these evaluation studies to our own. Table 4.3 contains characteristics that describe the selected related works, for comparison with our own study. Regarding the amount of missing data in the datasets, as mentioned earlier, it is common for longitudinal datasets to have a high number 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 datasets used in our experiments have 68 (TILDA), 125 (ELSA-core) and 138 (ELSA-nurse) features with missing values.

Among the cited related works, (Engels and Diehr 2003) has the most similar approach for evaluating MVR 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 26 (TILDA), 30 (ELSA-core) and 45 (ELSA-nurse) longitudinal features, representing a wider diversity of feature types and distributions. In addition, our work includes several nominal features over all datasets, which are treated differently from the numeric features by our MVR methods. In their conclusion, the authors mention that a method able to select the best-fitting MVR 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 4.4.

The missing value replacement methods compared in each of the aforementioned studies are shown in Table 4.4[4], for comparison with our work. The mean imputation and previous observation (usually LOCF) methods are the most common approaches for estimating missing values in longitudinal datasets, and among the more complex methods the Linear Regression and KNN algorithms are often

[4]In Table 4.4, 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 4.3: Number of waves, features with missing values and percentage of missing data in the related works about comparing MVR methods for longitudinal datasets in a classifier-independent scenario. The names in the first four rows refer to the first authors in the references used in the comparison, respectively: (Engels and Diehr 2003), (Belger et al. 2016), (Gad and Abdelkhalek 2017), and (Zhu 2014).

Reference	**Waves**	**Number of Features**			**Missing Values**	**Type of Datasets**
		Numeric	**Nominal**	**Total**		
Engels	10	40	0	40	21.8%	Real
Zhu	5	2	0	2	4-22%	Artificial
Belger	4	1	0	1	10-40%	Artificial
Gad	6	1	1	2	45.6%	Artificial and Real
This study: ELSA-nurse	4	99	39	138	38.5%	Real
This study: ELSA-core	8	7	117	125	19.1%	Real
This study: TILDA	4	39	29	68	9.5%	Real

used. As discussed in Section 4.3, 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 complex methods, based on machine learning (KNN) and our proposed Data-Driven approach, combining these 5 methods. This selection was made to include representative methods from very different approaches to missing value estimation in our experiments.

4.6 Comparing MVR Methods on a Classifier-Independent Scenario

We performed a series of experiments to evaluate the estimation accuracy of the six missing value replacement (MVR) methods (the five methods described in Section 4.3 and the proposed Data-Driven missing value replacement approach described in Section 4.4), in the classifier-independent scenario. The setup used in these experiments can be replicated for comparing any number of missing value

Table 4.4: Missing value replacement 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), (Belger et al. 2016), (Gad and Abdelkhalek 2017), and (Zhu 2014).

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

replacement methods, even outside of the area of classification.

For each MVR 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 are obtained through the same process used in the Data-Driven approach (Section 4.4) to rank the methods for each feature in the dataset: a 5-fold cross-validation where we create known data subsets for each feature, then hide the known values of each feature in the validation fold in turn, and estimate these hidden values using each of the MVR methods, comparing the estimated values to the known values to get an error value.

Regarding the applicability of each method, the 7NN method could not be applied to features that did not have repeated measurements in other waves (2/138 features with missing values in ELSA-nurse, 0/125 in ELSA-core and 3/68 in TILDA), 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 ELSA-nurse and TILDA 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.

All experiments were ran with the datasets derived from the 3 types of data sources used in this research (ELSA-nurse, ELSA-core and TILDA), and the results of this classifier-independent evaluation is shown separately for each data source, in Sections 4.6.1, 4.6.2 and 4.6.3. Note that, since this set of classifier-independent experiments does not depend at all on the class label variable, we only need to run it once for each set of 10 datasets from each data source, as all datasets from the same source share the same predictive features.

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 row of Tables 4.5, 4.6 and 4.7 in the following Sections. These Tables also present the mean error rate over the nominal features, the mean absolute error over the numeric features and over all features, for each MVR 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 feature in the dataset has had its values normalised before the missing value replacement methods were applied. For numerical features, the normalisation method used was min-max, where the normalised value of a

feature is given by the formula: $(f - f_{min})/(f_{max} - f_{min})$, where f is the raw (non-normalized) value of the feature, f_{min} and f_{max} are the minimum and maximum values of the feature in the dataset. Note that this formula produces normalised values in the [0..1] range. For nominal features, we created equidistant values in the [0..1] range for all response options. 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.

4.6.1 ELSA-nurse classifier-independent scenario results

Table 4.5 shows the results of the classifier-independent experiments for the ELSA-nurse dataset.

Table 4.5: Classifier-independent scenario: Elsa-nurse error rates (in [0..1]) of the MVR 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 row shows the applicability (%) of each method. The best result for each row is shown in boldface font.

ELSA-nurse	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Nominal (39)	0.068	0.078	0.055	**0.048**	0.049	**0.048**
Numeric (99)	0.082	0.083	0.078	**0.075**	0.083	0.077
Total (138)	0.078	0.082	0.07	**0.068**	0.077	**0.068**
Applicability	**100%**	97.08%	35.57%	2.95%	81.79%	**100%**

The Data-Driven and PrevNext methods obtained the smallest average errors overall, with both methods getting the same average when considering only nominal features, and when considering all features together. For numeric features, PrevNext had the upper hand by a small margin. However, these values need to be interpreted together with the applicability of each method.

The Prev method had a 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 in ELSA-nurse datasets. 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. 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%.

It is worthwhile to mention that, although it did not obtain the best results, the 7NN method also performed remarkably well, with its performance for nominal features being surpassed only by the proposed Data-Driven approach and the PrevNext method. 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. Note that 7NN achieved an applicability of 81.78%.

Considering both the applicability and the mean error results, the most adequate method is clearly the Data-Driven approach, as it obtained low error values while also successfully estimating every missing value in the ELSA-nurse datasets. The Data-Driven approach makes use of the advantages presented by different methods, and is able to reliably choose, in feature-wise manner, which out of a set of missing value replacement methods is the most effective. However, this may be due to the characteristics of the ELSA-nurse datasets, so I performed this same set of experiments with the ELSA-core and TILDA dataset to confirm whether this pattern would be the same.

4.6.2 ELSA-core classifier-independent scenario results

Table 4.6 shows the results of the classifier-independent experiments for the ELSA-core dataset. The ELSA-core datasets have more waves than our other datasets (7 feature waves, with the class variables being set on the 8th and final wave), which yielded greater applicability to the MVR methods that use longitudinal information.

For the ELSA-core datasets, we see a similar pattern of the MVR methods devised specifically for longitudinal data having small average error rates, while

Table 4.6: Classifier-independent scenario: Elsa-core error rates (in [0..1]) of the MVR 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 117 features) and for numeric features each value is the mean absolute error (over 7 features). The last row shows the applicability (%) of each method. The best result for each row is shown in boldface font.

ELSA-core	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Nominal (117)	0.08	0.083	0.059	**0.054**	0.071	0.06
Numeric (7)	0.104	0.105	**0.096**	0.097	0.101	**0.096**
Total (125)	0.082	0.086	0.06	**0.058**	0.073	0.062
Applicability	**100%**	**100%**	44.61%	5.02%	91.78%	**100%**

also having lower applicability. The proposed Data-Driven approach is, again, one of the best methods regarding its average error rates, and has 100% applicability by definition.

The PrevNext method obtained the smallest error rates considering only nominal features, and considering all features together. The Prev and Data-Driven methods tied for smallest error in the numeric features, which are only 7 out of the 125 predictive features with missing values in the ELSA-core dataset.

Considering both the average error rates and applicability, the Data-Driven approach remains arguably the best choice, even though the applicability of Prev, PrevNext and KNN increased in the ELSA-core datasets, as mentioned.

4.6.3 TILDA classifier-independent scenario results

Table 4.7 shows the results for the classifier-independent experiments with the feature set from the TILDA dataset. This dataset had the lowest applicability for the methods that use longitudinal information. Noticeably, the KNN method's applicability was reduced to 59.45%, from 81.79% on ELSA-nurse and 91.78% on ELSA-core. This is mainly due to the several features in TILDA that were measured in only two waves. Those features were still eligible for replacement using our implementation of KNN, which needs a known value for at least one different measure of the conceptual feature. If we only have one other measurement of a conceptual feature, and its value is also missing, the KNN method could not be applied. Conversely, in a dataset such as ELSA-core, which has several measurements of each feature, the method is much more likely to be applicable.

Table 4.7: Classifier-independent scenario: TILDA error rates (in [0..1]) of the MVR 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 117 features) and for numeric features each value is the mean absolute error (over 7 features). The last row shows the applicability (%) of each method. The best result for each row is shown in boldface font.

TILDA	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Nominal (29)	0.085	0.091	0.06	**0.058**	0.07	0.061
Numeric (39)	0.102	0.102	0.092	**0.091**	0.101	0.094
Total (68)	0.095	0.097	0.077	**0.076**	0.088	0.081
Applicability	**100%**	96.17%	16.82%	2.41%	59.45%	**100%**

In this set of experiments PrevNext got the smallest average error in all cases, followed by the Prev method, then by the Data-Driven approach in third place. However, as mentioned earlier, the applicabilities of the longitudinal MVR methods were hindered by characteristics of the TILDA dataset, with only 2.41% and 16.82% of the missing values being replaced by the PrevNext and Prev methods, respectively.

As the average error values of the proposed Data-Driven approach were still considerably close to the winner method, and it is applicable to every single missing value in the dataset, the recommendation for this method as the best choice remains, even for the TILDA feature set.

4.6.4 Classifier-independent results summary

To summarise, in this Section we performed a classifier-independent comparison of a set of missing value replacement methods. The experimental results showed that each of the six tested MVR methods was the most accurate for some features in the datasets, which corroborates the notion that no single MVR 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 superiority of the Data-Driven approach can be summarised by focusing on the error rate over all features (i.e. both nominal and numerical features) across all three tables with results for the classifier-independent scenario – i.e. Tables 4.5, 4.6 and 4.7 – and focusing only on the approaches

that had an applicability rate greater than 50%. In this context, the Data-Driven approach always achieved the smallest error among all such approaches in all the three aforementioned Tables, often by a substantial difference with respect to the error rate of the second best method with applicability rate over 50%, which was always KNN. More precisely, over all features, in Table 4.5 the Data-Driven approach obtained an error rate of 6.8% against 7.7% of KNN; in Table 4.6 the Data-Driven approach and KNN obtained error rates of 6.2% and 7.3% respectively; and in Table 4.7 they obtained error rates of 8.1% and 8.8% respectively.

As mentioned earlier, the two methods devised specifically for longitudinal data (Prev and PrevNext) had very low applicability. However, they had some of the smallest average error rates in all datasets. That, together with the good performance of the 7NN method (which also had longitudinal characteristics in our implementation), shows the value of considering the often ignored temporal aspect of the data when handling missing values in a longitudinal dataset.

4.7 Comparing MVR 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 4.3 and 4.4, on the predictive accuracy of Random Forest (RF) classifiers, using 10-fold cross-validation in all our 30 datasets.

All datasets had their missing values replaced in a data preprocessing step. For all experiments in this Section, each missing value replacement (MVR) 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 MVR methods compared in the classifier-independent scenario (Section 4.6), for the experiments in this current Section with the Random Forest (RF) classifier we added a baseline approach of not using any of the MVR 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 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 MVR methods compared in this Section are each of the five methods described in Section 4.3 and our Data-Driven approach (Section 4.4), 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.

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), as discussed in Section 2.2.5. The IR value is an indication of how imbalanced the class distribution of a dataset is, and our 30 datasets have very different levels of class imbalance, with IR values ranging from 1.35 (Arthritis on ELSA-nurse datasets) to 160.3 (Parkinson's Disease on ELSA-nurse datasets), depending on how rare the age-related disease is and the distribution of the data available.

As mentioned earliler, the RFs were trained and tested using the Weka. The

RFs were trained 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.

4.7.1 Comparing undersampling strategies on the ELSA-nurse datasets

Because of the class imbalance problem present in all our datasets, before comparing the MVR methods we needed to decide on a strategy to reduce the bias towards the majority class in our datasets. Thus, we performed experiments using the ELSA-nurse datasets, with two random undersampling methods that bring the ratio of positive to negative instances in the training set down to a 1:1 ratio. This 1:1 ratio (for each instance of the minority class in the training set, only one instance of the majority class is kept) 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 2017).

The set of experiments reported in this Section was not executed with the ELSA-core and TILDA datasets because those were prepared at a later stage of our project, and it would be too time-consuming to have a separate set of experiments with each data source for this part of the thesis. We deemed this not cost-effective because we are using standard class-balancing strategies which are well established in the literature, since proposing a class-balancing strategy is not an objective in this project.

In the class imbalance experiments, the ELSA-nurse training sets were balanced through the following strategies: a) removing instances from the majority class in a data 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. We compare these two methods in this Section, and apply the chosen method in all experiments in this thesis.

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 ration (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 UBB method would generate classifiers that are more overfitted to a part of the training set than classifiers generated with the BRF. In order to confirm that notion, and to analyse the overfitting in our RFs, we performed experiments comparing the UBB and BRF methods.

In all experiments comparing classification models in this thesis, the RF classifiers were evaluated based on the following metrics: Sensitivity (True Positive Rate), Specificity (True Negative Rate), Accuracy (percentage of correct classifications) and GMean (Geometric mean between Sensitivity and Specificity). These metrics were chosen based on (Malley, Malley and Pajevic 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. 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).

Tables 4.8 and 4.9 show the average Sensitivity (True Positive rate) and Specificity (True Negative rate) of the RF models, over a 10-fold cross-validation. Recall that in this work the positive (negative) class is the majority (minority) class. 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, meant each method got 0.5 'win' points.

Table 4.8: ELSA-nurse average Sensitivity values for the UBB and BRF undersampling methods for each dataset/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	Baseline		Globalmean		Agebased		Prev		PrevNext		KNN		Data-Driven	
(IR)	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB
Art. (1.35)	**0.72**	0.695	0.661	**0.664**	0.649	**0.678**	**0.729**	0.665	**0.729**	0.645	0.658	**0.681**	0.667	**0.679**
HBP (1.49)	0.641	**0.652**	0.656	**0.692**	0.653	**0.705**	**0.647**	**0.647**	**0.644**	**0.644**	0.642	**0.700**	0.659	**0.697**
Cat. (2.06)	**0.663**	0.660	0.626	**0.673**	0.612	**0.670**	**0.693**	0.659	**0.671**	0.630	0.611	**0.676**	0.629	**0.672**
Dia. (6.5)	**0.744**	0.654	**0.84**	0.765	**0.83**	0.752	**0.794**	0.680	**0.746**	0.653	**0.841**	0.782	**0.841**	0.781
Ost. (9.85)	**0.632**	**0.632**	0.647	**0.688**	0.656	**0.700**	**0.65**	0.638	**0.641**	0.630	0.655	**0.685**	0.653	**0.689**
Str. (15.86)	0.602	**0.616**	0.675	**0.699**	0.681	**0.705**	**0.625**	0.622	**0.601**	0.574	0.672	**0.697**	0.675	**0.691**
H. A. (16.7)	**0.657**	0.626	0.681	**0.699**	0.671	**0.696**	**0.642**	0.615	**0.664**	0.624	0.7	**0.702**	0.678	**0.718**
Ang. (26.51)	**0.619**	0.611	0.659	**0.698**	0.657	**0.694**	**0.641**	0.633	**0.625**	0.611	0.678	**0.689**	0.653	**0.702**
Dem. (56.96)	0.695	**0.703**	0.728	**0.745**	0.75	**0.755**	**0.696**	0.675	**0.691**	0.671	0.729	**0.764**	0.728	**0.753**
P. D. (160.3)	**0.625**	0.567	0.624	**0.656**	0.632	**0.664**	**0.606**	0.541	**0.618**	0.537	0.588	**0.660**	0.616	**0.633**
Wins	**6.5**	3.5	1	**9**	1	**9**	**9.5**	0.5	**9.5**	0.5	1	**9**	1	**9**

Table 4.9: ELSA-nurse average Specificity values for the UBB and BRF undersampling methods for each dataset/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	Baseline		Globalmean		Agebased		Prev		PrevNext		KNN		Data-Driven	
(IR)	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB
Art. (1.35)	0.542	**0.548**	**0.593**	0.510	**0.587**	0.501	0.531	**0.545**	0.521	**0.556**	**0.591**	0.502	**0.598**	0.507
HBP (1.49)	**0.738**	0.701	**0.747**	0.629	**0.742**	0.601	**0.717**	0.666	**0.727**	0.679	**0.744**	0.601	**0.752**	0.610
Cat. (2.06)	0.662	**0.674**	**0.702**	0.601	**0.733**	0.596	**0.652**	0.637	0.645	**0.694**	**0.728**	0.596	**0.715**	0.599
Dia. (6.5)	**0.87**	0.795	**0.864**	0.698	**0.84**	0.655	**0.883**	0.803	**0.871**	0.826	**0.871**	0.661	**0.878**	0.686
Ost. (9.85)	**0.735**	0.716	**0.723**	0.606	**0.688**	0.601	**0.703**	0.673	**0.73**	0.702	**0.713**	0.624	**0.714**	0.578
Str. (15.86)	**0.737**	0.727	**0.715**	0.596	**0.667**	0.572	**0.774**	0.696	0.721	**0.734**	**0.72**	0.577	**0.732**	0.558
H. A. (16.7)	**0.735**	0.718	**0.731**	0.564	**0.728**	0.581	**0.781**	0.736	**0.716**	0.703	**0.736**	0.586	**0.723**	0.611
Ang. (26.51)	**0.737**	0.729	**0.721**	0.585	**0.682**	0.496	**0.767**	0.667	0.708	**0.721**	**0.686**	0.566	**0.733**	0.570
Dem. (56.96)	0.754	**0.791**	**0.736**	0.662	**0.73**	0.676	**0.804**	0.784	**0.754**	0.743	**0.709**	0.635	**0.743**	0.615
P. D. (160.3)	**0.672**	0.591	**0.682**	0.561	**0.667**	0.545	**0.621**	**0.621**	**0.689**	0.561	**0.636**	0.485	**0.636**	0.470
Wins	**7**	3	**10**	0	**10**	0	**8.5**	1.5	**6**	4	**10**	0	**10**	0

Table 4.8 shows a noticeable trend for higher Sensitivity values when using the UBB method, except for the Baseline MVR (no replacement). For all other MVR method the Sensitivity values of UBB models were higher in at least 9 out of 10 cases.

On the other hand, as seen in Table 4.9, overall higher Specificity values were observed when using the BRF method, for all 7 MVR methods. The BRF won for all 10 datasets in 4 of these methods, showing a clear trend of better Specificity values.

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.

As global measures of the RF models' performances, their average Accuracy values are reported in Table 4.10 and the GMean values are reported in Table 4.11.

Table 4.10: ELSA-nurse average Accuracy values for the UBB and BRF undersampling methods for each dataset/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	Baseline		Globalmean		Agebased		Prev		PrevNext		KNN		Data-Driven	
(IR)	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB
Art. (1.35)	0.641	**0.672**	0.632	**0.667**	0.623	**0.664**	0.645	**0.655**	0.637	**0.658**	0.629	**0.66**	0.638	**0.662**
HBP (1.49)	**0.681**	0.632	**0.693**	0.599	**0.689**	0.603	**0.675**	0.614	**0.679**	0.607	**0.683**	0.605	**0.697**	0.606
Cat. (2.06)	0.663	**0.665**	**0.651**	0.649	**0.651**	0.645	**0.68**	0.652	**0.662**	0.651	0.649	**0.65**	**0.657**	0.648
Dia. (6.5)	**0.762**	0.673	**0.843**	0.756	**0.831**	0.739	**0.805**	0.696	**0.763**	0.676	**0.845**	0.766	**0.846**	0.768
Ost. (9.85)	**0.642**	0.639	0.654	**0.68**	0.659	**0.691**	**0.655**	0.641	**0.649**	0.636	0.661	**0.679**	0.658	**0.679**
Str. (15.86)	0.61	**0.615**	0.677	**0.694**	0.68	**0.687**	**0.634**	**0.634**	0.608	**0.615**	0.675	**0.684**	0.678	**0.697**
H. A. (16.7)	**0.662**	0.623	0.684	**0.693**	0.675	**0.697**	**0.65**	0.627	**0.667**	0.584	**0.702**	0.69	0.681	**0.683**
Ang. (26.51)	0.624	**0.632**	0.661	**0.692**	0.658	**0.689**	**0.645**	0.622	0.628	**0.629**	0.678	**0.695**	0.656	**0.712**
Dem. (56.96)	0.696	**0.704**	0.728	**0.743**	0.749	**0.754**	**0.698**	0.677	**0.692**	0.672	0.728	**0.761**	0.728	**0.75**
P. D. (160.3)	**0.625**	0.568	0.624	**0.655**	0.632	**0.663**	**0.606**	0.542	**0.619**	0.537	0.589	**0.658**	0.616	**0.632**
Wins	**5**	**5**	3	**7**	3	**7**	**8.5**	1.5	**7**	3	3	**7**	3	**7**

Table 4.11: ELSA-nurse average GMean values for the UBB and BRF undersampling methods for each dataset/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	Baseline		Globalmean		Agebased		Prev		PrevNext		KNN		Data-Driven	
(IR)	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB	BRF	UBB
Art. (1.35)	**0.625**	0.617	**0.626**	0.582	**0.617**	0.583	**0.622**	0.602	**0.616**	0.598	**0.623**	0.585	**0.632**	0.587
HBP (1.49)	**0.688**	0.676	**0.7**	0.660	**0.696**	0.651	**0.681**	0.657	**0.684**	0.661	**0.691**	0.648	**0.704**	0.652
Cat. (2.06)	0.663	**0.667**	**0.663**	0.636	**0.669**	0.632	**0.672**	0.648	0.658	**0.661**	**0.667**	0.634	**0.671**	0.635
Dia. (6.5)	**0.804**	0.721	**0.852**	0.731	**0.835**	0.702	**0.837**	0.739	**0.806**	0.734	**0.856**	0.719	**0.86**	0.732
Ost. (9.85)	**0.682**	0.672	**0.684**	0.645	**0.672**	0.649	**0.676**	0.655	**0.684**	0.665	**0.683**	0.654	**0.683**	0.631
Str. (15.86)	0.666	**0.669**	**0.694**	0.646	**0.674**	0.635	**0.696**	0.658	**0.658**	0.649	**0.695**	0.634	**0.703**	0.621
H. A. (16.7)	**0.695**	0.671	**0.706**	0.628	**0.699**	0.636	**0.708**	0.673	**0.69**	0.662	**0.717**	0.641	**0.7**	0.663
Ang. (26.51)	**0.676**	0.667	**0.689**	0.639	**0.67**	0.587	**0.701**	0.650	**0.665**	0.664	**0.682**	0.624	**0.692**	0.632
Dem. (56.96)	0.724	**0.745**	**0.732**	0.702	**0.74**	0.714	**0.748**	0.727	**0.721**	0.706	**0.719**	0.697	**0.736**	0.681
P. D. (160.3)	**0.648**	0.579	**0.652**	0.606	**0.649**	0.602	**0.613**	0.580	**0.652**	0.549	**0.612**	0.566	**0.626**	0.545
Wins	**7**	3	**10**	0	**10**	0	**10**	0	**9**	1	**10**	0	**10**	0

In the Accuracy analysis, the UBB method performs better than the BRF class-balancing method with 4 out of 7 MVR methods. The trend in favour of UBB is slightly less clear, with BRF getting more wins for the Prev and PrevNext methods and a tie of 5 wins for the Baseline method. It is important to note that Accuracy favours the Sensitivity results, as the positive (majority) class has a bigger weight in its values, thus the trend in favour of UBB was to be expected, since it got the best Sensitivity results overall.

The other global measure, GMean, clearly trends towards the BRF method, with more wins for all MVR methods, including wins for all 10 datasets for 5 out of 7 MVR methods. This means that, when we consider the classification of both positive and negative cases as equally important, the BRF models were clearly superior.

Therefore, when analysing the performance of the RF models, the BRF method was overall superior to the UBB method. Note that, when applying the BRF method, the models are 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. Note that this also applies to the ELSA-core and TILDA datasets, in addition to the ELSA-nurse datasets we experimented with in this Section.

4.7.2 Comparing the MVR Methods

Once we have chosen the BRF undersampling as the default method for handling the class imbalance in all our datasets, as discussed in the previous Section, in this Section we analyse which of the missing value replacement (MVR) methods is the most adequate for our datasets.

For this set of experiments, we will again present the results for the ELSA-nurse, ELSA-core and TILDA datasets separately, in the following Subsections.

Elsa-nurse classifier-dependent results

The Sensitivity and Specificity results obtained on ELSA-nurse datasets by each MVR method (with BRF undersampling) are presented in Tables 4.12 and 4.13, 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 (average between ranks 1 and 2).

Table 4.12: Elsa-nurse: average Sensitivity 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Arthritis (1.35)	0.720	0.661	0.649	**0.729**	**0.729**	0.658	0.667
HBP (1.49)	0.641	0.656	0.653	0.647	0.644	0.642	**0.659**
Cataract (2.06)	0.663	0.626	0.612	**0.693**	0.671	0.611	0.629
Diabetes (6.5)	0.744	0.840	0.830	0.794	0.746	**0.841**	**0.841**
Osteoporosis (9.85)	0.632	0.647	**0.656**	0.650	0.641	0.655	0.653
Stroke (15.86)	0.602	0.675	**0.681**	0.625	0.601	0.672	0.675
HeartAttack (16.7)	0.657	0.681	0.671	0.642	0.664	**0.700**	0.678
Angina (26.51)	0.619	0.659	0.657	0.641	0.625	**0.678**	0.653
Dementia (56.96)	0.695	0.728	**0.750**	0.696	0.691	0.729	0.728
Parkinsons (160.3)	0.625	0.624	**0.632**	0.606	0.618	0.588	0.616
Average Rank	5.4	3.3	**3.1**	4.4	5.0	3.8	3.2

Table 4.13: Elsa-nurse: average Specificity 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Arthritis (1.35)	0.542	0.593	0.587	0.531	0.521	0.591	**0.598**
HBP (1.49)	0.738	0.747	0.742	0.717	0.727	0.744	**0.752**
Cataract (2.06)	0.662	0.702	**0.733**	0.652	0.645	0.728	0.715
Diabetes (6.5)	0.870	0.864	0.840	**0.883**	0.871	0.871	0.878
Osteoporosis (9.85)	**0.735**	0.723	0.688	0.703	0.730	0.713	0.714
Stroke (15.86)	0.737	0.715	0.667	**0.774**	0.721	0.720	0.732
HeartAttack (16.7)	0.735	0.731	0.728	**0.781**	0.716	0.736	0.723
Angina (26.51)	0.737	0.721	0.682	**0.767**	0.708	0.686	0.733
Dementia (56.96)	0.754	0.736	0.730	**0.804**	0.754	0.709	0.743
Parkinsons (160.3)	0.672	0.682	0.667	0.621	**0.689**	0.636	0.636
Average Rank	3.4	3.8	5.2	3.7	4.5	4.2	**3.3**

Regarding the Sensitivity values (Table 4.12), the Age-based mean/mode method

obtained the lowest (best) average rank (3.1) and 4 best values across the 10 datasets, closely followed by the Data-Driven method with the second lowest average rank (3.2) and 2 best values.

Regarding the Specificity results (Table 4.13), the Age-based mean/mode method obtained the highest (worst) average rank, showing the typical trade-off between Sensivity and Specificity. However, the Data-Driven approach still presented good results for Specificity, with the smallest (best) average rank (3.3), followed by the Baseline approach (3.4). This indicates that the classifiers created with the Data-Driven approach were balanced enough to obtain the best Specificity and the second best Sensitivity among the 7 MVR methods, despite the trade-off usually associated with these two metrics.

For the global analysis of the classifiers generated with ELSA-nurse datasets, we present the models' Accuracy and GMean results in Tables 4.14 and 4.15.

Table 4.14: Elsa-nurse: 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Arthritis (1.35)	0.641	0.632	0.623	**0.645**	0.637	0.629	0.638
HBP (1.49)	0.681	0.693	0.689	0.675	0.679	0.683	**0.697**
Cataract (2.06)	0.663	0.651	0.651	**0.680**	0.662	0.649	0.657
Diabetes (6.5)	0.762	0.843	0.831	0.805	0.763	0.845	**0.846**
Osteoporosis (9.85)	0.642	0.654	0.659	0.655	0.649	**0.661**	0.658
Stroke (15.86)	0.610	0.677	**0.680**	0.634	0.608	0.675	0.678
HeartAttack (16.7)	0.662	0.684	0.675	0.650	0.667	**0.702**	0.681
Angina (26.51)	0.624	0.661	0.658	0.645	0.628	**0.678**	0.656
Dementia (56.96)	0.696	0.728	**0.749**	0.698	0.692	0.728	0.728
Parkinsons (160.3)	0.625	0.624	**0.632**	0.606	0.619	0.589	0.616
Average Rank	5.0	3.4	3.2	4.6	5.4	3.6	**2.9**

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.

Table 4.15: Elsa-nurse: average GMean 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Arthritis (1.35)	0.625	0.626	0.617	0.622	0.616	0.623	**0.632**
HBP (1.49)	0.688	0.700	0.696	0.681	0.684	0.691	**0.704**
Cataract (2.06)	0.663	0.663	0.669	**0.672**	0.658	0.667	0.671
Diabetes (6.5)	0.804	0.852	0.835	0.837	0.806	0.856	**0.860**
Osteoporosis (9.85)	0.682	**0.684**	0.672	0.676	**0.684**	0.683	0.683
Stroke (15.86)	0.666	0.694	0.674	0.696	0.658	0.695	**0.703**
HeartAttack (16.7)	0.695	0.706	0.699	0.708	0.690	**0.717**	0.700
Angina (26.51)	0.676	0.689	0.670	**0.701**	0.665	0.682	0.692
Dementia (56.96)	0.724	0.732	0.740	**0.748**	0.721	0.719	0.736
Parkinsons (160.3)	0.648	**0.652**	0.649	0.613	**0.652**	0.612	0.626
Average Rank	5.2	3.0	4.5	3.5	5.6	4.0	**2.4**

In the Accuracy results the Data-Driven approach had the smallest average rank (2.9), followed by the proposed Age-based mean/mode (average rank of 3.2), a similar trend to what we observed when analysing Sensitivity (Table 4.13). When considering Accuracy (Table 4.14), the Baseline models (learned from datasets without any missing value replacement) only outperformed the PrevNext method.

The GMean results (Table 4.15) also have the Data-Driven approach as the method with the smallest average rank (2.4). This other global performance metric puts the same weight in the Sensitivity and Specificity of the classifier, meaning it is not very affected by the class imbalance in the datasets. Based on these results we can claim that models created with the proposed Data-Driven approach tend to have better performance, which corroborates our previous results from the classifier-independent evaluation, which had the proposed method as the best choice overall, considering applicability and error rate. Importantly, the Baseline approach of doing no missing value replacement at all is outperformed by the proposed method in all four metrics.

ELSA-core classifier-dependent results

We ran the same set of classifier-dependent experiments using the 10 ELSA-core datasets, to investigate whether the proposed Data-Driven approach would still

outperform the other methods in a different scenario. As mentioned earlier, ELSA-core datasets have more waves than our other datasets, which yielded greater applicability to the MVR methods that use longitudinal information.

Tables 4.16 and 4.17 show the Sensitivity and Specificity results for the ELSA-core datasets, respectively.

Table 4.16: Elsa-core: average Sensitivity 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Arthritis (2.52)	0.775	0.750	0.749	**0.803**	0.773	0.748	0.754
HBP (2.58)	0.638	0.644	0.630	**0.700**	0.632	0.634	0.641
Cataract (3.38)	0.694	0.598	0.607	**0.740**	0.696	0.607	0.600
Diabetes (7.80)	0.634	0.670	0.668	**0.690**	0.640	0.677	0.679
Osteoporosis (11.84)	0.721	0.697	0.695	**0.748**	0.726	0.700	0.695
Stroke (18.35)	0.660	0.689	0.690	**0.737**	0.667	0.694	0.690
HeartAttack (19.06)	0.673	0.674	0.671	**0.721**	0.692	0.676	0.668
Angina (29.49)	0.695	0.710	0.713	**0.735**	0.696	0.715	0.709
Dementia (52.20)	0.716	0.761	0.765	**0.791**	0.717	0.760	0.763
Parkinsons (112.07)	0.723	0.694	0.702	**0.761**	0.722	0.694	0.692
Average Rank	4.7	4.55	4.75	**1**	4.2	4.1	4.7

Table 4.17: Elsa-core: average Specificity 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Arthritis (2.52)	0.680	0.714	**0.718**	0.676	0.681	**0.718**	0.714
HBP (2.58)	0.658	**0.671**	**0.671**	0.614	0.654	0.663	0.667
Cataract (3.38)	0.564	0.730	**0.755**	0.569	0.560	0.730	0.736
Diabetes (7.80)	**0.791**	0.746	0.745	0.725	0.787	0.758	0.737
Osteoporosis (11.84)	0.626	0.668	0.665	0.592	0.622	0.665	**0.673**
Stroke (18.35)	0.719	**0.707**	0.729	0.659	0.696	0.699	0.718
Heart Attack (19.06)	0.677	0.683	0.676	0.642	0.661	**0.696**	0.680
Angina (29.49)	**0.766**	0.733	0.761	0.730	0.742	0.737	0.758
Dementia (52.20)	**0.821**	0.789	0.764	0.733	0.807	0.776	0.783
Parkinson's (112.07)	0.754	0.707	0.733	0.653	0.738	0.720	**0.760**
Average Rank	3.3	3.55	3.05	6.8	4.7	3.65	**2.95**

For these models, the Prev method (which has a 44.61% applicability in the ELSA-core datasets) got the best Sensitivity results in all 10 datasets. The Sensitivity values from the other methods were closer together, with average ranks

varying from 4.1 (KNN) to 4.75 (Age-based mean/mode). The high Sensitivity of the Prev models is, in this case, an indication of a tendency to classify instances as the majority class, which increases the True Positive Rate while simultaneously decreasing the True Negative Rate.

This is made clear on the Specificity results in Table 4.17, where Prev has one of the highest (worst) average ranks (6.8). For this metric, the Data-Driven method obtained the best results (average rank of 2.95), although the pattern of 'wins' was not as clear as in the ELSA-nurse datasets.

The Accuracy and GMean results for the ELSA-core datasets are presented in Tables 4.18 and 4.19, as global performance measures.

Table 4.18: Elsa-core: 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Arthritis (2.52)	0.734	0.736	0.737	**0.752**	0.734	0.736	0.738
HBP (2.58)	0.647	0.654	0.646	**0.667**	0.641	0.646	0.651
Cataract (3.38)	0.649	0.637	0.651	**0.689**	0.650	0.644	0.640
Diabetes (7.80)	0.656	0.679	0.677	**0.694**	0.660	0.687	0.687
Osteoporosis (11.84)	0.712	0.694	0.692	**0.735**	0.717	0.697	0.693
Stroke (18.35)	0.663	0.690	0.692	**0.733**	0.668	0.695	0.691
Heart Attack (19.06)	0.673	0.674	0.671	**0.717**	0.690	0.677	0.668
Angina (29.49)	0.697	0.711	0.714	**0.735**	0.698	0.716	0.711
Dementia (52.20)	0.718	0.761	0.765	**0.790**	0.719	0.760	0.763
Parkinson's (112.07)	0.723	0.694	0.702	**0.760**	0.722	0.694	0.692
Average Rank	5.25	4.55	4.05	**1**	4.75	3.9	4.5

As mentioned earlier, the Accuracy metric is more heavily influenced by the Sensitivity (as the positive class is the majority class), and this led to the same pattern: the Prev method obtained the best result for every single ELSA-core dataset. The other methods in Table 4.18 have average ranks ranging from 3.9 (KNN) to 5.25 (Baseline), a slightly broader range when compared to the Sensitivity results in Table 4.16.

Regarding the GMean results (Table 4.19), which put the same weight into correctly classifying majority and minority class instances, we see the Data-driven approach as the best method (average rank 2.8). In this case, as the Prev models seem to have a heavy bias towards the majority class, the GMean metric is the most indicated out of the two to show which method generated the models with

Table 4.19: Elsa-core: average GMean 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
Arthritis (2.52)	0.726	0.732	0.733	**0.736**	0.726	0.733	0.734
HBP (2.58)	0.648	**0.657**	0.650	0.656	0.643	0.649	0.654
Cataract (3.38)	0.625	0.661	**0.677**	0.649	0.624	0.666	0.664
Diabetes (7.80)	0.708	0.707	0.705	0.707	0.710	**0.716**	0.708
Osteoporosis (11.84)	0.671	0.682	0.680	0.665	0.672	0.682	**0.684**
Stroke (18.35)	0.689	0.698	**0.709**	0.697	0.681	0.697	0.704
Heart Attack (19.06)	0.675	0.678	0.673	0.680	0.677	**0.686**	0.674
Angina (29.49)	0.730	0.722	**0.737**	0.733	0.719	0.726	0.733
Dementia (52.20)	0.767	**0.775**	0.765	0.761	0.761	0.768	0.773
Parkinson's (112.07)	**0.738**	0.700	0.717	0.705	0.730	0.707	0.725
Average Rank	4.8	3.8	3.75	4.2	5.4	3.25	**2.8**

better predictive accuracy.

TILDA classifier-dependent results

Finally, we also ran the classifier-dependent scenario experiments with the 10 TILDA datasets. The Irish study datasets have the lowest applicabilities for the methods that use longitudinal information, which means the Prev and PrevNext have very similar effect to using the Baseline approach, since every instance for which the method is not applicable stays with a missing value to be handled by the classification algorithm.

The Sensitivity and Specificity results for the models created with TILDA datasets are shown in Tables 4.20 and 4.21, respectively.

In these experiments, the Data-Driven approach obtained the lowest average rank for Specificity (2.95), although the 'wins' were relatively well-distributed, with Data-Driven getting the best Sensitivity value only twice. The Age-based and Global mean/mode methods follow with average ranks of 3.25 and 3.3, respectively.

For the Specificity results, the Baseline method got the lowest average rank (2.95), with our proposed Data-Driven approach having the second-to-highest rank (4.65). This may be an indication of a small bias in favour of the majority class in the Data-Driven models, but it is not nearly as strong as the one observed in the Prev models from the ELSA-core experiments. The highest Specificity

Table 4.20: TILDA: average Sensitivity 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
HBP (2.38)	0.635	0.677	0.673	0.625	0.633	0.653	**0.684**
Arthritis (2.92)	**0.747**	0.724	0.726	0.742	0.737	0.734	0.734
Osteoporosis (9.53)	**0.688**	0.680	0.676	0.680	0.672	0.681	0.677
Cataract (10.83)	0.710	0.698	**0.715**	0.703	0.633	0.691	0.706
Diabetes (13.44)	0.744	**0.781**	0.776	0.741	0.746	0.749	0.776
Cancer (17.02)	0.573	0.551	0.553	**0.583**	0.572	0.557	0.562
Angina (20.70)	0.729	**0.751**	0.750	0.730	0.729	0.735	0.749
Heart Attack (25.24)	0.728	0.747	**0.749**	0.727	0.733	0.733	0.747
Mini-stroke (50.74)	0.696	0.707	0.707	0.707	0.688	0.704	**0.711**
Stroke (79.62)	0.705	**0.724**	0.722	0.706	0.708	0.702	0.713
Average Rank	4.15	3.3	3.25	4.45	5.3	4.6	**2.95**

Table 4.21: TILDA: average Specificity 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
HBP (2.38)	0.795	0.760	0.767	**0.802**	0.792	0.789	0.764
Arthritis (2.92)	0.646	0.650	**0.651**	0.649	0.636	0.649	0.649
Osteoporosis (9.53)	0.801	0.797	0.796	0.781	0.794	**0.807**	0.796
Cataract (10.83)	0.736	0.743	0.743	0.738	0.720	**0.749**	0.724
Diabetes (13.44)	0.826	0.823	0.818	0.836	0.823	**0.842**	0.834
Cancer (17.02)	**0.612**	0.595	0.592	0.576	0.595	0.599	0.599
Angina (20.70)	**0.908**	0.892	0.896	**0.908**	**0.908**	0.900	0.884
Heart Attack (25.24)	0.912	0.873	0.873	**0.917**	0.907	0.912	0.878
Mini-stroke (50.74)	**0.794**	0.725	0.765	0.735	0.765	0.755	0.755
Stroke (79.62)	0.754	0.738	**0.785**	0.754	**0.785**	0.738	0.754
Average Rank	**2.95**	5.05	4.15	3.8	4.3	3.1	4.65

values are well-distributed between methods, with the Baseline approach getting 3 out of 10 wins.

The Accuracy and GMean results for the TILDA datasets are shown in Tables 4.22 and 4.23, respectively.

The Accuracy results have the Data-Driven approach with the lowest average rank (2.9), which is expected since it had the best Sensitivity results as well. However, this time the GMean results have the Baseline approach as the lowest ranked method on average (3.2), very closely followed by the Age-based approach (3.3) and the Data-Driven approach (3.4).

Table 4.22: TILDA: 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
HBP (2.38)	0.696	0.709	0.709	0.693	0.693	0.704	**0.714**
Arthritis (2.92)	**0.716**	0.701	0.703	0.713	0.706	0.708	0.707
Osteoporosis (9.53)	**0.698**	0.691	0.687	0.690	0.684	0.693	0.689
Cataract (10.83)	0.712	0.702	**0.717**	0.706	0.640	0.696	0.707
Diabetes (13.44)	0.750	**0.784**	0.778	0.748	0.751	0.755	0.780
Cancer (17.02)	0.575	0.553	0.555	**0.582**	0.574	0.559	0.564
Angina (20.70)	0.737	**0.757**	0.756	0.738	0.736	0.743	0.755
Heart Attack (25.24)	0.734	0.752	**0.753**	0.734	0.739	0.739	0.751
Mini-stroke (50.74)	0.698	0.707	0.708	0.707	0.690	0.705	**0.711**
Stroke (79.62)	0.705	**0.724**	0.723	0.707	0.709	0.702	0.714
Average Rank	4.15	3.3	3.15	4.45	5.6	4.45	**2.9**

Table 4.23: TILDA: average GMean 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.

Class (IR)	Baseline	GlobalMean	AgeBased	Prev	PrevNext	KNN	DataDriven
HBP (2.38)	0.711	0.717	0.719	0.708	0.708	0.718	**0.723**
Arthritis (2.92)	**0.695**	0.686	0.688	0.694	0.685	0.690	0.690
Osteoporosis (9.53)	**0.742**	0.736	0.733	0.729	0.730	0.741	0.734
Cataract (10.83)	0.723	0.720	**0.729**	0.721	0.675	0.719	0.715
Diabetes (13.44)	0.784	0.802	0.797	0.787	0.784	0.794	**0.804**
Cancer (17.02)	**0.592**	0.573	0.572	0.579	0.584	0.577	0.580
Angina (20.70)	0.814	0.818	**0.820**	0.814	0.813	0.814	0.814
Heart Attack (25.24)	0.815	0.808	0.809	0.816	0.816	**0.818**	0.810
Mini-stroke (50.74)	**0.743**	0.716	0.735	0.721	0.726	0.729	0.732
Stroke (79.62)	0.729	0.731	**0.753**	0.730	0.745	0.720	0.733
Average Rank	**3.2**	4.5	3.3	4.55	5.15	3.9	3.4

Therefore, for the TILDA datasets, the proposed Data-Driven missing value replacement approach still presented good results in the comparison against the 6 other approaches, but it is not a clear best choice when compared to the Baseline approach of not doing any imputation.

Statistical analysis

To further investigate the difference between the RF models' performances, we compared their results over all 30 datasets using two non-parametric statistical

significance tests, as follows. We applied the Friedman's test (multiple simultaneous comparisons) to the results of all 4 metrics, for the 7 missing value-handling techniques, with the usual significance level of $\alpha = 0.05$. The tests resulted in the following *p-values*: $1.529e-05$, $2.554e-08$, $1.112e-09$ and $1.48e-10$, for Sensitivity, Specificity, Accuracy and GMean respectively. This means that, for all four measures, there was enough evidence to reject the null hypothesis (that the models' performances were equivalent).

As a post-hoc statistical test, using the Data-Driven approach as a control classifier (i.e., the one classifier we intend to compare against each of the others), we applied the Wilcoxon signed-rank test to compare the control against the other methods pairwise, and adjusted the α values using Holm's procedure for multiple tests (Wilcoxon 1992; Holm 1979), as recommended in (Demšar 2006).

Figures 4.2, 4.3, 4.4 and 4.5 presents the Critical Difference diagrams, for each metric. In each diagram, the methods connected to the Data-Driven control by a horizontal line were those for which the null hypothesis of the statistical test was not rejected, meaning their differences in performance were not statistically significant.

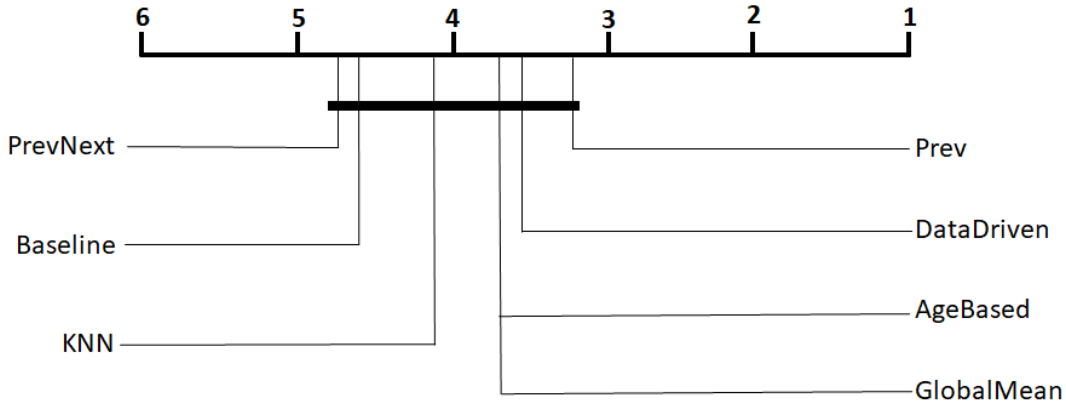

Figure 4.2: Critical Difference Diagram for Sensitivity metric. No method was significantly different from the Data-Driven approach.

Summary of classifier-dependent results

In this Section we compared the effects of employing different strategies to handle missing values in our longitudinal datasets. We analysed the performances of

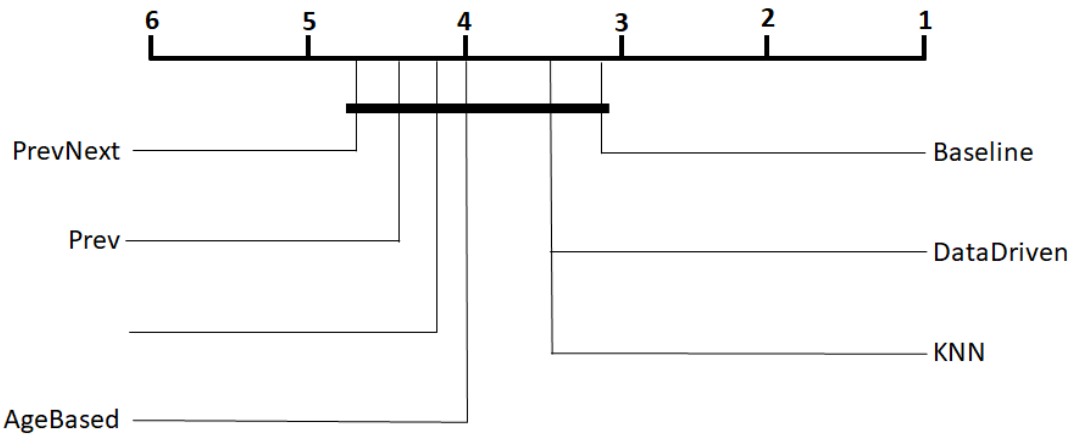

Figure 4.3: Critical Difference Diagram for Specificity metric. No method was significantly different from the Data-Driven approach.

RF models trained with these datasets using four metrics: Sensitivity, Specificity, Accuracy and GMean.

Overall, the proposed Data-Driven missing value replacement approach achieved the best results. To be more precise, it was the lowest-ranked method on average in all three sets of experiments (with three dataset sources) for the Sensitivity measure, and in two out of three for the other metrics: Specificity (ELSA-nurse and ELSA-core), Accuracy (TILDA and ELSA-nurse), and GMean (ELSA-nurse, ELSA-core). In our statistical analysis, the Data-Driven approach was deemed significantly superior to all methods but the PrevNext regarding its ranking, for all measures.

The proposed approach selects the best MVR method for every feature in the dataset, and guarantees that every missing value will be replaced. This is relevant for other experiments later in this thesis, where it is beneficial to have a fully imputed dataset, with no missing values. This is because we will create temporal features from the longitudinal values of the conceptual features in our datasets (e.g.: creating features reporting whether a conceptual feature's values have monotonically increased or decreased throughout the waves) in Chapter 5, and the creation of these temporal features is enabled by a dataset without missing values.

Thus, based on the results of our classifier-independent experiments in Section 4.6, and the classifier-dependent experiments in this current Section, we chose to

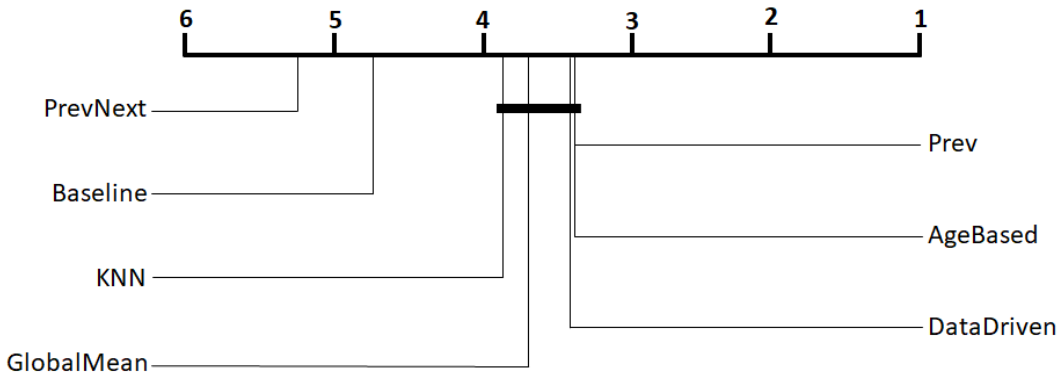

Figure 4.4: Critical Difference Diagram for Accuracy metric. The Data-Driven approach significantly outperformed the Baseline (p-value 0.01108) and PrevNext (p-value 0.00438) methods.

adopt the Data-Driven approach as our default method for handling missing values in all 30 of our datasets (ELSA-nurse, ELSA-core and TILDA datasets). For the experiments performed in later Chapters of this thesis, all datasets will have their missing values imputed on a preprocessing step, using the proposed Data-Driven approach (with the same set of base MVR methods from Section 4.3), and the training sets will be undersampled using the Balanced Random Forest method, as discussed in Section 4.7.1.

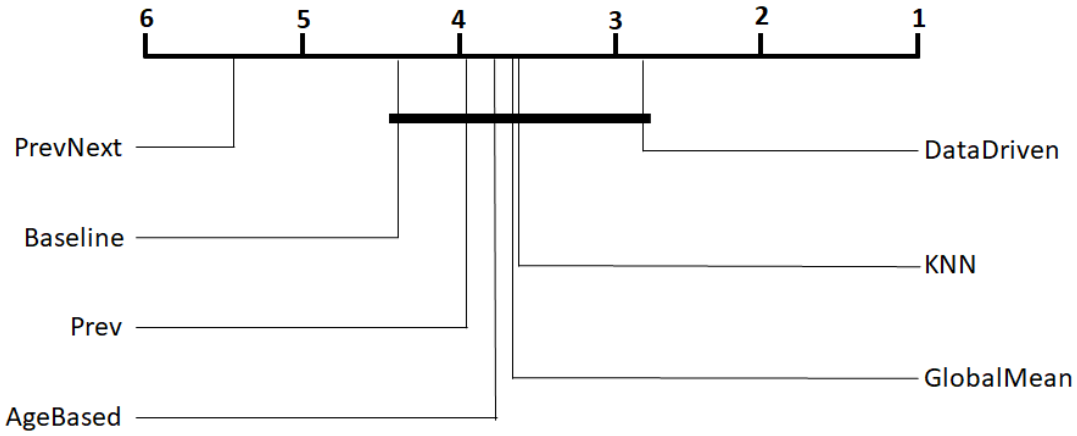

Figure 4.5: Critical Difference Diagram for GMean metric. The Data-Driven approach significantly outperformed the PrevNext method (p-value 0.0009).

Chapter 5

Constructed Temporal Features for Longitudinal Classification

Longitudinal datasets contain multiple measures of a set of features taken over different time-points, following the same group of instances. Because standard classification algorithms do not cope directly with the temporality of longitudinal data, they disregard time-related information that may be relevant to the problem. One way to address this issue in a data preparation step is to explicitly add a representation of the time-related information, inherent to longitudinal data, as additional features in the dataset.

In this Chapter we discuss the creation of additional features from longitudinal datasets, as a strategy to increase predictive accuracy of our classifiers. This is a data preparation approach, using both the UKLI (union keeping longitudinal information) and AGG (aggregated data) representations (see taxonomy on Chapter 3), the latter referring to the new constructed features. The performance of the proposed Constructed Temporal Features (CTFs) was evaluated on the 30 real-world datasets created from the English Longitudinal Study of Ageing (ELSA) and Irish Longitudinal Study of Ageing (TILDA) databases, after their preprocessing, carried out as discussed in Chapter 4.

Constructing temporal features in a preprocessing step is only one of the possible approaches for considering temporal patterns in classification problems. Other strategies include Structural Pattern Detection Morid et al. (2020), Recurrent Neural Networks (often Long-Short Term Memory) Aghili et al. (2018), and Deep Learning Luo et al. (2020). Although we are using only standard classifiers in our

experiments, our approach may be combined with more sophisticated algorithms tailored to longitudinal analysis.

By itself, CTF creation has the advantages of being simple to implement and adapt, and generating interpretable features that clearly represent temporal patterns. This is in contrast e.g. to deep neural networks, where the constructed features are not directly interpretable by users.

We performed experiments adding 6 types of CTF to the 30 longitudinal datasets, and compare them using four predictive performance metrics and two classification algorithms (Random Forests and the C4.5 Decision Tree algorithm). The experiments compared the results of those algorithms on different versions of the datasets, with and without the CTFs, in order to determine the impact of those CTFs on predictive performance. The contributions in this Chapter were published in a conference paper (Ribeiro and Freitas 2021a).

5.1 Adding CTFs in a Data Preprocessing Step

For each set of features representing measurements of the same variable at different time-points (e.g. for the features with values of "cholesterol" in different waves), we will create 6 types of CTFs to represent changes in their values across waves. These constructed features can be relevant information for class prediction, for example, an increase in cholesterol level across waves can indicate a higher risk of a heart attack for an individual.

The CTFs are created in a data preprocessing step, before running the classification algorithm, meaning they can be used with any classifier. Note that, in order to precisely calculate the CTF values, ideally the values of all temporally related (source) features should be known. For our experiments, all missing values in the datasets have been replaced by estimations using the Data-Driven missing value imputation approach, discussed in Chapter 4.

Note that the amount of temporal information available in our different datasets varies. For the Elsa-nurse dataset, as mentioned before, we have a total of 4 waves (each separated by 4 years from the next), where most original features are measured repeatedly. For the Elsa-core dataset there are 8 waves (separated by 2 years), and we use up to 7 measurements of each conceptual feature (as we did not include predictive features measured in the 8th wave). For the TILDA dataset,

although it also has 4 waves, about half of the conceptual features have only two measurements, as they are taken from the equivalent of the nurse-data interviews in ELSA, which happen every two waves. Because of this, we are able to analyse the impact of CTFs in longitudinal datasets with a small (TILDA), average (ELSA-nurse) or high (ELSA-core) amount of temporal information available.

5.1.1 Related work on CTFs

Niemann et al. (2015) generated evolution features by comparing instance clustering results at different time-points in a longitudinal dataset. The CTFs used in this study included the differences between feature measurements at each wave and cluster metrics taken from the clustering results. They performed experiments with different classifiers, and concluded that the added temporal features improved predictive accuracy for most classifiers.

In addition, Buizza et al. (2018) created longitudinal pattern features by comparing distances and means related to two subsequent images (PET/CT scans). The authors claimed that single time-point features were limited in describing local tumour characteristics that may affect treatment outcome. In their experiments, classification models created from datasets with the longitudinal pattern features generally outperformed those without them.

Both works are quite different from our study since they focus on specific types of CTFs involving clustering results and images, which are out the scope for this thesis.

In a more similar context to ours, Pomsuwan and Freitas (2017) have also used CTFs for longitudinal data, using datasets created from the ELSA-nurse database. The author added three types of CTFs to their original dataset: Up (whether the feature value increased between two consecutive waves), Monotonicity and Diff. The latter two features are also used in our study, and are described in detail in Section 5.2. However, their study focused on proposing a new longitudinal feature selection algorithm, rather than on the CTFs. Hence, they did not perform controlled experiments comparing the predictive accuracy of classification algorithms with and without CTFs, as performed in this thesis. Rather, they simply reported the relative frequencies with which the CTFs were selected by the random forest algorithm. The conclusion from their experiments was that CTFs can have a

positive impact on the predictive accuracy of longitudinal classifiers, but not all CTFs were selected often.

The work presented in this Chapter differs from these related studies as it is the first that focuses specifically on the creation of CTFs for longitudinal data (including the proposal of new types of CTFs) and the evaluation of their impact on the predictive accuracy of the classifiers by performing controlled experiments with and without the CTFs. We also propose a larger number of types of CTFs (6 types, as opposed to 3 types in Pomsuwan and Freitas (2017)), experiment with a larger number of datasets (30 datasets from real-world longitudinal data, as opposed to 10 datasets in Pomsuwan and Freitas (2017)).

5.2 The Proposed Constructed Temporal Features

In this section, we define six types of Constructed Temporal Features (CTFs), that represent temporal patterns that might be otherwise disregarded by a classification algorithm applied to longitudinal data. We describe each CTF's calculation for the numeric and ordered nominal features in our datasets.

However, before defining our proposed CTFs, it is important to recall our definition of conceptual features in longitudinal datasets. Features in a longitudinal dataset are often repeated measures of the same variable across waves (time-points). As mentioned earlier, the evolution of these features' values throughout the study's waves represents temporal information we intend to exploit, because the changes in values can represent relevant information.

We use the term "conceptual feature" to refer to the abstract definition of a feature, without specifying the wave where the feature was measured. For instance, *cholesterol* is a conceptual feature; whilst, in concrete terms, the dataset will contain different *cholesterol* features which are distinguished by the time-points (waves) where they were measured. The proposed CTFs are calculated using values of different measurements of the same conceptual feature. Thus, all features that have multiple measurements are eligible for the creation of CTFs.

Regarding the novelty of our proposed CTFs, the Monotonicity and Diff have been applied to similar longitudinal datasets in Pomsuwan and Freitas (2017),

as mentioned earlier. We could not find an example of the Ratio feature being applied to longitudinal datasets, but it is a small variation from the Diff feature. The Percentile and the two CTFs based on the age mean/mode, DiffAgeMean and AvgDiffAgeMean, are novel contributions.

Throughout the next sections, we will use Table 5.1 to give examples of CTF calculations, using real data from Elsa-nurse instances, of ldl (Blood LDL cholesterol level in mmol/l) measurements over the 4 waves in the dataset. The most common age value in this sample Table is 68, so we boldfaced the measurements for this age value, which will be useful in several of the examples used.

Table 5.1: A small sample from Elsa-Nurse data for examples of CTF calculation.

Instance	ldl_w2	Age_w2	ldl_w4	Age_w4	ldl_w6	Age_w6	ldl_w8	Age_w8
1	0.125	65	**0.548**	**68**	0.421	72	0.618	76
2	0.271	57	0.479	60	0.605	64	**0.632**	**68**
3	0.271	61	0.589	64	**0.579**	**68**	0.632	71
4	**0.146**	**68**	0.315	72	0.276	76	0.324	79
5	0.208	55	0.274	58	0.342	61	0.338	64
6	0.25	57	0.384	61	0.289	64	**0.353**	**68**
7	0.229	57	0.411	61	0.395	64	**0.632**	**68**
8	0.25	53	0.589	57	0.263	60	0.309	64
9	0.521	59	0.151	62	0.237	66	**0.279**	**68**
10	0.458	60	0.534	64	**0.303**	**68**	0.338	71

5.2.1 Monotonicity

A monotonic increase or decrease of a feature's values over its consecutive measurements in a longitudinal dataset may be a temporal pattern useful for predicting the value of the class variable. To represent these patterns, we created a Monotonicity CTF with three possible values: $+1$ indicating a monotonic increase in the feature's values across all its measurements, -1 indicating a monotonic decrease and 0 indicating no monotonicity pattern.

The calculation of a Monotonicity feature value is shown in Equation 5.1, where $F_{i,t}$ denotes the value of the i-th feature at time-point (wave) t. Note that it first checks if all feature values are equal; if so, it assigns a 0 value. Hence, the rules for the -1 and 1 values apply only if there has been at least one change to

the feature's values over time.

$$Monotonicity(F_i) = 0 : F_{i,0} = F_{i,1} = \cdots = F_{i,T} \\ OR \begin{cases} 1 : F_{i,0} \leq F_{i,1} \leq \cdots \leq F_{i,T} \\ -1 : F_{i,0} \geq F_{i,1} \geq \cdots \geq F_{i,T} \\ 0 : \text{ otherwise} \end{cases} \tag{5.1}$$

Note that we do not use a strict monotonicity definition with $<$ and $>$ operators. Rather, we use a more flexible monotonicity definition with the $\leq$ and $\geq$ operators. Hence, if the feature's value increased or decreased at least once, as long as the feature's values do not change in the opposite direction in other waves, we consider this a monotonic change.

The motivation for this more flexible definition is that it can be applied to both numeric and ordered nominal features. Our datasets contain several ordered nominal features taking between 2 and 8 possible values, and the above strict definition of monotonicity would not be flexible enough to cope with such ordered nominal features. For example, if a dataset has 4 waves but the feature can take only two ordered values, say "low" and "high", it is impossible to detect a monotonic change according to the strict definition, but a sequence of feature values such as "low", "low", "high", "high" would be recognised as a monotonic increase, a potentially useful pattern for classification.

In Table 5.1, Instance 2 would get the value 1 for its Monotonicity for the ldl conceptual feature, indicating its values for ldl cholesterol have steadily increased over the 4 consecutive measurements, from wave 2 to wave 8. All other instances would get the value 0, indicating no Monotonicity pattern.

5.2.2 Diff - difference between last two measurements

Feature measurements taken closer in time to the class wave (the last wave) arguably have more impact on the model's output, as they are likely more closely related to the class variable than measurements of the same feature taken further in the past (earlier waves). In this context, we consider that the most recent changes to a feature's value may represent an important temporal trend. Thus, we created the Diff CTF to measure the numerical difference between the conceptual

features last and second to last measurements.

The calculation of the Diff CTF is shown in Equation 5.2, where T is the index of the class variable's wave (the last wave). For ordered nominal features, Diff represents the degree of difference between the nominal values. This is only possible because all nominal features in our datasets are ordered, so we can assign numerical values to them and calculate the difference between these values as a degree of difference. Note that this degree of difference measurement is precise only for cases where the response options in the nominal features are equidistant, and we do not make that assumption, as we did not design the data. However, after inspecting all nominal features in our datasets, we decided that their values can be considered similarly distant enough that the Diff calculation would be acceptable. The same decision was made for Percentile, DiffAgeMean and AvgDiffAgeMean, where we also calculate degrees of difference for nominal features.

$$Diff(F_i) = F_{i,T} - F_{i,T-1} \tag{5.2}$$

In Table 5.1, the ldl Diff value for Instance 1 would be $0.618 - 0.421 = 0.197$, the positive value indicating an increase in the feature's value from wave 6 to wave 8. Conversely, Instance 5 would get a ldl Diff of $0.338 - 0.342 = -0.004$, with a negative value indicating that the feature's value has decreased from wave 6 to wave 8.

5.2.3 Ratio between last two measurements

The Ratio CTF functions similarly to the Diff CTF. However, instead of the difference, it calculates the result of dividing the value of the conceptual features last measurement by its second to last measurement. We chose not to calculate this CTF for nominal features, as the assumption of equidistance between the possible values is much more important for the Ratio CTF, which is more sensitive to changes in the values of a feature than the Diff CTF. As stated before, we do not assume equidistance in the values of all our nominal features, although they are similar enough that we considered plausible to calculate degrees of difference between them using subtraction (for the Diff CTF). Hence, in this work the Ratio CTF is used only for numeric features.

Note that the Ratio CTF can capture patterns quite different from patterns captured by the Diff CTF. For example, Diff has the same value (0.2) for the feature value pairs (0.2, 0.4) and (0.6, 0.8), whilst Ratio has value 2 for the former pair and 1.33 for the latter.

The calculation of the Ratio CTF is shown in Equation 5.3. To avoid a division by zero error, before performing the division we add 1 to both feature values. Note that this change can have different effects depending on the unit of measure. In our case, as all values in the dataset are normalised between 0 and 1, this changes the range of Ratio value beyond this range, and may make the interpretation of these values more challenging. Another possibility to address this issue would be to add a small epsilon value to the denominator; this alternative approach could be investigated in future work.

$$Ratio(F_i) = \frac{F_{i,T} + 1}{F_{i,T-1} + 1} \tag{5.3}$$

Using the same example as the one used for Diff, in Table 5.1 we have Instance 1 with a ldl Ratio value of $0.618/0.421 = 1.4679$, and Instance 5 with a value of $0.338/0.342 = 0.9883$. A value smaller than 1 denotes a decrease in the measurements from wave 6 to wave 8, and vice-versa. The main point of Ratio is that smaller differences will produce values closer to 1, so the further from 1 a result is, the bigger the difference found in the last two measurements of a feature.

5.2.4 DiffAgeMean - last measurement's difference from age-based mean/mode

The age of the subjects is intuitively a very relevant variable in our datasets, as they contain data about human ageing. In the experiments with missing value replacement in these datasets, we have found that the mean (or mode, for nominal features) value of subjects of the same age as the current instance is a good estimation for an expected value of a feature (Ribeiro and Freitas 2021b). Therefore, we propose a CTF to calculate the difference between a feature's value in the last wave (the class variable's wave) and its "expected" value, which is an age-based mean/mode.

The calculation of the DiffAgeMean CTF is shown in Equation 5.4. To calculate the expected value for a feature F_i's last measurement ($F_{i,T}$) for each subject (instance), we get the value of that subject's age at wave T ($Age_{i,T}$) and calculate the mean over all measurements of F_i, over all waves and subjects where a subject's age equals $Age_{i,T}$. For nominal features, the expected value becomes the mode among individuals of the same age, instead of the mean, and DiffAgeMean measures the degree of difference (instead of the numerical difference) from that mode.

$$DiffAgeMean(F_i) = F_{i,T} - Exp(F_i, Age_{i,T}) \tag{5.4}$$

For the DiffAgeMean example in Table 5.1, we will consider Instance 2, which has the most common age value in the Table, 68. First, the CTF calculates the average ldl value for all individuals aged 68, across all waves (0.434, averaged from individuals 1, 2, 3, 4, 6, 7, 9 and 10). Then, the ldl value for Instance 2 at wave 8 is compared to this average, getting a DiffAgeMean value of $0.632 - 0.434 = 0.198$, indicating this ELSA participant has a higher ldl cholesterol value than the average of other individuals of their age.

5.2.5 AvgDiffAgeMean - average difference from age-based mean/mode

As an expansion of the DiffAgeMean feature, we calculate the DiffAgeMean for all different measurements of a feature, then average these results (dividing the sum of DiffAgeMean's by the number of measurements), to get an average difference from the expected values. Note that at each wave of the study, the subject's age changes, so we need to recalculate the expected value for each measurement of the current feature.

The AvgDiffAgeMean CTF is calculated as shown in Equation 5.5. Again, for nominal features, we use the mode as the expected values, instead of the mean.

$$AvgDiffAgeMean(F_i) = \frac{\sum_{k=1}^{T} F_{i,k} - Exp(F_{i,k}, Age_{i,k})}{T} \tag{5.5}$$

Using the same example from DiffAgeMean in Table 5.1, the calculation for AvgDiffAgeMean for Instance 2 is as follows. We have shown that their DiffAgeMean for wave 8 is $0.632 - 0.434 = 0.198$. For Wave 6, we calculate the average measurements for individuals aged 64, and calculate the DiffAgeMean for Instance 2, getting $0.605 - 0.437 = 0.168$. The same procedure is followed for Wave 4, using individuals aged 60, getting $0.479 - 0.4 = 0.079$, and for Wave 2, using individuals aged 57, getting $0.271 - 0.335 = -0.064$. The ldl AvgDiffAgeMean value for Instance 2 is the mean of these 4 DiffAgeMean values: $(0.198 + 0.168 + 0.079 - 0.064)/4 = 0.095$. This value indicates that this ELSA participant tends to have a ldl cholesterol value slightly higher (about 9.5%, as these values are normalised between 0 and 1) than other participant of their age, considering all age values of the current ELSA participant across all waves.

5.2.6 Age-based Percentile

This proposed CTF is also based on the measurement taken from subjects with the same age as the current subject ($Age_{i,T}$). However, instead of choosing one expected value, we rank all values of the current conceptual feature from all subjects with $age = Age_{i,T}$, and compute in what Percentile the current subject's last measurement for the conceptual feature is. We consider the last measurement of the feature, as it is intuitively the most relevant.

This CTF was inspired by the percentile feature used in Al-Otaibi et al. (2015), but that work did not use any other variable to compute percentiles and did not use longitudinal datasets. By contrast, in this work we compute age-based percentiles and adapt DiffAgeMean's calculation to cope with a feature's multiple measurements across time-points in longitudinal datasets.

Thus, the Age-based Percentile CTF indicates what percentage of the other subjects with the same age as the current subject had measurements with lower values than the current subject's measurement. For example, the Percentile 30% for a feature means that, among the subjects of the same age as the current subject's age in wave T, only 30% of those subjects have a feature value lower than the current subject's feature value in wave T. The temporal aspect of the Percentile CTF is the calculation of the Ranks, which happens over all different measurements of the current feature.

The calculation of the Percentile CTF is shown in Equation 5.6. Note that the term $Age_{i,T}$ in this equation is indexed by T because we compute the rank of a subject's feature value at wave T, considering all subjects with the same age as the current subject's age at wave T. However, when computing the rank, we consider any measurement from subjects of that age, regardless of the wave. In Equation 5.6, $NValues(F_{i,T}, Age_{i,T})$ is the number of values used to compute the ranking. This CTF is calculated for numeric and ordered nominal features in the same way.

$$Percentile(F_i) = \frac{Rank(F_{i,T}, Age_{i,T})}{NValues(F_{i,T}, Age_{i,T})} \tag{5.6}$$

For the Percentile example, let's consider Instances 2, 6 and 9 in Table 5.1. First, we rank the 8 values of ldl for individuals aged 68 in this sample table, from lowest to highest. The Percentile value is simply the rank of an individual's value divided by the number of samples, in this case 8. As Instance 2's ldl value is ranked 7.5 (tied for last place alongside Instance 7), both Instances 2 and 7 get a Percentile value of $7.5/8 = 0.9375$. For Instance 6, its value is the 4th lowest, so it gets a ldl Percentile value of $4/8 = 0.5$. For Instance 9, that has the second lowest value, it's Percentile is $2/8 = 0.25$. Thus, we can interpret that Instances 2 and 7 have high ldl cholesterol values, when compared to other participants of their age, while Instance 6 has a medium value, and Instance 9 has lower cholesterol than most people of their age.

5.3 Experimental Setup

For our experiments with the proposed CTFs, we created classification models using the Random Forest (RF) (Breiman 2001) and the C4.5 Decision Tree (Quinlan 1993) algorithms. The results from the latter were moved to Appendix C to reduce the size of this Chapter.

This work is the first to test CTFs for longitudinal data using the RF algorithm. As discussed in Chapter 2, The RF algorithm is among the state-of-the-art classification algorithms (Fernández-Delgado et al. 2014), while still maintaining some interpretability of its models, mainly through feature importance measures.

In addition, RFs handle well datasets with a high ratio of features to instances, which are prone to overfitting (Scornet et al. 2015). This is desirable as our proposal can add up to 6 CTFs for each conceptual feature in the longitudinal datasets.

Recall that, because of the class imbalance problem in our datasets, we decided to apply a majority class undersampling strategy to our training datasets. In the RF experiments, all training sets were balanced using the Balanced Random Forest (BRF) method, defined in section 2.2.5. For the C4.5 experiments, we performed random undersampling of majority class instances in the training dataset, to obtain a ratio of 1:1 for the frequencies of the two classes.

The RFs were trained and tested using the Weka data mining toolkit, with the default parameters $ntrees = 100$ (number of 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, and $\lfloor x \rfloor$ is the "floor" of x, i.e., the biggest integer which is smaller than or equal to x. The C4.5 decision trees were also trained with default parameters $C = 0.25$, which is the confidence factor used for pruning the trees, and $M = 2$, which is the minimum number of instances that can constitute a leaf node.

As defined in Section 4.7.1 in the previous Chapter, all experiments comparing classifiers in the thesis were evaluated using four metrics: Sensitivity (True Positive Rate), Specificity (True Negative Rate), GMean (geometric mean between Sensitivity and Specificity) and Accuracy (percentage of correct classifications). The experiments used the well-known 10-fold cross-validation procedure, and we compared the results of three feature sets (defined in the following Sections 5.3.1 and 5.3.2) for each metric using two statistical significance tests, as follows.

As we did in Section 4.7.2 in the previous Chapter, we first applied the Friedman's test, which compares all features sets at once. If this test indicated the results are significantly different, we then applied the Nemenyi post-hoc test in a pairwise, using Holm's procedure to correct the significance level for multiple tests, to determine which combinations of methods were significantly different. The tests were applied with the usual initial significance level $\alpha = 0.05$.

In the following sections, we describe the two scenarios created for our analysis of the effect of adding CTFs to longitudinal datasets in the predictive accuracy of classifiers. The same experimental setup described here was applied in both

scenarios, for each of our 30 longitudinal datasets.

5.3.1 Scenario 1 - controlled experiments with only eligible original features

For these experiments, our objective was to evaluate the potential increase in predictive accuracy for classifiers learned from baseline datasets containing the proposed CTFs as added features. First, we identify all conceptual features that can be used in the creation of the CTFs, called the set of "eligible" features. This set consists of all conceptual features that had at least two measurements (across waves).

We experimented with each CTF separately first, then with adding all 6 CTFs simultaneously. In order to have a fair evaluation of the proposed CTFs in a controlled experiment, we created datasets from the Elsa-nurse, Elsa-core and TILDA databases with the following three different feature sets:

- **Base-el**: The eligible original features for creating the CTF being evaluated; no CTF.
- **CTFs-only**: The proposed CTF being evaluated (AvgDiffAgeMean, DiffAgeMean, Diff, Monotonicity, Age-based Percentile, Ratio, or all 6 at once), created for each eligible conceptual feature; no original features.
- **Base-el+CTFs**: Both the above feature sets combined, i.e. both eligible original and CTF features.

These three feature sets represent different strategies for handling the construction of temporal features for longitudinal dataset. The baseline is simply not creating them, and trusting that the temporal information inherent to the longitudinal data is not relevant for the classification task. The second strategy is to completely replace the original features by the CTFs, which is not recommended, as they represent different information, but we are using it to investigate how much the information represented by the CTFs can contribute towards classification on its own. The third strategy is what we propose, adding the CTFs in a preprocessing step to the existing features in the dataset. Our hypothesis is

that the latter is the best strategy out of the three in general, for longitudinal classification problems.

Note that, depending on the CTF being evaluated, the baseline changed based on how many measurements of the conceptual features were used in the CTF creation: for the Percentile and DiffAgeMean features, only the last measurement of each conceptual feature is used; for the Diff and Ratio features, the last two measurements are used; for the Monotonicity and AvgDiffAgeMean features, we use all measurements available (up to 4 in Elsa-core and TILDA datasets and up to 7 in Elsa-nurse datasets). This means that, for example, the Base-el feature set for the Percentile analysis corresponds only to the last measurement of each eligible conceptual feature in the original dataset.

All result tables shown in this Section and in Section 5.3.2 have the same structure: each column corresponds to the feature set that composes the dataset (Base-el, CTFs only, and both the Base-el and the CTFs combined), and each row shows the results for one dataset – defined by a combination of a class variable and a data source (EN for Elsa-nurse, EC for Elsa-core and TI for TILDA datasets). In each row, the best result is shown in boldface. In the last four rows, we show the average rank for each feature set, over the Elsa-nurse, Elsa-core and TILDA datasets, and over all 30 datasets, respectively. When ties happened, the rank was divided among the tied feature sets (i.e., if two sets are tied for first place, they would both get a 1.5 rank). Each result table reports two measurements, either Sensitivity and Specificity or Accuracy and GMean.

5.3.2 Scenario 2 - experiments including both eligible and ineligible original features

In a second set of experiments, we evaluated the use of CTFs in a less controlled scenario than Scenario 1. Arguably, in real-world applications it would make sense to use all available features, and so the features ineligible for CTF calculation (i.e., those with only one measurement in the longitudinal dataset) would be kept in the dataset regardless of whether or not CTF features are added to the dataset. So, we decided to make a separate set of experiments (Scenario 2) where this is the case. Thus, instead of comparing feature sets using only the "eligible" original features (from which CTFs are constructed), the new experiments in this current

section compare the following three feature sets:

- **Base-el-in:** All original features, i.e. both the eligible original features for creating the CTF being evaluated and the original features ineligible for CTF creation (features measured just once); no CTF.
- **CTFs+in:** The proposed CTF being evaluated (AvgDiffAgeMean, DiffAgeMean, Diff, Monotonicity, Age-based Percentile, Ratio, or all 6 at once), created for each eligible conceptual feature, plus the features ineligible for CTF creation (features measured just once).
- **Base-el-in+CTFs:** Both the above feature sets combined.

The reason for having the two separate Scenarios for our experiments with CTFs is as follows. The features that are ineligible for CTF creation include important features such as the age and sex of the respondent, demographic features that have high predictive power and were considered among the most relevant (based on feature importance metrics) in previous models we created. Thus, Scenario 2 is more realistic than Scenario 1 because the former uses the ineligible features in the evaluated feature sets (exploiting as much information from the data as possible), unlike Scenario 1. However, Scenario 2 involves a less controlled experiment to measure the effectiveness of CTFs, since in this Scenario the predictive accuracy of CTFs is measured together with ineligible features, unlike Scenario 1, where one of the feature subsets consists of CTFs only.

5.4 Random Forest Experimental Results

In this Section, we first briefly discuss the overall results of experiments with each individual CTF (the detailed results and discussion for these sets experiments using RFs and DTs are presented in Appendices B and C, respectively). Then, we report the RFs and DT results for experiments using Scenarios 1 and 2 (defined in Sections 5.3.1 and 5.3.2), for all 6 CTFs combined. After discussing the overall results for the Sensitivity, Specificity, Accuracy and GMean metrics, we also report the results of the Friedman's rank-based test and (if applicable) the Nemenyi post-hoc test for each experiment.

5.4.1 Summary of the RF Results for Individual CTF Experiments

The Diff, Ratio and Monotonicity CTFs, the simplest among the CTFs tested in our experiments, were not proposed in this thesis. All three of these CTFs had similar results to the baseline approach, when comparing it to the BL+CTFs (Base-el+CTFs and Base-el-in+CTFs, in each Scenario, i.e., our proposal of adding the CTFs to the original features) feature set, for both classifiers.

The three CTFs proposed in this work had better RF results in the individual experiments compared to the other three. DiffAgeMean and Percentile obtained better results for the BL+CTFs feature sets overall for both scenarios. For AvgDiffAgeMean there was not a clear winner in either scenario, but this CTF notably got the best results overall for the CTFs-in feature set, which included only constructed and ineligible features: smallest average rank for Specificity and GMean, in ELSA-core datasets.

In summary, individually most of the proposed CTFs did not lead to a clear increase in predictive accuracy, with the exception of DiffAgeMean and Percentile. However, as we believe the information represented by each CTF to be possibly relevant for classification, and not a detriment to the classification algorithm, we still included all six of them in our main approach, of adding all CTFs to the longitudinal dataset.

5.4.2 Results for all 6 types of CTFs combined

The set of experiments presented in this Section is the most important one for this Chapter, as our goal was always to propose the addition of all 6 CTFs to a longitudinal dataset, as opposed to choosing one of them. As each CTF represents a different type of trend calculated from longitudinal data, we believe that they can be used in tandem to improve the predictive accuracy of classifiers, when added to longitudinal datasets. Thus, we included both the Random Forest and the Decision Tree results in this section.

RF results for all 6 types of CTFs

The Random Forest results for the experiments including all 6 CTFs in the datasets, for Scenario 1, are shown in Tables 5.2 (for Sensitivity and Specificity) and 5.3 (for Accuracy and GMean). The results for Scenario 2 are shown in Tables 5.4 (for Sensitivity and Specificity) and 5.5 (for Accuracy and GMean).

For Scenario 1, the BL+CTFs feature set had the smallest average ranks overall for all 4 metrics. The Friedman test p-values for Scenario 1 were 0.0121, 0.0780, 0.0006 and 0.0322 for Sensitivity, Specificity, Accuracy and GMean, respectively; so we ran the post-hoc Nemenyi test for all metrics except Specificity. For Sensitivity, the only pair with a significant p-value was BL+CTFs vs. CTFs-only (0.0102). For Accuracy, both Base-el and BL+CTFs had significant p-values when compared to CTFs-only (0.0184 and 0.001, respectively). For GMean, again, only BL+CTFs vs. CTFs-only had a significant p-value of 0.0266.

The Base-el-in feature set results were slightly improved in Scenario 2, with the inclusion of the highly predictive ineligible features, with the Base-el-in set achieving the smallest average rank overall for Sensitivity and Accuracy, and BL+CTFs winning for Specificity and GMean. The Friedman test p-values for Scenario 2 had p-values 0.0215, 0.2328, 0.0481 and 0.0247 for Sensitivity, Specificity, Accuracy and GMean, respectively; so again we did not run Nemenyi post-hoc tests for Specificity. In the post-hoc tests, for Sensitivity, only the pair Base-el-in vs. CTFs-in got a significant result with p-value 0.0221. For Accuracy none of the pairs had significant p-values. For GMean, only the pair BL+CTFs vs. CTFs-in had a significant p-value, 0.0184.

Notably, the Base-el and Base-el-in were the best for TILDA datasets in all 4 metrics, but its average rank was always second or third for Elsa-nurse and Elsa-core datasets. This may be due to the reduced temporal information represented in the TILDA dataset which, as discussed before, has on average fewer measurements in its longitudinal features. It is also important to highlight the good results in Scenario 1 from CTFs-only and BL+CTFs in the Elsa-core datasets, with the former achieving the smallest average ranks for Specificity (1.6, against 1.7 of BL+CTFs), and the latter winning for the other 3 metrics. This is important because the Elsa-core datasets have the most temporal information in their features, due to their highest number of waves.

The Random Forest results for adding all 6 CTFs to the longitudinal datasets

showed that the predictive performance of classifiers was improved in the majority of the cases, although that majority was not large enough to be deemed significant when comparing the BL+CTFs and the baseline feature sets. Having more or less temporal information in the datasets seemed to influence the effectiveness of the CTF, which is a good indication that they do represent the temporal trends we hoped, and can be a criterion for deciding on their inclusion in other longitudinal datasets. In addition, although the results of these experiments were not the best overall results obtained by the BL+CTFs feature sets, we believe that in principle adding all CTFs combined could be a preferable (more robust) strategy to only adding, for example, the DiffAgeMean CTF, which had the best individual results.

Table 5.2: All 6 CTFs Sensitivity and Specificity results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	0.680	0.672	**0.683**	0.678	0.709	**0.713**
EN_Arthritis	0.669	0.655	**0.670**	0.594	**0.604**	0.593
EN_Cataract	**0.615**	0.576	0.601	0.723	**0.754**	0.729
EN_Dementia	0.737	0.729	**0.752**	**0.716**	0.709	0.703
EN_Diabetes	0.843	0.841	**0.846**	0.865	**0.870**	0.863
EN_HBP	**0.653**	0.647	0.650	0.747	0.730	**0.751**
EN_Heartattack	**0.698**	**0.698**	0.698	**0.731**	0.696	0.718
EN_Osteoporosis	**0.655**	0.629	0.643	0.699	0.716	**0.717**
EN_Parkinsons	0.604	0.630	**0.634**	0.636	0.591	**0.652**
EN_Stroke	0.667	**0.679**	0.678	**0.710**	0.694	0.701
EC_Angina	**0.710**	0.691	0.706	0.723	0.765	**0.772**
EC_Arthritis	0.741	**0.752**	0.750	0.717	0.721	**0.726**
EC_Cataract	0.601	**0.626**	0.617	0.675	0.750	**0.751**
EC_Dementia	0.757	0.771	**0.773**	0.727	**0.832**	0.776
EC_Diabetes	0.671	**0.691**	0.674	0.750	0.748	**0.759**
EC_HBP	0.625	**0.640**	0.634	0.662	**0.671**	0.669
EC_Heartattack	**0.673**	0.654	0.669	0.689	**0.721**	0.705
EC_Osteoporosis	0.690	0.680	**0.691**	0.635	**0.661**	0.638
EC_Parkinsons	0.685	0.714	**0.715**	0.720	0.720	**0.747**
EC_Stroke	0.689	0.685	**0.692**	0.694	**0.758**	0.747
TI_Angina	**0.748**	0.695	0.743	**0.876**	0.772	0.848
TI_Arthritis	**0.731**	0.686	0.710	0.652	0.614	**0.655**
TI_Cancer	0.542	**0.548**	0.526	**0.595**	0.497	0.526
TI_Cataract	0.660	0.695	**0.698**	0.705	0.676	**0.734**
TI_Diabetes	0.737	0.712	**0.744**	**0.795**	0.732	**0.795**
TI_HBP	**0.678**	0.630	0.662	**0.763**	0.657	0.750
TI_Heartattack	**0.750**	0.707	0.745	**0.863**	0.751	0.829
TI_Ministroke	0.704	0.660	**0.704**	**0.765**	0.696	0.745
TI_Osteoporosis	**0.670**	0.638	0.662	**0.772**	0.681	0.762
TI_Stroke	**0.717**	0.657	0.694	0.738	0.631	**0.754**
AvgRank E-Nurse	1.9	2.6	**1.5**	2.0	2.1	**1.9**
AvgRank E-Core	2.4	2.0	**1.6**	2.9	1.7	**1.5**
AvgRank TILDA	**1.5**	2.7	1.9	**1.4**	3.0	1.7
AvgRank Overall	1.9	2.4	**1.7**	2.1	2.3	**1.7**

Table 5.3: All 6 CTFs Accuracy and GMean results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	0.680	0.673	**0.684**	0.679	0.690	**0.698**
EN_Arthritis	**0.637**	0.633	**0.637**	**0.630**	0.629	0.630
EN_Cataract	**0.650**	0.634	0.643	**0.667**	0.659	0.662
EN_Dementia	0.736	0.729	**0.751**	0.726	0.719	**0.727**
EN_Diabetes	0.846	0.844	**0.848**	0.854	**0.855**	0.854
EN_HBP	**0.691**	0.681	0.691	0.699	0.688	**0.699**
EN_Heartattack	**0.700**	0.698	0.699	**0.714**	0.697	0.708
EN_Osteoporosis	**0.659**	0.637	0.649	0.676	0.671	**0.679**
EN_Parkinsons	0.605	0.629	**0.634**	0.620	0.610	**0.643**
EN_Stroke	0.670	**0.680**	**0.680**	0.688	0.686	**0.689**
EC_Angina	**0.711**	0.694	0.708	0.716	0.727	**0.738**
EC_Arthritis	0.731	0.740	**0.741**	0.729	0.737	**0.738**
EC_Cataract	0.623	**0.663**	0.657	0.637	**0.685**	0.681
EC_Dementia	0.757	0.772	**0.773**	0.742	**0.801**	0.774
EC_Diabetes	0.681	**0.699**	0.685	0.709	**0.719**	0.715
EC_HBP	0.639	**0.652**	0.647	0.643	**0.656**	0.651
EC_Heartattack	**0.674**	0.658	0.671	0.681	**0.687**	0.687
EC_Osteoporosis	0.686	0.678	**0.687**	0.662	**0.670**	0.664
EC_Parkinsons	0.685	0.714	**0.715**	0.702	0.717	**0.731**
EC_Stroke	0.689	0.689	**0.695**	0.692	**0.721**	0.719
TI_Angina	**0.753**	0.699	0.748	**0.809**	0.733	0.794
TI_Arthritis	**0.706**	0.664	0.693	**0.690**	0.649	0.682
TI_Cancer	**0.545**	0.545	0.526	**0.568**	0.522	0.526
TI_Cataract	0.664	0.694	**0.701**	0.682	0.685	**0.716**
TI_Diabetes	0.741	0.713	**0.747**	0.765	0.722	**0.769**
TI_HBP	**0.710**	0.640	0.695	**0.719**	0.644	0.704
TI_Heartattack	**0.755**	0.708	0.748	**0.805**	0.729	0.786
TI_Ministroke	0.705	0.661	**0.705**	**0.734**	0.678	0.724
TI_Osteoporosis	**0.680**	0.642	0.671	**0.719**	0.659	0.710
TI_Stroke	**0.717**	0.657	0.694	**0.728**	0.644	0.723
AvgRank E-Nurse	1.8	2.8	**1.5**	1.9	2.7	**1.5**
AvgRank E-Core	2.5	2.1	**1.5**	3.0	**1.4**	1.7
AvgRank TILDA	**1.4**	2.8	1.9	**1.3**	2.9	1.8
AvgRank Overall	1.9	2.5	**1.6**	2.1	2.3	**1.6**

Table 5.4: All 6 CTFs Sensitivity and Specificity results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	**0.684**	0.673	0.681	**0.702**	**0.702**	0.686
EN_Arthritis	**0.671**	0.658	**0.671**	0.586	**0.609**	0.594
EN_Cataract	**0.620**	0.593	0.605	0.723	**0.751**	0.736
EN_Dementia	0.729	**0.748**	0.743	0.709	0.696	**0.736**
EN_Diabetes	0.841	0.836	**0.845**	0.866	0.863	**0.868**
EN_HBP	**0.651**	0.644	0.650	**0.749**	0.724	0.745
EN_Heartattack	**0.700**	0.694	**0.700**	**0.738**	0.711	0.713
EN_Osteoporosis	**0.649**	0.633	0.643	0.696	**0.723**	0.716
EN_Parkinsons	0.628	**0.650**	0.627	0.712	0.652	**0.727**
EN_Stroke	0.670	**0.681**	0.674	**0.724**	0.698	0.720
EC_Angina	**0.711**	0.699	0.706	0.723	**0.765**	0.761
EC_Arthritis	0.749	0.747	**0.756**	0.717	**0.722**	0.720
EC_Cataract	0.609	**0.623**	0.613	0.717	0.760	**0.764**
EC_Dementia	0.764	0.765	**0.766**	0.770	**0.845**	0.807
EC_Diabetes	0.674	**0.686**	0.682	0.747	0.743	**0.758**
EC_HBP	0.641	**0.642**	0.637	0.662	**0.677**	0.673
EC_Heartattack	0.678	0.669	**0.680**	0.692	0.714	**0.717**
EC_Osteoporosis	0.700	0.687	**0.702**	0.676	**0.696**	0.666
EC_Parkinsons	0.697	**0.714**	0.694	0.693	0.693	**0.733**
EC_Stroke	0.694	0.683	**0.696**	0.721	**0.751**	0.747
TI_Angina	**0.748**	0.697	0.739	**0.916**	0.756	0.896
TI_Arthritis	**0.729**	0.685	0.721	**0.646**	0.623	0.642
TI_Cancer	0.549	0.554	**0.555**	0.579	0.526	**0.599**
TI_Cataract	**0.706**	**0.706**	0.699	0.724	0.686	**0.741**
TI_Diabetes	**0.775**	0.752	0.755	**0.831**	0.756	0.818
TI_HBP	**0.678**	0.625	0.669	**0.765**	0.679	0.754
TI_Heartattack	**0.751**	0.719	0.743	**0.878**	0.761	0.849
TI_Ministroke	**0.712**	0.668	0.706	**0.735**	0.647	0.716
TI_Osteoporosis	**0.677**	0.665	0.670	**0.807**	0.761	0.759
TI_Stroke	**0.728**	0.663	0.708	**0.800**	0.600	0.754
AvgRank E-Nurse	**1.7**	2.4	1.9	2.0	2.3	**1.8**
AvgRank E-Core	2.2	2.1	**1.7**	2.8	**1.6**	1.7
AvgRank TILDA	**1.3**	2.8	2.0	**1.2**	2.9	1.9
AvgRank Overall	**1.7**	2.4	1.9	2.0	2.2	**1.8**

Table 5.5: All 6 CTFs Accuracy and GMean results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	**0.684**	0.675	0.681	**0.693**	0.687	0.684
EN_Arthritis	0.635	0.637	**0.639**	0.627	**0.633**	0.632
EN_Cataract	**0.654**	0.645	0.648	**0.670**	0.667	0.667
EN_Dementia	0.729	**0.747**	0.742	0.719	0.722	**0.740**
EN_Diabetes	0.845	0.840	**0.848**	0.854	0.849	**0.856**
EN_HBP	**0.690**	0.676	0.688	**0.698**	0.683	0.696
EN_Heartattack	**0.702**	0.695	0.701	**0.719**	0.702	0.707
EN_Osteoporosis	**0.654**	0.641	0.650	0.672	0.677	**0.679**
EN_Parkinsons	0.629	**0.650**	0.628	0.669	0.651	**0.675**
EN_Stroke	0.674	**0.682**	0.677	**0.697**	0.690	**0.697**
EC_Angina	**0.711**	0.701	0.708	0.717	0.731	**0.733**
EC_Arthritis	0.736	0.737	**0.741**	0.733	0.734	**0.738**
EC_Cataract	0.641	**0.663**	0.658	0.661	**0.688**	0.685
EC_Dementia	0.764	**0.767**	**0.767**	0.767	**0.804**	0.787
EC_Diabetes	0.684	**0.693**	0.692	0.710	0.714	**0.719**
EC_HBP	0.649	**0.655**	0.651	0.651	**0.659**	0.655
EC_Heartattack	0.679	0.671	**0.682**	0.685	0.691	**0.698**
EC_Osteoporosis	0.698	0.688	**0.699**	0.688	**0.692**	0.684
EC_Parkinsons	0.697	**0.713**	0.694	0.695	0.703	**0.713**
EC_Stroke	0.695	0.686	**0.699**	0.707	0.716	**0.721**
TI_Angina	**0.756**	0.700	0.746	**0.828**	0.726	0.814
TI_Arthritis	**0.703**	0.666	0.697	**0.686**	0.654	0.681
TI_Cancer	0.550	0.552	**0.557**	0.564	0.540	**0.576**
TI_Cataract	**0.707**	0.704	0.702	0.715	0.696	**0.719**
TI_Diabetes	**0.779**	0.752	0.760	**0.803**	0.754	0.786
TI_HBP	**0.711**	0.645	0.701	**0.720**	0.651	0.710
TI_Heartattack	**0.756**	0.720	0.747	**0.812**	0.740	0.794
TI_Ministroke	**0.712**	0.668	0.706	**0.724**	0.658	0.711
TI_Osteoporosis	**0.689**	0.674	0.679	**0.739**	0.711	0.713
TI_Stroke	**0.728**	0.662	0.709	**0.763**	0.631	0.731
AvgRank E-Nurse	**1.8**	2.3	1.9	1.9	2.5	**1.7**
AvgRank E-Core	2.4	2.0	**1.7**	2.9	1.6	**1.5**
AvgRank TILDA	**1.2**	2.8	2.0	**1.2**	3.0	1.8
AvgRank Overall	**1.8**	2.4	1.9	2.0	2.4	**1.7**

Decision tree results for all 6 types of CTFs

The C4.5 Decision Tree results for the experiments including all 6 CTFs in the datasets, for Scenario 1, are shown in Tables 5.6 (for Sensitivity and Specificity) and 5.7 (for Accuracy and GMean). The results for Scenario 2 are shown in Tables 5.8 (for Sensitivity and Specificity) and 5.9 (for Accuracy and GMean).

In the Decision Tree experiments, the baseline approach usually got a clear advantage when compared to the other two feature sets, obtaining the smallest average ranks for all 4 metrics in both Scenarios, with only two ties with BL+CTFs: Specificity for Scenario 2 and GMean for Scenario 1, both in the Elsa-core datasets. The same trend of better baseline results for TILDA datasets, and better CTF results in Elsa-core datasets (when compared to the BL+CTFs and CTFs-only results with Elsa-nurse and TILDA datasets) we saw in the Random Forest experiments can be observed in these Decision Tree experiments. However, we believe that adding CTFs to the longitudinal dataset (BL+CTFs feature set) did not have the desired effect in the decision tree classifiers mainly because of the dimensionality increase caused by creating new features. As the depth of the tree increases, the sample size diminishes, so the tree has fewer data points to make a decision about its split point, and when it is considering all features at once, as is the case in C4.5, having too many features to choose from can negatively impact the performance of the resulting model.

The Friedman test p-values for Scenario 1 had p-values 0.0001, 0.0272, 9E−5 and 0.0007 for Sensitivity, Specificity, Accuracy and GMean, respectively. In the Nemenyi post-hoc test, for Sensitivity, the pair Base-el vs. BL+CTFs got a p-value of 0.0125, and the pair Base-el vs. CTFs-only got a p-value of 0.001. For Specificity, only Base-el vs. CTFs-only had a significant p-value, 0.0266. For Accuracy, both the pair Base-el vs. CTFs-only and the pair Base-el vs. BL+CTFs got significant p-values, of 0.001 and 0.0184 respectively. The same for GMean, with a p-value of 0.001 for Base-el vs. CTFs-only, and a p-value of 0.0377 for Base-el vs. BL+CTFs.

The Friedman test p-values for Scenario 2 had p-values $6e-05$, 0.3710, 0.0002 and .0322 for Sensitivity, Specificity, Accuracy and GMean, respectively; so we did not run the Nemenyi post-hoc test for Specificity. The post-hoc results for Sensitivity and Accuracy had significant p-values for the pair Base-el-in vs. CTFs-in (0.001 for both metrics) and for the pair Base-el-in vs. BL+CTFs (0.0011 and

0.0055, respectively). For GMean, the only pair with a significant p-value was Base-el-in vs. CTFs-in, with 0.0266.

The statistical analysis of the results with decision tree classifiers show the baseline sets as the clear winners for all 6 CTFs together, which agrees with what we observed in the Tables. Therefore, we cannot claim that adding constructed features to the original longitudinal dataset is a good strategy when the classification algorithm is sensitive to increasing the dataset's dimensionality. It is worthwhile to highlight that the BL+CTFs feature set did have comparable results to the baseline in some cases, having no significant difference in most comparisons, and achieving a tie in average rank for Specificity in Scenario 1 and GMean in Scenario 2, both for Elsa-core datasets (which have more temporal information available). This indicates that the strategy may be feasible in scenarios with a high amount of temporal information available.

Table 5.6: All 6 CTFs Sensitivity and Specificity results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	**0.611**	0.576	0.603	**0.570**	0.535	**0.570**
EN_Arthritis	**0.558**	0.551	0.551	0.555	0.553	**0.564**
EN_Cataract	0.594	**0.596**	**0.596**	0.573	0.578	**0.581**
EN_Dementia	0.643	**0.662**	0.660	**0.655**	0.608	0.601
EN_Diabetes	**0.812**	0.791	0.804	**0.792**	0.782	0.789
EN_HBP	**0.624**	0.603	0.617	**0.615**	0.592	0.597
EN_Heartattack	**0.643**	0.610	0.618	0.631	**0.648**	0.581
EN_Osteoporosis	**0.598**	0.582	0.589	**0.622**	0.598	0.607
EN_Parkinsons	0.587	**0.596**	0.557	0.515	0.470	**0.576**
EN_Stroke	**0.630**	0.629	0.620	**0.594**	0.606	0.572
EC_Angina	**0.688**	0.681	0.672	**0.660**	**0.660**	**0.660**
EC_Arthritis	**0.724**	0.695	0.701	**0.679**	0.661	0.674
EC_Cataract	0.599	0.638	**0.642**	0.571	**0.622**	0.604
EC_Dementia	0.684	0.732	**0.729**	**0.714**	0.702	0.702
EC_Diabetes	**0.672**	0.658	0.657	**0.684**	0.635	0.647
EC_HBP	**0.621**	0.582	0.600	**0.590**	0.587	0.574
EC_Heartattack	**0.636**	0.607	0.625	0.576	**0.605**	0.592
EC_Osteoporosis	**0.651**	0.592	0.615	0.569	**0.579**	0.555
EC_Parkinsons	0.642	**0.655**	0.652	0.573	0.600	**0.667**
EC_Stroke	**0.665**	0.644	0.654	0.600	0.598	**0.640**
TI_Angina	**0.745**	0.675	0.738	0.732	0.612	**0.768**
TI_Arthritis	0.603	0.580	**0.612**	**0.614**	0.610	0.603
TI_Cancer	**0.533**	0.530	0.510	0.543	0.539	**0.595**
TI_Cataract	**0.691**	0.648	0.659	0.628	0.628	**0.669**
TI_Diabetes	**0.768**	0.744	0.763	**0.758**	0.738	0.738
TI_HBP	**0.652**	0.609	0.640	**0.640**	0.603	**0.640**
TI_Heartattack	**0.760**	0.671	0.737	**0.751**	0.673	0.693
TI_Ministroke	**0.684**	0.638	0.679	**0.676**	0.578	0.667
TI_Osteoporosis	**0.671**	0.650	0.654	0.650	**0.667**	0.622
TI_Stroke	**0.687**	0.589	0.610	**0.646**	0.631	0.631
AvgRank E-Nurse	**1.5**	2.3	2.2	**1.7**	2.4	2.0
AvgRank E-Core	**1.6**	2.3	2.1	**1.9**	2.1	2.1
AvgRank TILDA	**1.1**	2.9	2.0	**1.5**	2.6	2.0
AvgRank Overall	**1.4**	2.5	2.1	**1.7**	2.3	2.0

Table 5.7: All 6 CTFs Accuracy and GMean results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	**0.609**	0.574	0.602	**0.590**	0.555	0.586
EN_Arthritis	**0.557**	0.552	0.556	**0.557**	0.552	**0.557**
EN_Cataract	0.587	0.590	**0.591**	0.583	0.587	**0.588**
EN_Dementia	0.644	**0.661**	0.658	**0.649**	0.634	0.630
EN_Diabetes	**0.809**	0.789	0.802	**0.802**	0.786	0.796
EN_HBP	**0.620**	0.598	0.609	**0.619**	0.597	0.607
EN_Heartattack	**0.643**	0.612	0.616	**0.637**	0.629	0.599
EN_Osteoporosis	**0.601**	0.584	0.591	**0.610**	0.590	0.598
EN_Parkinsons	0.586	**0.595**	0.557	0.550	0.529	**0.566**
EN_Stroke	**0.628**	0.627	0.617	0.612	**0.617**	0.596
EC_Angina	**0.687**	0.680	0.671	**0.674**	0.670	0.666
EC_Arthritis	**0.706**	0.681	0.691	**0.701**	0.678	0.688
EC_Cataract	0.591	**0.633**	0.631	0.585	**0.630**	0.623
EC_Dementia	0.684	**0.731**	0.729	0.699	**0.717**	0.715
EC_Diabetes	**0.673**	0.655	0.656	**0.678**	0.646	0.652
EC_HBP	**0.609**	0.584	0.590	**0.605**	0.585	0.587
EC_Heartattack	**0.633**	0.607	0.623	0.605	0.606	**0.608**
EC_Osteoporosis	**0.644**	0.591	0.610	**0.608**	0.585	0.584
EC_Parkinsons	0.641	**0.654**	0.652	0.606	0.627	**0.659**
EC_Stroke	**0.662**	0.642	0.653	0.632	0.621	**0.647**
TI_Angina	**0.744**	0.672	0.739	0.738	0.643	**0.753**
TI_Arthritis	0.606	0.589	**0.609**	**0.608**	0.595	**0.608**
TI_Cancer	**0.533**	0.530	0.515	0.538	0.535	**0.551**
TI_Cataract	**0.686**	0.647	0.659	0.659	0.638	**0.664**
TI_Diabetes	**0.768**	0.744	0.761	**0.763**	0.741	0.750
TI_HBP	**0.648**	0.607	0.640	**0.646**	0.606	0.640
TI_Heartattack	**0.760**	0.671	0.736	**0.756**	0.672	0.715
TI_Ministroke	**0.684**	0.637	0.679	**0.680**	0.607	0.673
TI_Osteoporosis	**0.669**	0.652	0.651	**0.661**	0.658	0.638
TI_Stroke	**0.687**	0.589	0.610	**0.666**	0.609	0.620
AvgRank E-Nurse	**1.5**	2.4	2.1	**1.5**	2.5	2.1
AvgRank E-Core	**1.6**	2.3	2.1	**1.9**	2.2	1.9
AvgRank TILDA	**1.1**	2.8	2.1	**1.4**	2.9	1.75
AvgRank Overall	**1.4**	2.5	2.1	**1.6**	2.5	1.9

Table 5.8: All 6 CTFs Sensitivity and Specificity results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	**0.616**	0.592	0.600	**0.593**	0.535	0.574
EN_Arthritis	**0.570**	0.562	0.567	0.555	0.569	**0.580**
EN_Cataract	0.591	**0.612**	0.586	0.584	**0.591**	0.568
EN_Dementia	**0.675**	0.673	0.665	**0.628**	0.601	**0.628**
EN_Diabetes	**0.809**	0.794	0.803	**0.797**	0.773	0.785
EN_HBP	**0.624**	0.600	0.617	**0.618**	0.589	0.596
EN_Heartattack	**0.643**	0.627	0.630	**0.633**	0.608	0.579
EN_Osteoporosis	**0.614**	0.603	0.603	**0.593**	0.581	0.592
EN_Parkinsons	**0.631**	0.630	0.629	0.500	**0.652**	0.545
EN_Stroke	0.615	0.622	**0.634**	0.591	**0.610**	0.575
EC_Angina	0.669	**0.679**	0.661	**0.681**	0.674	0.663
EC_Arthritis	**0.727**	0.693	0.709	**0.677**	0.660	0.668
EC_Cataract	**0.647**	0.641	0.635	**0.653**	0.624	0.612
EC_Dementia	**0.742**	**0.742**	0.728	0.758	0.720	**0.801**
EC_Diabetes	**0.678**	0.654	0.657	**0.676**	0.635	0.655
EC_HBP	**0.621**	0.602	0.603	0.582	0.584	**0.586**
EC_Heartattack	**0.637**	0.628	**0.637**	0.626	**0.635**	0.596
EC_Osteoporosis	**0.687**	0.667	0.676	0.606	**0.634**	0.625
EC_Parkinsons	**0.664**	0.651	0.632	0.547	0.627	**0.667**
EC_Stroke	0.638	0.655	**0.656**	0.616	0.587	**0.635**
TI_Angina	**0.751**	0.675	0.737	0.724	0.612	**0.784**
TI_Arthritis	**0.614**	0.580	0.610	0.598	0.610	**0.605**
TI_Cancer	**0.547**	0.530	0.527	0.572	0.539	**0.592**
TI_Cataract	0.617	**0.648**	0.626	0.619	**0.628**	0.596
TI_Diabetes	0.719	**0.744**	0.696	0.730	**0.738**	0.704
TI_HBP	**0.653**	0.609	0.642	**0.646**	0.603	0.640
TI_Heartattack	**0.742**	0.671	0.728	**0.688**	0.673	0.683
TI_Ministroke	**0.693**	0.638	0.673	**0.706**	0.578	0.676
TI_Osteoporosis	**0.667**	0.650	0.640	0.624	**0.667**	0.641
TI_Stroke	**0.682**	0.589	0.609	**0.631**	**0.631**	0.600
AvgRank E-Nurse	**1.3**	2.5	2.3	**1.7**	2.2	2.2
AvgRank E-Core	**1.4**	2.4	2.3	**1.9**	2.2	**1.9**
AvgRank TILDA	**1.3**	2.4	2.3	**1.9**	2.1	2.1
AvgRank Overall	**1.3**	2.4	2.3	**1.8**	2.2	2.1

Table 5.9: All 6 CTFs Accuracy and GMean results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	**0.615**	0.590	0.599	**0.604**	0.563	0.587
EN_Arthritis	0.563	0.565	**0.573**	0.562	0.565	**0.574**
EN_Cataract	0.589	**0.605**	0.580	0.587	**0.602**	0.577
EN_Dementia	**0.674**	0.672	0.664	**0.651**	0.636	0.646
EN_Diabetes	**0.808**	0.791	0.800	**0.803**	0.783	0.794
EN_HBP	**0.622**	0.596	0.608	**0.621**	0.595	0.606
EN_Heartattack	**0.642**	0.626	0.627	**0.638**	0.618	0.604
EN_Osteoporosis	**0.612**	0.601	0.602	**0.604**	0.592	0.597
EN_Parkinsons	**0.630**	**0.630**	0.628	0.562	**0.641**	0.586
EN_Stroke	0.614	0.621	**0.631**	0.603	**0.616**	0.604
EC_Angina	0.670	**0.679**	0.662	0.675	**0.676**	0.662
EC_Arthritis	**0.707**	0.680	0.693	**0.701**	0.676	0.688
EC_Cataract	**0.649**	0.636	0.628	**0.650**	0.632	0.623
EC_Dementia	**0.742**	0.742	0.729	0.750	0.731	**0.764**
EC_Diabetes	**0.677**	0.652	0.657	**0.677**	0.645	0.656
EC_HBP	**0.606**	0.595	0.596	**0.601**	0.593	0.594
EC_Heartattack	**0.637**	0.629	0.635	**0.632**	0.632	0.617
EC_Osteoporosis	**0.680**	0.664	0.672	0.645	**0.650**	**0.650**
EC_Parkinsons	**0.663**	0.651	0.632	0.603	0.639	**0.649**
EC_Stroke	0.637	0.651	**0.654**	0.627	0.620	**0.645**
TI_Angina	**0.750**	0.672	0.739	0.737	0.643	**0.760**
TI_Arthritis	**0.609**	0.589	**0.609**	0.606	0.595	**0.608**
TI_Cancer	**0.548**	0.530	0.531	**0.559**	0.535	0.559
TI_Cataract	0.617	**0.647**	0.623	0.618	**0.638**	0.611
TI_Diabetes	0.719	**0.744**	0.696	0.724	**0.741**	0.700
TI_HBP	**0.651**	0.607	0.641	**0.650**	0.606	0.641
TI_Heartattack	**0.740**	0.671	0.727	**0.714**	0.672	0.705
TI_Ministroke	**0.694**	0.637	0.673	**0.700**	0.607	0.675
TI_Osteoporosis	**0.663**	0.652	0.640	0.645	**0.658**	0.640
TI_Stroke	**0.682**	0.589	0.609	**0.656**	0.609	0.604
AvgRank E-Nurse	**1.6**	2.4	2.1	**1.7**	2.2	2.1
AvgRank E-Core	**1.4**	2.4	2.3	**1.8**	2.3	2.0
AvgRank TILDA	**1.4**	2.5	2.2	**1.6**	2.3	2.2
AvgRank Overall	**1.4**	2.4	2.2	**1.7**	2.3	2.1

5.5 Discussion

5.5.1 Feature importance analysis

In order to further evaluate the impact of adding CTFs to the baseline dataset, we used a feature importance metric to analyse how often the proposed CTFs were selected as the best features in the RF models. For this, we used the Scenario 2 Base-el-in+CTFs datasets for the experiments with all 6 CTFs, as it includes all original features and all proposed constructed features at once.

For this analysis, we considered the feature importance metric implemented for RF in the Weka data mining tool, which is based on the average class-impurity decrease over all nodes where the feature was selected. We selected the top 10 features with the highest average impurity decrease for the RF produced in each of the 10 folds in the cross-validation process, totalling 100 top-ranking features, for each of the 10 classes in each dataset. Tables 5.10, 5.11 and 5.12 show how many times the original (baseline) features and each type of CTF were selected in the Elsa-nurse, Elsa-core and TILDA datasets, respectively. The numbers in brackets after each class (dataset) name in these tables are the corresponding class imbalance ratios for each dataset.

Table 5.10: Feature importance analysis summary, Elsa-nurse datasets.

Class	Original Fs	CTFs	Diff	Ratio	Monot.	DiffAM	AvgDiffAM	Percentile
Arthritis (1.35)	96	4	3	0	0	1	0	0
HBP (1.49)	98	2	1	0	0	1	0	0
Cataract (2.06)	93	7	3	0	0	3	1	0
Diabetes (6.5)	82	18	7	0	4	4	1	2
Osteoporosis (9.85)	90	10	5	0	1	0	4	0
Stroke (15.86)	88	12	4	1	2	3	1	1
Heart Attack (16.7)	72	28	10	0	11	2	2	3
Angina (26.51)	74	26	5	0	7	6	2	6
Dementia (56.96)	64	36	5	1	22	3	1	4
Parkinson's D. (160.3)	55	45	8	5	20	3	5	4
TOTAL	812	188	51	7	67	26	17	20

The trends shown in these Tables confirm our hypothesis that the datasets with more temporal information available seem to benefit more from the added CTFs. The Elsa-core datasets (Table 5.11), which have more measurements for its conceptual features, had more CTF selections overall, while the opposite case is seen in TILDA datasets (Table 5.12), which have less temporal information. Note

Table 5.11: Feature importance analysis summary, Elsa-core datasets.

Class	Original Fs	CTFs	Diff	Ratio	Monot.	DiffAM	AvgDiffAM	Percentile
Arthritis (2.52)	2	98	1	8	12	8	0	69
HBP (2.58)	2	98	1	9	4	8	0	76
Cataract (3.38)	6	94	0	4	7	11	0	72
Diabetes (7.80)	16	84	3	5	14	13	0	49
Osteoporosis (11.84)	14	86	3	6	21	16	0	40
Stroke (18.35)	25	75	12	1	23	12	4	23
Heart Attack (19.06)	22	78	9	3	16	23	0	27
Angina (29.49)	29	71	17	0	15	13	9	17
Dementia (52.20)	33	67	13	0	21	15	7	11
Parkinson's D. (112.07)	36	64	10	0	23	18	5	8
TOTAL	185	815	69	36	156	137	25	392

Table 5.12: Feature importance analysis summary, TILDA datasets.

Class	Original Fs	CTFs	Diff	Ratio	Monot.	DiffAM	AvgDiffAM	Percentile
HBP (2.38)	100	0	0	0	0	0	0	0
Arthritis (2.92)	100	0	0	0	0	0	0	0
Osteoporosis (9.53)	86	14	0	1	0	8	0	5
Cataract (10.83)	92	8	0	0	0	3	0	5
Diabetes (13.44)	89	11	1	0	0	4	0	6
Cancer (17.02)	89	11	0	1	0	4	0	6
Angina (20.70)	81	19	2	3	3	7	0	4
Heart Attack (25.24)	83	17	0	1	1	10	0	5
Ministroke (50.74)	80	20	0	1	4	7	0	8
Stroke (79.62)	74	26	4	3	1	8	0	10
TOTAL	874	126	7	10	9	51	0	49

that a selection, in this context, means that the feature was among the 10 best-ranked features in the classifier, across all features in the dataset. Overall, 18.8%, 81.5% and 12.6% of the best ranked features were CTFs, for the ELSA-nurse, ELSA-core and TILDA datasets, respectively.

For the Elsa-nurse and TILDA datasets, as the class imbalance ratio increased the number of CTFs selected as the top-ranked features also increased in general. Notably, for Arthritis and High Blood Pressure, the classes with more information on the minority class available for training, the CTFs were chosen very few times for the Elsa-nurse datasets, and not at all for TILDA. This indicates that when the classifier has enough training information it tends to select the original features, which is a negative point against the effectiveness of the CTFs. However, we must also consider that the opposite scenario happened for Elsa-core datasets, where increased imbalance reduced the number of CTFs chosen. One possible reason for that is that Elsa-core original features were not very predictive, compared to Elsa-nurse and TILDA original features, but it is also likely that the added temporal information in Elsa-core contributed towards this result.

Regarding the frequency each type of CTF was selected, for classifiers trained with Elsa-nurse datasets the Monotonicity was selected in total 67 times (roughly a third of the times CTFs were selected for these datasets), followed by Diff (51 times) and DiffAgeMean (26 times). For classifiers trained with Elsa-core datasets the Percentile was selected most often (392 times in total, almost half of all selections), followed by Monotonicty (156) and DiffAgeMean (137). Finally, for the classifier trained with TILDA datasets, the best CTFs were DiffAgeMean (51) and Percentile (49), with Ratio being the third best with only 10 selections. The AvgDiffAgeMean CTF was selected the least often for all cases.

From these results, we can conclude that Percentile, DiffAgeMean and Monotonicity were the most successful CTFs in this analysis, even though the latter did not perform well by itself. Percentile and DiffAgeMean both focus on the most recent measurement of a feature, and compare it to the measurements of other individuals of the same age of the respondent. Monotonicity aims to identify upwards or downwards trends in the values of a feature over all measurements.

However, although there is a big difference between the selection frequencies of the different types of CTFs, it is important to highlight that each of them was selected among the best in some cases (except AvgDiffAgeMean in TILDA datasets). Hence, there is no incentive for removing any of them in a data preprocessing phase if we are using Random Forests or other classifiers that are robust against high dimensionality data, as such classifiers' performance is not in principle significantly hindered by the addition of features. Note that these temporal trends would be ignored by the classification algorithm applied to the original dataset, so adding the proposed CTFs to the dataset in a preprocessing phase is an effective and computationally non-expensive approach.

5.5.2 Summary of the results

We have proposed 3 new types of Constructed Temporal Features (CTFs) and investigated whether adding 6 different CTFs to longitudinal datasets increases predictive accuracy. CTFs are inherent to longitudinal data, as they relate to changes over time captured by different measurements of the same features, but they are ignored by standard supervised machine learning algorithms.

In our experiments, we used 30 real-world datasets created from the English

and Irish Longitudinal Studies of Ageing. The datasets have both numeric and nominal features, so we adapted our proposed CTFs to handle both types of data whenever possible. To assess the effect of adding the proposed CTFs to longitudinal datasets, we ran two sets of experiments.

First, we ran a controlled experiment to measure the impact of the CTFs in predictive accuracy. These experiments compared three different feature sets: (a) a baseline set with only the original features used for constructing the CTF being tested, (b) the proposed CTFs only (no original features), and (c) an extended feature set combining sets (a) and (b). In the second set of experiments, we included the original features that were ineligible for CTF creation in all three feature sets. These include highly predictive features such as age and sex, which improved the learned RF models.

Table 5.13 contains a summary of our main experimental results with the RF classifier. The Diff, Ratio and Monotonicity CTFs, which were not proposed in this work and are conceptually simpler, did not perform as well as the DiffAgeMean, AvgDiffAgeMean and Percentile CTFs proposed in this thesis, when evaluated individually. When we added all 6 types of CTFs to the dataset, there was in general an increase in predictive performance for all metrics in Scenario 1, and for Specificity and GMean in Scenario 2. However, when performing the same set of experiments using the C4.5 decision tree algorithm, the baseline feature sets (no CTF addition) outperformed the alternatives in the majority of the cases.

Table 5.14 shows a summary of the statistical significance results obtained by applying the Friedman and Nemenyi tests to the experiments with the DiffAgeMean CTFs, AvgDiffAgeMean CTFs, Percentile CTFs and all 6 types of CTFs together. The first two columns of this table show the type of CTF and the scenario (1 or 2). The last column shows the predictive performance measure and the comparisons whose results were statistically significant when applying the Nemenyi post-hoc test (ran after a significant p-value in the Friedman's test), using the symbol $>$ to indicate significantly better than.

As shown in Table 5.14, the vast majority of the statistically significant results involve the CTFs-only and CTFs-in feature sets being significantly outperformed by either the baseline or BL+CTFs feature sets. There are only four cases in the table where there was a statistically significant difference between the Base-el

Table 5.13: Best feature subset for each combination of Scenario and predictive performance measure, considering the Overall Average Rank results (30 datasets, including all 3 data sources), for the random forest classifier. In the Table, BL represents the Base-el and Base-el-in datasets (for Scenarios 1 and 2, respectively), which include only original features used for generating the CTF, and BL+CTFs represents Base-el+CTFs and Base-el-in+CTFs (for Scenarios 1 and 2, respectively), which includes both original features and the proposed CTFs. Ineligible features that cannot be used for CTF creation are included in all feature sets in Scenario 2, and excluded in Scenario 1.

CTF Type	Scenario 1				Scenario 2			
	Sensitivity	Specificity	Accuracy	GMean	Sensitivity	Specificity	Accuracy	GMean
Diff	BL	BL	BL	BL	BL	BL	BL	BL
Ratio	BL+CTFs	BL+CTFs	BL+CTFs	BL+CTFs	BL	BL+CTFs	BL	BL
Monotonicity	BL	BL+CTFs	BL	Tie: BL, BL+CTFs	BL	BL	BL	BL
DiffAgeMean	BL+CTFs	BL+CTFs	BL+CTFs	BL+CTFs	BL+CTFs	BL	BL+CTFs	BL+CTFs
AvgDiffAgeMean	BL	BL+CTFs	Tie: BL, BL+CTFs	BL+CTFs	BL	BL+CTFs	BL	Tie: BL, BL+CTFs
Percentile	BL+CTFs	BL+CTFs	BL+CTFs	BL+CTFs	BL+CTFs	BL+CTFs	Tie: BL, BL+CTFs	Tie: BL, BL+CTFs
All 6 CTFs	BL+CTFs	BL+CTFs	BL+CTFs	BL+CTFs	BL	BL+CTFs	BL	BL+CTFs

and BL+CTFs sets (Scenario 1), and in all these four cases BL+CTFs significantly outperformed the Base-el (i.e. the reverse was not true in any case). These four significant results were obtained by DiffAgeMean for three performance metrics (Sensitivity, Accuracy and GMean) and by AvgDiffAgeMean for one metric (Specificity).

We also looked at feature importance measurements in the Random Forests generated with the full datasets combined with all 6 types of CTFs. Considering the 10 best-ranked features in each RF learned in the 10-fold cross-validation process, for all datasets, we verified that the Percentile, Monotonicity and DiffAgeMean CTFs were the most commonly selected types of CTF. Percentile and DiffAgeMean are new contributions of this work, whilst Monotonicity was proposed by Pomsuwan and Freitas (2017) for numerical features only, and in this work they were also extended to ordered nominal features. In the feature importance analysis we also observed that the datasets with more temporal information (i.e., more consecutive measurements of conceptual features) had benefited considerably more from the added CTFs.

Note that, for the majority of our RF experiments, and all DT experiments, the proposed approach was not able to significantly outperform the Baseline approach of not including CTFs in the dataset. Therefore, we cannot currently claim

Table 5.14: Summary of statistical significance results when using the RF classifier, by type of CTF, experimental scenario and performance metric (in the last column, '>' denotes 'significantly better than'.

CTF	Scenario	Statistical significance results by performance metric
DiffAgeMean	1	Sensitivity: BL+CTFs >{Base-el, CTFs}, Specificity: {Base-el, BL+CTFs} >CTFs, Accuracy: BL+CTFs >{Base-el, CTFs}, G-mean: BL+CTFs >Base-el >CTFs
	2	Sensitivity: BL+CTFs >CTFs-in, Specificity: {Base-el-in, BL+CTFs} >CTFs-in, Accuracy: BL+CTFs >CTFs-in, G-mean: {Base-el-in, BL+CTFs} >CTFs-in
AvgDiffAgeMean	1	Sensitivity: {Base-el, BL+CTFs} >CTFs, Specificity: BL+CTFs >{Base-el, CTFs}, Accuracy: {Base-el, BL+CTFs} >CTFs, G-mean: BL+CTFs >CTFs
	2	Sensitivity: {Base-el-in, BL+CTFs} >CTFs-in, Accuracy: {Base-el-in, BL+CTFs} >CTFs-in
Percentile	1	Sensitivity: BL+CTFs >CTFs, Specificity: {Base-el, BL+CTFs} >CTFs, Accuracy: BL+CTFs >CTFs, G-mean: BL+CTFs >CTFs
	2	Sensitivity: {Base-el-in, BL+CTFs} >CTFs-in
All 6 CTFs	1	Sensitivity: BL+CTFs >CTFs, Accuracy: {Base-el, BL+CTFs} >CTFs, G-mean: BL+CTFs >CTFs
	2	Sensitivity: Base-el-in >CTFs-in, G-mean: BL+CTFs >CTFs-in

that including CTFs in a preprocessing step will lead to better models for other longitudinal datasets. However, our results do have enough promise in them, we believe, to encourage further efforts in this direction, as the BL+CTFs set did perform well in some scenarios, and more longitudinal data seems to benefit this approach, based on our observations.

The results in this Chapter show that adding features representing temporal information into longitudinal datasets is a feasible, albeit still not completely matured, strategy to mitigate the issue of having this information ignored by classification algorithms. A more sophisticated version of this CTF creation approach would be implementing a data-driven selection of which CTFs increase the predictive accuracy of a target longitudinal dataset, adapting the proposal to have only the CTFs that reached a set threshold of performance being added to the dataset. In addition, it would be interesting to ask healthcare professionals to

analyse our proposed CTFs and give feedback about their clinical validity and interest, possibly proposing new features based on what type of temporal pattern would be considered relevant for clinical or healthcare research purposes.

Chapter 6

A New Lexicographic Split Criterion for Decision Tree-based Classification Algorithms

As discussed in previous Chapters, 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. Supervised machine learning (ML) methods can be adapted to 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 Chapter we propose an algorithm adaptation approach, using the UKLI representation of longitudinal data (see taxonomy on Chapter 3). The proposal is an adaptation to decision tree-based classification algorithms that uses the time-related information of longitudinal data to increase predictive accuracy.

The contribution of this Chapter, described in Section 6.1, is 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 process for learning a decision tree. In essence, the proposed lexicographic approach gives priority to select features with a higher information gain ratio, but when candidate features have approximately the same highest gain ratio, the most recent feature among those is selected. The contributions in this Chapter were published in a conference paper (Ribeiro and Freitas 2020).

6.1 Lexicographic Approach Definition

The lexicographic split adaptation for tree-based classifiers 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. More precisely, when choosing the feature to be used in a node's split, the decision trees in our adapted algorithms (C4.5 decision tree and Random Forests) will consider 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 this bi-objective feature evaluation is that we intend to add a bias favouring more recent information. 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. Intuitively, the further in the past a feature value was measured, the less it is related to the class label. However, we always prioritise gain ratio over the time index as 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.

Another argument for using the proposed lexicographic split is that it leads to classification models that are less dependent on older data. This is desirable because longitudinal datasets created for classification problems, especially in the ageing studies used as data sources in this thesis, tend to have more missing values in the earlier waves. As many instances are added to study as it has new waves added, and instances from participants who left the study will not be present in the target wave (so they don't have a class value, and must be discarded for the classification datasets), the tendency is that the closer to the target wave, the less likely a feature is to have a missing value due to attrition (naturally, this has no effect on other reasons for missing data). Note that, although all missing values in the datasets used in our experiments have been imputed using the approach proposed in Chapter 4, the estimated values are inherently less precise, thus this argument is still valid for datasets with no missing values left in them.

This approach of optimising objectives in priority order is sometimes called the lexicographic approach (Freitas 2004), and it has been used in decision tree algorithms for conventional (non-longitudinal) classification before (Basgalupp et al.

2009). However, to the best of our knowledge, a lexicographic approach such as the one proposed in this Chapter has never been used for longitudinal classification before. However, a similar strategy of using time-related information in the split decision was used in (Deng et al. 2013), where the authors combine entropy gain and a time-related distance measure in their split criteria, for an application in time series datasets.

We implemented and tested the proposed lexicographic split approach for Random Forest and C4.5 decision trees, but it can be applied to any decision tree-based classification algorithm. In order to make our approach compatible with but independent from previous contributions of this thesis, we propose two versions of the lexicographic approach, i.e., for datasets with and without the constructed temporal features proposed in Chapter 5. We report the results of a series of experiments using RF on Sections 6.2 and 6.3, using Baseline datasets with only the original features and datasets with added Constructed Temporal Features (CTFs), respectively. The decision tree results are presented in Appendix D.

6.1.1 Lexicographic approach for Baseline datasets

The base version of the proposed lexicographic split approach involves longitudinal datasets that do not have Constructed Temporal Features (CTFs). Most features in these datasets (with the exception of some demographic features such as sex, which are set with the most recent feature wave as their time-index by default) take a value associated with a single wave of the study, regardless of whether they have multiple measurements or not. For instance, cholesterol is a feature measured in multiple waves, taking a value for each wave. By contrast, in general a CTF does not have a single time index, since each CTF takes a value that is typically calculated from the values of original features in multiple waves (multiple time indexes), which complicates the definition of the lexicographic approach, as discussed in more detail later. Thus, the lexicographic approach for baseline datasets (the focus of this Section) is defined as follows.

For the C4.5 decision tree algorithm, the standard split-feature selection considers every feature of the dataset in each node of the tree, ordering them based on their Information gain ratio $g(f_{i,j})$ (feature i measured at time j) for that node, selecting the feature with the greater gain value for splitting the data.

In the standard split-feature selection used by Random Trees in the RF algorithm, instead of using all available features, the algorithm first randomly samples a set of candidate features S from the dataset ($|S| = mtry$, with mtry being a user-defined parameter for how many of the features are sampled). Then, it orders the features in S based on their information gain ratio and selects the one with 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 eligible (i.e., all features in C4.5 decision trees and the randomised pool of features sampled for the current node in Random Trees) features that were considered 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 6.1. 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 6.1 Base version of the Lexicographic Split Feature Selection function, applied at each node of a decision tree, for a dataset without CTFs. It receives a set of eligible features S and a user-specified tie-threshold th, and returns the selected $splitfeature$, based on gain ratio and the feature's time index.

```
1: function LexicographicSplitFeatureSelection(S, th)
2:     S.DescendingOrder(gainratio)
3:     splitfeature ← S[0]
4:     CandidateFeatures.add(splitfeature)
5:     pos ← 1
6:     while |g(splitfeature) − g(S[pos])| < th AND pos < S.length do
7:         CandidateFeatures.add(S[pos])
8:         pos + +
9:     end while
10:    CandidateFeatures.DescendingOrder(time-index)
11:    splitfeature ← CandidateFeatures[0]
12:     return splitfeature
13: end function
```

As an example of how the lexicographic feature-split approach works, consider a set S (the set of eligible features to be selected on a given node split) 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 (giving it a greater time-index value). 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 case of a tie for both the gain ratio value and the time-index 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 only the training set instances. This internal cross-validation creates classifiers using 11 possible threshold values (0.0, to 0.05, with 0.005 increments), and chooses the value that yields the model with the best average GMean over its 5 folds.

Note that a th value of 0.0 does not mean that the lexicographic split would not be applied (i.e., two features would never be considered tied). The gain values are often very close, with differences small enough that a subtraction operation in Java would return a 0 value. For nodes in lower depths of a decision tree, where the number of instances in the dataset is very low, exact ties happen often and are detected even with a 0.0 threshold. The threshold value th is selected using this data-driven approach for each fold in the external cross-validation process. In our experiments, we first show a comparison of using this data-driven threshold selection against fixing the threshold value as each of the 11 values tested in the approach, to show that this automated selection is superior to asking the user to define the parameter.

6.1.2 Lexicographic approach for Baseline+CTF datasets

In the case of the Constructed Temporal Features (CTFs) proposed in Chapter 5, they do not have a single, precise time index. Even though CTFs are calculated

using measurements of original features, meaning we could analyse them in the context of how recent they are, or have other reasoning for considering a particular time index for a CTF (such as its performance in previous experiments), we decided to disregard them in the lexicographic split approach. The rationale for this choice was that attributing a time index for the CTFs would be subjective, and would require separate experiments to justify each choice (which we leave for future research).

For CTFs or any other features without a single, precise time index, the lexicographic approach disregards them as candidate features for the tie-breaking criterion of the bi-objective split. In essence, if a feature without a single, precise time index is selected as the split feature for having the greatest information gain ratio, the split is done using that feature (no tie threshold is considered). However, if it would be considered by the lexicographic approach as a candidate feature (i.e., if it has a gain ratio difference within the threshold against the currently selected split feature), it will be disregarded as well. Thus, features without time indexes are never replaced nor used as a replacement by the lexicographic approach. This modified version of the lexicographic approach split, which considers CTFs, is described in Algorithm 6.2.

Algorithm 6.2 Modified version of the Lexicographic Split Feature Selection function, applied at each node of a decision tree, for a dataset with added CTFs. Receives a set of eligible features S and a tie-threshold th (set by the user), and returns the selected $splitfeature$, based on gain ratio and the feature's time-index.

```
1: function LexicographicSplitFeatureSelection(S, th)
2:     S.DescendingOrder(gainratio)
3:     splitfeature ← S[0]
4:     if !IsCTF(splitfeature) then
5:         CandidateFeatures.add(splitfeature)
6:         pos ← 1
7:         while |g(splitfeature) − g(S[pos])| < th AND pos < S.length do
8:             pos + +
9:             if !IsCTF(S[pos]) then
10:                CandidateFeatures.add(S[pos])
11:            end if
12:        end while
13:        CandidateFeatures.DescendingOrder(time-index)
14:        splitfeature ← CandidateFeatures[0]
15:    end if
16:    return splitfeature
17: end function
```

6.2 RF Results for Baseline Datasets

We compared the RF with the lexicographic approach against the standard RF (without the lexicographic approach) in experiments using the ELSA-nurse, ELSA-core and TILDA datasets, both without (Baseline datasets) and with (Baseline+CTF datasets) the addition of the 6 constructed feature types proposed in Chapter 5. For each set of experiments, we also compared fixing the threshold value as each of the values tried by the data-driven threshold selection to using it.

For all experiments, we report the Sensitivity, Specificity, Accuracy and GMean results for each of our 30 datasets, as well as the average ranks obtained by each approach, both for each type of dataset (ELSA-nurse, ELSA-core and TILDA) separately and for all datasets together. The experimental setup is the one used in our previous experiments: all missing values in the datasets were estimated using our data-driven approach proposed in Chapter 4, all training datasets were undersampled using the Balanced Random Forest method (i.e., individual undersampling for each tree in the RF), and all reported results are the average

values from a 10-fold cross-validation process, with the default RF parameters ($ntrees = 100$ and $mtry = \lfloor log_2(d) \rfloor + 1$).

The lexicographic split approach requires a threshold value for defining gain ratio ties (i.e., how close the difference between the gain values must be for them to be considered equivalent), and this added parameter would need to be defined by the user. However, we propose a data-driven automated threshold value selection that uses an internal cross-validation process with training set instances and chooses a threshold value for each RF (each fold of the external, main cross-validation). In order to confirm that this automated process is preferable to having the user define the threshold value, we compared fixing the threshold value as each of the 11 values tested by the automated approach (0.0 to 0.05 with 0.005 increments, range defined after preliminary experiments) against using it. Tables 6.1 and 6.2 contain the Accuracy and GMean results of those experiments, respectively. The Sensitivity and Specificity results for the threshold experiments are presented in Appendix D.1.

Table 6.1: Accuracy results for threshold selection experiments in the Baseline datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**Data-Driven**
EN_Angina	0.678	0.683	0.681	0.682	0.686	0.685	0.695	0.683	0.687	0.685	0.684	0.693
EN_Arthritis	0.633	0.634	0.636	0.634	0.633	0.632	0.623	0.634	0.633	0.632	0.625	0.635
EN_Cataract	0.655	0.655	0.657	0.659	0.657	0.659	0.67	0.652	0.656	0.661	0.656	0.66
EN_Dementia	0.732	0.733	0.733	0.733	0.737	0.737	0.734	0.733	0.734	0.733	0.732	0.74
EN_Diabetes	0.847	0.847	0.846	0.848	0.848	0.847	0.856	0.845	0.847	0.846	0.845	0.845
EN_HBP	0.687	0.689	0.689	0.686	0.68	0.683	0.695	0.685	0.685	0.679	0.676	0.688
EN_HeartAttack	0.704	0.705	0.702	0.699	0.701	0.699	0.707	0.701	0.701	0.701	0.699	0.705
EN_Osteoporosis	0.656	0.658	0.661	0.657	0.657	0.653	0.674	0.654	0.652	0.657	0.658	0.66
EN_Parkinsons	0.633	0.633	0.632	0.632	0.634	0.636	0.658	0.646	0.642	0.645	0.644	0.634
EN_Stroke	0.674	0.673	0.673	0.672	0.672	0.682	0.686	0.677	0.677	0.678	0.683	0.679
EC_Angina	0.711	0.711	0.713	0.713	0.713	0.705	0.708	0.714	0.714	0.714	0.712	0.71
EC_Arthritis	0.735	0.737	0.735	0.734	0.736	0.738	0.736	0.74	0.736	0.734	0.735	0.739
EC_Cataract	0.644	0.642	0.643	0.644	0.645	0.649	0.65	0.652	0.654	0.651	0.652	0.651
EC_Dementia	0.765	0.764	0.766	0.766	0.767	0.769	0.767	0.769	0.77	0.772	0.772	0.768
EC_Diabetes	0.686	0.682	0.683	0.685	0.681	0.679	0.68	0.679	0.672	0.677	0.675	0.684
EC_HBP	0.64	0.649	0.65	0.647	0.645	0.647	0.642	0.642	0.649	0.645	0.647	0.645
EC_HeartAttack	0.68	0.679	0.682	0.682	0.687	0.685	0.684	0.684	0.683	0.684	0.681	0.684
EC_Osteoporosis	0.695	0.694	0.697	0.693	0.695	0.701	0.699	0.696	0.697	0.696	0.697	0.699
EC_Parkinsons	0.695	0.698	0.704	0.699	0.698	0.7	0.702	0.701	0.701	0.701	0.704	0.702
EC_Stroke	0.693	0.694	0.699	0.695	0.694	0.695	0.697	0.699	0.698	0.699	0.699	0.699
TI_Angina	0.755	0.756	0.755	0.755	0.756	0.754	0.751	0.75	0.748	0.748	0.745	0.755
TI_Arthritis	0.711	0.702	0.706	0.709	0.699	0.7	0.698	0.686	0.694	0.695	0.694	0.706
TI_Cancer	0.558	0.556	0.544	0.545	0.541	0.54	0.54	0.544	0.55	0.541	0.541	0.542
TI_Cataract	0.705	0.701	0.703	0.708	0.704	0.701	0.7	0.701	0.697	0.7	0.696	0.701
TI_Diabetes	0.78	0.782	0.78	0.783	0.781	0.779	0.779	0.778	0.772	0.772	0.772	0.776
TI_HBP	0.71	0.709	0.706	0.707	0.706	0.705	0.704	0.709	0.702	0.701	0.701	0.711
TI_HeartAttack	0.757	0.755	0.756	0.754	0.754	0.755	0.753	0.753	0.755	0.749	0.749	0.753
TI_Ministroke	0.709	0.707	0.712	0.705	0.705	0.704	0.708	0.71	0.705	0.7	0.701	0.707
TI_Osteoporosis	0.693	0.691	0.69	0.69	0.688	0.682	0.683	0.678	0.675	0.677	0.678	0.683
TI_Stroke	0.726	0.728	0.728	0.726	0.722	0.724	0.721	0.721	0.72	0.714	0.711	0.718
AvgRank Elsanurse	8.20	6.50	6.55	7.60	6.60	6.75	**2.45**	7.70	6.70	6.70	8.25	4.00
AvgRank Elsacore	9.55	8.80	5.75	8.05	7.20	5.95	6.55	5.00	5.05	5.85	5.55	**4.70**
AvgRank TILDA	2.35	**3.35**	3.65	3.90	5.55	7.25	7.95	7.25	8.90	10.60	11.00	6.25
AvgRank Overall	6.70	6.22	5.32	6.52	6.45	6.65	5.65	6.65	6.88	7.72	8.27	**4.98**

Table 6.2: GMean results for threshold selection experiments in the Baseline datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**Data-Driven**
EN_Angina	0.687	0.695	0.678	0.697	0.695	0.689	0.695	0.683	0.69	0.705	0.685	0.692
EN_Arthritis	0.626	0.628	0.629	0.627	0.625	0.625	0.623	0.628	0.627	0.626	0.618	0.628
EN_Cataract	0.67	0.669	0.672	0.674	0.672	0.673	0.67	0.666	0.67	0.675	0.669	0.674
EN_Dementia	0.734	0.725	0.731	0.731	0.736	0.73	0.734	0.728	0.728	0.725	0.728	0.732
EN_Diabetes	0.853	0.852	0.852	0.858	0.856	0.855	0.856	0.851	0.853	0.85	0.852	0.853
EN_HBP	0.695	0.696	0.697	0.694	0.687	0.691	0.695	0.693	0.693	0.687	0.684	0.696
EN_HeartAttack	0.708	0.71	0.719	0.712	0.714	0.709	0.707	0.712	0.71	0.711	0.71	0.722
EN_Osteoporosis	0.676	0.68	0.676	0.678	0.673	0.675	0.674	0.679	0.675	0.678	0.676	0.684
EN_Parkinsons	0.671	0.671	0.664	0.649	0.643	0.644	0.658	0.649	0.661	0.67	0.67	0.635
EN_Stroke	0.7	0.688	0.697	0.682	0.686	0.699	0.686	0.69	0.692	0.689	0.698	0.695
EC_Angina	0.718	0.725	0.734	0.73	0.741	0.724	0.73	0.746	0.743	0.741	0.738	0.723
EC_Arthritis	0.732	0.733	0.731	0.73	0.733	0.735	0.732	0.734	0.73	0.729	0.73	0.735
EC_Cataract	0.662	0.66	0.662	0.66	0.662	0.665	0.666	0.668	0.671	0.666	0.668	0.668
EC_Dementia	0.762	0.767	0.762	0.759	0.766	0.769	0.766	0.775	0.77	0.762	0.771	0.772
EC_Diabetes	0.716	0.71	0.711	0.716	0.712	0.71	0.709	0.71	0.704	0.707	0.709	0.717
EC_HBP	0.644	0.653	0.653	0.649	0.647	0.648	0.645	0.645	0.652	0.648	0.651	0.649
EC_HeartAttack	0.692	0.689	0.69	0.694	0.697	0.693	0.69	0.702	0.69	0.691	0.688	0.686
EC_Osteoporosis	0.688	0.682	0.684	0.677	0.678	0.688	0.683	0.683	0.689	0.687	0.686	0.689
EC_Parkinsons	0.687	0.702	0.699	0.696	0.709	0.703	0.717	0.717	0.711	0.717	0.705	0.717
EC_Stroke	0.7	0.704	0.707	0.706	0.704	0.705	0.713	0.709	0.705	0.71	0.71	0.713
TI_Angina	0.824	0.828	0.822	0.822	0.819	0.822	0.817	0.812	0.811	0.82	0.815	0.812
TI_Arthritis	0.692	0.688	0.689	0.695	0.684	0.686	0.686	0.674	0.683	0.685	0.684	0.691
TI_Cancer	0.577	0.582	0.564	0.555	0.548	0.561	0.567	0.571	0.581	0.56	0.554	0.544
TI_Cataract	0.715	0.713	0.721	0.716	0.721	0.722	0.716	0.712	0.704	0.718	0.707	0.707
TI_Diabetes	0.8	0.808	0.807	0.806	0.804	0.806	0.806	0.803	0.8	0.805	0.809	0.802
TI_HBP	0.718	0.717	0.716	0.717	0.715	0.714	0.714	0.718	0.712	0.71	0.711	0.72
TI_HeartAttack	0.806	0.809	0.812	0.807	0.811	0.82	0.817	0.819	0.812	0.81	0.813	0.809
TI_Ministroke	0.722	0.716	0.728	0.71	0.72	0.719	0.721	0.722	0.72	0.713	0.708	0.73
TI_Osteoporosis	0.742	0.737	0.741	0.741	0.737	0.736	0.739	0.735	0.734	0.732	0.736	0.732
TI_Stroke	0.732	0.725	0.725	0.717	0.715	0.723	0.73	0.73	0.729	0.719	0.709	0.713
AvgRank Elsanurse	5.80	6.10	5.20	5.20	6.95	7.30	7.30	7.65	7.25	6.55	8.40	**4.30**
AvgRank Elsacore	8.45	7.90	7.05	8.20	6.70	6.20	6.75	4.25	5.50	6.65	6.20	**4.15**
AvgRank TILDA	4.55	5.10	**4.10**	6.25	7.40	5.55	5.30	6.40	8.45	8.25	8.65	8.00
AvgRank Overall	6.27	6.37	**5.45**	6.55	7.02	6.35	6.45	6.10	7.07	7.15	7.75	5.48

The results in these tables show that the proposed automated threshold selection approach is the most consistent, as it often obtains the smallest average rank values. Considering all 30 datasets, the only measure where the automated approach did not have the best (smallest) average rank was Accuracy (including Sensitivity and Specificity, see Tables in Appendix D.1), where it was the second best with a 6.02 average rank against 5.97 for the 0.04 threshold. Note that we are comparing a single approach against 11 others, so the consistency the data-driven automated threshold selection obtained is impressive. Even though in some situations a specific threshold value obtained considerably smaller average ranks (e.g., the 2.35 average Accuracy rank for the 0.0 threshold in TILDA datasets, compared to the 6.25 obtained by the automated selection), it is clear that fixing the value for this parameter is an unnecessary risk that most users would not want to take. This result is positive for our proposed lexicographic approach; even though it does require an additional user-defined parameter, the value of this parameter can be reliably chosen via an automated approach.

Although the sample sizes for comparing 12 different approaches is arguably small, we ran Friedman's rank-based tests (comparing all methods simultaneously) to investigate whether differences between these approaches can be considered statistically significant. In the cases where the Friedman p-value was smaller than the 0.05 threshold, we ran the post-hoc Nemenyi test (pairwise comparison), but only report here on the results comparing the proposed data-driven automated threshold selection (referred to as DD in these results) to each fixed threshold value, as differences between different fixed values were not relevant for this analysis. For this set of experiments with Baseline datasets, the statistical analysis results were as follows.

In the comparison for all 30 datasets, we had a significant Friedman p-value (0.0233) for the Accuracy measure, and the post-hoc test only had a significant p-value (0.0215) when comparing DD vs the 0.05 threshold. Regarding the Elsa-nurse datasets, again, Accuracy had a significant Friedman p-value of 0.0095, but none of the DD comparisons in the post-hoc test had a significant result. For Elsa-core datasets we had two significant Friedman p-values: 0.0128 for Sensitivity and 0.0276 for Accuracy, but both post-hoc tests did not have significant p-values with DD comparisons. Finally, when testing for the TILDA datasets, we had significant Friedman p-values for Sensitivity ($< 1E-16$), which had one significant post-hoc

p-value (0.0318) when comparing DD and the 0.05 threshold, and for Accuracy ($< 1E-16$), which did not have any significant post-hoc p-values for DD.

After these initial experiments confirmed that the automated threshold selection is the most reliable way to define the value for the tie threshold parameter, we performed experiments comparing using the proposed lexicographic approach (with the data-driven threshold selection) against not using it (we named these approaches Lexic and NoLexic, respectively). The results are reported in Table 6.3, where, for each metric, the best average ranks are shown in bold font.

Table 6.3: Comparison of Lexic and NoLexic approaches for Baseline datasets.

Datasets	SENSITIVITY		SPECIFICITY		ACCURACY		GMEAN	
	Lexic	NoLexic	Lexic	NoLexic	Lexic	NoLexic	Lexic	NoLexic
EN_Angina	0.693	0.684	0.69	0.702	0.693	0.684	0.692	0.693
EN_Arthritis	0.669	0.671	0.589	0.586	0.635	0.635	0.628	0.627
EN_Cataract	0.63	0.62	0.72	0.723	0.66	0.654	0.674	0.67
EN_Dementia	0.74	0.729	0.723	0.709	0.74	0.729	0.732	0.719
EN_Diabetes	0.843	0.841	0.863	0.866	0.845	0.845	0.853	0.854
EN_HBP	0.647	0.651	0.749	0.749	0.688	0.69	0.696	0.698
EN_HeartAttack	0.703	0.7	0.741	0.738	0.705	0.702	0.722	0.719
EN_Osteoporosis	0.654	0.649	0.716	0.696	0.66	0.654	0.684	0.672
EN_Parkinsons	0.634	0.628	0.636	0.712	0.634	0.629	0.635	0.669
EN_Stroke	0.677	0.67	0.713	0.724	0.679	0.674	0.695	0.697
EC_Angina	0.709	0.711	0.737	0.723	0.71	0.711	0.723	0.717
EC_Arthritis	0.752	0.749	0.719	0.717	0.739	0.736	0.735	0.733
EC_Cataract	0.625	0.609	0.715	0.717	0.651	0.641	0.668	0.661
EC_Dementia	0.768	0.764	0.776	0.77	0.768	0.764	0.772	0.767
EC_Diabetes	0.672	0.674	0.764	0.747	0.684	0.684	0.717	0.71
EC_HBP	0.633	0.641	0.665	0.662	0.645	0.649	0.649	0.651
EC_HeartAttack	0.683	0.678	0.689	0.692	0.684	0.679	0.686	0.685
EC_Osteoporosis	0.701	0.7	0.677	0.676	0.699	0.698	0.689	0.688
EC_Parkinsons	0.701	0.697	0.733	0.693	0.702	0.697	0.717	0.695
EC_Stroke	0.697	0.694	0.729	0.721	0.699	0.695	0.713	0.707
TI_Angina	0.749	0.748	0.88	0.916	0.755	0.756	0.812	0.828
TI_Arthritis	0.729	0.729	0.654	0.646	0.706	0.703	0.691	0.686
TI_Cancer	0.542	0.549	0.546	0.579	0.542	0.55	0.544	0.564
TI_Cataract	0.699	0.706	0.715	0.724	0.701	0.707	0.707	0.715
TI_Diabetes	0.771	0.775	0.834	0.831	0.776	0.779	0.802	0.803
TI_HBP	0.679	0.678	0.764	0.765	0.711	0.711	0.72	0.72
TI_HeartAttack	0.749	0.751	0.873	0.878	0.753	0.756	0.809	0.812
TI_Ministroke	0.706	0.712	0.755	0.735	0.707	0.712	0.73	0.724
TI_Osteoporosis	0.671	0.677	0.799	0.807	0.683	0.689	0.732	0.739
TI_Stroke	0.718	0.728	0.708	0.8	0.718	0.728	0.713	0.763
AvgRank Elsanurse	**1.20**	1.80	1.55	**1.45**	**1.20**	1.80	**1.50**	**1.50**
AvgRank Elsacore	**1.30**	1.70	**1.20**	1.80	**1.25**	1.75	**1.10**	1.90
AvgRank TILDA	1.75	**1.25**	1.70	**1.30**	1.85	**1.15**	1.75	**1.25**
AvgRank Overall	**1.42**	1.58	**1.48**	1.52	**1.43**	1.57	**1.45**	1.55

For the Lexic and NoLexic comparison across all 30 datasets, for all 4 metrics the Lexic overall average rank was smaller, albeit only slightly. The greatest difference between the overall average rank values values was 0.1, for GMean.

When considering each data source separately, we observed a pattern where datasets with more measurements for their conceptual features had better results for the Lexic approach. Thus, considering only the Elsa-core datasets, with 7

feature waves, Lexic wins for all 4 metrics with large differences in the average ranks. Regarding the Elsa-nurse datasets, with 4 feature waves (that are 4 years apart from each other, instead of 2), Lexic wins for Sensitivity and Accuracy, but loses for Specificity and ties for GMean. Finally, regarding the TILDA datasets, which have 4 feature waves (2 years apart from each other) but do not have 4 measurements for most features, NoLexic wins for all 4 metrics.

This pattern corroborates the core principle of the lexicographic split approach, that adding a bias in favour of more recent features would increase predictive accuracy. However the impact of this approach on longitudinal datasets is proportional to the number of waves of data available. The reason for this is twofold: first, the gap in time between a selected feature and its replacement in the lexicographic approach is, on average, larger for datasets with more waves (for example, in Elsa-core datasets it can get to 14 years, between waves 1 to 7), so making a replacement can be more meaningful in that regard; second, a replacement is more likely if there are more recent features to choose from, so the lexicographic approach makes more difference in datasets that have a greater likelihood of ties and replacements (for example, in TILDA datasets there are usually only two measurements of a conceptual feature, so the likelihood that a more recent measurement is equivalent to the one selected is smaller, compared to the 7 measurements in Elsa-core features).

We also compared the ranks of the Lexic and NoLexic approaches using the Wilcoxon signed-rank test. For this analysis, we ran the Wilcoxon test with all 30 datasets, then considering each data source separately. In the overall results analysis, none of the p-values were significant. When considering only Elsa-nurse datasets, Lexic was significantly better than NoLexic for Sensitivity (p-value of 0.0248) and Accuracy (0.0170). In Elsa-core datasets there was a significant difference in Specificity (0.0364) and GMean (0.0142), both in favour of Lexic. Finally, TILDA datasets had a significant result in favour of NoLexic in the Sensitivity (0.0205) Accuracy (0.0204) metrics.

We also measured the effect the proposed lexicographic split approach has on the resulting Random Forest models, i.e., how different the models generated with this split were from the baseline models. For this, we counted in every RF model the proportion of nodes where a tie happened (nodes where more than one

candidate feature had equivalent information gain ratios, according to the tie-threshold parameter) and the proportion of nodes where a replacement happened (nodes where the tie led to a different, more recent feature being selected for the split function).

For the baseline datasets used in these experiments, in the ELSA-nurse models we had an average of 50.3% of nodes where a tie occurred (at least one candidate feature had an equivalent information gain ratio to the first-ranked feature), making them eligible for changing the split feature based on the secondary objective, the time-index. About half of these nodes switched the chosen feature for a more recent feature, resulting in final models that were 26.6% different from the baseline models (standard split function), for these datasets. In the ELSA-core and TILDA datasets we had less frequent ties in the nodes (42.9% and 39.6% average nodes with ties, for ELSA-core and TILDA respectively), thus the final models were not changed as much (23.1% and 25.4% average difference, for ELSA-core and TILDA respectively).

Even though there was clearly a significant change in the final Random Forests, for all data sources, the resulting reflection on predictive accuracy is not expected to be large. This is because the split features eligible for replacement are, by design, equivalent from each other in terms of information gain. However, the chosen split feature dictates the division of the instances among the child nodes, so there is some cascading effect of changing a split feature as the instance subsets going into the child nodes are changed.

In summary, when experimenting with only original features (i.e. no constructed feature) we observed that the proposed lexicographic split approach has a positive impact on predictive accuracy for longitudinal datasets with many measurements of their conceptual features (Elsa-core, Elsa-nurse), but this impact is small or nonexistent in datasets with fewer measurements in their conceptual features (TILDA). Therefore, our recommendation is that the decision about whether to apply this approach be based on the number of waves in the dataset, and the time-gap between those waves, with higher number of waves and larger time gaps increasing the benefits of using the lexicographic split.

6.3 RF Results for Baseline+CTF Datasets

Our second set of experiments combines the original features in the datasets with all 6 types of Constructed Temporal Features (CTFs) proposed in Chapter 5. As explained earlier, the lexicographic approach disregards the CTFs as they do not have a specific, well-defined time-index.

As with the previous experiments, we first compared using the automated data-driven threshold selection approach to fixing the threshold values. Tables 6.4 and 6.5 show the Accuracy and GMean results for these experiments, respectively. The Sensitivity and Specificity results for the threshold experiments are presented in Appendix D.1.

Table 6.4: Accuracy results for threshold selection experiments in the Baseline+CTF datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**Data-Driven**
EN_Angina	0.682	0.684	0.688	0.687	0.686	0.684	0.686	0.682	0.677	0.677	0.68	0.684
EN_Arthritis	0.636	0.644	0.638	0.641	0.639	0.646	0.637	0.637	0.634	0.638	0.639	0.637
EN_Cataract	0.649	0.645	0.65	0.657	0.648	0.651	0.646	0.649	0.649	0.644	0.647	0.654
EN_Dementia	0.742	0.743	0.743	0.741	0.741	0.741	0.743	0.742	0.74	0.742	0.743	0.745
EN_Diabetes	0.846	0.845	0.845	0.846	0.845	0.845	0.847	0.846	0.848	0.846	0.845	0.849
EN_HBP	0.687	0.688	0.684	0.69	0.679	0.683	0.682	0.68	0.684	0.689	0.686	0.686
EN_HeartAttack	0.7	0.701	0.704	0.704	0.705	0.704	0.701	0.702	0.703	0.704	0.706	0.703
EN_Osteoporosis	0.652	0.654	0.652	0.651	0.652	0.648	0.646	0.646	0.647	0.647	0.647	0.649
EN_Parkinsons	0.631	0.632	0.635	0.634	0.639	0.64	0.642	0.644	0.643	0.646	0.648	0.644
EN_Stroke	0.677	0.681	0.679	0.681	0.678	0.68	0.677	0.682	0.677	0.682	0.676	0.682
EC_Angina	0.71	0.706	0.707	0.709	0.708	0.705	0.706	0.709	0.709	0.707	0.705	0.707
EC_Arthritis	0.741	0.738	0.737	0.738	0.74	0.736	0.737	0.738	0.741	0.739	0.735	0.739
EC_Cataract	0.66	0.658	0.658	0.659	0.66	0.66	0.659	0.664	0.659	0.659	0.665	0.664
EC_Dementia	0.767	0.769	0.769	0.766	0.769	0.771	0.767	0.768	0.77	0.771	0.772	0.773
EC_Diabetes	0.691	0.687	0.688	0.692	0.689	0.691	0.688	0.684	0.686	0.689	0.689	0.684
EC_HBP	0.652	0.648	0.653	0.647	0.653	0.66	0.653	0.65	0.655	0.651	0.656	0.65
EC_HeartAttack	0.68	0.675	0.678	0.68	0.677	0.674	0.675	0.679	0.676	0.677	0.678	0.678
EC_Osteoporosis	0.694	0.698	0.695	0.695	0.692	0.692	0.69	0.693	0.695	0.694	0.692	0.694
EC_Parkinsons	0.697	0.695	0.695	0.697	0.698	0.697	0.7	0.701	0.7	0.701	0.703	0.709
EC_Stroke	0.698	0.699	0.699	0.702	0.698	0.699	0.7	0.701	0.697	0.699	0.701	0.698
TI_Angina	0.748	0.748	0.75	0.75	0.747	0.748	0.748	0.752	0.75	0.746	0.747	0.746
TI_Arthritis	0.693	0.698	0.694	0.696	0.695	0.693	0.69	0.695	0.694	0.69	0.694	0.697
TI_Cancer	0.554	0.549	0.548	0.548	0.553	0.555	0.558	0.551	0.554	0.555	0.552	0.557
TI_Cataract	0.704	0.7	0.696	0.702	0.702	0.704	0.702	0.704	0.707	0.705	0.702	0.702
TI_Diabetes	0.761	0.761	0.759	0.755	0.756	0.758	0.76	0.757	0.755	0.754	0.755	0.76
TI_HBP	0.698	0.698	0.703	0.704	0.7	0.699	0.7	0.697	0.695	0.695	0.693	0.701
TI_HeartAttack	0.745	0.746	0.744	0.745	0.745	0.748	0.746	0.745	0.751	0.746	0.747	0.747
TI_Ministroke	0.709	0.706	0.703	0.702	0.704	0.703	0.702	0.7	0.701	0.698	0.699	0.706
TI_Osteoporosis	0.677	0.678	0.678	0.683	0.681	0.679	0.68	0.682	0.681	0.677	0.677	0.679
TI_Stroke	0.705	0.704	0.704	0.705	0.706	0.709	0.708	0.706	0.708	0.706	0.706	0.709
AvgRank Elsanurse	7.90	6.25	5.60	4.65	6.90	6.35	7.70	7.30	8.25	6.20	6.55	**4.35**
AvgRank Elsacore	5.40	8.45	7.35	5.80	6.45	7.20	8.25	5.95	6.05	6.00	**5.30**	5.80
AvgRank TILDA	6.55	6.60	7.80	6.40	6.50	5.10	5.70	6.25	5.40	8.65	8.65	**4.40**
AvgRank Overall	6.62	7.10	6.92	5.62	6.62	6.22	7.22	6.50	6.57	6.95	6.83	**4.85**

Table 6.5: GMean results for threshold selection experiments in the Baseline+CTF datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**Data-Driven**
EN_Angina	0.68	0.683	0.698	0.675	0.686	0.679	0.686	0.686	0.683	0.687	0.685	0.692
EN_Arthritis	0.629	0.637	0.631	0.635	0.633	0.639	0.63	0.63	0.628	0.633	0.634	0.63
EN_Cataract	0.669	0.664	0.668	0.676	0.665	0.669	0.665	0.667	0.667	0.662	0.667	0.672
EN_Dementia	0.733	0.733	0.723	0.725	0.729	0.725	0.72	0.715	0.718	0.729	0.726	0.741
EN_Diabetes	0.854	0.851	0.85	0.854	0.853	0.854	0.853	0.85	0.855	0.85	0.849	0.855
EN_HBP	0.696	0.696	0.691	0.698	0.686	0.69	0.69	0.687	0.692	0.696	0.694	0.693
EN_HeartAttack	0.705	0.706	0.71	0.712	0.713	0.714	0.713	0.715	0.718	0.711	0.72	0.714
EN_Osteoporosis	0.681	0.682	0.678	0.676	0.679	0.674	0.673	0.669	0.671	0.671	0.67	0.677
EN_Parkinsons	0.67	0.684	0.686	0.664	0.674	0.675	0.676	0.677	0.662	0.671	0.672	0.64
EN_Stroke	0.689	0.69	0.69	0.698	0.69	0.69	0.693	0.699	0.689	0.694	0.685	0.694
EC_Angina	0.729	0.719	0.726	0.727	0.722	0.72	0.725	0.727	0.74	0.741	0.737	0.738
EC_Arthritis	0.737	0.735	0.735	0.735	0.737	0.733	0.734	0.735	0.737	0.735	0.731	0.735
EC_Cataract	0.686	0.683	0.684	0.685	0.685	0.687	0.685	0.69	0.684	0.685	0.69	0.686
EC_Dementia	0.79	0.791	0.793	0.783	0.775	0.774	0.79	0.79	0.803	0.794	0.798	0.802
EC_Diabetes	0.722	0.717	0.712	0.719	0.716	0.726	0.722	0.717	0.715	0.719	0.718	0.705
EC_HBP	0.655	0.653	0.657	0.651	0.655	0.663	0.657	0.653	0.658	0.654	0.659	0.655
EC_HeartAttack	0.689	0.683	0.684	0.688	0.697	0.693	0.7	0.698	0.698	0.69	0.695	0.684
EC_Osteoporosis	0.68	0.683	0.682	0.691	0.682	0.679	0.681	0.68	0.678	0.686	0.678	0.678
EC_Parkinsons	0.708	0.707	0.714	0.708	0.722	0.715	0.71	0.704	0.703	0.71	0.705	0.714
EC_Stroke	0.721	0.719	0.719	0.724	0.714	0.716	0.722	0.717	0.718	0.714	0.714	0.723
TI_Angina	0.808	0.806	0.809	0.802	0.801	0.804	0.803	0.81	0.807	0.803	0.802	0.814
TI_Arthritis	0.68	0.686	0.681	0.686	0.684	0.682	0.68	0.686	0.683	0.679	0.682	0.681
TI_Cancer	0.566	0.561	0.555	0.55	0.549	0.562	0.561	0.545	0.558	0.558	0.548	0.576
TI_Cataract	0.72	0.719	0.713	0.721	0.715	0.724	0.714	0.716	0.721	0.716	0.713	0.719
TI_Diabetes	0.785	0.788	0.782	0.784	0.778	0.785	0.785	0.786	0.785	0.782	0.78	0.786
TI_HBP	0.706	0.707	0.711	0.712	0.709	0.708	0.71	0.707	0.705	0.704	0.702	0.71
TI_HeartAttack	0.798	0.791	0.788	0.793	0.789	0.792	0.796	0.791	0.799	0.798	0.797	0.794
TI_Ministroke	0.712	0.711	0.709	0.718	0.724	0.714	0.714	0.712	0.713	0.712	0.702	0.711
TI_Osteoporosis	0.711	0.718	0.717	0.723	0.724	0.72	0.721	0.722	0.722	0.713	0.711	0.713
TI_Stroke	0.699	0.706	0.706	0.714	0.714	0.716	0.715	0.722	0.723	0.722	0.722	0.731
AvgRank Elsanurse	6.75	5.65	6.25	5.45	6.60	6.05	7.30	7.30	8.00	6.80	7.20	**4.65**
AvgRank Elsacore	**5.60**	8.40	6.95	6.55	6.70	6.50	5.70	6.95	6.10	5.80	6.55	6.20
AvgRank TILDA	6.80	6.30	8.40	5.25	7.35	5.25	6.30	5.75	**4.60**	7.80	9.40	4.80
AvgRank Overall	6.38	6.78	7.20	5.75	6.88	5.93	6.43	6.67	6.23	6.80	7.72	**5.22**

For the threshold comparison experiments with the Baseline+CTF datasets, the pattern in favour of the DD approach (data-driven threshold selection) was slightly less clear as it was for the Baseline datasets. The best (smallest) overall average rank for Specificity (see Appendix D.1 for Sensitivity and Specificity Tables) was obtained by the 0.015 threshold, while the DD overall average ranks were the best for Sensitivity, Accuracy and GMean. The consistency of the DD approach is still noticeable; its average ranks never reached very high values – the only DD average ranks above 6 were for Specificity (7) and Accuracy (6.20), both for Elsa-core datasets. Therefore, the automated selection of the best threshold value was considered the best approach for these datasets as well. In the statistical analysis with the Friedman's test, there were no significant p-values for any of the tests (for types of datasets or all datasets combined).

Thus, we kept the conclusion that the automated threshold selection is the most reliable way to define the value for the tie threshold parameter, and performed experiments comparing the Lexic and NoLexic approaches. The results for these experiments are reported in Table 6.6.

Table 6.6: Comparison of Lexic and NoLexic approaches for Baseline+CTF datasets.

Baseline+CTFs	SENSITIVITY		SPECIFICITY		ACCURACY		GMEAN	
Datasets	Lexic	NoLexic	Lexic	NoLexic	Lexic	NoLexic	Lexic	NoLexic
EN_Angina	0.683	0.681	0.702	0.686	0.684	0.681	0.692	0.684
EN_Arthritis	0.669	0.671	0.594	0.594	0.637	0.639	0.63	0.632
EN_Cataract	0.614	0.605	0.734	0.736	0.654	0.648	0.672	0.667
EN_Dementia	0.745	0.743	0.736	0.736	0.745	0.742	0.741	0.74
EN_Diabetes	0.846	0.845	0.864	0.868	0.849	0.848	0.855	0.856
EN_HBP	0.648	0.65	0.742	0.745	0.686	0.688	0.693	0.696
EN_HeartAttack	0.702	0.7	0.726	0.713	0.703	0.701	0.714	0.707
EN_Osteoporosis	0.643	0.643	0.713	0.716	0.649	0.65	0.677	0.679
EN_Parkinsons	0.644	0.627	0.636	0.727	0.644	0.628	0.64	0.675
EN_Stroke	0.681	0.674	0.708	0.72	0.682	0.677	0.694	0.697
EC_Angina	0.705	0.706	0.772	0.761	0.707	0.708	0.738	0.733
EC_Arthritis	0.751	0.756	0.719	0.72	0.739	0.741	0.735	0.738
EC_Cataract	0.627	0.613	0.75	0.764	0.664	0.658	0.686	0.685
EC_Dementia	0.772	0.766	0.832	0.807	0.773	0.767	0.802	0.787
EC_Diabetes	0.677	0.682	0.734	0.758	0.684	0.692	0.705	0.719
EC_HBP	0.632	0.637	0.678	0.673	0.65	0.651	0.655	0.655
EC_HeartAttack	0.677	0.68	0.692	0.717	0.678	0.682	0.684	0.698
EC_Osteoporosis	0.697	0.702	0.659	0.666	0.694	0.699	0.678	0.684
EC_Parkinsons	0.709	0.694	0.72	0.733	0.709	0.694	0.714	0.713
EC_Stroke	0.695	0.696	0.751	0.747	0.698	0.699	0.723	0.721
TI_Angina	0.739	0.742	0.896	0.88	0.746	0.748	0.814	0.808
TI_Arthritis	0.721	0.713	0.642	0.648	0.697	0.693	0.681	0.68
TI_Cancer	0.555	0.552	0.599	0.579	0.557	0.554	0.576	0.566
TI_Cataract	0.699	0.701	0.741	0.741	0.702	0.704	0.719	0.72
TI_Diabetes	0.755	0.757	0.818	0.813	0.76	0.761	0.786	0.785
TI_HBP	0.669	0.668	0.754	0.746	0.701	0.698	0.71	0.706
TI_HeartAttack	0.743	0.741	0.849	0.859	0.747	0.745	0.794	0.798
TI_Ministroke	0.706	0.709	0.716	0.716	0.706	0.709	0.711	0.712
TI_Osteoporosis	0.67	0.668	0.759	0.757	0.679	0.677	0.713	0.711
TI_Stroke	0.708	0.705	0.754	0.692	0.709	0.705	0.731	0.699
AvgRank Elsanurse	**1.25**	1.75	1.70	**1.30**	**1.30**	1.70	1.60	**1.40**
AvgRank Elsacore	1.70	**1.30**	1.60	**1.40**	1.70	**1.30**	**1.45**	1.55
AvgRank TILDA	**1.40**	1.60	**1.30**	1.70	**1.40**	1.60	**1.30**	1.70
AvgRank Overall	**1.45**	1.55	1.53	**1.47**	**1.47**	1.53	**1.45**	1.55

For the Lexic vs NoLexic comparison with Baseline+CTFs datasets, the Lexic approach got the smallest overall average rank for Sensitivity, Accuracy and GMean, and NoLexic got a smaller overall average rank for Specificity. Again, the greatest difference between the overall average rank values values was 0.1, indicating that the approaches had very similar performances.

When looking at each data source individually, we now see a different pattern than the one observed in the first set of experiments. In the Elsa-core datasets, which previously had benefited the most from the lexicographic split, NoLexic got smaller average ranks for Sensitivity, Specificity and Accuracy. Another difference between these experiments and the previous one was for the TILDA datasets, where Lexic won for all four metrics. The Elsa-nurse results had no clear winner, with Lexic getting smaller ranks for Sensitivity and Accuracy, but NoLexic winning for Specificity and GMean. In the Wilcoxon signed-rank test analysis for these experiments, none of the p-values were significant. We tested for each data source separately and for all 30 datasets together.

The reason for the reduced impact of the lexicographic approach is due to the added CTFs. The most important change in the results was the inversion for the Elsa-core datasets, where Lexic was still better for GMean (arguably the most important metric for evaluating a classifier out of the 4 considered), but lost for the other metrics. One explanation for this change is that, as shown in the feature importance experiments from Chapter 5, in Table 5.11, the CTFs are often the best-ranked features in Elsa-core classifiers (over 80% of the top-10 features in the Elsa-core classifiers were CTFs). Therefore, the lexicographic split was not as effective for Elsa-core datasets because most of the top-ranked features, which have a bigger impact in the classification, were disregarded because they were CTFs.

Regarding the effect of the proposed lexicographic split approach in the datasets used in this set of experiments, which have a large volume of added constructed features which are ignored by the approach, we observed an expected reduction in applicability for the replacements. In the RF models generated with the Baseline+CTF feature sets, the average percentage of nodes where a tie happened was 60.7%, for ELSA-nurse datasets, considerably higher than the 50.3% in the Baseline datasets. However, in less than a third of the nodes this resulted in a replacement of the chosen split feature, so there was a 16% average difference in the models, down from 26.6% in the Baseline ELSA-nurse datasets. The smaller impact of the lexicographic split in this second set of experiments is due to the large portion of CTFs in the dataset, as those are not eligible for being replaced or used as replacement even if their gain ratios are equivalent to the other candidate features in a node.

In the ELSA-core datasets, both the number of ties and the change in the final model were reduced. Ties happened in 31.7% of the nodes (down from 42.9%), less than half of them resulted in a different split feature being selected, as the final models were 13.7% (down from 23.1%) different from the ones trained without the lexicographic split function. As the ELSA-core datasets seem to select CTFs as split features more often (see Section 5.5), it is not surprising that the lexicographic approach was less effective in these models.

For the TILDA datasets, we had the same pattern as the ELSA-nurse with more ties but fewer replacements. In average, for 53.2% of the nodes (up from 39.6%) we had multiple candidate features to select from, but less than a third of these resulted in a change in the selected feature based on the secondary objective. The final TILDA models with Baseline+CTF datasets were 14.5% different from the models created without the lexicographic split.

However, the overall results of our experiments still favour the lexicographic split approach. Even though its impact is reduced when the dataset has added features without time-indexes, which are disregarded, the proposed lexicographic split still favours models based on more recent features, which led to slightly improved predictive performance.

6.4 Summarising and Comparing the Experimental Results Regarding Predictive Accuracy

In this Section, we summarise our experimental results in Tables comparing the two sets of experiments, with Baseline datasets and datasets with added CTFs. The summarised results for the threshold selection experiments are shown in Tables 6.7, 6.8, 6.9 and 6.10.

Table 6.7: RF Threshold selection, Sensitivity average rank results summary for Baseline and Baseline+CTF datasets.

BASELINE DATASETS (ORIGINAL FEATURES ONLY)												
Threshold	0	0.005	0.01	0.015	0.02	0.025	0.03	0.035	0.04	0.045	0.05	DD
AvgRank Elsanurse	8.30	5.95	6.80	8.00	5.80	6.25	6.05	7.30	6.20	6.25	7.20	**3.90**
AvgRank Elsacore	9.85	9.55	5.95	7.15	7.00	5.60	6.30	5.15	5.15	**4.70**	5.75	5.85
AvgRank TILDA	**2.60**	3.10	3.75	3.95	5.65	7.30	8.10	7.45	8.70	10.60	11.15	5.65
AvgRank Overall	6.92	6.20	5.50	6.37	6.15	6.38	6.82	6.63	6.68	7.18	8.03	**5.13**
BASELINE+CTF DATASETS (ADDED FEATURES WITHOUT TIME-INDEX)												
Threshold	0	0.005	0.01	0.015	0.02	0.025	0.03	0.035	0.04	0.045	0.05	DD
AvgRank Elsanurse	8.40	6.40	5.10	5.50	6.95	6.80	7.90	6.65	7.90	6.05	6.70	**3.65**
AvgRank Elsacore	5.40	7.85	7.45	6.10	6.15	7.45	8.95	6.15	6.20	5.95	**5.15**	5.20
AvgRank TILDA	6.05	7.25	7.45	6.50	5.75	5.95	5.90	6.50	5.30	8.85	8.10	**4.40**
AvgRank Overall	6.62	7.17	6.67	6.03	6.28	6.73	7.58	6.43	6.47	6.95	6.65	**4.42**

Table 6.8: RF Threshold selection, Specificity average rank results summary for Baseline and Baseline+CTF datasets.

BASELINE DATASETS (ORIGINAL FEATURES ONLY)												
Threshold	0	0.005	0.01	0.015	0.02	0.025	0.03	0.035	0.04	0.045	0.05	DD
AvgRank Elsanurse	5.10	6.00	5.75	**5.05**	6.45	6.20	9.80	7.65	6.30	6.55	7.80	5.35
AvgRank Elsacore	6.75	6.80	7.30	7.95	6.55	7.20	7.35	5.20	5.70	7.35	5.60	**4.25**
AvgRank TILDA	7.30	6.85	**5.15**	7.30	8.15	5.55	4.70	5.90	5.90	6.00	6.75	8.45
AvgRank Overall	6.38	6.55	6.07	6.77	7.05	6.32	7.28	6.25	5.97	6.63	6.72	**6.02**
BASELINE+CTF DATASETS (ADDED FEATURES WITHOUT TIME-INDEX)												
Threshold	0	0.005	0.01	0.015	0.02	0.025	0.03	0.035	0.04	0.045	0.05	DD
AvgRank Elsanurse	5.65	5.65	6.75	5.35	6.80	6.70	7.05	7.50	7.55	7.05	6.85	**5.10**
AvgRank Elsacore	5.60	7.45	6.25	6.40	7.55	6.10	**5.05**	6.65	6.85	6.10	7.00	7.00
AvgRank TILDA	7.55	6.25	8.55	5.20	7.75	5.40	6.30	5.20	**4.85**	6.40	9.20	5.35
AvgRank Overall	6.27	6.45	7.18	**5.65**	7.37	6.07	6.13	6.45	6.42	6.52	7.68	5.82

For the Sensitivity and Specificity comparisons, we can observe that the DD approach (data-driven automated threshold selection using an internal cross-validation process) got smaller average ranks for the Baseline+CTF datasets than for the Baseline datasets, in all cases. For Accuracy and GMean, this is true only for the TILDA datasets and the overall average ranks.

The overall average ranks of the DD approach were the smallest in most cases, with the exception of GMean for the Baseline datasets and Specificity for the Baseline+CTF datasets. However, it is still clear that in both cases the DD approach is the most reliable, when compared to having a user-chosen fixed value for the threshold parameter for defining ties.

Table 6.11 shows the summarised results for the Lexic vs NoLexic comparisons. For these, the only overall average rank result where NoLexic wins is for Specificity in the Baseline+CTF datasets. Over all comparisons, the overall average rank

Table 6.9: RF Threshold selection, Accuracy average rank results summary for Baseline and Baseline+CTF datasets

BASELINE DATASETS (ORIGINAL FEATURES ONLY)												
Threshold	0	0.005	0.01	0.015	0.02	0.025	0.03	0.035	0.04	0.045	0.05	DD
AvgRank Elsanurse	8.20	6.50	6.55	7.60	6.60	6.75	**2.45**	7.70	6.70	6.70	8.25	4.00
AvgRank Elsacore	9.55	8.80	5.75	8.05	7.20	5.95	6.55	5.00	5.05	5.85	5.55	**4.70**
AvgRank TILDA	2.35	**3.35**	3.65	3.90	5.55	7.25	7.95	7.25	8.90	10.60	11.00	6.25
AvgRank Overall	6.70	6.22	5.32	6.52	6.45	6.65	5.65	6.65	6.88	7.72	8.27	**4.98**
BASELINE+CTF DATASETS (ADDED FEATURES WITHOUT TIME-INDEX)												
Threshold	0	0.005	0.01	0.015	0.02	0.025	0.03	0.035	0.04	0.045	0.05	DD
AvgRank Elsanurse	7.90	6.25	5.60	4.65	6.90	6.35	7.70	7.30	8.25	6.20	6.55	**4.35**
AvgRank Elsacore	5.40	8.45	7.35	5.80	6.45	7.20	8.25	5.95	6.05	6.00	**5.30**	5.80
AvgRank TILDA	6.55	6.60	7.80	6.40	6.50	5.10	5.70	6.25	5.40	8.65	8.65	**4.40**
AvgRank Overall	6.62	7.10	6.92	5.62	6.62	6.22	7.22	6.50	6.57	6.95	6.83	**4.85**

Table 6.10: RF Threshold selection, GMean average rank results summary for Baseline and Baseline+CTF datasets

BASELINE DATASETS (ORIGINAL FEATURES ONLY)												
Threshold	0	0.005	0.01	0.015	0.02	0.025	0.03	0.035	0.04	0.045	0.05	DD
AvgRank Elsanurse	5.80	6.10	5.20	5.20	6.95	7.30	7.30	7.65	7.25	6.55	8.40	**4.30**
AvgRank Elsacore	8.45	7.90	7.05	8.20	6.70	6.20	6.75	4.25	5.50	6.65	6.20	**4.15**
AvgRank TILDA	4.55	5.10	**4.10**	6.25	7.40	5.55	5.30	6.40	8.45	8.25	8.65	8.00
AvgRank Overall	6.27	6.37	**5.45**	6.55	7.02	6.35	6.45	6.10	7.07	7.15	7.75	5.48
BASELINE+CTF DATASETS (ADDED FEATURES WITHOUT TIME-INDEX)												
Threshold	0	0.005	0.01	0.015	0.02	0.025	0.03	0.035	0.04	0.045	0.05	DD
AvgRank Elsanurse	6.75	5.65	6.25	5.45	6.60	6.05	7.30	7.30	8.00	6.80	7.20	**4.65**
AvgRank Elsacore	**5.60**	8.40	6.95	6.55	6.70	6.50	5.70	6.95	6.10	5.80	6.55	6.20
AvgRank TILDA	6.80	6.30	8.40	5.25	7.35	5.25	6.30	5.75	**4.60**	7.80	9.40	4.80
AvgRank Overall	6.38	6.78	7.20	5.75	6.88	5.93	6.43	6.67	6.23	6.80	7.72	**5.22**

results are very close (maximum 0.1 difference), which is not unexpected as we are comparing the rankings of two similar approaches over 30 datasets.

There were, however, cases where the ranks were very distinct, getting significant Wilcoxon p-values. All of these cases were in the first set of experiments, with Baseline datasets. Four of the significantly different results were in favour of Lexic (namely Sensitivity and Accuracy for Elsa-nurse datasets and Specificity and GMean for Elsa-core datasets), and two were in favour of NoLexic (Sensitivity and Accuracy for TILDA datasets).

Table 6.11: RF Lexic vs NoLexic comparison, summary of average rank results for Baseline and Baseline+CTF datasets.

BASELINE DATASETS (ORIGINAL FEATURES ONLY)								
Dataset	SENSITIVITY		SPECIFICITY		ACCURACY		GMEAN	
	Lexic	NoLexic	Lexic	NoLexic	Lexic	NoLexic	Lexic	NoLexic
AvgRank Elsanurse	**1.20**	1.80	1.55	**1.45**	**1.20**	1.80	**1.50**	**1.50**
AvgRank Elsacore	**1.30**	1.70	**1.20**	1.80	**1.25**	1.75	**1.10**	1.90
AvgRank TILDA	1.75	**1.25**	1.70	**1.30**	1.85	**1.15**	1.75	**1.25**
AvgRank Overall	**1.42**	1.58	**1.48**	1.52	**1.43**	1.57	**1.45**	1.55
BASELINE+CTF DATASETS (ADDED FEATURES WITHOUT TIME-INDEX)								
Dataset	SENSITIVITY		SPECIFICITY		ACCURACY		GMEAN	
	Lexic	NoLexic	Lexic	NoLexic	Lexic	NoLexic	Lexic	NoLexic
AvgRank Elsanurse	**1.25**	1.75	1.70	**1.30**	**1.30**	1.70	1.60	**1.40**
AvgRank Elsacore	1.70	**1.30**	1.60	**1.40**	1.70	**1.30**	**1.45**	1.55
AvgRank TILDA	**1.40**	1.60	**1.30**	1.70	**1.40**	1.60	**1.30**	1.70
AvgRank Overall	**1.45**	1.55	1.53	**1.47**	**1.47**	1.53	**1.45**	1.55

6.5 Interpreting the Best Random Forest and Decision Tree Classification Models

As our case-study classification problem of predicting age-related diseases has health applications, it is important to discuss what insight we can get from analysing our classification models. Thus, as an additional contribution, in this Section we interpret the best C4.5 decision trees (as a directly interpretable type of model) and report the most important features in the best RF models, as an "indirect" interpretation of RF models. The list of top-ranked features for the RF models not discussed in this Section are presented in Appendix E.

Decision tree models are very interpretable in general (Quinlan 1993; Freitas 2014), unless the tree is too large. In particular, since the top nodes (closer to the root node) are used to classify more instances than lower nodes, and the features selected to split the data in top nodes were chosen using more data, we can consider that features at those top nodes have more predictive power for labelling instances in the test set. In addition, we can create classification rules based on the split decisions at each node leading to a leaf node, and determine combinations of feature values that lead to a classification.

In the case of RF models, directly interpreting each random tree in the forest is not feasible, due to the large number of trees. However, we can calculate feature importance measures such as the average value of information gain ratio across

the nodes where the feature was selected (used in our RFs), which allows the user to see which features are considered most important for classification across all trees in the forest.

For selecting the best classification models to be interpreted, we considered the average GMean results from the models from the first scenario (Section 6.2), trained using the proposed lexicographic split approach. These models combine two of our main contributions (the data-driven MVR approach from Chapter 4 and the lexicographic split from this Chapter). We use GMean for this analysis because it is a global measure that summarises predictive accuracy in both classes, and is not as sensitive to class imbalance as the Accuracy measure (the other global measure of accuracy used in this thesis).

In this section, for each data source (ELSA-Nurse, ELSA-Core and TILDA), we report the interpretation of the best RF and the best decision tree model among the 10 datasets from that data source. These are the Diabetes models (classifiers) from the ELSA-nurse and TILDA datasets, and the Dementia models from the ELSA-core dataset. The GMean values of the chosen RF models were: 0.855 (ELSA-nurse Diabetes), 0.802 (ELSA-core Dementia), and 0.786 (TILDA Diabetes). For the C4.5 decision tree models, the GMean values were: 0.805 (ELSA-nurse Diabetes), 0.771 (ELSA-core Dementia), and 0.756 (TILDA Diabetes).

After choosing the best models based on their GMean values, we trained new models using the entire datasets (no training and test set division), to ensure the model-interpretability analysis would consider all data available. The full datasets were undersampled to a 1:1 ratio using the Balanced Random Forest approach, and had their missing values replaced using the data-driven MVR approach from Chapter 4. The C4.5 decision trees were trained with the default Weka parameter settings, including $C = 0.2$ (confidence factor used in pruning). For the RF models, we kept the default setting of $mtry$ but increased the number of trees from the default 100 to 1000, to have a better representation of the top features. All Figures and Tables in this Section correspond to these full-dataset models.

ELSA-nurse Diabetes models

Figure 6.1 shows a summarised version of the Diabetes decision tree classifier trained with the ELSA-nurse dataset, displaying the root node and the features in

the top 2 levels (as features higher in the tree are used to classify more instances). In this figure – and other similar figures in this Section – a node at the third level of the tree may be either a real leaf node (with the positive or negative class label) or a "summarised" node, which contains an ellipsis ('...') followed by an indication of a class (positive for no diagnosis, and negative for diagnosis of the target disease). These summarised nodes represent a subtree that was omitted from this representation, with multiple internal and leaf nodes, and the class label in these summarised nodes indicates the label assigned to the majority of instances from the undersampled training set that were in the path leading to that summarised node. Note that all summarised subtrees have real leaf nodes with both class labels, but we are only indicating the class of the majority of instances within that summarised subtree.

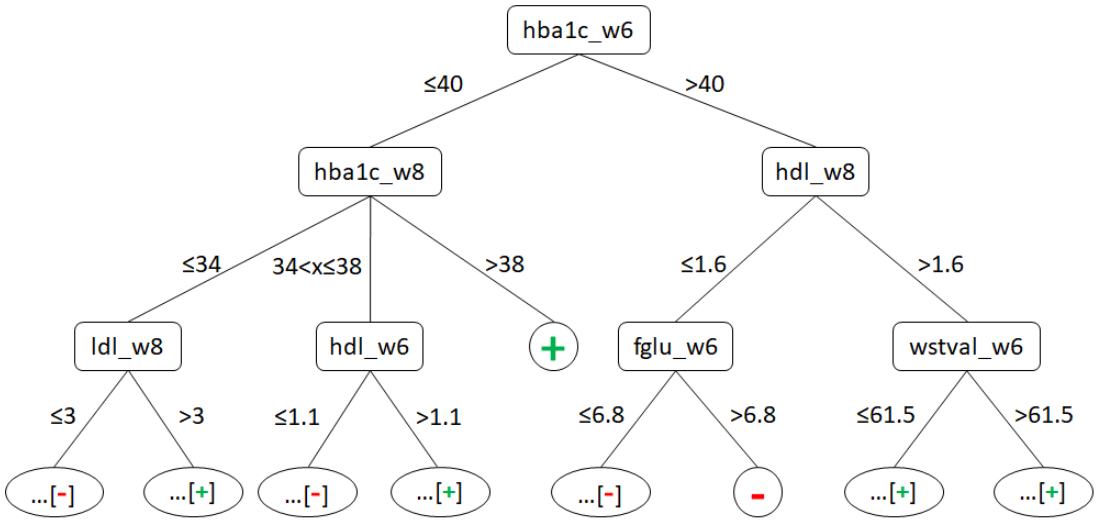

Figure 6.1: ELSA-nurse Diabetes C4.5 summarised decision tree model.

In the ELSA-nurse Diabetes tree, only four conceptual features were chosen in the top levels. The first is the *hba1c*, a blood glycated haemoglobin level (mmol/mol), with its two most recent measurements (waves 6 and 8) being used in two of the three highest nodes of the tree. This is not surprising, as the *hba1c* is a measurement of blood sugar, and it is used to diagnose diabetes, although the *hba1c* value by itself is likely not sufficient for an accurate diagnosis (Cavagnolli et al. 2011). The other feature in one of the three highest nodes of the decision tree was *hdl*, representing blood high-density lipoprotein level (mmol/l), and this feature is known to be correlated with type 2 diabetes (Farbstein and Levy 2012).

Two other top level features in the decision tree classifier were *ldl*, *fglu* and *wstval*. The first is the blood LDL cholesterol level (mmol/l), which is commonly connected to heart diseases instead of diabetes. The *fglu* feature is a measurement of blood glucose level while fasting (mmol/L), another measurement of blood sugar (notably a high level of fglu lead to a negative class label, meaning diabetes diagnosis). The third feature is *wstval*, which represents the waist measurement (cm) of the participant. The waist measurement is connected with obesity, and has been used as a predictor of diabetes previously (Janiszewski, Janssen and Ross 2007).

Table 6.12 shows the 10 best-ranked features for the Diabetes RF classifier trained with the ELSA-nurse dataset. The ranking is based on the average impurity decrease (AID, the arithmetic mean of information gain ratio), calculated over all nodes where the feature was selected, in all trees in the RF. This measure represents the predictive power associated with the feature in the trees.

Table 6.12: ELSA-nurse Diabetes RF model, 10 features with the greatest average impurity decrease (AID) values.

Feature	**Description**	**AID**
sex	Sex of the participant	0.6
indager_w8	Age at wave 8	0.51
cfib_w8	Blood Fibrinogen level (g/l)	0.49
chestin_w6	Whether had any respiratory infection in last 3 weeks	0.48
clotb_w8	Blood sample: whether has clotting disorder	0.48
hgb_w8	Blood haemoglobin level (g/dl)	0.46
diaval_w8	Mean diastolic blood pressure	0.46
igf1_w8	Blood insulin-like growth factor (IGF-1) level (nmol/l)	0.46
hscrp_w8	Blood C-reactive protein (CRP) level (mg/l)	0.46
mmgsnavg_w8	Mean grip strength with non-dominant hand	0.46

For the ELSA-nurse RF model, the age (indager_w8) and sex features were the highest ranked. Naturally, all age-related diseases are correlated with the age feature, and diabetes is more prevalent among men (Gale and Gillespie 2001). There are 5 blood sample features in this set of top-ranked features, namely *cfib*, *clotb*, *hgb*, *igf1* and *hscrp*, and one blood pressure feature, *diaval*. Although the blood glucose level is the simplest way to detect this disease, diabetes is also known to increase the chance of heart diseases, so it is possible the RF models detected patterns among ELSA respondents with heart or blood pressure problems. The

other two features in the list are *chestin*, related to respiratory infections, which may be correlated to type 1 diabetes (Lönnrot et al. 2017), and *mmgsnavg*, mean grip strength, a test used as a general health indicator, which has been connected to diabetes and hypertension (Mainous III et al. 2015).

Notably, most of the top-ranked features in the RF models are wave 8 measurements (the class wave in ELSA-nurse), with the exception of *chestin*. This is in part due to the use of our lexicographic split approach, which adds a bias in favour of more recent features.

ELSA-core Dementia models

Figure 6.2 shows a summarised version of the Dementia decision tree classifier trained with the ELSA-core dataset, displaying the features in its root node and top 2 levels.

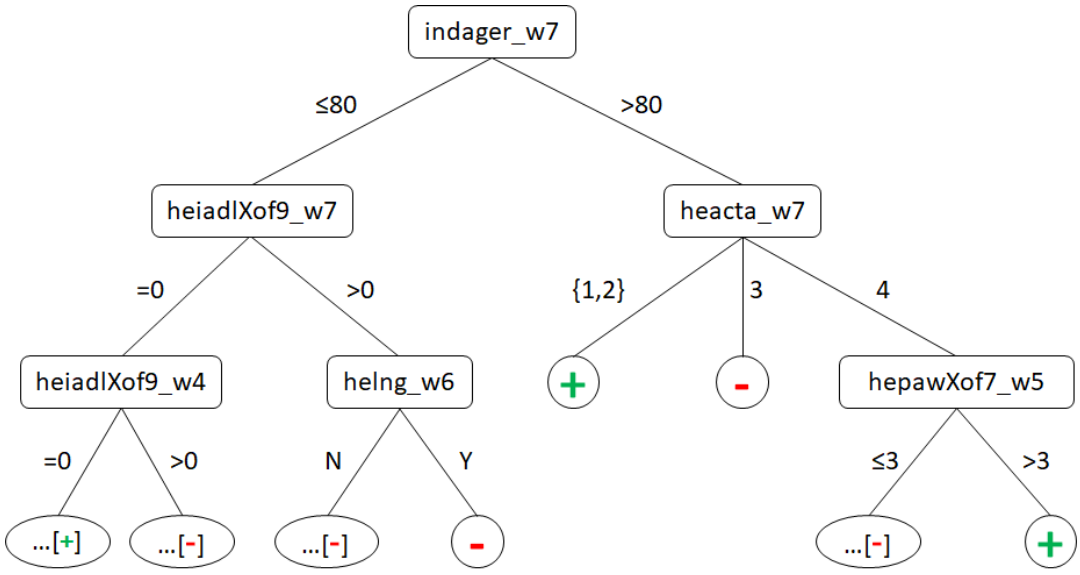

Figure 6.2: ELSA-core Dementia C4.5 summarised decision tree model.

The age feature is selected at the root node, followed by *heiadlXof9* and *heacta*, all measured at the last feature wave (wave 7 for ELSA-core). The *heiadlXof9* counts how many out of the 9 Instrumental Activities of Daily Living (IADL) in the questionnaire the respondent reported having difficulty with. The ADL and IADL measurements are related with disability, a defining feature of dementia

(Desai, Grossberg and Sheth 2004). The *heacta* feature measures frequency of vigorous sports and activities, and the responses corresponding to high frequency resulted in a positive class (no dementia diagnosis) prediction. The other two top level features are *helng*, a binary feature reporting whether the respondent takes medication for a lung condition (resulting in a negative class label if so), and *hepawXof7*, a count of body parts the respondent reported often felling pain in. Pain has been correlated with dementia in the sense that patients may report pain less often if they are suffering from cognitive decline, due to a difficulty in communicating or recognising pain (McAuliffe, Brown and Fetherstonhaugh 2012).

Table 6.13 shows the 10 best-ranked features for the Diabetes RF classifier trained with the ELSA-core dataset.

Table 6.13: ELSA-core Dementia RF model, 10 features with the greatest average impurity decrease (AID) values.

Feature	**Description**	**AID**
helng_w1	Whether taking medication for lung condition	0.82
hecanaa_w2	Organ or part of body which cancer started	0.66
hefrac_w6	Whether has fractured hip	0.64
sex	Sex of the participant	0.61
cesd_w7	Depression questionnaire score	0.61
indager_w7	Age at wave 7	0.61
headlXof6_w6	Reported ADL difficulties (count)	0.59
dicdnm_w7	Cause of death of mother of respondent	0.59
cfmetper_w7	Perception of memory compared to 2 years ago	0.59
hechm_w4	Cholesterol: whether taking cholesterol medication	0.58

The age and sex features are, again, among the top-ranked features in the RF classifier. Interestingly, there is no consensus regarding a direct effect of the sex of an individual on the likelihood of dementia (Ruitenberg et al. 2001), but it has been connected to other environmental factors such as loneliness, which is correlated to diagnosis (Zhou, Wang and Fang 2018). The feature with the highest AID was *helng*, and it was also an important feature in the decision tree model, used to confirm a negative class (dementia diagnosis) label. Two mental health features, *cesd* and *cfmetper*, were included in the set of top-ranked features. The latter is obviously associated with the class, and the former was also correlated

to an increased likelihood of dementia (Bennett and Thomas 2014). The other top-ranked features are about the medical history and current health status of the respondent, namely *helng*, *hecanaa*, *hefrac*, *dicdnm* and *hechm*.

TILDA Diabetes models

Figure 6.3 shows a summarised version of the Diabetes decision tree classifier trained with the TILDA dataset, displaying the features in the root node and top 2 levels. Although the class variable is the same as the one in the ELSA-nurse model, some of the features in the datasets are different, so it is interesting to see that both models performed well for predicting diabetes.

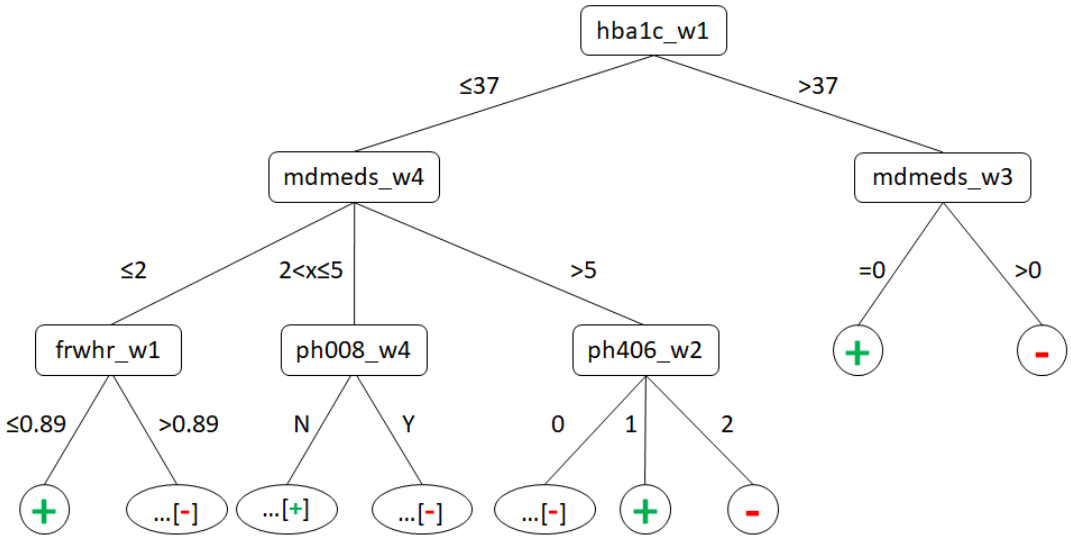

Figure 6.3: TILDA Diabetes C4.5 summarised decision tree model.

For the TILDA decision tree, we again see *hba1c* as the root node feature, unsurprisingly. The next nodes both split on the *mdmeds* feature, which counts how many medications the respondent reported taking currently, excluding nutrition supplements. In the case of respondents with *hba1c* above 37, taking no medication led to a positive class label (no diabetes diagnosis), and otherwise they were labelled as diabetic. The other three features in the top levels of the decision tree are *frwhr* (waist-to-hip ratio), *ph008* (whether the respondent has recently lost weight), and *ph406* (how many times the respondent has fainted in this last year). The former two are related to weight, which is connected to diabetes' symptoms

and diagnosis (Kim et al. 2018). Notably, a low waist-to-hip ratio led to a positive class label (no diabetes). We were not able to find a direct correlation between fainting and diabetes diagnosis in the literature, however. It is possible that this feature was associated with additional factors which are correlated to the class.

Table 6.14 shows the 10 best-ranked features for the diabetes RF classifier trained with the TILDA dataset.

Table 6.14: TILDA Diabetes RF model, 10 features with the greatest average impurity decrease (AID) values.

Feature	Description	AID
ph505_w3	Whether takes pain medication to control pain	0.45
indager_w4	Age at wave 4	0.43
bh107_w3	Hours spent sitting in a typical day	0.43
ph505_w1	Whether takes pain medication to control pain	0.43
behalc_freq_week_w1	Average amount of time respondent drinks a week	0.43
bphypertension_w3	Objective measured hypertension	0.42
ipaqmetminutes_w3	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	0.41
ph505_w2	Whether takes pain medication to control pain	0.41
ph601_w1	Did you lose any urine beyond your control in the last year	0.41
behcage_w1	Count of CAGE questionnaire responses (measures alcoholism)	0.41

The TILDA RF has age (indager_w4) among the top-ranked features, but not sex. The *ph505* binary feature was selected remarkably often, with three of its measurements appearing in the top-10. Frequent pain is more prevalent among older people with diabetes (Karjalainen et al. 2018), and it is associated with several comorbidities. The *bh107* and *ipaqmetminutes* features are measurements of physical fitness and activity, which are often correlated with weight and diabetes (Gill and Cooper 2008). The *ph601* is related to urinary incontinence, which was correlated to type 2 diabetes (Lifford et al. 2005). Finally, *behalc* and *behcage* are related to alcohol consumption, commonly associated with diabetes risk (DPPRG 2009).

6.6 Summary of the Results

In this Chapter, we proposed a novel adaptation to tree-based classifiers for longitudinal datasets. The proposed adaptation is a lexicographic bi-objective split approach, which uses time-related information available in longitudinal data. We

performed experiments comparing two versions of this adaptation to the standard classifiers, using our 30 real-world longitudinal classification datasets. The proposed lexicographic approach improved the predictive accuracy of RF classifiers, when compared to the standard split criterion based only on the information gain ratio, in the majority of the experiments.

The proposed approach can be summarised as follows. 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. More recent features are also less likely to have missing data due to attrition, so adding a bias in their favour reduces the chances of using missing or estimated data in the classification task.

The lexicographic approach led to the choice of a different feature in a substantial number of nodes (on average 25% over all 30 RFs in the Baseline datasets, and 14.7% over all 30 Baseline+CTF datasets), which shows that the added bias in favour of more recent features was a noticeable change to the RFs algorithm, for our longitudinal datasets.

In our experiments with Baseline datasets (without CTFs), we observed that the datasets with more waves benefited the most from our proposed approach, likely because of the greater time-gap covered in the dataset and the greater likelihood of the lexicographic approach causing a change on the selected split features. It is important to highlight that longitudinal datasets tend to grow over time, so the impact of the lexicographic approach also tends to increase as new waves are added to longitudinal datasets.

Regarding the experiments with datasets with added CTFs, we saw a reduced increase in predictive performance overall. This was expected, as the lexicographic approach has a diminished impact in these datasets because there are fewer opportunities to cause changes in the classification model. Even though the lexicographic approach still obtained slightly better results, meaning the two approaches (creating CTFs and applying the lexicographic split) can be applied together, our main recommendation from these results is applying the Lexicographic approach on its own. This follows from our results in the previous Chapter, where we concluded that the CTFs approach needs to be further explored before being considered fully matured.

Finally, we interpreted the best models as an additional contribution. We were

able to find several existing connections between the high-ranking features in RF models and decision tree models and peer-reviewed medical research.

Chapter 7

Conclusions

Developing supervised ML methods for longitudinal data is an under-explored area with many opportunities for research. The temporal nature of the data allows for different analysis techniques, considering the changes that the values of each conceptual feature undergo throughout the study's waves. It also presents new challenges, such as increased dimensionality and different forms of dependency between features, i.e., not only dependency between different features in the same wave, but also dependency between different values of a conceptual feature across waves.

In this thesis, we focused on proposing new supervised ML approaches for longitudinal data and applying them to longitudinal datasets of human ageing studies, due to the rising importance of this type of study, considering that the proportion of old people in the world population is increasing and old age is the greatest risk factor for a number of diseases.

Overall, we succeeded in exploring different strategies for handling longitudinal data in supervised ML. Our contributions have considerable limitations, and may require further experiments to be optimised to their full potential, but throughout all our experiments we always found evidence that considering temporal information had a positive impact.

In this Chapter we start by summarising our contributions and results. Then, we propose extensions for this work, and new research directions following the results of this thesis.

7.1 Summary of Contributions

7.1.1 A taxonomy of longitudinal dataset representations

We reviewed studies that have applied supervised ML techniques to longitudinal datasets from human ageing studies, highlighting the techniques and data representation strategies each study employed. In our review we observed that most studies adapted their data preparation process, whereas few studies proposed algorithm adaptations that made the ML algorithm consider the temporal information in the data.

The data preparation approaches have the advantage of being simpler and less dependent on the algorithm choice. However, algorithm adaptations are more robust, requiring less human-made decisions and data-specific preprocessing. Naturally, both types of approach can be used in combination – in addition to changes in the preparation process, researchers might also employ adapted algorithms for longitudinal analysis.

We believe there is a gap in the literature for classification algorithms adapted to cope with longitudinal data, handling the specific characteristics that this type of dataset brings to the knowledge discovery task. In our review, we noted that most of the works in the literature that proposed algorithm adaptations or new algorithms for longitudinal analysis focus on regression algorithms, rather than classification ones. Notably, there are several decision support tasks in health and ageing research that are framed as classification problems, such as diagnosis prediction, identifying risk groups, or predicting needs for specialised care. Thus, we highlight the need to propose new or adapt existing classification algorithms to learn from longitudinal datasets, which are common in health and ageing research.

As a contribution derived from our review, we introduced a new taxonomy that characterises four ways to represent longitudinal data for supervised ML problems, and divides the adopted approaches into data preparation or algorithm adaptation approaches. The taxonomy can help future researchers in their own literature reviews, facilitating the selection of relevant works and identification of gaps in the literature, such as the aforementioned lack of classifiers adapted for longitudinal data input. For our research, we use the UKLI (union keeping longitudinal information representation) and AGG (aggregated data – when we create constructed features) approaches for representing the datasets, and propose both

a data preparation and an algorithm adaptation for our longitudinal classification problem.

7.1.2 Data-driven missing value replacement approach

Longitudinal datasets often have a high ratio of missing values due to attrition. Thus, the choice of missing value replacement (MVR) strategy is particularly important in ML applications with longitudinal datasets. In our case the overall percentage of missing data was 38.5%, 19.1% and 9.5% for the ELSA-nurse, ELSA-core and TILDA datasets, respectively. This varies considerably between different features in the same dataset, and there were several cases where over half of the values are missing. Thus, our first specific objective was to propose a MVR approach for longitudinal data inputs.

Different replacement methods can achieve more accurate estimations depending on the situation, as the performance of the MVR method depends on several aspects, including whether the data is longitudinal or not. Therefore, the choice of MVR method is not a trivial one. As a contribution of this thesis, we developed a new approach to address this issue.

The proposed approach is a data-driven MVR method created with a rationale that the known feature values are a good source of information to determine how accurate a method's estimations are. Strictly speaking, this holds true for data that is missing completely at random, which is often not the case for real-world data collected through interviews and exams, such as our datasets. However, we believe that the available known data can still be used as a "heuristic" approach for estimating missing values in our datasets, which has been empirically confirmed by the good results obtained by the proposed data-driven MVR approach in our experiments.

In our implementation of the proposed data-driven MVR approach, we selected a set of five MVR methods. These are: the global mean/mode; the age-based mean/mode; a longitudinal implementation of the k-nearest-neighbours algorithm; the previous observation from the same instance; the mean/mode of both the previous and next observations of the same instance. This set of MVR methods was chosen because it represents different strategies, namely traditional statistics,

machine learning, and methods devised for longitudinal data. Notably, the data-driven approach can be used with any set of MVR methods, limited only by the time available for preprocessing the data and by the method's applicability to the dataset.

The proposed approach performs a feature-wise ranking of the MVR methods, based on their estimation error for known values of each feature, performing the following steps for each MVR method. First, the instances with missing values for the current target feature are removed from the dataset. Then, the resulting subset is divided into an estimation set and a validation set, the former being used to calculate the estimated values for the hidden values in the latter, for the current feature. Finally, the estimated values in the validation sets are compared to the previously hidden ground truth, and an error value is calculated (average absolute difference for numerical features or mismatch ratio for nominal features). Thus, the method calculates an estimation error value for each MVR method, for each feature. The methods are then ranked feature-wise and applied starting from the best-ranked one, until all missing values in the feature are replaced or there are no more possible methods to apply.

We ran two sets of experiments to compare our proposed data-driven MVR approach to the alternatives. The first was a classifier-independent comparison of each MVR method and the proposed data-driven approach, which calculated the average estimation error and applicability (percentage of missing values replaced) of each method. This analysis showed that the methods devised specifically for longitudinal data had low estimation errors, but they were not applicable in many of the cases. Overall, the data-driven approach outperformed the individual methods, considering both error rate and applicability, with KNN performing the best among individual methods.

The second set of experiments was a classifier-dependent comparison of the predictive accuracy of models generated with each strategy to handle missing data, including a baseline of not performing any replacement. For this, we trained RF classifiers with each individual method, the data-driven approach and the baseline, adopting an undersampling strategy that was successful in preliminary experiments comparing two undersampling strategies for RFs. For this comparison, the proposed data-driven approach outperformed the other methods in most cases.

In summary, the proposed data-driven MVR approach is recommendable for any dataset, longitudinal or not, where the known data can be used to determine how well a MVR method is able to calculate its estimations. It can be expanded with more MVR methods, and used for datasets with any number of features or volume of missing data. The lower-bound applicability of the approach approach is the upper-bound of the most applicable method in the chosen set of MVR methods, which is usually 100% as many simple methods, such as the global mean/mode, are applicable in any case. An important limitation is that, although the data-driven MVR approach can be used for any set of MVR methods and any dataset, it is the most computationally expensive of the contributions in this thesis, as it combines the run times of all selected MVR methods and requires them to be run multiple times, in the feature-wise comparisons.

7.1.3 Constructed temporal features for longitudinal datasets

The second specific objective set in our research regarded modifying the dataset, before training the supervised ML model, to include temporal information that will be considered by the algorithm during training. For this, we proposed a data preparation approach of creating constructed temporal features (CTFs), using the conceptual features in the longitudinal datasets, and adding those to the original feature set in a preprocessing step.

We experimented with adding six different types of CTFs. Three of them had been used before (Pomsuwan and Freitas 2017): the Monotonicity, which represents the presence and direction of a monotonic change in the feature values over time; the Diff, which is the result of a simple subtraction between the two last measurements of a feature, representing actual difference or degree of difference, for numeric and nominal features respectively; and the Ratio (there was no specific mention of the Ratio CTF in previous work, but it is conceptually very similar to Diff), applicable only to numeric features, which divides the last measurement of a feature by the penultimate, and also represents recent change in values.

The other three types of CTFs are original contributions of this research: the DiffAgeMean is a difference between the last measurement of a feature and the mean/mode over all measurements taken from instances with the same age of the

current individual (which we consider a viable expected value); the AvgDiffAgeMean expands the former to include the average differences of each measurement and the expected values for the ages at time of measurement; the Percentile ranks the feature values of all individuals of the same age, and places the current value in that rank, representing how much an individual's measurement was low or high compared to similar individuals.

We ran experiments with each type of CTF individually, then with all 6 combined, comparing three feature sets in two scenarios. We reported the RF experiments in the main text, and C4.5 decision tree experiments in Appendix C.

The first experimental scenario excluded, from all feature sets, the original features that were ineligible for CTF creation (features that were not measured multiple times), as a way to make more a controlled experiment comparing the impact of the CTFs in predictive accuracy. The second scenario includes ineligible features in all feature sets. The three feature sets compared in each experiment are: (a) a baseline set with the original features used for creating the type of CTF being tested (e.g., only the last two measurements in the Diff test); (b) a CTF set with only the constructed features, without the original features used for creating them; and (c) a combination of the two previous sets, with both original and constructed features (the proposed approach).

For the RF experiments, four of the CTFs did not increase predictive accuracy in general on their own, namely the Diff, Ratio and Monotonicity and AvgDiffAgeMean; while the Diff and Percentile did increase the predictive accuracy in most cases. The main experiments, with all six types of CTFs combined, had good RF results for the proposed approach of adding CTFs to the original dataset. For DT experiments, the results tended in favour of the Baseline approach of not adding constructed features in a preprocessing step.

In addition to the predictive accuracy of the models, we also reported the feature importance analysis results, to investigate whether the CTFs were having a significant impact in the models themselves. We noted that the frequency of CTFs selected among the top 10 features in a RF (considering the average feature importance over all trees in the forest) seems related to how much temporal information is available in the dataset. To be precise: the TILDA dataset had 12.6% CTFs among the top features, and it is the dataset with the least temporal information; the ELSA-nurse had 18.8% CTFs among the top features; the

ELSA-core had 87.4%, and it has the most temporal information (up to 7 measurements of each feature). Interestingly, although its individual results were not good, the Monotonicity was among the CTFs appearing the most often among the top-ranked features, possibly due to a bias in favour of categorical features in the tree split function (all other CTFs have continuous numeric values).

Although the predictive accuracy results were not very consistent in our experiments with CTFs, we believe that this data preparation approach is a promising, computationally inexpensive way to modify longitudinal datasets in a preprocessing step and increase the predictive accuracy of supervised ML models. We believe that the temporal patterns in longitudinal data can become important predictors, particularly for datasets that have more temporal data available, as we have seen that those benefit the most from the CTFs. The main limitation of this approach is the significant increase in dimensionality caused by the added features, which can hinder the performance of some classification algorithms. One important advantage is that, as a data preparation approach, it does not depend on the ML algorithm, so it is applicable even for regression or unsupervised ML problems. Note, however, that evaluating the CTFs on the latter types of tasks is out of the scope of this thesis.

7.1.4 Lexicographic split for tree-based classifiers

The third specific objective posed for this research was about adapting standard supervised ML algorithms. As the model's interpretability is particularly important in health applications such as diagnosis prediction, we chose to focus on tree-based classifiers. Thus, we proposed an adaptation to the split function of tree-based classifiers, which select which feature will be used for splitting the data at each node.

The proposed algorithm is based on a new lexicographic bi-objective split function which, as a secondary objective for feature selection, prioritises more recent features. The feature's time-index as secondary objective is used as a tie-breaking criterion, when candidate features being compared have approximately the same predictive power, i.e., their information gain ratio differences fall within a predefined threshold. In order to avoid having an added user-specified parameter, we also proposed a data-driven automated threshold selection, which uses an

internal cross-validation process to choose an adequate threshold value based on the training data.

We tested the proposed approach using mainly the RF classifier, as it tends to generate high-quality models, compared to other state-of-the-art classifiers, while also maintaining some partial interpretability, mainly indirectly through feature importance metrics. Additional experiments with the C4.5 decision tree algorithm were added to Appendix D.

We ran two sets of experiments, in two scenarios. The first set of experiments compared the automated threshold selection to fixed threshold values in the same range of the values tested internally by the automated selection, and the second set of experiments compared using the proposed lexicographic approach (with automated threshold selection) to using the unchanged algorithm. In the first scenario, we used only the original features in our datasets, most of which have time indexes. In a second scenario, we used datasets with added CTFs, which do not have time indexes, in an adapted version of the lexicographic approach that ignores CTFs in its internal logic.

The results of these experiments showed that the lexicographic approach slightly improved the predictive performance in most cases, especially for datasets that had more temporal data (i.e., more consecutive measurements of the same features). This pattern of better performance when more temporal data is available corroborates the rationale for this adaptation, that data measured closer to the target wave is more relevant for classification. Note, however, that when we combined both the data preparation approach of adding CTFs and the algorithm adaptation approach of the lexicographic split, the latter had worse results compared to the first scenario of using only the original features. As CTFs do not have inherent time indexes, they are ignored by the lexicographic split, reducing its impact on the generated models.

Overall, the proposed adaptation increased the predictive performance for the majority of the RF classifiers. It can be implemented for any tree-based classifier, with a small increase to computational cost (as the standard split function already compares the information gain ratios of the candidate features by default), with the caveat of an added parameter. The proposed automated threshold selection is our recommended approach for selecting this parameter's value, and although

it does require an internal cross-validation using the training data, this is not prohibitive for tree-based classifiers, that tend to have short training times compared to, for example, support vector machines and neural network classifiers.

7.1.5 Evaluation of the proposed approaches in longitudinal datasets of human ageing

Our final contribution was applying and evaluating our proposed approaches using real-world datasets, and comparing them to baseline models using 4 predictive accuracy measures. For this, we created 30 longitudinal classification datasets using data from the nurse-data and core-data questionnaires from the ELSA (UK study) and TILDA (Irish study) databases. The longitudinal datasets have 7097/141/4, 8405/172/7 and 5715/81/4 instances/features/waves for ELSA-nurse, ELSA-core and TILDA sources, respectively (the feature sets include repeated measures of conceptual features, see Appendix A).

All datasets produced in this work include data from up until 2018 (wave 8 from the ELSA study and wave 4 from the TILDA study). TILDA has concluded its fifth wave in 2018, but the data for it is still not available for research, and they are currently collecting data for wave 6. ELSA has published its 9th wave in 2020, which so far only includes the core questionnaire data (a special nurse-data questionnaire was conducted on wave 9, but its data is currently not available), so the ELSA-core datasets can be updated using the most recent data. As both the ELSA and TILDA studies are ongoing, our datasets can be incrementally updated as new waves are released, following the chosen conceptual features and updating the target variables to reflect the new last wave of the dataset. Notably, as more temporal information is aggregated into the dataset, the techniques proposed in this thesis become more relevant and have a bigger impact on the results.

To create these datasets, we did a manual feature selection for each data source, starting from the full database with thousands of features and discarding those that were unrelated to the target variables, or unusable for ML, and performing data transformations such as merging similar features together to reduce the dimensionality of the final datasets. The chosen predictive features represent mainly biomedical information collected by health professionals (ELSA-nurse and TILDA) and self-reported mental and physical health data answered by the study

participants (ELSA-core), measured repeatedly over the waves of the source studies. For each source, we have 10 binary class variables (thus, 10 datasets for each source) that represent the diagnosis, on the last wave of the dataset, of an age-related disease. Importantly, the datasets also have different amounts of temporal information represented, as their conceptual features can be measured over fewer or more waves (from 2 to 7 measurements).

For every main contribution in this work, we ran experiments using these datasets and evaluated the performance of the models created using our proposed approaches. The models were compared in terms of Sensitivity (true positive rate), Specificity (true negative rate), Accuracy and GMean, with focus on the two latter measures as they are global measures of performance. We also ran non-parametric statistical tests to compare the results of different approaches, to determine whether the changes in predictive accuracy were significant.

In the human ageing study context, interpreting the classification models generated from these datasets can bring new insights regarding how the predictive features relate to the target variables. Therefore, as an additional contribution in the field of human ageing research, we analysed the best RF and decision tree classification models in each data source. In this analysis, we highlighted the most important features (top nodes of a decision tree and top-ranked features in the RF) in the models, and referenced existing works in the literature that link these features to the target classes. As mentioned earlier, being able to observe how the labelling of an unseen instance is done (decision trees) or what features are commonly selected and have a significant impact in the labelling (Random Forests) is an important advantage of more interpretable classifiers that is particularly relevant for health applications. In addition to corroborating previous results in the medicine literature, our analysis may motivate new research to investigate connections between different predictive features, such as the medications a patient is currently taking, associated with weight loss, being used as a predictor of diabetes (see the TILDA decision tree model, Figure 6.3).

7.2 Future Work

The contributions in this thesis can be improved and further explored in future works, increasing their positive impact on predictive accuracy. In this Section we

highlight some ideas for further research.

7.2.1 Extensions to the contributions

As mentioned, our contributions can all be used together in any longitudinal classification project, and we hope to expand this framework to include new data preparation and algorithm adaptation approaches. We can also expand the existing proposals to make this framework more robust and applicable to datasets from different problems.

Extending the data-driven MVR approach

The data-driven MVR approach is the most widely applicable of our contributions, as it can be used for any dataset with missing data. To improve this approach, we would like to perform new experiments using more MVR methods. One possible extension is to use our classifier-independent comparison method to recommend a set of MVR methods for a given input dataset, considering the time required to run the approach in the feature-wise ranking. In the case of longitudinal datasets, we will also include more methods devised specifically for longitudinal data, as the three of these methods in our work (Prev, PrevNext, and our implementation of KNN) had low estimated errors.

Extending the constructed temporal features

The CTF addition proposal can be expanded by testing new CTFs, and by including a data-driven approach for automatically selecting a set of CTFs that work well with a given longitudinal dataset input, considering the dimensionality increase trade-off. To be specific, we are most interested in CTFs that reflect the evolution of a feature's value over consecutive measurements, possibly with a numerical CTF similar to Monotonicity that reflects change over time considering all waves. We believe such temporal patterns can be interesting predictive features and might be used in tandem with the original biomedical features in the dataset, in decision support applications.

Extending the adaptations to tree-based classifiers

Finally, we can propose further algorithm adaptations to tree-based classifiers for coping with longitudinal data. As our results combining the CTF addition approach and the lexicographic split were not as good as we hoped, it would be interesting to propose adaptations that try to combine the strengths of both contributions. The current lexicographic split could be changed to, instead of ignoring CTFs as they don't have time-indexes, incorporate a ranking system that can be used for both original and constructed features. This would require experimenting with different ranking strategies that consider the temporal information in the CTFs, and how it compares to the time-indexes of the original features.

Another idea we are interested in exploring is creating specialised nodes (or specialised trees within the RF) that deal differently with original and constructed features. For example, some of the nodes could select only constructed features as their candidates. This is an interesting way we could promote diversity within a RF, as trees with different ratios of specialised nodes would use different feature sets. However, this approach would require careful testing to ensure that the artificially reduced randomness does not negatively impact the RF classifier.

7.2.2 Experiments with other datasets

There are several populational studies of ageing available for research, and our framework (including the data preparation process) can be replicated to their longitudinal data. Importantly, as some of the ageing studies being conducted in other countries have compatible frameworks to the ELSA, we could create datasets combining data from different populations, and also perform cohort effect studies to create more generalised models.

Some of the longitudinal studies of ageing we are interested in are: The Survey of Health Ageing and Retirement in Europe (SHARE), which has 8 published waves and includes data from over 20 countries (Börsch-Supan et al. 2013); the Chinese Longitudinal Healthy Longevity Survey (CLHLS) which has 8 published waves and has data on many 90+ year old participants (Gu et al. 2020); the Wisconsin Longitudinal Study (WLS), which has data collected since 1957, and could provide insights on the passage of great periods of time (Sewell et al. 2003).

In addition to creating entirely new datasets, we can also test with different

classification problems using our current datasets. For this, we could use other predictive features in ELSA and TILDA and create new class variables, such as health status variables or risk level classifications.

7.2.3 Experiments with other techniques

Decision tree classifiers output a defined classification for each input instance, but they are also able to output class probabilities using the information in each leaf node. That is, if a leaf node has, for example, 8 out of 10 training instances in class 0, it will classify any input instance that lands on that leaf node as class 0, but it could instead output the 80% probability attached to that classification. This information can be used in tandem with classification accuracy to evaluate the performance of a decision tree classifier, possibly leading to a more robust classifier. Several techniques were proposed taking class probabilities into account (Jiang, Li and Cai 2009; Liu et al. 2010), and in general they tend to outperform standard DTs for imbalance datasets.

As future work, we would like to perform experiments with this type of decision tree in the future, as they might perform better for our data, and the adaptations proposed in this work can still be applied to them. In addition to probability-based decision trees, we would also like to experiment with decision trees that use skew-insensitive split criteria, which were designed for imbalanced data (Cieslak et al. 2012; Mulyar and Krawczyk 2018).

7.2.4 Deeper analysis of the ML results

As we focused in our contributions to the supervised ML process in this thesis, we only briefly analysed the best models from each data source, in Section 6.5. We would like to invest more time analysing all classification models and their contribution to ageing research. The RF and decision tree classifiers could go through parameter optimisation processes, to generate more accurate models that we could analyse, possibly with help from domain experts (health professionals).

We do not claim that our models are adequate decision support applications for diagnosis prediction, as this would likely require more specialised data and fully optimised classifiers. However, discovering features that have a connection to the target variables could prompt further research into the dynamics of ageing

and the development of age-related diseases. It would be especially interesting to investigate the constructed features as predictors, as if the temporal patterns represented in them are connected to accurate predictions, this could lead to new data collection methodologies in the studies themselves.

7.2.5 Multi-label classification

Traditional classification problems assume that every instance can be classified as a single label from a predefined set of labels. Problems where that assumption is not met because an instance can be attributed to more than one labels at the same time are named multi-label classification problems (Tsoumakas, Katakis and Vlahavas 2009; Al-Otaibi, Flach and Kull 2014). Naturally, simultaneously considering whether an instance belongs or not to each of the class labels considerably changes the way the classification is approached.

There are two main strategies to address a multi-label classification problem: (a) transforming the problem into a set of single-label classification problems, and applying standard classification algorithms to those, merging the results afterwards (i.e., not performing multi-label classification), or (b) adapting classification algorithms to make them predict a set of labels for each instance (Tsoumakas, Katakis and Vlahavas 2009). The second strategy has the advantage of taking into account the correlations between labels, but as the number of labels grows, considering all correlations between labels becomes exponentially more difficult.

The age-related disease prediction problem can be considered a multi-label classification problem, as different diseases may be connected. Our strategy so far has been to treat each classification problem separately, removing all class labels except one for each dataset. However, it would be interesting to investigate multi-label longitudinal classifiers in our age-related prediction context, as it could lead to discoveries about how the target diseases influence one another. The datasets we prepared from each data source share the same predictive features, so we already have three multi-label datasets we could use for this possible extension.

Appendix A

Dataset Feature Descriptions

Tables A.1, A.2 and A.3 show the conceptual features selected for the longitudinal datasets used in our experiments. For each feature we state in which waves of the study it was measured, and its data type. Note that, for the age variable (indager), we only used the most recent measurement as a predictive feature when training the models, but some methods such as the age-based mean/mode missing value replacement used the indager values for other waves.

Table A.1: Description of the selected features for the ELSA-nurse datasets.

Feature name	**Description**	**W2**	**W4**	**W6**	**W8**	**Type**
indager	Age of the participant at a given wave				X	Numeric
sex	Sex of the participant (male/female)	Not applicable				Binary
sysval	Mean systolic blood pressure	X	X	X	X	Numeric
diaval	Mean diastolic blood pressure	X	X	X	X	Numeric
pulval	Pulse pressure	X	X	X	X	Numeric
mapval	Mean arterial pressure	X	X	X	X	Numeric
mmgsd_avg	Mean grip strenght with dominant hand	X	X	X	X	Numeric
mmgsn_avg	Mean grip strenght with non-dominant hand	X	X	X	X	Numeric
clotb	Blood sample: whether has clotting disorder	X	X	X	X	Binary
cfib	Blood Fibrinogen level (g/l)	X	X	X	X	Numeric
chol	Blood total cholesterol level (mmol/l)	X	X	X	X	Numeric
hdl	Blood High-density lipoprotein (HDL) level (mmol/l)	X	X	X	X	Numeric
trig	Blood triglyceride level (mmol/l)	X	X	X	X	Numeric

Table A.1 continued from previous page

Feature name	Description	W2	W4	W6	W8	Type
ldl	Blood LDL cholesterol level (mmol/l)	X	X	X	X	Numeric
fglu	Blood glucose level while fasting (mmol/L)	X	X	X	X	Numeric
rtin	Blood ferritin level (ng/ml)	X	X	X	X	Numeric
hscrp	Blood C-reactive protein (CRP) level (mg/l)	X	X	X	X	Numeric
hgb	Blood haemoglobin level (g/dl)	X	X	X	X	Numeric
hba1c	Blood glycated haemoglobin level (mmol/mol)	X	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 WHO definitions	X	X	X		Ordered Nominal (4)
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		Ordered Nominal (3)
mmstre	Outcome of semi-tandem stand	X	X	X		Ordered Nominal (3)

Table A.1 continued from previous page

Feature name	Description	W2	W4	W6	W8	Type
mmftre2	Outcome of full tandem stand according to age	X	X	X		Ordered Nominal (5)
mmlore	Leg raise (eyes open): Outcome	X	X	X		Ordered Nominal (3)
mmlsre	Leg raise (eyes shut): Outcome	X	X	X		Ordered Nominal (3)
mmcrre	Single chair rise outcome	X	X	X		Ordered Nominal (3)
mmrroc	Outcome of multiple chair rises, split by age	X	X	X		Ordered Nominal (5)
igf1	Blood insulin-like growth factor (IGF-1) level (nmol/l)		X	X	X	Numeric
wbc	White blood cell count (x 10^9 cells/litre)		X	X	X	Numeric
mch	Blood mean corpuscular haemoglobin level (pg/cell)		X	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	X	Numeric

Table A.2: Description of the selected features for the ELSA-core datasets.

Feature name	Description	W1	W2	W3	W4	W5	W6	W7	Type
indager	Age of the participant at a given wave							X	Numeric
sex	Sex of the participant (male/female)	Not applicable							Binary
cesd	Depression questionnaire score	X	X	X	X	X	X	X	Numeric
cfmetm	Self-rated memory	X	X	X	X			X	Ordered Nominal (5)
cfmetper	Perception of memory compared to 2 years ago		X	X	X			X	Ordered Nominal (5)
dicdnf	Cause of death of father of respondent	Not applicable							Unordered Nominal (7)
dicdnm	Cause of death of mother of respondent	Not applicable							Unordered Nominal (7)
heacta	Frequency does vigorous sports or activities	X	X	X	X	X	X	X	Ordered Nominal (4)
heactb	Frequency does moderate sports or activities	X	X	X	X	X	X	X	Ordered Nominal (4)
heactc	Frequency does mild sports or activities	X	X	X	X	X	X	X	Ordered Nominal (4)
headlno	Reported difficulty with ADL or IADL			X	X	X	X	X	Binary
headlxof6	Reported ADL difficulties (count)			X	X	X	X	X	Numeric
heam	Whether taking medication for asthma	X	X	X	X	X	X	X	Binary
hecanaa	Organ or part of body which cancer started	X	X	X	X	X	X	X	Ordered Nominal (9)
hecanb	Cancer: whether received treatment in last 2 years	X	X	X	X	X	X	X	Binary
hechm	Cholesterol: whether taking cholesterol medication			X	X	X	X	X	Binary
hefrac	Whether has fractured hip	X	X	X	X	X	X	X	Binary
hehelf	Self-reported general health	X	X		X	X	X	X	Ordered Nominal (5)
heiadlxof9	Reported IADL difficulties (count)			X	X	X	X	X	Numeric
heill	Whether has self-reported long-standing illness	X	X	X	X	X	X	X	Binary
heji	Whether had joint replacement	X	X	X	X	X	X	X	Binary
helng	Whether taking medication for lung condition	X	X	X	X	X	X	X	Binary
hemobno	Reported difficulties with mobility			X	X	X	X	X	Binary
hemobxof10	Reported mobility issues (count)			X	X	X	X	X	Numeric
hepaa	Severity of pain most of the time	X	X	X	X	X	X	X	Ordered Nominal (4)
hepain	Whether often troubled with pain	X	X	X	X	X	X	X	Binary
hepawxof7	Pain reported (count)			X	X	X	X	X	Numeric
hepsyxof9	Psychiatric problems reported (count)			X	X	X	X	X	Numeric
heyrc	Experienced psychiatric problems in last 2 years	X	X	X	X	X	X	X	Binary
memtotb	Index of memory function (0-29)	X							Numeric
scako	How often had alcoholic drinks in last 12 months		X	X	X	X	X	X	Ordered Nominal (8)
smokerstat	Smoker status (past or present)	X	X	X	X	X	X	X	Ordered Nominal (5)

Table A.3: Description of the selected features for the TILDA datasets.

Feature name	Description	W1	W2	W3	W4	Type
indager	Age of the participant at a given wave				X	Numeric
sex	Sex of the participant (male/female)	Not Applicable				Binary
behalc_freq_week	Average amount of time respondent drinks a week	X				Numeric
bh004	For how many years have you smoked altogether	X				Numeric
hba1c_w1	Blood glycated haemoglobin level (mmol/mol)	X				Numeric
ph008	Have you lost at least 4.5kg without trying in the past year	X	X	X	X	Binary
ph402	How many times have you fallen in this last year	X	X	X	X	Ordered Nominal (3)
ph406	How many times have you fainted in this last year	X	X	X	X	Binary
ph415	Had any joint replacements	X	X	X	X	Binary
ph505	Takes pain medication to control pain	X	X	X	X	Binary
ph601	Did you lose any urine beyond your control in the last year	X	X	X	X	Binary
mdmeds_excl_supps	Number of medications reported by respondent (excluding supplements)	X	X	X	X	Numeric
disimpairments	Physical impairments count (activities the respondent can't do)	X	X	X	X	Ordered Nominal (4)
disadl	Count of ADLs the respondent reported difficulty with (top coded at 5)	X	X	X	X	Ordered Nominal (3)
disiadl	Count of IADLs the respondent reported difficulty with (top coded at 5)	X	X	X	X	Ordered Nominal (3)
behcage	Score of CAGE questionnaire responses (measures alcoholism)	X	X	X	X	Numeric
ipaqmetminutes	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	X	X	X	X	Numeric
bh107	Hours spent sitting in a typical day	X		X	X	Numeric
bh202	How often do you have trouble sleeping	X	X			Ordered Nominal (3)
frbmi	Body-mass index	X		X		Numeric
frwhr	Waist-to-Hip ratio	X		X		Numeric
frgripstrengthd	Mean grip strength for dominant hand (kg)	X		X		Numeric
frgripstrengthnd	Mean grip strength for non-dominant hand (kg)	X		X		Numeric
frtugtimesec	Timed "Up and Go" mobility test where respondent needs to get up from a chair	X		X		Numeric
bpseatedsystolicmean	Mean seated systolic blood pressure (mm Hg)	X		X		Numeric
bpseateddiastolicmean	Mean seated diastolic blood pressure (mm Hg)	X		X		Numeric
bphypertension	Objective measured hypertension	X		X		Binary
bloods_chol	Cholesterol (mmol/l -millimoles per litre)	X		X		Numeric
bloods_hdl	HDL (mmol/l -millimoles per litre)	X		X		Numeric
bloods_ldl	LDL (mmol/l -millimoles per litre)	X		X		Numeric
bloods_trig	Triglycerides (mmol/l -millimoles per litre)	X		X		Numeric

Appendix B

Detailed Random Forests results for the individual CTF experiments

In Tables B.1 to B.24, we report on the individual CTF experiments results using the RF classifier. For each CTF, there are four tables, reporting Sensitivity and Specificity results, then Accuracy and GMean results, for scenarios 1 and 2, in this order. The setup is exactly the same for these experiments, with two scenarios (i.e., with and without ineligible features in the dataset) comparing three feature sets, namely a Baseline set with only the features used to create the CTFs, a CTFs-only set without original features, and a BL+CTFs set combining both original and constructed features.

B.1 Diff Results for Random Forests

The Diff CTF is a representation of the most recent change in the value of a longitudinal feature, either as a real difference value (for numeric features) or a degree of difference (for ordered nominal features). Using the sign of the Diff value, one can interpret the changes in the value of the feature from the penultimate wave to the last wave as possible temporal trends in the Diff values (upwards or downwards). The results for Scenario 1 are shown in Tables B.1 (Sensitivity and Specificity results) and B.2 (Accuracy and GMean results). The results for

Scenario 2 are shown in Tables B.3 (Sensitivity and Specificity results) and B.4 (Accuracy and GMean results).

For the Scenario 1 experiments, where ineligible features were not used, the CTFs+inel set got the worst result (largest average rank) in most experiments, for all 4 metrics. The Baseline and BL+CTFs sets had closer results, with the former achieving the smallest (best) average ranks overall – except Specificity for ELSA-nurse datasets, where BL+CTFs had a smaller rank, as well as Specificity for ELSA-core datasets, Accuracy for ELSA-core datasets, and GMean for TILDA datasets, where both tied. This indicates that, by itself, the Diff CTF does not represent enough information to work as a feasible substitute for the original information represented in the longitudinal dataset, although it can still be a good addition to a longitudinal dataset. The Friedman test p-values for Scenario 1 were all 1E−16. Thus, we ran the post-hoc Nemenyi pairwise test for all pairs of feature sets, and got significant values for all 4 metrics in the comparison of the Baseline set against the CTFs-only set, and BL+CTFs against CTFs-only, with p-value 0.001 for all 4 metrics.

The Scenario 2 experiments had similar results, but the addition of ineligible features with high predictive power, such as age and gender, slightly improved the results of the CTFs+inel feature set. The Friedman and Nemenyi test p-values for Scenario 2 were the same as Scenario 1 for all metrics except Specificity, which got a Friedman p-value of 0.0027. In the Nemenyi tests the comparison of Baseline and CTFs-only got a p-value of 0.0036, and the comparison of BL+CTFs and CTFs-only got a p-value of 0.0266, both still rejecting the null hypothesis. As in Scenario 1, the comparison of the Baseline set against the BL+CTFs did not get significant p-values for any of the metrics.

Table B.1: Diff Sensitivity and Specificity results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	0.678	0.639	**0.691**	**0.721**	0.609	0.690
EN_Arthritis	**0.667**	0.568	0.659	0.580	**0.587**	0.586
EN_Cataract	**0.654**	0.584	0.648	0.672	0.670	**0.680**
EN_Dementia	0.724	0.701	**0.741**	**0.750**	0.615	0.723
EN_Diabetes	**0.843**	0.686	0.839	0.868	0.681	**0.872**
EN_HBP	**0.631**	0.613	0.621	0.715	0.609	**0.723**
EN_Heartattack	0.689	0.635	**0.697**	0.711	0.606	**0.721**
EN_Osteoporosis	**0.652**	0.614	0.651	0.683	0.633	**0.694**
EN_Parkinsons	**0.627**	0.552	0.622	**0.697**	0.576	0.621
EN_Stroke	0.667	0.664	**0.676**	**0.732**	0.596	0.715
EC_Angina	0.692	0.625	**0.694**	**0.761**	0.656	0.754
EC_Arthritis	**0.730**	0.659	0.724	**0.711**	0.705	0.710
EC_Cataract	**0.584**	0.545	0.575	**0.631**	0.588	0.629
EC_Dementia	0.758	0.728	**0.759**	0.745	0.720	**0.770**
EC_Diabetes	0.653	0.553	**0.655**	**0.760**	0.607	0.754
EC_HBP	**0.627**	0.531	0.615	0.656	0.581	**0.661**
EC_Heartattack	**0.659**	0.592	0.657	**0.687**	0.603	0.680
EC_Osteoporosis	0.663	0.622	**0.671**	**0.648**	0.644	0.646
EC_Parkinsons	**0.705**	0.643	0.698	0.693	0.680	**0.747**
EC_Stroke	0.686	0.639	**0.687**	0.710	0.657	**0.731**
TI_Angina	**0.737**	0.689	**0.737**	0.828	0.752	**0.864**
TI_Arthritis	**0.694**	0.669	0.685	**0.644**	0.583	0.640
TI_Cancer	**0.529**	**0.529**	0.527	**0.566**	0.503	0.559
TI_Cataract	**0.669**	0.631	0.671	**0.690**	0.646	0.674
TI_Diabetes	**0.739**	0.702	0.732	**0.782**	0.719	**0.782**
TI_HBP	0.641	0.622	**0.644**	0.738	0.635	**0.745**
TI_Heartattack	0.740	0.706	**0.746**	**0.834**	0.780	0.829
TI_Ministroke	0.690	0.649	**0.695**	**0.745**	0.667	**0.745**
TI_Osteoporosis	**0.633**	0.608	0.630	0.751	0.672	**0.755**
TI_Stroke	0.709	0.651	**0.695**	**0.738**	0.569	0.708
AvgRank E-Nurse	**1.4**	3.0	1.6	1.7	2.8	**1.5**
AvgRank E-Core	**1.5**	3.0	**1.5**	**1.4**	3.0	1.6
AvgRank TILDA	**1.5**	2.9	1.7	**1.4**	3.0	1.6
AvgRank Overall	**1.5**	3.0	1.6	**1.5**	2.9	1.6

Table B.2: Diff Accuracy and GMean results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	0.679	0.638	**0.691**	**0.699**	0.624	0.690
EN_Arthritis	**0.630**	0.576	0.628	**0.622**	0.578	0.621
EN_Cataract	**0.660**	0.612	0.659	0.663	0.626	**0.664**
EN_Dementia	0.725	0.699	**0.741**	**0.737**	0.657	0.732
EN_Diabetes	**0.847**	0.685	0.844	**0.856**	0.683	**0.856**
EN_HBP	**0.665**	0.611	0.662	**0.672**	0.611	0.670
EN_Heartattack	0.690	0.633	**0.698**	0.700	0.620	**0.709**
EN_Osteoporosis	**0.655**	0.616	**0.655**	**0.667**	0.623	0.672
EN_Parkinsons	**0.628**	0.552	0.622	**0.661**	0.564	0.621
EN_Stroke	0.671	0.660	**0.678**	**0.698**	0.629	0.695
EC_Angina	0.694	0.626	**0.696**	**0.726**	0.640	0.724
EC_Arthritis	**0.723**	0.677	0.719	**0.721**	0.682	0.717
EC_Cataract	**0.598**	0.558	0.591	**0.607**	0.566	0.602
EC_Dementia	0.758	0.728	**0.760**	0.752	0.724	**0.765**
EC_Diabetes	0.666	0.560	**0.667**	**0.704**	0.579	0.703
EC_HBP	**0.638**	0.550	0.633	**0.641**	0.555	0.637
EC_Heartattack	**0.661**	0.592	0.658	**0.673**	0.597	0.669
EC_Osteoporosis	0.662	0.624	**0.669**	0.655	0.633	**0.659**
EC_Parkinsons	**0.705**	0.643	0.699	0.699	0.661	**0.722**
EC_Stroke	0.687	0.640	**0.690**	0.698	0.648	**0.709**
TI_Angina	0.741	0.692	**0.742**	0.781	0.720	**0.798**
TI_Arthritis	0.678	0.642	**0.671**	**0.668**	0.625	0.662
TI_Cancer	**0.531**	0.528	0.528	**0.547**	0.516	0.543
TI_Cataract	**0.671**	0.632	**0.671**	**0.680**	0.639	0.672
TI_Diabetes	**0.742**	0.703	0.736	**0.760**	0.711	0.757
TI_HBP	0.678	0.627	**0.682**	0.687	0.629	**0.693**
TI_Heartattack	0.743	0.708	**0.749**	0.785	0.742	**0.786**
TI_Ministroke	0.691	0.650	**0.696**	0.717	0.658	**0.720**
TI_Osteoporosis	**0.644**	0.614	0.642	0.689	0.639	**0.690**
TI_Stroke	**0.709**	0.650	0.695	**0.723**	0.609	0.701
AvgRank E-Nurse	**1.5**	3.0	1.6	**1.4**	3.0	1.7
AvgRank E-Core	**1.5**	3.0	**1.5**	**1.4**	3.0	1.6
AvgRank TILDA	**1.5**	3.0	1.6	**1.5**	3.0	**1.5**
AvgRank Overall	**1.5**	3.0	1.6	**1.4**	3.0	1.6

Table B.3: Diff Sensitivity and Specificity results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	**0.681**	0.624	0.675	**0.733**	0.671	0.721
EN_Arthritis	**0.672**	0.600	0.663	**0.587**	0.603	0.585
EN_Cataract	0.650	0.614	**0.651**	0.702	**0.714**	0.692
EN_Dementia	0.736	0.775	**0.743**	**0.770**	0.682	0.723
EN_Diabetes	**0.847**	0.684	0.842	**0.874**	0.687	0.873
EN_HBP	0.627	0.601	**0.631**	0.718	0.649	**0.726**
EN_Heartattack	**0.697**	0.648	0.686	0.733	0.646	**0.738**
EN_Osteoporosis	**0.660**	0.624	0.653	0.699	**0.709**	0.700
EN_Parkinsons	**0.654**	0.627	0.647	0.697	**0.727**	0.515
EN_Stroke	0.671	0.669	**0.680**	**0.703**	0.648	0.701
EC_Angina	0.695	0.640	**0.702**	0.751	0.656	**0.772**
EC_Arthritis	**0.740**	0.669	0.732	0.709	0.715	**0.716**
EC_Cataract	0.627	0.621	**0.630**	0.730	**0.744**	0.729
EC_Dementia	0.770	0.758	**0.775**	**0.801**	0.783	0.776
EC_Diabetes	**0.657**	0.572	**0.657**	0.759	**0.616**	0.753
EC_HBP	**0.637**	0.568	0.626	0.660	0.604	**0.672**
EC_Heartattack	**0.671**	0.622	0.669	**0.739**	0.719	0.698
EC_Osteoporosis	0.684	0.656	**0.687**	0.717	**0.725**	0.685
EC_Parkinsons	**0.715**	0.677	0.713	0.720	0.707	**0.733**
EC_Stroke	**0.694**	0.659	**0.694**	**0.731**	0.712	**0.731**
TI_Angina	**0.743**	0.702	0.741	0.852	0.744	**0.836**
TI_Arthritis	0.690	0.674	**0.694**	0.641	0.602	**0.649**
TI_Cancer	0.531	0.546	**0.534**	0.543	0.493	**0.553**
TI_Cataract	**0.710**	0.698	0.708	0.718	0.713	**0.732**
TI_Diabetes	0.778	**0.779**	0.769	**0.821**	0.779	**0.821**
TI_HBP	0.645	0.627	**0.646**	**0.742**	0.652	0.724
TI_Heartattack	**0.755**	0.725	0.747	**0.834**	0.800	0.829
TI_Ministroke	**0.695**	0.675	0.693	0.696	0.657	**0.706**
TI_Osteoporosis	**0.648**	0.641	0.645	0.797	0.788	**0.805**
TI_Stroke	**0.712**	0.663	0.703	**0.738**	0.615	0.692
AvgRank E-Nurse	**1.5**	2.8	1.7	**1.7**	2.2	2.1
AvgRank E-Core	**1.5**	3.0	**1.5**	**1.8**	2.3	2.0
AvgRank TILDA	**1.5**	2.6	1.9	1.6	3.0	**1.5**
AvgRank Overall	**1.5**	2.8	1.7	**1.7**	2.5	1.8

Table B.4: Diff Accuracy and GMean results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	**0.683**	0.626	0.677	**0.706**	0.647	0.698
EN_Arthritis	**0.636**	0.601	0.630	**0.628**	0.601	0.623
EN_Cataract	**0.667**	0.647	0.665	**0.676**	0.662	0.671
EN_Dementia	0.736	**0.773**	0.742	**0.753**	0.727	0.733
EN_Diabetes	**0.851**	0.684	0.846	**0.861**	0.685	0.857
EN_HBP	0.664	0.620	**0.669**	0.671	0.624	**0.677**
EN_Heartattack	**0.699**	0.648	**0.689**	**0.715**	0.647	0.712
EN_Osteoporosis	**0.663**	0.632	0.657	**0.679**	0.665	0.676
EN_Parkinsons	**0.655**	0.628	0.646	**0.675**	**0.675**	0.577
EN_Stroke	0.673	0.668	**0.681**	0.687	0.659	**0.690**
EC_Angina	0.696	0.640	**0.704**	0.722	0.648	**0.736**
EC_Arthritis	**0.728**	0.687	0.726	**0.724**	0.691	**0.724**
EC_Cataract	0.658	0.657	**0.659**	0.676	0.680	**0.678**
EC_Dementia	0.771	0.759	**0.775**	**0.785**	0.770	0.776
EC_Diabetes	0.670	0.577	**0.669**	**0.706**	0.593	0.703
EC_HBP	**0.646**	0.582	0.644	0.648	0.586	**0.649**
EC_Heartattack	**0.674**	0.627	0.671	**0.704**	0.668	0.684
EC_Osteoporosis	**0.687**	0.662	**0.687**	**0.700**	0.690	0.686
EC_Parkinsons	**0.715**	0.677	0.713	**0.717**	0.691	0.723
EC_Stroke	**0.696**	0.662	**0.696**	**0.713**	0.685	0.712
TI_Angina	**0.748**	0.704	0.745	**0.796**	0.723	0.787
TI_Arthritis	0.675	0.652	**0.680**	0.665	0.637	**0.671**
TI_Cancer	0.532	**0.544**	0.535	0.537	0.519	**0.543**
TI_Cataract	**0.710**	0.700	**0.710**	0.714	0.706	**0.720**
TI_Diabetes	**0.781**	0.779	0.773	**0.799**	0.779	0.795
TI_HBP	**0.682**	0.637	0.676	**0.692**	0.640	0.684
TI_Heartattack	**0.757**	0.728	0.750	**0.793**	0.762	0.787
TI_Ministroke	**0.695**	0.675	0.693	0.695	0.666	**0.699**
TI_Osteoporosis	**0.662**	0.655	0.660	0.719	0.711	**0.721**
TI_Stroke	**0.712**	0.663	0.703	**0.725**	0.639	0.698
AvgRank E-Nurse	**1.4**	2.8	1.8	**1.3**	2.9	1.9
AvgRank E-Core	**1.4**	3.0	1.6	**1.6**	2.7	1.8
AvgRank TILDA	**1.4**	2.7	2.0	**1.5**	3.0	**1.5**
AvgRank Overall	**1.4**	2.8	1.8	**1.4**	2.9	1.7

B.2 Ratio Results for Random Forests

Similarly to the Diff, The Ratio CTF is also related to the most recent change in the value of a longitudinal feature (change from the last but one to the last wave), but this type of CTF was created only for numeric features. The Ratio is more sensitive to changes in the values than the Diff, so it might be able to capture some trends more effectively. The results for Scenario 1 are shown in Tables B.5 (for Sensitivity and Specificity) and B.6 (for Accuracy and GMean). The results for Scenario 2 are shown in Tables B.7 (for Sensitivity and Specificity) and B.8 (for Accuracy and GMean).

For the Scenario 1 experiments, the BL+CTFs set had the smallest (best) average ranks overall in most cases – except for the Accuracy in TILDA datasets, where Baseline had the smallest rank, and both tied for Sensitivity in TILDA datasets, Accuracy in Elsa-core datasets and GMean in Elsa-nurse datasets. The better performance of the Ratio CTF compared to Diff may be due to the lack of nominal features in the datasets from these experiments, which considerably reduced the information available for the classifier, giving more importance to the added features overall. The Friedman test p-values for Scenario 1 were 1E−16 for all 4 metrics. We ran the Nemenyi post-hoc test and confirmed that the Baseline and BL+CTFs sets had significantly superior results to CTFs+inel for all 4 metrics, with p-values 0.001 in all cases. The p-values for the comparison between the Baseline and BL+CTFs sets were not significant in any of the metrics.

The apparent superiority of BL+CTFs in Scenario 1 changed for Scenario 2. The Baseline set got the smallest average ranks more often, and the results were closer, with the CTFs+inel set still getting the worst results. The Friedman tests returned significant p-values for all 4 metrics in this scenario as well, with p-values 1E−16 for Sensitivity, Accuracy and GMean, and 0.00011 for Specificity. Thus, we ran the Nemenyi post-hoc test for all 4 metrics. For all 4 metrics, the test confirmed that the Baseline set was superior to the CTFs+inel set (p-values 0.001 for Sensitivity, Accuracy and GMean, and 0.00286 for Specificity), and that the BL+CTFs set was also superior to CTFs+inel (p-values 0.001 for Specificity, Accuracy and GMean, and 0.00143 for Sensitivity). This inversion in the Ratio results from one Scenario to the other gives more weight to our initial assumption that the lack of nominal features in the dataset heavily influenced the classifiers.

Table B.5: Ratio Sensitivity and Specificity results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	**0.678**	0.606	**0.678**	**0.721**	0.597	0.705
EN_Arthritis	**0.667**	0.561	0.660	0.580	0.592	**0.584**
EN_Cataract	**0.654**	0.563	0.651	0.672	0.664	**0.679**
EN_Dementia	0.724	0.680	**0.744**	**0.750**	0.669	0.723
EN_Diabetes	0.843	0.687	**0.845**	0.868	0.688	**0.871**
EN_HBP	0.631	0.601	**0.633**	0.715	0.615	**0.717**
EN_Heartattack	0.689	0.624	**0.696**	0.711	0.613	**0.716**
EN_Osteoporosis	0.652	0.616	**0.656**	0.683	0.642	**0.699**
EN_Parkinsons	0.627	0.578	**0.630**	**0.697**	0.515	0.652
EN_Stroke	0.667	0.636	**0.679**	**0.732**	0.601	0.715
EC_Angina	**0.692**	0.615	**0.692**	0.761	0.632	**0.765**
EC_Arthritis	**0.730**	0.724	0.724	0.711	0.607	**0.715**
EC_Cataract	0.584	**0.624**	0.575	**0.631**	0.510	0.622
EC_Dementia	0.758	0.700	**0.762**	0.745	0.683	**0.752**
EC_Diabetes	0.653	0.631	**0.655**	0.760	0.519	**0.762**
EC_HBP	0.627	**0.654**	0.630	0.656	0.458	**0.657**
EC_Heartattack	**0.659**	0.585	0.657	**0.687**	0.560	0.673
EC_Osteoporosis	0.663	0.637	**0.671**	0.648	0.590	**0.661**
EC_Parkinsons	**0.705**	0.574	0.694	0.693	0.640	**0.720**
EC_Stroke	0.686	0.627	**0.687**	0.710	0.618	**0.738**
TI_Angina	**0.737**	0.689	0.733	0.828	0.748	**0.864**
TI_Arthritis	0.694	0.671	**0.695**	**0.644**	0.591	0.629
TI_Cancer	**0.529**	0.529	0.519	**0.566**	0.507	0.516
TI_Cataract	0.669	0.633	**0.673**	**0.690**	0.646	0.686
TI_Diabetes	**0.739**	0.694	0.736	0.782	0.719	**0.784**
TI_HBP	0.641	0.621	**0.646**	0.738	0.633	**0.746**
TI_Heartattack	0.740	0.708	**0.749**	0.834	0.771	**0.849**
TI_Ministroke	**0.690**	0.647	**0.690**	0.745	0.676	**0.765**
TI_Osteoporosis	0.633	0.596	**0.635**	**0.751**	0.650	0.733
TI_Stroke	**0.709**	0.649	0.695	0.738	0.646	**0.754**
AvgRank E-Nurse	1.8	3.0	**1.3**	1.7	2.8	**1.5**
AvgRank E-Core	1.8	2.6	**1.7**	1.8	3.0	**1.2**
AvgRank TILDA	1.6	2.9	**1.6**	1.6	3.0	**1.4**
AvgRank Overall	1.7	2.8	**1.5**	1.7	2.9	**1.4**

Table B.6: Ratio Accuracy and GMean results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	**0.679**	0.606	**0.679**	**0.699**	0.602	0.692
EN_Arthritis	**0.630**	0.574	0.628	**0.622**	0.576	0.621
EN_Cataract	**0.660**	0.596	**0.660**	0.663	0.612	**0.665**
EN_Dementia	0.725	0.680	**0.744**	**0.737**	0.674	0.733
EN_Diabetes	0.847	0.687	**0.849**	0.856	0.688	**0.858**
EN_HBP	0.665	0.607	**0.667**	0.672	0.608	**0.674**
EN_Heartattack	0.690	0.623	**0.697**	0.700	0.619	**0.706**
EN_Osteoporosis	0.655	0.618	**0.660**	**0.667**	0.629	**0.677**
EN_Parkinsons	0.628	0.577	**0.630**	**0.661**	0.545	0.641
EN_Stroke	0.671	0.634	**0.681**	**0.698**	0.618	0.697
EC_Angina	**0.694**	0.615	**0.694**	0.726	0.623	**0.727**
EC_Arthritis	**0.723**	0.678	0.721	**0.721**	0.663	0.720
EC_Cataract	0.598	0.590	**0.589**	**0.607**	0.564	0.598
EC_Dementia	0.758	0.700	**0.762**	0.752	0.692	**0.757**
EC_Diabetes	0.666	0.617	**0.669**	0.704	0.572	**0.706**
EC_HBP	0.638	0.578	**0.641**	0.641	0.547	**0.643**
EC_Heartattack	**0.661**	0.584	0.658	**0.673**	0.573	0.665
EC_Osteoporosis	0.662	0.633	**0.670**	0.655	0.613	**0.666**
EC_Parkinsons	**0.705**	0.574	0.694	0.699	0.606	**0.707**
EC_Stroke	0.687	0.626	**0.689**	0.698	0.622	**0.712**
TI_Angina	**0.741**	0.691	0.738	0.781	0.718	**0.796**
TI_Arthritis	**0.678**	0.646	0.674	**0.668**	0.630	0.661
TI_Cancer	**0.531**	0.528	0.519	**0.547**	0.518	0.518
TI_Cataract	0.671	0.634	**0.674**	**0.680**	0.639	0.679
TI_Diabetes	**0.742**	0.696	0.740	**0.760**	0.707	**0.760**
TI_HBP	0.678	0.625	**0.684**	0.687	0.627	**0.694**
TI_Heartattack	0.743	0.711	**0.753**	0.785	0.739	**0.797**
TI_Ministroke	0.691	0.648	**0.692**	0.717	0.662	**0.726**
TI_Osteoporosis	**0.644**	0.601	**0.644**	**0.689**	0.622	0.682
TI_Stroke	**0.709**	0.649	0.696	0.723	0.648	**0.724**
AvgRank E-Nurse	1.8	3.0	**1.2**	**1.5**	3.0	**1.5**
AvgRank E-Core	**1.6**	2.9	**1.6**	1.7	3.0	**1.3**
AvgRank TILDA	**1.5**	2.9	1.7	1.6	3.0	**1.5**
AvgRank Overall	1.6	2.9	**1.5**	1.6	3.0	**1.4**

Table B.7: Ratio Sensitivity and Specificity results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	**0.681**	0.617	0.680	**0.733**	0.647	0.702
EN_Arthritis	**0.672**	0.599	0.668	0.587	**0.616**	0.579
EN_Cataract	**0.650**	0.607	0.648	0.702	**0.719**	0.697
EN_Dementia	0.736	**0.771**	0.742	**0.770**	0.709	0.709
EN_Diabetes	**0.847**	0.685	0.844	**0.874**	0.677	0.862
EN_HBP	0.627	0.600	**0.630**	0.718	0.643	**0.728**
EN_Heartattack	**0.697**	0.639	0.693	**0.733**	0.658	**0.733**
EN_Osteoporosis	**0.660**	0.608	**0.660**	0.699	0.706	**0.711**
EN_Parkinsons	**0.654**	0.609	0.640	**0.697**	0.652	0.606
EN_Stroke	0.671	0.655	**0.678**	0.703	0.639	**0.705**
EC_Angina	0.695	0.610	**0.698**	0.751	0.621	**0.775**
EC_Arthritis	**0.740**	0.646	0.735	0.709	0.689	**0.716**
EC_Cataract	**0.627**	0.624	0.620	0.730	0.722	**0.733**
EC_Dementia	0.770	0.753	**0.782**	0.801	0.789	**0.832**
EC_Diabetes	0.657	0.531	**0.660**	0.759	0.574	**0.763**
EC_HBP	**0.637**	0.555	0.634	0.660	0.589	**0.671**
EC_Heartattack	**0.671**	0.608	**0.671**	0.739	0.698	**0.753**
EC_Osteoporosis	0.684	0.642	**0.688**	**0.717**	0.699	0.689
EC_Parkinsons	**0.715**	0.612	0.704	**0.720**	0.680	**0.720**
EC_Stroke	**0.694**	0.642	0.691	0.731	0.699	**0.734**
TI_Angina	**0.743**	0.699	0.741	**0.852**	0.764	0.836
TI_Arthritis	**0.690**	0.673	0.684	**0.641**	0.617	**0.641**
TI_Cancer	0.531	**0.541**	0.532	**0.543**	0.516	0.523
TI_Cataract	**0.710**	0.701	0.696	0.718	**0.728**	0.722
TI_Diabetes	0.778	**0.780**	0.771	**0.821**	0.779	**0.821**
TI_HBP	**0.645**	0.627	0.644	0.742	0.656	**0.748**
TI_Heartattack	**0.755**	0.724	0.748	**0.834**	0.776	0.805
TI_Ministroke	**0.695**	0.669	0.693	0.696	**0.716**	0.696
TI_Osteoporosis	0.648	0.642	**0.649**	0.797	0.785	**0.807**
TI_Stroke	**0.712**	0.661	0.696	0.738	0.646	**0.754**
AvgRank E-Nurse	**1.5**	2.8	1.8	**1.7**	2.4	2.0
AvgRank E-Core	**1.5**	2.9	1.7	1.9	2.9	**1.3**
AvgRank TILDA	**1.4**	2.5	2.1	1.8	2.6	**1.7**
AvgRank Overall	**1.4**	2.7	1.8	1.8	2.6	**1.6**

Table B.8: Ratio Accuracy and GMean results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	**0.683**	0.618	0.681	**0.706**	0.632	0.691
EN_Arthritis	**0.636**	0.606	0.630	**0.628**	0.607	0.622
EN_Cataract	**0.667**	0.644	0.664	**0.676**	0.661	0.672
EN_Dementia	0.736	**0.770**	0.741	**0.753**	0.740	0.726
EN_Diabetes	**0.851**	0.684	0.846	**0.861**	0.681	0.853
EN_HBP	0.664	0.617	**0.670**	0.671	0.621	**0.678**
EN_Heartattack	**0.699**	0.640	0.695	**0.715**	0.649	0.713
EN_Osteoporosis	0.663	0.617	**0.664**	0.679	0.656	**0.685**
EN_Parkinsons	**0.655**	0.610	0.640	**0.675**	0.630	0.623
EN_Stroke	0.673	0.654	**0.680**	0.687	0.647	**0.692**
EC_Angina	0.696	0.611	**0.701**	0.722	0.616	**0.736**
EC_Arthritis	**0.728**	0.663	**0.728**	0.724	0.667	**0.726**
EC_Cataract	**0.658**	0.653	0.654	**0.676**	0.671	0.674
EC_Dementia	0.771	0.754	**0.783**	0.785	0.771	**0.807**
EC_Diabetes	0.670	0.537	**0.673**	0.706	0.552	**0.710**
EC_HBP	0.646	0.568	**0.649**	0.648	0.572	**0.653**
EC_Heartattack	0.674	0.613	**0.675**	0.704	0.652	**0.711**
EC_Osteoporosis	0.687	0.646	**0.688**	**0.700**	0.669	0.689
EC_Parkinsons	**0.715**	0.613	0.704	**0.717**	0.645	0.712
EC_Stroke	**0.696**	0.645	0.694	**0.713**	0.670	0.712
TI_Angina	**0.748**	0.702	0.745	**0.796**	0.731	0.787
TI_Arthritis	**0.675**	0.656	0.671	**0.665**	0.644	0.662
TI_Cancer	0.532	**0.540**	0.532	**0.537**	0.529	0.528
TI_Cataract	**0.710**	0.703	0.698	**0.714**	**0.714**	0.709
TI_Diabetes	**0.781**	0.780	0.774	**0.799**	0.779	0.795
TI_HBP	0.682	0.638	**0.683**	0.692	0.642	**0.694**
TI_Heartattack	**0.757**	0.726	0.750	**0.793**	0.750	0.776
TI_Ministroke	**0.695**	0.670	0.693	**0.695**	0.692	0.694
TI_Osteoporosis	0.662	0.655	**0.664**	0.719	0.710	**0.723**
TI_Stroke	**0.712**	0.661	0.696	**0.725**	0.653	0.724
AvgRank E-Nurse	**1.5**	2.8	1.7	**1.3**	2.8	1.9
AvgRank E-Core	1.7	3.0	**1.4**	1.6	3.0	**1.4**
AvgRank TILDA	**1.4**	2.6	2.1	**1.3**	2.8	2.0
AvgRank Overall	**1.5**	2.8	1.7	**1.4**	2.9	1.8

B.3 Monotonicity Results for Random Forests

This CTF represents clear temporal patterns of monotonic increase or decrease over all stored values of a longitudinal feature. The Monotonicity values represent the pattern itself, so this feature can be an important asset in identifying temporal trends in data, such as a tendency to decreasing values of consecutive measures of a conceptual feature for instances of the positive class, for example. The Monotonicity results for Scenario 1 are shown in Tables B.9 (for Sensitivity and Specificity) and B.10 (for Accuracy and GMean). The results for Scenario 2 are shown in Tables B.11 (for Sensitivity and Specificity) and B.12 (for Accuracy and GMean).

As with the Diff and Ratio, the CTFs+inel feature set got the worst results overall, and the other two feature sets got closer results. The Friedman test p-values for Scenario 1 were 1E−16 for all metrics, and the Nemenyi test results confirmed that the Baseline and BL+CTFs sets were superior to CTFs+inel (p-values 0.001 in all cases), but not to each other (no significant p-values).

Once again, adding the highly predictive ineligible features in Scenario 2 further skewed the results in favour of the Baseline feature set, which got the lowest average ranks in all cases except Specificity for Elsa-nurse and Elsa-core datasets. The Friedman test p-values for Scenario 2 were 1E−16 for Sensitivity, Accuracy and GMean, and 0.0022 for Specificity. In the Nemenyi post-hoc test, for Sensitivity, the p-value for comparing Baseline and CTFs+inel was 0.0010, and for comparing BL+CTFs to CTFs+inel it was 0.0014. For Specificity, the only pair with a significant p-value was BL+CTFs vs CTFs+inel, with p-value 0.01028. For Accuracy, both Baseline and BL+CTFs, when compared to CTFs+inel, got a p-value of 0.001. For GMean, Baseline and CTFs+inel had a p-value of 0.001, and BL+CTFs and CTFs+inel got 0.0023. For none of the post-hoc tests the Baseline and BL+CTF tests had significant p-values in their comparison with each other.

From these results, we can surmise that the Monotonicity CTF did not get good results by itself, likely because the information it represents is very coarse-grained, and decision tree classifiers tend to select the features with more fine-grained information, such as numerical features. However, the BL+CTF feature set was still comparable to the Baseline in most cases, so the addition of this CTF by itself was arguably not a detriment to the classifiers.

Table B.9: Monotonicity Sensitivity and Specificity results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is bold-faced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	0.680	0.588	**0.681**	0.678	0.574	**0.682**
EN_Arthritis	0.669	0.543	**0.672**	0.594	0.591	**0.597**
EN_Cataract	**0.615**	0.548	0.614	0.723	0.711	**0.729**
EN_Dementia	**0.737**	0.679	0.733	0.716	0.676	**0.736**
EN_Diabetes	**0.843**	0.674	0.841	**0.865**	0.682	0.862
EN_HBP	**0.653**	0.560	0.647	0.747	0.625	**0.755**
EN_Heartattack	**0.698**	0.603	0.697	0.731	0.643	**0.733**
EN_Osteoporosis	**0.655**	0.581	0.653	**0.699**	0.602	0.690
EN_Parkinsons	**0.604**	0.594	0.596	**0.636**	0.621	0.621
EN_Stroke	**0.667**	0.631	0.666	0.710	0.651	**0.713**
EC_Angina	**0.710**	0.645	**0.710**	**0.723**	0.635	0.716
EC_Arthritis	**0.741**	0.640	0.740	0.717	0.575	**0.719**
EC_Cataract	**0.601**	0.584	0.597	0.675	0.541	**0.680**
EC_Dementia	**0.757**	0.657	0.746	**0.727**	0.634	0.720
EC_Diabetes	**0.671**	0.588	0.669	**0.750**	0.601	0.743
EC_HBP	0.625	0.585	**0.630**	0.662	0.540	**0.669**
EC_Heartattack	**0.673**	0.596	0.671	**0.689**	0.601	0.685
EC_Osteoporosis	**0.690**	0.600	0.687	0.635	0.562	**0.651**
EC_Parkinsons	0.685	0.633	**0.705**	**0.720**	0.627	0.680
EC_Stroke	0.689	0.625	**0.692**	0.694	0.583	**0.710**
TI_Angina	**0.748**	0.639	0.743	**0.876**	0.584	0.872
TI_Arthritis	**0.731**	0.608	0.722	**0.652**	0.559	0.648
TI_Cancer	**0.542**	0.531	0.539	**0.595**	0.477	0.592
TI_Cataract	**0.660**	0.608	0.658	0.705	0.548	**0.713**
TI_Diabetes	0.737	0.582	**0.743**	0.795	0.535	**0.808**
TI_HBP	**0.678**	0.557	0.677	0.763	0.534	**0.765**
TI_Heartattack	**0.750**	0.584	0.746	**0.863**	0.532	**0.863**
TI_Ministroke	0.704	0.643	**0.711**	0.765	0.667	**0.775**
TI_Osteoporosis	**0.670**	0.566	0.654	0.772	0.551	**0.777**
TI_Stroke	0.717	0.610	**0.718**	**0.738**	0.538	0.692
AvgRank E-Nurse	**1.2**	3.0	1.8	1.7	3.0	**1.4**
AvgRank E-Core	**1.4**	3.0	1.7	**1.5**	3.0	**1.5**
AvgRank TILDA	**1.3**	3.0	1.7	1.6	3.0	**1.5**
AvgRank Overall	**1.3**	3.0	1.7	1.6	3.0	**1.4**

Table B.10: Monotonicity Accuracy and GMean results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	0.680	0.588	**0.681**	0.679	0.581	**0.681**
EN_Arthritis	0.637	0.563	**0.641**	0.630	0.566	**0.634**
EN_Cataract	0.650	0.601	**0.652**	0.667	0.624	**0.669**
EN_Dementia	**0.736**	0.679	0.733	0.726	0.678	**0.735**
EN_Diabetes	**0.846**	0.675	0.844	**0.854**	0.678	0.851
EN_HBP	**0.691**	0.586	0.690	**0.699**	0.592	**0.699**
EN_Heartattack	**0.700**	0.605	0.699	0.714	0.623	**0.715**
EN_Osteoporosis	**0.659**	0.583	0.656	**0.676**	0.591	0.671
EN_Parkinsons	**0.605**	0.594	0.597	**0.620**	0.607	0.609
EN_Stroke	**0.670**	0.633	0.669	0.688	0.641	**0.689**
EC_Angina	**0.711**	0.645	0.710	**0.716**	0.640	0.713
EC_Arthritis	0.731	0.614	**0.732**	**0.729**	0.607	**0.729**
EC_Cataract	**0.623**	0.571	0.622	0.637	0.562	**0.638**
EC_Dementia	**0.757**	0.657	0.745	**0.742**	0.645	0.733
EC_Diabetes	**0.681**	0.590	0.679	**0.709**	0.594	0.705
EC_HBP	0.639	0.568	**0.645**	0.643	0.562	**0.649**
EC_Heartattack	**0.674**	0.597	0.672	**0.681**	0.599	0.678
EC_Osteoporosis	**0.686**	0.597	0.684	0.662	0.581	**0.669**
EC_Parkinsons	0.685	0.633	**0.704**	**0.702**	0.630	0.692
EC_Stroke	0.689	0.623	**0.693**	0.692	0.604	**0.701**
TI_Angina	**0.753**	0.637	0.749	**0.809**	0.611	0.805
TI_Arthritis	**0.706**	0.593	0.699	**0.690**	0.583	0.684
TI_Cancer	**0.545**	0.528	0.542	**0.568**	0.503	0.565
TI_Cataract	**0.664**	0.603	0.663	0.682	0.577	**0.685**
TI_Diabetes	0.741	0.578	**0.747**	0.765	0.558	**0.775**
TI_HBP	**0.710**	0.548	**0.710**	0.719	0.545	**0.720**
TI_Heartattack	**0.755**	0.582	0.750	**0.805**	0.557	0.802
TI_Ministroke	0.705	0.643	**0.712**	0.734	0.655	**0.742**
TI_Osteoporosis	**0.680**	0.565	0.665	**0.719**	0.558	0.713
TI_Stroke	**0.717**	0.609	**0.717**	**0.728**	0.573	0.705
AvgRank E-Nurse	**1.3**	3.0	1.7	1.7	3.0	**1.4**
AvgRank E-Core	**1.4**	3.0	1.6	**1.5**	3.0	1.6
AvgRank TILDA	**1.3**	3.0	1.7	**1.4**	3.0	1.6
AvgRank Overall	**1.3**	3.0	1.7	**1.5**	3.0	**1.5**

Table B.11: Monotonicity Sensitivity and Specificity results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is bold-faced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	**0.684**	0.595	0.678	**0.702**	0.632	0.686
EN_Arthritis	**0.671**	0.574	0.662	0.586	**0.612**	0.593
EN_Cataract	**0.620**	0.605	0.615	0.723	**0.734**	**0.734**
EN_Dementia	0.729	0.719	**0.734**	0.709	0.743	**0.757**
EN_Diabetes	**0.841**	0.673	0.839	**0.866**	0.680	0.854
EN_HBP	**0.651**	0.567	0.650	**0.749**	0.641	0.746
EN_Heartattack	**0.700**	0.631	0.689	**0.738**	0.686	0.736
EN_Osteoporosis	0.649	0.626	**0.652**	0.696	0.691	**0.708**
EN_Parkinsons	**0.628**	0.612	0.604	0.712	**0.727**	0.712
EN_Stroke	**0.670**	0.657	0.664	**0.724**	0.677	0.708
EC_Angina	**0.711**	0.629	0.707	0.723	0.667	**0.758**
EC_Arthritis	**0.749**	0.629	0.747	**0.717**	0.646	0.714
EC_Cataract	0.609	**0.615**	0.609	0.717	**0.739**	0.718
EC_Dementia	**0.764**	0.739	0.759	0.770	**0.789**	0.758
EC_Diabetes	0.674	0.581	**0.675**	0.747	0.619	**0.763**
EC_HBP	**0.641**	0.580	0.631	0.662	0.607	**0.666**
EC_Heartattack	**0.678**	0.620	0.675	0.692	0.683	**0.694**
EC_Osteoporosis	**0.700**	0.650	0.699	0.676	**0.711**	0.652
EC_Parkinsons	**0.697**	0.661	0.696	0.693	**0.747**	0.680
EC_Stroke	0.694	0.633	**0.696**	0.721	0.703	**0.725**
TI_Angina	**0.748**	0.681	0.747	**0.916**	0.736	0.888
TI_Arthritis	**0.729**	0.644	0.724	0.646	0.612	**0.650**
TI_Cancer	0.549	0.550	**0.557**	0.579	0.530	**0.589**
TI_Cataract	**0.706**	0.700	0.691	0.724	0.720	**0.741**
TI_Diabetes	**0.775**	0.768	0.766	**0.831**	0.730	0.829
TI_HBP	**0.678**	0.592	0.676	**0.765**	0.595	0.760
TI_Heartattack	**0.751**	0.671	0.744	**0.878**	0.727	0.859
TI_Ministroke	0.712	0.673	**0.716**	**0.735**	0.657	0.706
TI_Osteoporosis	**0.677**	0.633	0.675	**0.807**	0.779	0.799
TI_Stroke	**0.728**	0.636	0.719	**0.800**	0.615	0.785
AvgRank E-Nurse	**1.2**	2.9	1.9	1.9	2.4	**1.8**
AvgRank E-Core	**1.4**	2.8	1.9	2.0	2.2	**1.8**
AvgRank TILDA	**1.3**	2.7	2.0	**1.3**	3.0	1.7
AvgRank Overall	**1.3**	2.8	1.9	**1.7**	2.5	1.8

Table B.12: Monotonicity Accuracy and GMean results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs+inel	BL+CTFs	Baseline	CTFs+inel	BL+CTFs
EN_Angina	**0.684**	0.596	0.678	**0.693**	0.613	0.682
EN_Arthritis	**0.635**	0.590	0.633	**0.627**	0.592	0.626
EN_Cataract	**0.654**	0.647	**0.654**	0.670	0.666	**0.672**
EN_Dementia	0.729	0.720	**0.734**	0.719	0.731	**0.745**
EN_Diabetes	**0.845**	0.674	0.841	**0.854**	0.676	0.846
EN_HBP	**0.690**	0.597	0.689	**0.698**	0.603	0.696
EN_Heartattack	**0.702**	0.634	0.691	**0.719**	0.658	0.712
EN_Osteoporosis	0.654	0.632	**0.657**	0.672	0.658	**0.679**
EN_Parkinsons	**0.629**	0.613	0.605	**0.669**	0.667	0.656
EN_Stroke	**0.674**	0.658	0.667	**0.697**	0.667	0.686
EC_Angina	**0.711**	0.630	0.709	0.717	0.647	**0.732**
EC_Arthritis	**0.736**	0.636	0.734	**0.733**	0.637	0.730
EC_Cataract	0.641	**0.651**	0.641	0.661	**0.674**	0.661
EC_Dementia	**0.764**	0.740	0.759	**0.767**	0.763	0.758
EC_Diabetes	0.684	0.586	**0.686**	0.710	0.600	**0.718**
EC_HBP	**0.649**	0.590	0.645	**0.651**	0.593	0.648
EC_Heartattack	**0.679**	0.623	0.676	**0.685**	0.650	0.685
EC_Osteoporosis	**0.698**	0.655	0.695	**0.688**	0.680	0.675
EC_Parkinsons	**0.697**	0.662	0.695	0.695	**0.702**	0.688
EC_Stroke	0.695	0.636	**0.698**	0.707	0.667	**0.711**
TI_Angina	**0.756**	0.683	0.754	**0.828**	0.708	0.815
TI_Arthritis	**0.703**	0.634	0.701	**0.686**	0.628	0.686
TI_Cancer	0.550	0.549	**0.559**	0.564	0.540	**0.573**
TI_Cataract	**0.707**	0.702	0.695	**0.715**	0.710	**0.715**
TI_Diabetes	**0.779**	0.765	0.770	**0.803**	0.749	0.796
TI_HBP	**0.711**	0.593	0.708	**0.720**	0.593	0.717
TI_Heartattack	**0.756**	0.673	0.748	**0.812**	0.699	0.799
TI_Ministroke	0.712	0.672	**0.716**	**0.724**	0.665	0.711
TI_Osteoporosis	**0.689**	0.647	0.687	**0.739**	0.702	0.735
TI_Stroke	**0.728**	0.636	0.719	**0.763**	0.625	0.751
AvgRank E-Nurse	**1.3**	2.9	1.9	**1.4**	2.8	1.8
AvgRank E-Core	**1.4**	2.8	1.9	**1.6**	2.4	2.0
AvgRank TILDA	**1.2**	2.9	1.9	**1.2**	3.0	1.8
AvgRank Overall	**1.3**	2.9	1.9	**1.4**	2.7	1.9

B.4 Results for DiffAgeMean

This CTF, proposed in this thesis, measures the deviation of a feature's last measure from an expected value computed from individuals of the same age – the expected value is the mean for numerical features and the mode for nominal features. As our datasets are from ageing studies, the age of the participant is arguably the most important feature to determine what a "normal" value would be for the features being measured, thus this CTF can help identify recent trends of deviation from the norm.

The DiffAgeMean results for Scenario 1 are shown in Tables B.13 (for Sensitivity and Specificity) and B.14 (for Accuracy and GMean). The results for Scenario 2 are shown in Tables B.15 (for Sensitivity and Specificity) and B.16 (for Accuracy and GMean).

DiffAgeMean obtained better results for the BL+CTFs feature sets overall, in both Scenarios, which is encouraging. The CTF feature sets (CTFs-only and CTFs+in, in each Scenario) had the worst results overall.

For Scenario 1, the BL+CTFs feature set had the smallest average rank in all cases except Specificity for the ELSA-nurse datasets, where the Base-el approach had an average rank of 1.5, against BL+CTF's 1.7 average. The Friedman test p-values for Scenario 1 were 0.0006, 1E−5, , 2E−5 and 1E−16 (the smallest threshold for a number to be shown in our implementation for these tests) for Sensitivity, Specificity, Accuracy and GMean, respectively. In the Nemenyi post-hoc test, the Sensitivity metric had significant p-values when comparing the BL+CTFs set to the baseline (0.0084) and to the CTFs-only set (0.001). The Specificity metric had significant Nemenyi p-values (0.001) for the comparisons of Base-el vs. CTFs-only and BL+CTFs vs. CTFs-only. For Accuracy, BL+CTFs had significant p-values when compared to both the Base-el (0.0036) and CTFs-only (0.001) sets. Finally, for GMean, all 3 set comparisons had significant p-values: BL+CTFs and CTFs got 0.001, BL+CTFs and Base-el got 0.0266, and Base-el and CTFs got 0.0266.

For Scenario 2, the best Specificity average ranks were all from the Base-el-in feature set, but for the Sensitivity, Accuracy and GMean metrics the BL+CTFs approach had the smallest average ranks in all cases. The Friedman test p-values for Scenario 2 were 0.0022, 0.0004, 0.0015 and 8E−5 for Sensitivity, Specificity, Accuracy and GMean, respectively. In the Nemenyi post-hoc tests, for Sensitivity,

only the BL+CTFs vs. CTFs-in comparison got a significant p-value (0.0014). For Specificity, the significant p-values were obtained for Base-el-in vs. CTFs-in (0.001) and BL+CTFs vs. CTFs-in (0.0222). For Accuracy, only the BL+CTFs vs. CTFs-in got a significant p-value (0.001). Finally, for GMean, the significant p-values were obtained for Base-el-in vs. CTFs-in (0.0036) and BL+CTFs vs. CTFs-in (0.001).

These these results are a clear indication that the addition of the DiffAgeMean CTF to the datasets led to an overall increased predictive accuracy for the RF classifiers, although the comparison between BL+CTFs and the Base-el-in set did not get significant results in the Nemenyi post-hoc tests.

Table B.13: DiffAgeMean Sensitivity and Specificity results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is bold-faced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	**0.672**	0.662	0.670	**0.729**	0.674	**0.729**
EN_Arthritis	**0.671**	0.653	0.669	**0.579**	0.570	0.576
EN_Cataract	0.667	0.622	**0.674**	0.646	0.612	**0.649**
EN_Dementia	0.714	0.695	**0.722**	0.743	0.669	**0.757**
EN_Diabetes	0.844	0.847	**0.850**	**0.864**	0.847	0.856
EN_HBP	0.619	**0.634**	0.630	**0.701**	0.668	0.698
EN_Heartattack	**0.692**	0.688	0.692	0.726	0.688	**0.741**
EN_Osteoporosis	**0.647**	0.590	0.637	**0.702**	0.691	0.682
EN_Parkinsons	0.637	0.631	**0.652**	**0.667**	0.636	0.652
EN_Stroke	0.669	0.669	**0.674**	0.727	0.653	**0.734**
EC_Angina	0.677	**0.693**	0.689	0.744	0.740	**0.754**
EC_Arthritis	0.696	0.712	**0.724**	**0.705**	0.688	0.693
EC_Cataract	0.564	**0.627**	0.625	0.622	0.723	**0.743**
EC_Dementia	0.763	0.770	**0.783**	0.764	0.801	**0.832**
EC_Diabetes	0.635	**0.671**	0.649	**0.742**	0.690	0.735
EC_HBP	0.583	0.614	**0.617**	0.648	0.644	**0.660**
EC_Heartattack	0.639	0.639	**0.655**	**0.723**	0.694	0.705
EC_Osteoporosis	0.660	0.663	**0.667**	0.638	**0.670**	0.666
EC_Parkinsons	0.702	0.716	**0.723**	0.667	**0.707**	0.680
EC_Stroke	0.673	0.677	**0.682**	0.714	0.742	**0.755**
TI_Angina	**0.715**	0.606	0.715	**0.820**	0.672	0.808
TI_Arthritis	0.665	0.616	**0.688**	0.629	0.559	**0.645**
TI_Cancer	0.509	**0.556**	0.548	**0.543**	0.530	0.526
TI_Cataract	0.664	0.684	**0.701**	0.688	0.630	**0.703**
TI_Diabetes	**0.734**	0.571	0.729	**0.761**	0.545	0.753
TI_HBP	0.631	0.557	**0.636**	0.721	0.547	**0.727**
TI_Heartattack	**0.732**	0.618	0.731	**0.785**	0.620	0.761
TI_Ministroke	0.677	0.617	**0.678**	0.706	0.588	**0.716**
TI_Osteoporosis	0.626	0.557	**0.639**	0.729	0.584	**0.751**
TI_Stroke	**0.701**	0.554	0.679	0.662	0.569	**0.708**
AvgRank E-Nurse	1.9	2.7	**1.5**	**1.5**	2.9	1.7
AvgRank E-Core	3.0	1.8	**1.3**	2.2	2.3	**1.5**
AvgRank TILDA	1.9	2.7	**1.5**	1.6	2.9	**1.5**
AvgRank Overall	2.2	2.4	**1.4**	1.8	2.7	**1.6**

Table B.14: DiffAgeMean Accuracy and GMean results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	**0.674**	0.663	0.673	**0.700**	0.668	0.699
EN_Arthritis	**0.632**	0.618	0.629	**0.623**	0.610	0.620
EN_Cataract	0.660	0.618	**0.666**	0.656	0.617	**0.661**
EN_Dementia	0.715	0.695	**0.722**	0.729	0.682	**0.739**
EN_Diabetes	0.847	0.847	**0.851**	**0.854**	0.847	0.853
EN_HBP	0.652	0.648	**0.657**	0.659	0.651	**0.663**
EN_Heartattack	0.694	0.688	**0.695**	0.709	0.688	**0.716**
EN_Osteoporosis	**0.652**	0.600	0.641	**0.674**	0.639	0.659
EN_Parkinsons	0.637	0.631	**0.652**	**0.652**	0.634	0.652
EN_Stroke	0.672	0.668	**0.677**	0.697	0.661	**0.703**
EC_Angina	0.680	**0.694**	0.691	0.710	0.716	**0.721**
EC_Arthritis	0.700	0.702	**0.712**	0.701	0.700	**0.709**
EC_Cataract	0.581	0.655	**0.660**	0.593	0.673	**0.682**
EC_Dementia	0.763	0.771	**0.784**	0.764	0.786	**0.808**
EC_Diabetes	0.649	**0.673**	0.660	0.686	0.680	**0.690**
EC_HBP	0.608	0.626	**0.634**	0.615	0.629	**0.638**
EC_Heartattack	0.644	0.642	**0.657**	**0.680**	0.666	**0.680**
EC_Osteoporosis	0.658	0.663	**0.667**	0.649	**0.667**	**0.667**
EC_Parkinsons	0.702	0.716	**0.722**	0.684	**0.711**	0.701
EC_Stroke	0.675	0.681	**0.686**	0.693	0.709	**0.718**
TI_Angina	**0.720**	0.609	0.719	**0.766**	0.638	0.760
TI_Arthritis	0.654	0.599	**0.675**	0.647	0.587	**0.666**
TI_Cancer	0.511	**0.554**	0.546	0.526	**0.543**	0.537
TI_Cataract	0.666	0.680	**0.701**	0.676	0.656	**0.702**
TI_Diabetes	**0.736**	0.570	0.730	**0.747**	0.558	0.741
TI_HBP	0.665	0.553	**0.671**	0.675	0.552	**0.680**
TI_Heartattack	**0.734**	0.618	0.732	**0.758**	0.619	0.746
TI_Ministroke	0.677	0.616	**0.678**	0.691	0.602	**0.696**
TI_Osteoporosis	0.636	0.560	**0.650**	0.676	0.570	**0.693**
TI_Stroke	**0.701**	0.554	0.679	0.681	0.561	**0.693**
AvgRank E-Nurse	1.8	3.0	**1.3**	1.6	3.0	**1.5**
AvgRank E-Core	2.9	1.9	**1.2**	2.7	2.2	**1.2**
AvgRank TILDA	1.8	2.7	**1.5**	1.8	2.8	**1.4**
AvgRank Overall	2.2	2.5	**1.3**	2.0	2.7	**1.4**

Table B.15: DiffAgeMean Sensitivity and Specificity results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is bold-faced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	0.666	0.671	**0.674**	0.736	0.709	**0.752**
EN_Arthritis	**0.671**	0.661	0.666	0.586	**0.604**	0.584
EN_Cataract	0.642	0.622	**0.649**	0.704	**0.724**	0.700
EN_Dementia	0.709	**0.730**	0.719	**0.757**	0.736	**0.757**
EN_Diabetes	0.843	0.845	**0.850**	**0.857**	0.849	0.855
EN_HBP	0.622	0.624	**0.625**	**0.698**	0.680	0.691
EN_Heartattack	0.690	**0.694**	0.692	**0.756**	0.728	0.731
EN_Osteoporosis	**0.649**	0.629	0.648	0.714	**0.722**	0.716
EN_Parkinsons	0.660	**0.668**	**0.668**	0.682	**0.697**	**0.667**
EN_Stroke	0.663	**0.681**	0.677	**0.720**	0.689	**0.720**
EC_Angina	0.694	0.691	**0.698**	**0.775**	0.754	0.758
EC_Arthritis	0.727	0.729	**0.732**	0.693	0.689	**0.697**
EC_Cataract	**0.629**	0.627	0.623	0.752	0.748	**0.768**
EC_Dementia	**0.780**	0.766	0.777	**0.826**	0.801	**0.826**
EC_Diabetes	0.642	**0.674**	0.657	**0.749**	0.724	0.746
EC_HBP	**0.622**	0.615	0.621	**0.664**	0.658	0.657
EC_Heartattack	0.668	0.653	**0.669**	0.737	0.739	**0.741**
EC_Osteoporosis	0.678	0.670	**0.686**	**0.738**	0.715	0.707
EC_Parkinsons	0.714	0.719	**0.724**	0.720	0.720	**0.733**
EC_Stroke	**0.687**	0.670	0.685	0.736	**0.762**	0.731
TI_Angina	0.719	0.645	**0.723**	**0.804**	0.708	0.800
TI_Arthritis	0.680	0.637	**0.685**	0.636	0.577	**0.641**
TI_Cancer	0.541	**0.558**	0.552	0.536	0.526	**0.586**
TI_Cataract	0.717	0.698	**0.720**	**0.730**	0.686	0.705
TI_Diabetes	**0.788**	0.766	0.783	**0.803**	0.722	0.800
TI_HBP	0.635	0.581	**0.641**	**0.721**	0.584	0.720
TI_Heartattack	**0.747**	0.655	0.741	0.795	0.688	**0.820**
TI_Ministroke	**0.679**	0.645	0.675	**0.716**	0.657	0.686
TI_Osteoporosis	0.654	0.649	**0.659**	**0.792**	0.768	**0.792**
TI_Stroke	**0.702**	0.585	0.676	**0.708**	0.569	0.646
AvgRank E-Nurse	2.5	2.0	**1.6**	**1.7**	2.2	2.1
AvgRank E-Core	1.9	2.5	**1.6**	**1.7**	2.5	1.9
AvgRank TILDA	1.7	2.8	**1.5**	**1.4**	3.0	1.7
AvgRank Overall	2.0	2.4	**1.6**	**1.6**	2.6	1.9

Table B.16: DiffAgeMean Accuracy and GMean results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	**0.669**	0.672	0.676	0.700	0.690	**0.712**
EN_Arthritis	0.635	0.637	**0.631**	0.627	**0.632**	0.623
EN_Cataract	**0.663**	0.655	0.666	0.673	0.671	**0.674**
EN_Dementia	0.710	**0.730**	0.720	0.733	0.733	**0.738**
EN_Diabetes	0.845	0.846	**0.851**	0.850	0.847	**0.852**
EN_HBP	**0.653**	0.647	0.651	**0.659**	0.652	0.657
EN_Heartattack	**0.694**	0.696	0.694	**0.722**	0.711	0.711
EN_Osteoporosis	**0.655**	0.637	0.655	0.681	0.674	**0.681**
EN_Parkinsons	0.661	**0.668**	0.668	0.671	**0.682**	0.668
EN_Stroke	0.667	**0.682**	0.679	0.691	0.685	**0.698**
EC_Angina	**0.697**	0.693	0.700	**0.734**	0.722	0.727
EC_Arthritis	0.713	0.713	**0.718**	0.710	0.709	**0.714**
EC_Cataract	0.665	**0.663**	0.666	0.688	0.685	**0.692**
EC_Dementia	0.781	0.766	**0.778**	**0.803**	0.783	0.801
EC_Diabetes	0.656	**0.680**	0.668	0.693	0.698	**0.700**
EC_HBP	0.638	**0.632**	0.635	**0.643**	0.636	0.639
EC_Heartattack	0.672	0.658	**0.672**	0.702	0.695	**0.704**
EC_Osteoporosis	0.683	0.673	**0.688**	**0.707**	0.692	0.696
EC_Parkinsons	0.714	**0.719**	0.724	0.717	0.719	**0.729**
EC_Stroke	0.690	0.675	**0.687**	0.711	**0.715**	0.708
TI_Angina	**0.723**	0.648	0.726	**0.760**	0.676	**0.760**
TI_Arthritis	**0.666**	0.619	0.671	0.658	0.606	**0.663**
TI_Cancer	0.541	**0.556**	0.554	0.539	0.542	**0.569**
TI_Cataract	0.718	0.697	**0.719**	**0.724**	0.692	0.713
TI_Diabetes	**0.789**	0.763	0.784	**0.795**	0.744	0.791
TI_HBP	**0.668**	0.582	0.671	0.677	0.582	**0.679**
TI_Heartattack	**0.749**	0.656	0.744	0.771	0.671	**0.779**
TI_Ministroke	**0.679**	0.645	0.676	**0.697**	0.651	0.681
TI_Osteoporosis	**0.667**	0.661	0.671	0.720	0.706	**0.722**
TI_Stroke	**0.702**	0.585	0.676	**0.705**	0.577	0.661
AvgRank E-Nurse	2.4	1.9	**1.8**	1.8	2.5	**1.7**
AvgRank E-Core	1.9	2.7	**1.5**	1.8	2.6	**1.6**
AvgRank TILDA	1.7	2.8	**1.5**	1.7	2.9	**1.5**
AvgRank Overall	2.0	2.4	**1.6**	1.8	2.7	**1.6**

B.5 Results for AvgDiffAgeMean

The AvgDiffAgeMean is an extension of the DiffAgeMean, which considers all measurements of a longitudinal feature instead of only the most recent, comparing each measurement to the expected value for the age of the respondent at the time of measurement. The value of the trends identified by this feature, compared to the DiffAgeMean, hinges on the importance of older values and on the importance of a consistent deviation from the expected value. The AvgDiffAgeMean results for Scenario 1 are shown in Tables B.17 (for Sensitivity and Specificity) and B.18 (for Accuracy and GMean). The results for Scenario 2 are shown in Tables B.19 (for Sensitivity and Specificity) and B.20 (for Accuracy and GMean).

For the AvgDiffAgeMean CTF, the Scenario 1 results did not have a clear winner between the Base-el and the BL+CTFs feature sets. The Base-el had the smallest average ranks overall for Sensitivity and Accuracy (although the average rank for the latter is tied between the Base-el and BL+CTFs sets), while BL+CTFs won in the Specificity and GMean metrics. The Friedman test p-values were significant for all metrics in Scenario 1 (p-values 0.0072, 0.0005, 0.0015 and 0.0015 for Sensitivity, Specificity, Accuracy and GMean, respectively). In the Nemenyi post-hoc tests, for Sensitivity the significant p-values were obtained when comparing Base-el vs. CTFs-only (0.01254) and BL+CTFs vs. CTFs-only (0.02657). For Specificity, BL+CTFs was significantly superior to both other feature sets, with p-value 0.001 when compared to CTFs-only and 0.0266 when compared to the Base-el. For Accuracy, both the Base-el and BL+CTFs had significant p-values when compared to CTFs-only, with p-values 0.0023 and 0.0126, respectively. Finally, for GMean, only the BL+CTFs vs. CTFs-only comparison had a significant p-value (0.001).

The Scenario 2 results also did not have a clear winning feature set. Unlike most other experiments, the CTFs-in feature set achieved the smallest average ranks in two instances: for Specificity and GMean in Elsa-core datasets. This is an interesting indication that this CTF was able to generate feasible models when replacing the original features, in some cases. It is possible that the fact that Elsa-core datasets had more measurements of the longitudinal features (up to 7, against up to 4 from the Elsa-nurse and TILDA datasets) has increased the predictive power of the AvgDiffAgeMean CTF, for Elsa-core datasets. The

Friedman test p-values for Scenario 2 were 2E−5, 0.1799, 3E−5 and 0.1224 for Sensitivity, Specificity, Accuracy and GMean, respectively; so we only ran the Nemenyi post-hoc tests for Sensitivity and Accuracy. Both Sensitivity and Accuracy had significant p-values when comparing the Base-el-in vs. CTFs-in feature sets (p-values 0.001 for both measures), and BL+CTFs vs. CTFs-in sets (p-values 0.0036 and 0.0014, respectively).

The AvgDiffAgeMean results show that this CTF is also promising for increasing predictive accuracy, but it seems to be more situational than the DiffAgeMean, likely due to how it is constructed as an average of several measurements. It is important to highlight that this CTF achieved more competitive results for the CTFs-in feature sets (by comparison with experiments with other types of CTF), as the Friedman test did not have significant p-values for Specificity and GMean when comparing CTFs-in to the other two feature sets.

Table B.17: AvgDiffAgeMean Sensitivity and Specificity results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	0.680	0.670	**0.683**	0.678	0.674	**0.690**
EN_Arthritis	**0.669**	0.658	0.664	0.594	0.595	**0.598**
EN_Cataract	**0.615**	0.590	0.612	0.723	**0.738**	0.728
EN_Dementia	**0.737**	0.681	0.734	**0.716**	0.696	0.703
EN_Diabetes	**0.843**	0.839	0.842	0.865	0.836	**0.867**
EN_HBP	**0.653**	0.637	0.643	0.747	0.725	**0.748**
EN_Heartattack	**0.698**	0.685	0.694	**0.731**	0.691	0.728
EN_Osteoporosis	**0.655**	0.640	0.645	0.699	0.696	**0.706**
EN_Parkinsons	0.604	**0.612**	0.605	0.636	**0.742**	0.652
EN_Stroke	0.667	0.650	**0.667**	**0.710**	0.670	0.701
EC_Angina	**0.710**	0.692	0.703	0.723	0.751	**0.768**
EC_Arthritis	0.741	**0.757**	0.744	0.717	0.710	**0.718**
EC_Cataract	0.601	**0.635**	0.613	0.675	0.732	**0.734**
EC_Dementia	0.757	0.746	**0.760**	0.727	**0.832**	0.758
EC_Diabetes	0.671	**0.697**	0.674	**0.750**	0.735	**0.750**
EC_HBP	0.625	**0.646**	0.644	0.662	0.669	**0.670**
EC_Heartattack	**0.673**	0.662	0.672	0.689	0.698	**0.703**
EC_Osteoporosis	0.690	0.680	**0.692**	0.635	**0.661**	0.651
EC_Parkinsons	0.685	0.683	**0.696**	0.720	0.640	**0.733**
EC_Stroke	**0.689**	0.683	0.689	0.694	**0.751**	0.734
TI_Angina	**0.748**	0.500	0.747	0.876	0.500	**0.892**
TI_Arthritis	**0.731**	0.401	0.728	**0.652**	0.601	0.644
TI_Cancer	0.542	**0.699**	0.543	**0.595**	0.289	0.586
TI_Cataract	**0.660**	0.598	0.658	0.705	0.381	**0.711**
TI_Diabetes	0.737	0.600	**0.743**	0.795	0.408	**0.813**
TI_HBP	0.678	**0.695**	0.679	0.763	0.293	**0.767**
TI_Heartattack	**0.750**	0.699	0.747	0.863	0.268	**0.873**
TI_Ministroke	0.704	0.201	**0.711**	**0.765**	0.843	0.755
TI_Osteoporosis	**0.670**	0.499	0.658	0.772	0.490	**0.764**
TI_Stroke	0.717	**1.000**	0.710	0.738	0.000	**0.815**
AvgRank E-Nurse	**1.4**	2.8	1.9	2.0	2.5	**1.5**
AvgRank E-Core	2.2	2.2	**1.7**	2.7	2.0	**1.4**
AvgRank TILDA	**1.7**	2.4	1.9	1.7	2.8	**1.5**
AvgRank Overall	**1.7**	2.5	1.8	2.1	2.4	**1.5**

Table B.18: AvgDiffAgeMean Accuracy and GMean results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is bold-faced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	**0.680**	0.670	0.684	**0.679**	0.672	0.687
EN_Arthritis	**0.637**	0.631	0.636	**0.630**	0.626	**0.630**
EN_Cataract	**0.650**	0.639	**0.650**	0.667	0.660	**0.668**
EN_Dementia	**0.736**	0.682	0.734	**0.726**	0.689	0.718
EN_Diabetes	**0.846**	0.839	**0.846**	0.854	0.838	**0.855**
EN_HBP	**0.691**	0.673	0.685	**0.699**	0.680	0.693
EN_Heartattack	**0.700**	0.685	0.696	**0.714**	0.688	0.711
EN_Osteoporosis	**0.659**	0.645	0.651	**0.676**	0.667	0.675
EN_Parkinsons	0.605	**0.614**	0.605	0.620	**0.674**	0.628
EN_Stroke	**0.670**	0.651	0.669	**0.688**	0.660	0.684
EC_Angina	**0.711**	0.694	0.705	0.716	0.721	**0.735**
EC_Arthritis	0.731	**0.738**	0.733	0.729	**0.733**	0.731
EC_Cataract	0.623	**0.664**	0.648	0.637	**0.682**	0.670
EC_Dementia	0.757	0.748	**0.760**	0.742	**0.788**	0.759
EC_Diabetes	0.681	**0.702**	0.684	0.709	**0.716**	0.711
EC_HBP	0.639	**0.655**	0.654	0.643	**0.657**	**0.657**
EC_Heartattack	**0.674**	0.664	**0.674**	0.681	0.680	**0.687**
EC_Osteoporosis	0.686	0.679	**0.689**	0.662	0.670	**0.671**
EC_Parkinsons	0.685	0.682	**0.696**	0.702	0.661	**0.714**
EC_Stroke	0.689	0.687	**0.691**	0.692	**0.716**	0.711
TI_Angina	**0.753**	0.500	**0.753**	0.809	0.500	**0.816**
TI_Arthritis	**0.706**	0.463	0.702	**0.690**	0.491	0.684
TI_Cancer	0.545	**0.677**	0.545	**0.568**	0.450	0.564
TI_Cataract	**0.664**	0.580	0.662	0.682	0.477	**0.684**
TI_Diabetes	0.741	0.587	**0.748**	0.765	0.495	**0.777**
TI_HBP	0.710	0.542	**0.713**	0.719	0.451	**0.722**
TI_Heartattack	**0.755**	0.684	0.752	0.805	0.433	**0.808**
TI_Ministroke	0.705	0.212	**0.712**	**0.734**	0.411	0.733
TI_Osteoporosis	**0.680**	0.498	0.668	**0.719**	0.494	0.709
TI_Stroke	**0.717**	0.989	0.711	0.728	0.000	**0.761**
AvgRank E-Nurse	**1.4**	2.8	1.9	**1.6**	2.8	1.7
AvgRank E-Core	2.3	2.2	**1.6**	2.8	1.7	**1.6**
AvgRank TILDA	**1.6**	2.6	1.8	1.6	3.0	**1.4**
AvgRank Overall	**1.7**	2.5	**1.7**	2.0	2.5	**1.5**

Table B.19: AvgDiffAgeMean Sensitivity and Specificity results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	**0.684**	0.670	0.678	0.702	0.686	**0.705**
EN_Arthritis	**0.671**	0.660	0.668	0.586	**0.594**	0.586
EN_Cataract	**0.620**	0.612	0.613	0.723	0.727	**0.733**
EN_Dementia	0.729	0.716	**0.737**	0.709	**0.723**	0.716
EN_Diabetes	**0.841**	0.834	0.839	**0.866**	0.837	0.859
EN_HBP	**0.651**	0.641	0.649	0.749	0.730	**0.754**
EN_Heartattack	**0.700**	0.688	0.693	0.738	0.713	**0.743**
EN_Osteoporosis	0.649	0.647	**0.652**	0.696	**0.713**	0.697
EN_Parkinsons	0.628	**0.632**	0.612	**0.712**	**0.712**	0.636
EN_Stroke	**0.670**	0.655	0.669	**0.724**	0.691	0.708
EC_Angina	**0.711**	0.698	0.704	0.723	0.737	**0.744**
EC_Arthritis	0.749	**0.755**	0.754	**0.717**	0.715	0.715
EC_Cataract	0.609	**0.633**	0.609	0.717	**0.762**	0.749
EC_Dementia	**0.764**	0.745	0.761	0.770	**0.845**	0.776
EC_Diabetes	0.674	**0.697**	0.680	0.747	0.733	**0.750**
EC_HBP	0.641	**0.641**	0.637	0.662	0.672	**0.678**
EC_Heartattack	0.678	0.667	**0.685**	0.692	**0.735**	0.696
EC_Osteoporosis	0.700	0.678	**0.701**	0.676	**0.713**	0.662
EC_Parkinsons	**0.697**	0.673	0.692	**0.693**	0.667	0.653
EC_Stroke	0.694	0.670	**0.695**	0.721	**0.747**	0.721
TI_Angina	0.748	0.657	**0.749**	**0.916**	0.720	0.900
TI_Arthritis	**0.729**	0.582	0.728	0.646	0.554	**0.658**
TI_Cancer	**0.549**	0.546	0.546	**0.579**	0.572	**0.579**
TI_Cataract	**0.706**	0.686	0.696	0.724	0.701	**0.728**
TI_Diabetes	**0.775**	0.762	0.772	**0.831**	0.745	0.829
TI_HBP	**0.678**	0.552	0.672	**0.765**	0.548	0.757
TI_Heartattack	0.751	0.671	**0.753**	**0.878**	0.717	0.863
TI_Ministroke	**0.712**	0.635	0.711	0.735	**0.608**	0.745
TI_Osteoporosis	**0.677**	0.635	0.675	0.807	0.748	**0.812**
TI_Stroke	**0.728**	0.588	0.722	**0.800**	0.646	0.769
AvgRank E-Nurse	**1.3**	2.8	1.9	2.1	2.2	**1.8**
AvgRank E-Core	**1.9**	2.3	**1.9**	2.4	**1.7**	2.0
AvgRank TILDA	**1.2**	3.0	1.9	**1.5**	3.0	1.6
AvgRank Overall	**1.5**	2.7	1.9	2.0	2.3	**1.8**

Table B.20: AvgDiffAgeMean Accuracy and GMean results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is bold-faced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	**0.684**	0.670	0.679	**0.693**	0.678	0.692
EN_Arthritis	**0.635**	0.632	0.633	**0.627**	0.626	0.625
EN_Cataract	**0.654**	0.650	0.652	**0.670**	0.667	**0.670**
EN_Dementia	0.729	0.716	**0.737**	0.719	0.719	**0.727**
EN_Diabetes	**0.845**	0.834	0.842	**0.854**	0.835	0.849
EN_HBP	0.690	0.676	**0.691**	0.698	0.684	**0.700**
EN_Heartattack	**0.702**	0.689	0.696	**0.719**	0.700	0.718
EN_Osteoporosis	0.654	0.653	**0.656**	0.672	**0.679**	0.674
EN_Parkinsons	0.629	**0.633**	0.613	0.669	**0.671**	0.624
EN_Stroke	**0.674**	0.657	0.671	**0.697**	0.673	0.688
EC_Angina	**0.711**	0.699	0.706	0.717	0.717	**0.724**
EC_Arthritis	0.736	**0.739**	0.738	0.733	**0.735**	0.734
EC_Cataract	0.641	**0.671**	0.651	0.661	**0.694**	0.676
EC_Dementia	**0.764**	0.747	0.761	0.767	**0.793**	0.769
EC_Diabetes	0.684	**0.701**	0.689	0.710	**0.715**	0.714
EC_HBP	0.649	**0.653**	**0.653**	0.651	0.656	**0.657**
EC_Heartattack	0.679	0.671	**0.685**	0.685	**0.700**	0.690
EC_Osteoporosis	**0.698**	0.681	0.697	0.688	**0.695**	0.681
EC_Parkinsons	**0.697**	0.672	0.692	**0.695**	0.670	0.672
EC_Stroke	0.695	0.674	**0.696**	0.707	0.707	**0.708**
TI_Angina	**0.756**	0.660	**0.756**	**0.828**	0.688	0.821
TI_Arthritis	0.703	0.574	**0.707**	0.686	0.568	**0.692**
TI_Cancer	0.550	**0.547**	0.548	**0.564**	0.559	0.562
TI_Cataract	**0.707**	0.688	0.699	**0.715**	0.694	0.712
TI_Diabetes	**0.779**	0.761	0.776	**0.803**	0.754	0.800
TI_HBP	**0.711**	0.551	0.704	**0.720**	0.550	0.713
TI_Heartattack	0.756	0.672	**0.757**	**0.812**	0.694	0.806
TI_Ministroke	**0.712**	0.635	0.711	0.724	0.621	**0.728**
TI_Osteoporosis	**0.689**	0.646	0.688	0.739	0.689	**0.740**
TI_Stroke	**0.728**	0.589	0.723	**0.763**	0.617	0.745
AvgRank E-Nurse	**1.4**	2.8	1.8	**1.6**	2.5	2.0
AvgRank E-Core	2.0	2.3	**1.8**	2.6	**1.6**	1.8
AvgRank TILDA	**1.3**	3.0	1.8	**1.3**	3.0	1.7
AvgRank Overall	**1.6**	2.7	1.8	**1.8**	2.4	**1.8**

B.6 Results for Percentile

The last CTF proposed in this thesis is the most complex one regarding its definition, but it simply represents, for each conceptual feature, where the last value measured for an individual measures up when compared to other individuals of the same age. The highest percentile values indicate that the individual's measurement were the highest among individuals of that age, and vice-versa for the lowest percentile values, which can be interesting trends to consider in classification problems. The Percentile results for Scenario 1 are shown in Tables B.21 (for Sensitivity and Specificity) and B.22 (for Accuracy and GMean). The results for Scenario 2 are shown in Tables B.23 (for Sensitivity and Specificity) and B.24 (for Accuracy and GMean).

The Scenario 1 results for Percentile had the BL+CTFs feature set achieving the best overall average rank for all 4 metrics. The CTFs-only set tied with BL+CTFs for the smallest Sensitivity rank for Elsa-core datasets (1.9), and had the smallest GMean rank for Elsa-core datasets (1.6), this being the first time this feature set had competitive results in Scenario 1. The Friedman test p-values for Scenario 1 were 0.0291, 8E−5, 0.0082 and 0.0004 for Sensitivity, Specificity, Accuracy and GMean, respectively. In the Nemenyi post-hoc tests, for Sensitivity the only pair with a significant p-value was BL+CTFs and CTFs-only, which got 0.0221. For Specificity the significant p-values were 0.0018 for Base-el and CTFs-only, and 0.001 for BL+CTFs and CTFs-only. For Accuracy, again, only BL+CTFs and CTFs-only got a significant result of 0.0068. Same for GMean, with p-value 0.001 in the comparison between BL+CTFs and CTFs-only.

The Scenario 2 results had BL+CTFs still having the smallest average rank values across all 4 metrics, but the Base-el-in approach tied with it for both Accuracy and GMean. The CTFs-in had the smallest average rank for Specificity in Elsa-core datasets, and tied with the Base-el-in for the GMean metric, also in Elsa-core datasets (rank 2.0, considerably close to BL+CTFs' 2.1, so this was almost a three-way tie). The Friedman test p-values for Scenario 2 had significant p-values 0.0092, 0.1715, 0.0273 and 0.0273 for Sensitivity, Specificity Accuracy and GMean, respectively, so we did not run the post-hoc test for Specificity. In the Nemenyi post-hoc tests, the Sensitivity comparisons that got a significant p-value were Base-el-in and CTFs-in, with p-value 0.0377, and BL+CTFs and

CTFs-in, with p-value 0.0152. For the Accuracy and GMean post-hoc tests, none of the pairwise comparisons got significant p-values, so the null hypothesis that the feature sets's performances were different could not be rejected.

The Percentile results also show a promising CTF, especially considering that it managed to get good results by itself (CTFs-only set) in Scenario 1, without the added ineligible features to boost the predictive accuracy of its models. We believe that the Percentile feature could be used by health experts to get additional information when diagnosing a patient, so it is encouraging to see that it had an impact on the predictive accuracy of some models, although it seems situational enough that this impact was not positive across all experiments.

Table B.21: Percentile Sensitivity and Specificity results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	**0.672**	0.653	**0.672**	**0.729**	0.690	**0.729**
EN_Arthritis	**0.671**	0.655	0.664	0.579	0.572	**0.585**
EN_Cataract	0.667	0.644	**0.669**	0.646	0.614	**0.649**
EN_Dementia	0.714	**0.722**	0.721	**0.743**	0.689	**0.743**
EN_Diabetes	**0.844**	0.835	0.843	0.864	**0.851**	0.858
EN_HBP	0.619	**0.629**	0.623	**0.701**	0.685	0.694
EN_Heartattack	**0.692**	0.678	0.687	**0.726**	0.688	0.696
EN_Osteoporosis	**0.647**	0.613	0.646	**0.702**	0.694	0.700
EN_Parkinsons	0.637	0.637	**0.648**	**0.667**	0.591	0.606
EN_Stroke	0.669	0.659	**0.672**	**0.727**	0.667	0.724
EC_Angina	**0.677**	**0.677**	0.676	0.744	0.733	**0.779**
EC_Arthritis	0.696	**0.713**	0.708	**0.705**	0.696	0.697
EC_Cataract	0.564	0.564	**0.577**	0.622	**0.705**	0.691
EC_Dementia	0.763	**0.768**	**0.768**	0.764	**0.783**	**0.783**
EC_Diabetes	0.635	**0.652**	0.640	0.742	0.742	**0.754**
EC_HBP	0.583	**0.612**	0.600	0.648	0.653	**0.664**
EC_Heartattack	0.639	0.650	**0.653**	**0.723**	0.694	0.701
EC_Osteoporosis	**0.660**	0.653	0.658	0.638	**0.673**	0.669
EC_Parkinsons	**0.702**	0.690	0.687	0.667	**0.747**	0.733
EC_Stroke	0.673	0.669	**0.676**	0.714	**0.736**	0.718
TI_Angina	**0.715**	0.672	0.703	**0.820**	0.752	0.812
TI_Arthritis	0.665	0.670	**0.678**	0.629	0.604	**0.645**
TI_Cancer	0.509	**0.534**	0.529	**0.543**	0.487	0.520
TI_Cataract	0.664	**0.712**	0.702	0.688	0.669	**0.713**
TI_Diabetes	0.734	0.684	**0.735**	0.761	0.732	**0.777**
TI_HBP	0.631	0.608	**0.633**	**0.721**	0.640	0.712
TI_Heartattack	0.732	0.700	**0.736**	0.785	0.751	**0.824**
TI_Ministroke	**0.677**	0.654	0.673	0.706	0.647	**0.735**
TI_Osteoporosis	0.626	0.617	**0.637**	0.729	0.657	**0.727**
TI_Stroke	**0.701**	0.648	0.685	**0.662**	0.631	0.646
AvgRank E-Nurse	**1.8**	2.6	1.7	**1.3**	3.0	1.7
AvgRank E-Core	2.3	**1.9**	**1.9**	2.5	1.9	**1.7**
AvgRank TILDA	2.0	2.5	**1.5**	**1.5**	3.0	**1.5**
AvgRank Overall	2.0	2.3	**1.7**	1.8	2.6	**1.6**

Table B.22: Percentile Accuracy and GMean results for the Scenario 1 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Base-el	CTFs-only	BL+CTFs	Base-el	CTFs-only	BL+CTFs
EN_Angina	**0.674**	0.655	**0.674**	**0.700**	0.671	**0.700**
EN_Arthritis	**0.632**	0.620	0.630	**0.623**	0.612	**0.623**
EN_Cataract	0.660	0.634	**0.663**	0.656	0.629	**0.659**
EN_Dementia	0.715	**0.721**	**0.721**	0.729	0.705	**0.732**
EN_Diabetes	**0.847**	0.837	0.845	**0.854**	0.843	0.851
EN_HBP	**0.652**	0.651	**0.652**	**0.659**	0.656	0.658
EN_Heartattack	**0.694**	0.678	0.687	**0.709**	0.683	0.691
EN_Osteoporosis	**0.652**	0.621	0.651	**0.674**	0.652	0.672
EN_Parkinsons	0.637	0.637	**0.647**	**0.652**	0.614	0.627
EN_Stroke	0.672	0.659	**0.675**	0.697	0.663	**0.698**
EC_Angina	**0.680**	0.679	0.679	0.710	0.705	**0.725**
EC_Arthritis	0.700	**0.706**	0.703	0.701	**0.704**	0.702
EC_Cataract	0.581	0.606	**0.611**	0.593	**0.631**	**0.631**
EC_Dementia	0.763	**0.768**	**0.768**	0.764	**0.775**	**0.775**
EC_Diabetes	0.649	**0.663**	0.654	0.686	**0.696**	0.695
EC_HBP	0.608	**0.628**	0.625	0.615	**0.632**	0.631
EC_Heartattack	0.644	0.653	**0.656**	**0.680**	0.672	0.676
EC_Osteoporosis	0.658	0.654	**0.659**	0.649	**0.663**	**0.663**
EC_Parkinsons	**0.702**	0.690	0.687	0.684	**0.718**	0.710
EC_Stroke	0.675	0.673	**0.678**	0.693	**0.702**	0.697
TI_Angina	**0.720**	0.675	0.708	**0.766**	0.711	0.756
TI_Arthritis	0.654	0.650	**0.668**	0.647	0.636	**0.661**
TI_Cancer	0.511	**0.532**	0.528	**0.526**	0.510	0.524
TI_Cataract	0.666	**0.708**	0.703	0.676	0.690	**0.708**
TI_Diabetes	0.736	0.688	**0.737**	0.747	0.708	**0.755**
TI_HBP	**0.665**	0.620	0.663	0.675	0.624	**0.671**
TI_Heartattack	0.734	0.702	**0.739**	0.758	0.725	**0.779**
TI_Ministroke	**0.677**	0.654	0.674	0.691	0.650	**0.703**
TI_Osteoporosis	0.636	0.620	**0.645**	0.676	0.637	**0.681**
TI_Stroke	**0.701**	0.648	0.685	**0.681**	0.639	0.666
AvgRank E-Nurse	1.7	2.8	**1.6**	**1.4**	3.0	1.6
AvgRank E-Core	2.4	1.9	**1.7**	2.7	**1.6**	1.8
AvgRank TILDA	1.8	2.6	**1.6**	1.7	2.9	**1.4**
AvgRank Overall	2.0	2.4	**1.6**	1.9	2.5	**1.6**

Table B.23: Percentile Sensitivity and Specificity results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	0.666	**0.674**	0.673	**0.736**	0.729	**0.736**
EN_Arthritis	**0.671**	0.661	0.665	0.586	**0.590**	0.580
EN_Cataract	0.642	0.630	**0.656**	0.704	**0.723**	0.692
EN_Dementia	0.709	**0.749**	0.721	0.757	0.696	**0.784**
EN_Diabetes	0.843	0.838	**0.845**	0.857	0.858	**0.860**
EN_HBP	0.622	0.625	**0.626**	0.698	0.690	**0.704**
EN_Heartattack	0.690	0.687	**0.699**	**0.756**	0.723	0.723
EN_Osteoporosis	**0.649**	0.634	0.643	0.714	0.703	**0.719**
EN_Parkinsons	0.660	0.666	**0.666**	0.682	**0.697**	0.682
EN_Stroke	0.663	0.672	**0.674**	0.720	0.694	**0.722**
EC_Angina	**0.694**	0.682	0.685	0.775	0.775	**0.786**
EC_Arthritis	**0.727**	0.720	**0.727**	**0.693**	0.692	0.690
EC_Cataract	**0.629**	0.622	0.609	0.752	0.762	**0.763**
EC_Dementia	0.780	0.773	**0.783**	0.826	0.832	**0.839**
EC_Diabetes	0.642	**0.649**	0.643	0.749	**0.763**	0.761
EC_HBP	0.622	**0.632**	**0.632**	0.664	**0.666**	0.661
EC_Heartattack	0.668	0.662	**0.671**	**0.737**	0.732	0.732
EC_Osteoporosis	0.678	0.674	**0.681**	**0.738**	0.730	0.713
EC_Parkinsons	**0.714**	0.714	0.705	0.720	**0.787**	0.760
EC_Stroke	**0.687**	0.678	0.685	0.736	**0.766**	0.745
TI_Angina	**0.719**	0.693	0.708	0.804	0.776	**0.820**
TI_Arthritis	0.680	0.670	**0.683**	0.636	0.610	**0.640**
TI_Cancer	**0.541**	0.539	0.539	**0.536**	0.533	0.533
TI_Cataract	0.717	**0.723**	0.711	0.730	0.703	**0.734**
TI_Diabetes	**0.788**	0.780	0.778	**0.803**	0.784	0.797
TI_HBP	0.635	0.625	**0.640**	**0.721**	0.679	0.715
TI_Heartattack	**0.747**	0.727	0.741	0.795	0.780	**0.800**
TI_Ministroke	**0.679**	0.671	0.673	0.716	0.686	**0.745**
TI_Osteoporosis	0.654	0.653	**0.660**	**0.792**	0.781	0.775
TI_Stroke	**0.702**	0.652	0.693	**0.708**	0.631	0.677
AvgRank E-Nurse	2.3	2.3	**1.5**	2.0	2.3	**1.8**
AvgRank E-Core	**1.8**	2.4	**1.8**	2.3	**1.7**	2.1
AvgRank TILDA	**1.4**	2.7	2.0	**1.5**	2.9	1.7
AvgRank Overall	1.8	2.4	**1.7**	1.9	2.3	**1.8**

Table B.24: Percentile Accuracy and GMean results for the Scenario 2 experiments with Random Forest classifiers. The best result for each row is boldfaced, and the average ranks per type of dataset (ELSA-Nurse, ELSA-Core or TILDA) and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Base-el-in	CTFs-in	BL+CTFs	Base-el-in	CTFs-in	BL+CTFs
EN_Angina	0.669	**0.676**	0.675	0.700	0.701	**0.704**
EN_Arthritis	**0.635**	0.631	0.629	0.627	**0.624**	0.621
EN_Cataract	0.663	0.660	**0.667**	0.673	**0.674**	0.673
EN_Dementia	0.710	**0.747**	0.722	0.733	0.722	**0.752**
EN_Diabetes	0.845	0.841	**0.847**	0.850	0.848	**0.852**
EN_HBP	0.653	0.651	**0.657**	0.659	0.656	**0.664**
EN_Heartattack	0.694	0.689	**0.700**	0.722	0.705	**0.711**
EN_Osteoporosis	**0.655**	0.641	0.650	**0.681**	0.668	0.680
EN_Parkinsons	0.661	0.666	**0.667**	0.671	**0.681**	0.674
EN_Stroke	0.667	0.673	**0.676**	0.691	**0.682**	0.697
EC_Angina	**0.697**	0.685	0.689	**0.734**	0.727	**0.734**
EC_Arthritis	**0.713**	0.709	0.712	**0.710**	0.706	0.708
EC_Cataract	**0.665**	0.664	0.654	**0.688**	**0.688**	0.682
EC_Dementia	0.781	0.774	**0.784**	0.803	0.802	**0.810**
EC_Diabetes	0.656	**0.663**	0.658	0.693	**0.703**	0.699
EC_HBP	0.638	**0.645**	0.643	0.643	**0.649**	0.646
EC_Heartattack	0.672	0.666	**0.674**	**0.702**	0.696	0.701
EC_Osteoporosis	0.683	0.679	**0.684**	**0.707**	0.701	0.697
EC_Parkinsons	0.714	**0.715**	0.706	0.717	**0.749**	0.732
EC_Stroke	**0.690**	0.683	0.689	0.711	**0.721**	0.714
TI_Angina	**0.723**	0.697	0.713	0.760	0.733	**0.762**
TI_Arthritis	0.666	0.651	**0.670**	0.658	0.639	**0.661**
TI_Cancer	**0.541**	0.539	0.538	**0.539**	0.536	0.536
TI_Cataract	0.718	**0.721**	0.713	**0.724**	0.713	0.723
TI_Diabetes	**0.789**	0.781	0.780	**0.795**	0.782	0.788
TI_HBP	0.668	0.645	**0.669**	**0.677**	0.651	**0.677**
TI_Heartattack	**0.749**	0.729	0.743	**0.771**	0.753	0.770
TI_Ministroke	**0.679**	0.671	0.675	0.697	0.678	**0.708**
TI_Osteoporosis	0.667	0.665	**0.671**	**0.720**	0.714	0.716
TI_Stroke	**0.702**	0.652	0.693	**0.705**	0.641	0.685
AvgRank E-Nurse	2.2	2.3	**1.5**	2.0	2.4	**1.7**
AvgRank E-Core	**1.8**	2.3	1.9	**2.0**	**2.0**	2.1
AvgRank TILDA	**1.4**	2.6	2.0	**1.4**	3.0	1.7
AvgRank Overall	**1.8**	2.4	**1.8**	**1.8**	2.4	**1.8**

Appendix C

Constructed Feature Experiments with C4.5 Decision Trees

In Tables C.1 to C.24, we report on the individual CTF experiments results using the C4.5 decision tree classifier, instead of the Random Forest classifier. For each CTF, there are four tables, reporting Sensitivity and Specificity results, then Accuracy and GMean results, for scenarios 1 and 2, in this order. The setup is exactly the same for these experiments, with two scenarios (i.e., with and without ineligible features in the dataset) comparing three feature sets, namely a Baseline set with only the features used to create the CTFs, a CTFs-only set without original features, and a BL+CTFs set combining both original and constructed features.

Overall, in the decision tree experiments the baseline approach outperformed the CTFs-only and BL+CTFs feature sets in the majority of the cases. There is still little difference between the baseline and the proposed approach of adding constructed features to the original feature set (with the latter getting the smallest average ranks in some cases), but it seems that the added CTFs did not have the desired effect in these experiments. This points to a need for further developing CTFs for them to be able to compete on equal grounds with all original features, as in the C4.5 decision tree every single feature has its information gain tested in every node, and it seems like original features tend to be selected more often in this case. Note that, in our RF experiments in Chapter 5, we saw that constructed features were selected among the top-ranked features, and tended to improve predictive accuracy results.

Table C.1: Diff Sensitivity and Specificity results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.623**	0.553	**0.623**	**0.535**	0.531	0.523
EN_Arthritis	0.570	**0.594**	0.563	0.540	0.526	**0.556**
EN_Cataract	0.583	**0.618**	0.591	0.575	0.557	**0.577**
EN_Dementia	0.670	0.561	**0.678**	**0.649**	0.581	0.622
EN_Diabetes	0.809	0.682	**0.813**	**0.797**	0.584	0.789
EN_HBP	0.601	**0.609**	0.604	**0.574**	0.520	**0.574**
EN_Heartattack	0.636	0.624	**0.649**	**0.599**	0.539	0.594
EN_Osteoporosis	0.583	0.519	**0.589**	0.610	**0.630**	0.581
EN_Parkinsons	**0.560**	0.526	0.555	0.500	0.561	**0.636**
EN_Stroke	0.610	**0.657**	0.617	**0.637**	0.496	0.625
EC_Angina	**0.684**	0.608	0.666	0.681	0.533	**0.691**
EC_Arthritis	0.731	0.712	**0.733**	**0.674**	0.618	0.649
EC_Cataract	**0.610**	**0.610**	0.598	**0.547**	0.499	0.544
EC_Dementia	**0.738**	0.729	0.705	**0.708**	0.652	0.696
EC_Diabetes	0.678	0.594	**0.680**	**0.667**	0.545	0.661
EC_HBP	**0.636**	0.626	0.629	**0.596**	0.498	0.570
EC_Heartattack	**0.657**	0.646	0.649	**0.619**	0.517	0.612
EC_Osteoporosis	0.644	**0.656**	0.613	0.607	0.572	**0.611**
EC_Parkinsons	0.620	**0.679**	0.643	**0.787**	0.613	0.707
EC_Stroke	0.665	0.655	**0.669**	**0.616**	0.537	0.590
TI_Angina	**0.725**	0.629	0.719	**0.772**	0.676	0.684
TI_Arthritis	0.598	**0.640**	0.584	**0.595**	0.576	0.591
TI_Cancer	**0.533**	0.532	0.517	0.507	0.523	**0.559**
TI_Cataract	0.673	0.615	**0.675**	0.655	0.519	**0.665**
TI_Diabetes	**0.750**	0.654	**0.750**	**0.777**	0.644	0.748
TI_HBP	**0.634**	0.581	0.618	0.626	**0.640**	0.620
TI_Heartattack	**0.749**	0.675	0.710	**0.761**	0.717	**0.761**
TI_Ministroke	**0.674**	0.603	0.667	**0.647**	0.608	0.578
TI_Osteoporosis	0.661	0.601	**0.655**	0.648	0.567	**0.674**
TI_Stroke	**0.625**	0.615	**0.625**	**0.677**	0.508	0.569
AvgRank E-Nurse	2.2	2.2	**1.7**	**1.6**	2.6	1.9
AvgRank E-Core	**1.7**	2.4	2.0	**1.2**	3.0	1.8
AvgRank TILDA	**1.3**	2.7	2.0	**1.6**	2.6	1.9
AvgRank Overall	**1.7**	2.4	1.9	**1.4**	2.7	1.8

Table C.2: Diff Accuracy and GMean results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.620**	0.552	**0.620**	**0.577**	0.542	0.571
EN_Arthritis	0.557	**0.565**	0.560	0.555	0.559	**0.560**
EN_Cataract	0.581	**0.598**	0.586	0.579	**0.587**	0.584
EN_Dementia	0.669	0.561	**0.677**	**0.659**	0.571	0.649
EN_Diabetes	0.807	0.669	**0.810**	**0.803**	0.631	0.801
EN_HBP	0.590	0.573	**0.592**	0.587	0.563	**0.589**
EN_Heartattack	0.634	0.619	**0.646**	0.617	0.580	**0.621**
EN_Osteoporosis	0.585	0.529	**0.588**	**0.596**	0.572	0.585
EN_Parkinsons	**0.560**	0.526	0.556	0.529	0.543	**0.594**
EN_Stroke	0.611	**0.647**	0.617	**0.623**	0.571	0.621
EC_Angina	**0.684**	0.606	0.667	**0.682**	0.570	0.678
EC_Arthritis	**0.709**	0.675	0.700	**0.702**	0.663	0.690
EC_Cataract	**0.591**	0.577	0.582	**0.578**	0.552	0.570
EC_Dementia	**0.737**	0.727	0.705	**0.723**	0.689	0.700
EC_Diabetes	0.677	0.588	**0.678**	**0.673**	0.569	0.671
EC_HBP	**0.621**	0.576	0.606	**0.616**	0.558	0.599
EC_Heartattack	**0.655**	0.640	0.647	**0.638**	0.578	0.630
EC_Osteoporosis	0.641	**0.649**	0.613	**0.625**	0.613	0.612
EC_Parkinsons	0.622	**0.678**	0.644	**0.699**	0.645	0.674
EC_Stroke	0.662	0.649	**0.664**	**0.640**	0.593	0.628
TI_Angina	**0.727**	0.631	0.717	**0.748**	0.652	0.701
TI_Arthritis	0.597	**0.620**	0.586	0.597	**0.607**	0.587
TI_Cancer	**0.532**	**0.532**	0.520	0.520	0.528	**0.538**
TI_Cataract	0.671	0.607	**0.674**	0.664	0.565	**0.670**
TI_Diabetes	**0.752**	0.654	0.750	**0.763**	0.649	0.749
TI_HBP	**0.631**	0.603	0.618	**0.630**	0.610	0.619
TI_Heartattack	**0.749**	0.677	0.712	**0.755**	0.696	0.735
TI_Ministroke	**0.673**	0.603	0.666	**0.660**	0.605	0.621
TI_Osteoporosis	**0.660**	0.598	0.657	**0.655**	0.584	0.664
TI_Stroke	**0.626**	0.614	0.625	**0.651**	0.559	0.597
AvgRank E-Nurse	2.2	2.4	**1.5**	**1.8**	2.6	1.6
AvgRank E-Core	**1.5**	2.5	2.0	**1.0**	2.9	2.1
AvgRank TILDA	**1.3**	2.7	2.1	**1.5**	2.7	1.8
AvgRank Overall	**1.6**	2.5	1.9	**1.4**	2.7	1.8

Table C.3: Diff Sensitivity and Specificity results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.618**	0.572	0.615	**0.593**	0.570	0.535
EN_Arthritis	0.566	**0.588**	0.558	0.552	**0.579**	0.560
EN_Cataract	**0.615**	0.606	0.600	0.587	**0.658**	0.582
EN_Dementia	0.675	0.658	**0.684**	0.655	**0.703**	0.669
EN_Diabetes	0.810	0.645	**0.812**	**0.797**	0.614	0.779
EN_HBP	**0.604**	0.576	0.604	0.577	0.556	**0.590**
EN_Heartattack	0.652	0.601	**0.653**	0.611	**0.613**	0.589
EN_Osteoporosis	**0.605**	0.589	0.593	0.610	**0.641**	0.592
EN_Parkinsons	0.622	**0.623**	0.615	0.439	**0.667**	0.576
EN_Stroke	0.610	0.614	**0.622**	0.613	0.582	**0.618**
EC_Angina	0.667	0.604	**0.677**	**0.712**	0.607	0.691
EC_Arthritis	0.729	0.697	**0.737**	**0.660**	0.646	0.647
EC_Cataract	0.638	0.645	**0.651**	**0.699**	0.664	0.634
EC_Dementia	0.732	**0.752**	0.723	**0.826**	0.739	0.758
EC_Diabetes	**0.682**	0.580	0.669	0.665	0.539	**0.666**
EC_HBP	**0.629**	0.591	0.610	**0.614**	0.545	0.607
EC_Heartattack	**0.655**	0.616	0.646	**0.635**	0.594	0.626
EC_Osteoporosis	0.665	0.673	**0.680**	**0.682**	0.620	0.669
EC_Parkinsons	**0.678**	0.621	0.674	**0.747**	0.707	0.707
EC_Stroke	**0.658**	0.624	0.651	0.618	**0.624**	0.607
TI_Angina	0.731	0.658	**0.732**	**0.768**	0.636	0.716
TI_Arthritis	0.589	0.583	**0.595**	**0.592**	0.588	0.590
TI_Cancer	**0.534**	0.530	0.523	**0.533**	0.500	0.497
TI_Cataract	0.612	**0.640**	0.602	0.603	**0.644**	0.565
TI_Diabetes	0.699	**0.739**	0.701	0.701	**0.761**	0.699
TI_HBP	**0.642**	0.594	0.621	0.623	0.579	**0.638**
TI_Heartattack	**0.719**	0.676	0.710	0.737	0.659	**0.766**
TI_Ministroke	**0.667**	0.616	0.661	**0.676**	0.510	0.627
TI_Osteoporosis	0.628	**0.639**	0.619	0.635	**0.646**	0.632
TI_Stroke	**0.631**	0.616	0.622	**0.662**	0.492	0.508
AvgRank E-Nurse	**1.8**	2.4	1.9	2.1	**1.7**	2.2
AvgRank E-Core	**1.7**	2.6	**1.7**	**1.2**	2.7	2.2
AvgRank TILDA	**1.6**	2.3	2.1	**1.5**	2.3	2.2
AvgRank Overall	**1.7**	2.4	1.9	**1.6**	2.2	2.2

Table C.4: Diff Accuracy and GMean results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.617**	0.572	0.612	**0.605**	0.571	0.573
EN_Arthritis	0.560	**0.584**	0.559	0.559	0.584	**0.559**
EN_Cataract	0.606	**0.623**	0.594	0.601	**0.631**	0.591
EN_Dementia	0.674	0.659	**0.684**	0.665	**0.680**	0.676
EN_Diabetes	**0.808**	0.641	0.808	**0.803**	0.629	0.795
EN_HBP	0.593	0.568	**0.598**	0.590	0.566	**0.597**
EN_Heartattack	**0.650**	0.602	0.650	**0.631**	0.607	0.620
EN_Osteoporosis	**0.605**	0.594	0.593	0.607	**0.614**	0.593
EN_Parkinsons	0.621	**0.623**	0.614	0.523	**0.644**	0.595
EN_Stroke	0.611	0.612	**0.622**	0.612	0.598	**0.620**
EC_Angina	0.668	0.604	**0.678**	**0.689**	0.605	0.684
EC_Arthritis	**0.702**	0.677	0.701	**0.694**	0.671	0.691
EC_Cataract	**0.656**	0.651	0.646	**0.668**	0.654	0.643
EC_Dementia	0.734	**0.752**	0.724	**0.778**	0.746	0.740
EC_Diabetes	**0.680**	0.575	0.669	**0.673**	0.559	0.668
EC_HBP	**0.623**	0.573	0.609	**0.622**	0.568	0.608
EC_Heartattack	**0.654**	0.615	0.645	**0.645**	0.605	0.636
EC_Osteoporosis	0.667	0.668	**0.679**	0.673	0.646	**0.675**
EC_Parkinsons	**0.679**	0.622	0.674	**0.712**	0.663	0.690
EC_Stroke	**0.656**	0.624	0.649	**0.638**	0.624	0.629
TI_Angina	**0.732**	0.657	0.731	**0.749**	0.647	0.724
TI_Arthritis	**0.590**	0.584	0.594	**0.591**	0.585	0.593
TI_Cancer	**0.534**	0.528	0.522	**0.534**	0.515	0.510
TI_Cataract	0.611	**0.640**	0.599	0.607	**0.642**	0.583
TI_Diabetes	0.699	**0.741**	0.701	0.700	**0.750**	0.700
TI_HBP	**0.635**	0.588	0.627	**0.633**	0.586	0.629
TI_Heartattack	**0.720**	0.675	0.712	0.728	0.667	**0.738**
TI_Ministroke	**0.667**	0.615	0.660	**0.672**	0.561	0.644
TI_Osteoporosis	0.629	**0.640**	0.620	0.632	**0.643**	0.625
TI_Stroke	**0.631**	0.615	0.620	**0.646**	0.551	0.562
AvgRank E-Nurse	**1.8**	2.2	2.0	**2.0**	**2.0**	2.1
AvgRank E-Core	**1.4**	2.6	2.0	**1.1**	2.8	2.1
AvgRank TILDA	**1.5**	2.3	2.2	**1.6**	2.3	2.2
AvgRank Overall	**1.6**	2.4	2.1	**1.5**	2.4	2.1

Table C.5: Ratio Sensitivity and Specificity results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.623**	0.474	0.619	0.535	**0.578**	0.570
EN_Arthritis	**0.570**	0.564	0.550	0.540	**0.570**	0.566
EN_Cataract	0.583	0.520	**0.595**	0.575	**0.658**	0.571
EN_Dementia	**0.670**	0.444	0.663	0.649	**0.736**	0.615
EN_Diabetes	0.809	0.645	**0.812**	0.797	0.651	**0.808**
EN_HBP	0.601	**0.642**	0.606	0.574	0.529	**0.578**
EN_Heartattack	0.636	**0.671**	0.636	0.599	0.489	**0.626**
EN_Osteoporosis	0.583	0.468	**0.590**	0.610	**0.688**	0.587
EN_Parkinsons	0.560	**0.704**	0.566	0.500	0.318	**0.606**
EN_Stroke	0.610	0.578	**0.622**	**0.637**	0.553	0.618
EC_Angina	**0.684**	0.748	0.679	0.681	0.411	**0.684**
EC_Arthritis	**0.731**	0.691	0.727	0.674	**0.680**	0.668
EC_Cataract	**0.610**	0.520	0.601	0.547	**0.644**	0.553
EC_Dementia	0.738	**0.771**	0.722	0.708	0.602	**0.733**
EC_Diabetes	**0.678**	0.540	0.680	0.667	0.635	**0.668**
EC_HBP	**0.636**	0.545	0.632	0.596	**0.602**	**0.602**
EC_Heartattack	0.657	**0.680**	0.652	0.619	0.438	**0.626**
EC_Osteoporosis	0.644	**0.666**	0.632	**0.607**	0.576	0.606
EC_Parkinsons	0.620	**0.693**	0.624	**0.787**	0.360	0.773
EC_Stroke	**0.665**	0.604	**0.665**	0.616	**0.627**	0.616
TI_Angina	**0.725**	0.627	0.706	**0.772**	0.656	0.696
TI_Arthritis	0.598	**0.645**	0.589	**0.595**	0.567	0.589
TI_Cancer	0.533	**0.539**	0.515	0.507	0.507	**0.543**
TI_Cataract	0.673	0.603	**0.675**	0.655	0.542	**0.661**
TI_Diabetes	**0.750**	0.649	0.737	**0.777**	0.592	0.756
TI_HBP	**0.634**	0.583	0.622	0.626	**0.638**	0.601
TI_Heartattack	**0.749**	0.653	0.706	**0.761**	0.732	0.746
TI_Ministroke	**0.674**	0.608	0.661	**0.647**	0.539	0.588
TI_Osteoporosis	**0.661**	0.580	0.648	0.648	0.628	**0.681**
TI_Stroke	**0.625**	0.594	0.616	**0.677**	0.538	0.615
AvgRank E-Nurse	2.0	2.3	**1.8**	2.1	2.0	**1.9**
AvgRank E-Core	**1.8**	2.0	2.3	2.1	2.3	**1.7**
AvgRank TILDA	**1.3**	2.6	2.1	**1.5**	2.8	1.8
AvgRank Overall	**1.7**	2.3	2.0	1.9	2.3	**1.8**

Table C.6: Ratio Accuracy and GMean results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.620**	0.477	0.617	0.577	0.523	**0.594**
EN_Arthritis	0.557	**0.566**	0.557	0.555	**0.567**	0.558
EN_Cataract	0.581	0.565	**0.587**	0.579	**0.585**	0.583
EN_Dementia	**0.669**	0.450	0.662	**0.659**	0.572	0.638
EN_Diabetes	0.807	0.646	**0.811**	0.803	0.648	**0.810**
EN_HBP	0.590	**0.597**	0.595	0.587	0.583	**0.592**
EN_Heartattack	0.634	**0.661**	0.635	0.617	0.573	**0.631**
EN_Osteoporosis	0.585	0.488	**0.590**	**0.596**	0.567	0.588
EN_Parkinsons	0.560	**0.701**	0.566	0.529	0.473	**0.586**
EN_Stroke	0.611	0.577	**0.622**	**0.623**	0.566	0.620
EC_Angina	0.684	**0.737**	0.680	**0.682**	0.554	**0.682**
EC_Arthritis	**0.709**	0.687	0.703	**0.702**	0.686	0.697
EC_Cataract	**0.591**	0.556	0.587	0.578	**0.579**	0.577
EC_Dementia	0.737	**0.768**	0.722	0.723	0.682	**0.727**
EC_Diabetes	0.677	0.552	**0.678**	0.673	0.585	**0.674**
EC_HBP	**0.621**	0.567	0.620	0.616	0.573	**0.617**
EC_Heartattack	0.655	**0.667**	0.651	0.638	0.545	**0.639**
EC_Osteoporosis	0.641	**0.659**	0.630	**0.625**	0.620	0.619
EC_Parkinsons	0.622	**0.690**	0.625	**0.699**	0.500	0.694
EC_Stroke	0.662	0.605	**0.663**	**0.640**	0.615	**0.640**
TI_Angina	**0.727**	0.629	0.705	**0.748**	0.642	0.701
TI_Arthritis	0.597	**0.621**	0.589	0.597	**0.605**	0.589
TI_Cancer	0.532	**0.538**	0.517	0.520	0.523	**0.529**
TI_Cataract	0.671	0.598	**0.674**	0.664	0.572	**0.668**
TI_Diabetes	**0.752**	0.645	0.738	**0.763**	0.620	0.746
TI_HBP	**0.631**	0.604	0.614	**0.630**	0.610	0.612
TI_Heartattack	**0.749**	0.656	0.707	**0.755**	0.691	0.726
TI_Ministroke	**0.673**	0.606	0.659	**0.660**	0.572	0.623
TI_Osteoporosis	**0.660**	0.585	0.651	0.655	0.604	**0.664**
TI_Stroke	**0.626**	0.593	0.616	**0.651**	0.565	0.616
AvgRank E-Nurse	2.2	**2.2**	**1.7**	1.9	2.6	**1.5**
AvgRank E-Core	1.8	**2.0**	2.2	1.6	2.7	**1.7**
AvgRank TILDA	**1.3**	2.6	2.1	**1.5**	2.7	1.8
AvgRank Overall	**1.8**	2.3	2.0	**1.7**	2.7	**1.7**

Table C.7: Ratio Sensitivity and Specificity results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.618**	0.563	0.608	0.593	**0.628**	0.574
EN_Arthritis	0.566	**0.645**	0.571	0.552	0.545	**0.562**
EN_Cataract	**0.615**	0.609	0.610	0.587	**0.714**	0.587
EN_Dementia	**0.675**	0.670	0.674	0.655	**0.689**	0.676
EN_Diabetes	**0.810**	0.658	0.809	0.797	0.611	**0.798**
EN_HBP	0.604	0.584	**0.607**	0.577	**0.596**	0.582
EN_Heartattack	**0.652**	0.607	0.645	0.611	0.584	**0.628**
EN_Osteoporosis	**0.605**	0.555	0.602	0.610	**0.749**	0.612
EN_Parkinsons	**0.622**	0.608	0.617	0.439	**0.621**	0.530
EN_Stroke	0.610	0.589	**0.633**	0.613	**0.634**	0.580
EC_Angina	0.667	0.608	**0.687**	**0.712**	0.596	0.677
EC_Arthritis	**0.729**	0.716	0.725	**0.660**	0.623	0.652
EC_Cataract	0.638	**0.647**	0.638	0.699	**0.711**	0.682
EC_Dementia	0.732	**0.733**	0.724	0.826	**0.839**	0.820
EC_Diabetes	**0.682**	0.577	0.678	**0.665**	0.519	0.661
EC_HBP	**0.629**	0.597	0.625	**0.614**	0.539	0.613
EC_Heartattack	0.655	0.587	**0.657**	0.635	**0.676**	0.608
EC_Osteoporosis	0.665	0.630	**0.670**	0.682	**0.689**	0.677
EC_Parkinsons	**0.678**	0.569	0.676	**0.747**	0.693	0.720
EC_Stroke	0.658	0.591	**0.661**	0.618	**0.688**	0.596
TI_Angina	**0.731**	0.663	0.713	**0.768**	0.616	0.752
TI_Arthritis	0.589	**0.611**	0.587	0.592	0.573	**0.593**
TI_Cancer	0.534	**0.521**	0.520	**0.533**	0.507	0.490
TI_Cataract	0.612	**0.650**	0.600	0.603	**0.651**	0.590
TI_Diabetes	0.699	**0.733**	0.705	0.701	**0.761**	0.694
TI_HBP	**0.642**	0.588	0.625	0.623	0.582	**0.624**
TI_Heartattack	0.719	0.668	**0.726**	0.737	0.683	**0.741**
TI_Ministroke	**0.667**	0.612	0.652	**0.676**	0.500	0.618
TI_Osteoporosis	0.628	**0.660**	0.622	0.635	0.639	**0.648**
TI_Stroke	**0.631**	0.597	0.621	**0.662**	0.569	0.554
AvgRank E-Nurse	**1.4**	2.8	1.8	2.5	**1.6**	2.0
AvgRank E-Core	**1.7**	2.6	1.8	**1.5**	2.0	2.5
AvgRank TILDA	**1.6**	2.1	2.3	**1.7**	2.3	2.0
AvgRank Overall	**1.6**	2.5	2.0	**1.9**	2.0	2.2

Table C.8: Ratio Accuracy and GMean results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.617**	0.565	0.607	**0.605**	0.594	0.591
EN_Arthritis	0.560	**0.602**	0.567	0.559	**0.593**	0.567
EN_Cataract	0.606	**0.644**	0.603	0.601	**0.660**	0.599
EN_Dementia	**0.674**	0.670	0.674	0.665	**0.680**	0.675
EN_Diabetes	0.808	0.651	**0.808**	0.803	0.634	**0.804**
EN_HBP	0.593	0.589	**0.597**	0.590	0.590	**0.594**
EN_Heartattack	**0.650**	0.605	0.644	0.631	0.595	**0.636**
EN_Osteoporosis	**0.605**	0.573	0.603	0.607	**0.645**	0.607
EN_Parkinsons	**0.621**	0.608	0.616	0.523	**0.614**	0.572
EN_Stroke	0.611	0.591	**0.629**	**0.612**	0.611	0.605
EC_Angina	0.668	0.607	**0.686**	**0.689**	0.602	0.682
EC_Arthritis	**0.702**	0.679	0.696	**0.694**	0.668	0.688
EC_Cataract	0.656	**0.666**	0.651	0.668	**0.678**	0.660
EC_Dementia	0.734	**0.735**	0.726	0.778	**0.784**	0.770
EC_Diabetes	**0.680**	0.569	0.676	**0.673**	0.547	0.670
EC_HBP	**0.623**	0.575	0.621	**0.622**	0.568	0.619
EC_Heartattack	0.654	0.592	**0.654**	**0.645**	0.630	0.632
EC_Osteoporosis	0.667	0.635	**0.671**	0.673	0.659	**0.674**
EC_Parkinsons	**0.679**	0.570	0.676	**0.712**	0.628	0.697
EC_Stroke	0.656	0.596	**0.658**	0.638	**0.638**	0.628
TI_Angina	**0.732**	0.661	0.714	**0.749**	0.639	0.732
TI_Arthritis	0.590	**0.599**	0.589	0.591	**0.592**	0.590
TI_Cancer	**0.534**	0.521	0.519	**0.534**	0.514	0.505
TI_Cataract	0.611	**0.650**	0.599	0.607	**0.650**	0.595
TI_Diabetes	0.699	**0.735**	0.704	0.700	**0.747**	0.699
TI_HBP	**0.635**	0.586	0.625	**0.633**	0.585	0.625
TI_Heartattack	0.720	0.669	**0.727**	0.728	0.676	**0.734**
TI_Ministroke	**0.667**	0.610	0.652	**0.672**	0.553	0.635
TI_Osteoporosis	0.629	**0.658**	0.624	0.632	**0.649**	0.635
TI_Stroke	**0.631**	0.597	0.620	**0.646**	0.583	0.586
AvgRank E-Nurse	**1.6**	2.6	1.8	2.2	**1.8**	2.1
AvgRank E-Core	**1.6**	2.6	1.9	**1.4**	2.5	2.2
AvgRank TILDA	**1.6**	2.1	2.3	**1.6**	2.1	2.3
AvgRank Overall	**1.6**	2.4	2.0	**1.7**	2.1	2.2

Table C.9: Monotonicity Sensitivity and Specificity results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	0.611	0.571	**0.618**	0.570	0.508	**0.628**
EN_Arthritis	**0.558**	0.561	0.557	0.555	0.538	**0.566**
EN_Cataract	0.594	0.586	**0.604**	0.573	**0.647**	0.593
EN_Dementia	0.643	0.663	**0.684**	0.655	0.649	**0.662**
EN_Diabetes	**0.812**	0.641	0.808	0.792	0.628	**0.795**
EN_HBP	**0.624**	0.561	0.623	**0.615**	0.543	0.603
EN_Heartattack	**0.643**	0.601	**0.643**	0.631	0.554	**0.641**
EN_Osteoporosis	0.598	0.547	**0.609**	**0.622**	0.555	0.616
EN_Parkinsons	**0.587**	0.585	0.580	0.515	0.545	**0.636**
EN_Stroke	**0.630**	0.627	0.618	0.594	0.601	**0.608**
EC_Angina	0.688	0.668	**0.689**	0.660	0.544	**0.681**
EC_Arthritis	**0.724**	0.692	0.721	**0.679**	0.569	0.674
EC_Cataract	**0.599**	0.612	0.598	0.571	0.523	**0.572**
EC_Dementia	0.684	0.715	**0.723**	**0.714**	0.503	0.658
EC_Diabetes	0.672	0.636	**0.675**	**0.684**	0.551	0.681
EC_HBP	0.621	**0.637**	0.613	**0.590**	0.516	0.584
EC_Heartattack	0.636	**0.645**	0.625	0.576	0.501	**0.610**
EC_Osteoporosis	**0.651**	0.646	0.630	0.569	0.521	**0.592**
EC_Parkinsons	0.642	**0.677**	0.653	0.573	0.560	**0.600**
EC_Stroke	**0.665**	0.648	0.658	0.600	0.535	**0.611**
TI_Angina	**0.745**	0.621	0.744	0.732	0.500	**0.772**
TI_Arthritis	0.603	0.598	**0.626**	**0.614**	0.533	0.613
TI_Cancer	**0.533**	0.530	0.528	**0.543**	0.477	**0.543**
TI_Cataract	**0.691**	0.584	0.681	0.628	0.533	**0.651**
TI_Diabetes	**0.768**	0.539	**0.768**	0.758	0.558	**0.761**
TI_HBP	**0.652**	0.553	0.649	**0.640**	0.515	0.634
TI_Heartattack	**0.760**	0.547	0.751	**0.751**	0.459	**0.751**
TI_Ministroke	0.684	0.607	**0.691**	0.676	0.471	**0.725**
TI_Osteoporosis	**0.671**	0.567	0.666	**0.650**	0.516	0.648
TI_Stroke	**0.687**	0.612	0.630	0.646	0.523	**0.662**
AvgRank E-Nurse	**1.7**	2.5	1.9	2.1	2.6	**1.3**
AvgRank E-Core	**1.9**	2.0	2.1	1.6	3.0	**1.4**
AvgRank TILDA	**1.3**	2.9	1.9	1.6	3.0	**1.4**
AvgRank Overall	**1.6**	2.5	1.9	1.8	2.9	**1.4**

Table C.10: Monotonicity Accuracy and GMean results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	0.609	0.569	**0.618**	0.590	0.538	**0.623**
EN_Arthritis	0.557	0.551	**0.561**	0.557	0.550	**0.562**
EN_Cataract	0.587	**0.606**	0.600	0.583	**0.615**	0.598
EN_Dementia	0.644	0.663	**0.684**	0.649	0.656	**0.673**
EN_Diabetes	**0.809**	0.639	0.806	**0.802**	0.635	0.801
EN_HBP	**0.620**	0.554	0.615	**0.619**	0.552	0.613
EN_Heartattack	**0.643**	0.598	**0.643**	0.637	0.577	**0.642**
EN_Osteoporosis	0.601	0.548	**0.610**	0.610	0.551	**0.613**
EN_Parkinsons	**0.586**	0.585	0.580	0.550	0.565	**0.607**
EN_Stroke	**0.628**	0.625	0.618	0.612	**0.614**	0.613
EC_Angina	0.687	0.664	**0.689**	0.674	0.603	**0.685**
EC_Arthritis	**0.706**	0.643	0.702	**0.701**	0.627	0.697
EC_Cataract	**0.591**	0.585	0.590	**0.585**	0.565	**0.585**
EC_Dementia	0.684	0.711	**0.722**	**0.699**	0.600	0.690
EC_Diabetes	0.673	0.625	**0.676**	**0.678**	0.592	**0.678**
EC_HBP	**0.609**	0.590	0.602	**0.605**	0.573	0.598
EC_Heartattack	0.633	**0.637**	0.624	0.605	0.568	**0.617**
EC_Osteoporosis	**0.644**	0.636	0.627	0.608	0.580	**0.610**
EC_Parkinsons	0.641	**0.676**	0.653	0.606	0.616	**0.626**
EC_Stroke	**0.662**	0.642	0.656	0.632	0.589	**0.634**
TI_Angina	0.744	0.616	**0.745**	0.738	0.557	**0.758**
TI_Arthritis	0.606	0.578	**0.622**	0.608	0.565	**0.619**
TI_Cancer	**0.533**	0.527	0.529	**0.538**	0.503	0.535
TI_Cataract	**0.686**	0.579	0.678	0.659	0.558	**0.666**
TI_Diabetes	**0.768**	0.540	0.767	0.763	0.549	**0.764**
TI_HBP	**0.648**	0.539	0.643	**0.646**	0.534	0.641
TI_Heartattack	**0.760**	0.544	0.751	**0.756**	0.501	0.751
TI_Ministroke	0.684	0.604	**0.691**	0.680	0.534	**0.708**
TI_Osteoporosis	**0.669**	0.563	0.664	**0.661**	0.541	0.657
TI_Stroke	**0.687**	0.611	0.630	**0.666**	0.566	0.646
AvgRank E-Nurse	**1.8**	2.5	**1.8**	2.2	2.4	**1.4**
AvgRank E-Core	**1.7**	2.4	1.9	1.7	2.9	**1.4**
AvgRank TILDA	**1.3**	3.0	1.7	**1.5**	3.0	**1.5**
AvgRank Overall	**1.6**	2.6	1.8	1.8	2.8	**1.4**

Table C.11: Monotonicity Sensitivity and Specificity results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	0.616	0.567	**0.619**	0.593	0.539	**0.636**
EN_Arthritis	0.570	**0.575**	0.565	0.555	**0.566**	0.558
EN_Cataract	0.591	**0.635**	0.610	0.584	**0.659**	0.604
EN_Dementia	**0.675**	0.671	0.672	0.628	**0.703**	0.642
EN_Diabetes	**0.809**	0.644	0.805	**0.797**	0.636	0.791
EN_HBP	**0.624**	0.576	0.620	**0.618**	0.540	0.604
EN_Heartattack	**0.643**	0.619	0.632	0.633	0.611	**0.651**
EN_Osteoporosis	0.614	**0.629**	0.620	0.593	**0.641**	0.590
EN_Parkinsons	**0.631**	0.618	0.628	0.500	**0.561**	0.500
EN_Stroke	0.615	**0.641**	0.628	0.591	**0.608**	0.601
EC_Angina	**0.669**	0.627	0.668	**0.681**	0.663	0.667
EC_Arthritis	**0.727**	0.655	0.709	0.677	0.617	**0.683**
EC_Cataract	**0.647**	0.631	0.642	0.653	**0.721**	0.626
EC_Dementia	0.742	**0.765**	0.733	**0.758**	0.733	0.714
EC_Diabetes	**0.678**	0.605	0.664	**0.676**	0.558	0.671
EC_HBP	0.621	0.599	**0.622**	0.582	**0.587**	**0.587**
EC_Heartattack	**0.637**	0.622	0.636	0.626	0.610	**0.635**
EC_Osteoporosis	**0.687**	0.615	0.680	0.606	**0.745**	0.617
EC_Parkinsons	**0.664**	0.599	0.616	0.547	**0.667**	0.613
EC_Stroke	0.638	0.607	**0.651**	0.616	**0.657**	0.585
TI_Angina	**0.751**	0.660	0.732	0.724	0.592	**0.800**
TI_Arthritis	0.614	0.610	**0.626**	0.598	0.557	**0.617**
TI_Cancer	0.547	**0.569**	0.537	**0.572**	0.507	0.543
TI_Cataract	0.617	**0.680**	0.596	0.619	**0.657**	0.594
TI_Diabetes	0.719	**0.744**	0.708	**0.730**	0.706	0.709
TI_HBP	**0.653**	0.566	0.644	**0.646**	0.538	0.644
TI_Heartattack	**0.742**	0.637	0.726	0.688	0.688	**0.737**
TI_Ministroke	**0.693**	0.583	0.692	**0.706**	0.637	**0.706**
TI_Osteoporosis	**0.667**	0.656	0.641	0.624	**0.689**	0.621
TI_Stroke	**0.682**	0.656	0.639	**0.631**	0.492	0.585
AvgRank E-Nurse	**1.8**	2.2	2.0	2.3	**1.8**	2.0
AvgRank E-Core	**1.3**	2.8	1.9	**2.0**	**2.0**	2.1
AvgRank TILDA	**1.4**	2.2	2.4	**1.6**	2.6	1.9
AvgRank Overall	**1.5**	2.4	2.1	**2.0**	2.1	**2.0**

Table C.12: Monotonicity Accuracy and GMean results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	0.615	0.566	**0.620**	0.604	0.553	**0.627**
EN_Arthritis	0.563	**0.571**	0.562	0.562	**0.570**	0.561
EN_Cataract	0.589	**0.643**	0.608	0.587	**0.647**	0.607
EN_Dementia	**0.674**	0.672	0.672	0.651	**0.687**	0.657
EN_Diabetes	**0.808**	0.643	0.803	**0.803**	0.640	0.798
EN_HBP	**0.622**	0.562	0.613	**0.621**	0.558	0.612
EN_Heartattack	**0.642**	0.619	0.633	0.638	0.615	**0.641**
EN_Osteoporosis	0.612	**0.630**	0.617	0.604	**0.635**	0.605
EN_Parkinsons	**0.630**	0.618	0.626	0.562	**0.589**	0.560
EN_Stroke	0.614	**0.639**	0.627	0.603	**0.624**	0.615
EC_Angina	**0.670**	0.628	0.668	**0.675**	0.645	0.668
EC_Arthritis	**0.707**	0.640	0.699	**0.701**	0.636	0.696
EC_Cataract	0.649	**0.658**	0.637	0.650	**0.674**	0.634
EC_Dementia	0.742	**0.764**	0.732	**0.750**	0.749	0.723
EC_Diabetes	**0.677**	0.599	0.665	**0.677**	0.581	0.668
EC_HBP	0.606	0.594	**0.608**	0.601	0.593	**0.604**
EC_Heartattack	**0.637**	0.622	0.636	0.632	0.616	**0.636**
EC_Osteoporosis	**0.680**	0.626	0.675	0.645	**0.677**	0.648
EC_Parkinsons	**0.663**	0.599	0.616	0.603	**0.632**	0.614
EC_Stroke	0.637	0.610	**0.647**	0.627	**0.632**	0.617
TI_Angina	**0.750**	0.657	0.735	0.737	0.625	**0.765**
TI_Arthritis	0.609	0.594	**0.623**	0.606	0.583	**0.621**
TI_Cancer	0.548	**0.565**	0.538	**0.559**	0.537	0.540
TI_Cataract	0.617	**0.678**	0.596	0.618	**0.669**	0.595
TI_Diabetes	0.719	**0.741**	0.708	0.724	**0.725**	0.709
TI_HBP	**0.651**	0.556	0.644	**0.650**	0.552	0.644
TI_Heartattack	**0.740**	0.639	0.727	0.714	0.662	**0.731**
TI_Ministroke	**0.694**	0.584	0.692	**0.700**	0.609	0.699
TI_Osteoporosis	**0.663**	0.659	0.639	**0.645**	0.672	0.631
TI_Stroke	**0.682**	0.655	0.638	**0.656**	0.568	0.611
AvgRank E-Nurse	**1.8**	2.2	2.1	2.2	**1.8**	2.0
AvgRank E-Core	**1.4**	2.6	2.0	**1.8**	2.1	2.1
AvgRank TILDA	**1.4**	2.2	2.4	**1.6**	2.4	2.0
AvgRank Overall	**1.5**	2.3	2.2	**1.9**	2.1	2.0

Table C.13: DiffAgeMean Sensitivity and Specificity results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.612**	0.590	0.593	0.550	0.547	**0.601**
EN_Arthritis	0.627	**0.660**	0.580	0.530	0.482	**0.539**
EN_Cataract	**0.607**	0.600	0.583	0.565	0.530	**0.584**
EN_Dementia	**0.678**	0.596	0.666	**0.696**	0.554	0.649
EN_Diabetes	0.801	0.799	**0.807**	**0.813**	0.768	0.790
EN_HBP	0.589	**0.612**	0.576	0.571	0.581	**0.589**
EN_Heartattack	0.625	**0.659**	0.632	0.581	**0.626**	0.623
EN_Osteoporosis	**0.607**	0.568	0.590	0.583	**0.604**	0.580
EN_Parkinsons	0.561	0.552	**0.590**	**0.515**	0.455	0.470
EN_Stroke	**0.640**	0.626	0.625	0.629	0.587	**0.646**
EC_Angina	0.683	0.669	**0.694**	**0.730**	0.670	0.649
EC_Arthritis	**0.745**	0.703	0.727	**0.662**	0.635	0.648
EC_Cataract	0.594	**0.640**	0.635	0.578	0.633	**0.690**
EC_Dementia	0.742	0.727	**0.745**	0.745	**0.776**	0.752
EC_Diabetes	0.662	0.647	**0.655**	0.694	0.635	**0.706**
EC_HBP	**0.637**	0.606	0.620	**0.625**	0.585	0.592
EC_Heartattack	**0.669**	0.618	0.619	0.628	**0.639**	0.617
EC_Osteoporosis	**0.657**	0.628	0.626	**0.641**	0.559	0.597
EC_Parkinsons	0.621	**0.699**	0.660	**0.813**	0.587	0.773
EC_Stroke	0.644	0.653	**0.671**	**0.679**	0.644	0.620
TI_Angina	**0.700**	0.592	0.698	**0.716**	0.564	0.696
TI_Arthritis	0.594	**0.626**	0.603	0.600	0.507	**0.614**
TI_Cancer	0.521	**0.604**	0.522	**0.507**	0.474	0.493
TI_Cataract	**0.671**	0.656	0.654	0.644	0.552	**0.646**
TI_Diabetes	**0.770**	0.597	0.738	**0.740**	0.514	**0.740**
TI_HBP	0.611	0.544	**0.614**	**0.606**	0.526	0.602
TI_Heartattack	**0.726**	0.581	0.702	**0.746**	0.517	0.741
TI_Ministroke	**0.655**	0.583	0.616	0.657	0.588	**0.686**
TI_Osteoporosis	0.661	0.563	**0.669**	**0.656**	0.525	0.626
TI_Stroke	**0.622**	0.544	0.597	**0.662**	0.523	0.600
AvgRank E-Nurse	**1.6**	2.2	2.2	1.9	2.5	**1.6**
AvgRank E-Core	**1.8**	2.4	**1.8**	**1.6**	2.3	2.1
AvgRank TILDA	**1.6**	2.5	1.9	**1.4**	3.0	1.7
AvgRank Overall	**1.7**	2.4	2.0	**1.6**	2.6	1.8

Table C.14: DiffAgeMean Accuracy and GMean results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.609**	0.589	0.594	0.580	0.568	**0.597**
EN_Arthritis	**0.586**	0.584	0.563	**0.577**	0.564	0.559
EN_Cataract	**0.593**	0.577	0.584	**0.586**	0.564	0.584
EN_Dementia	**0.678**	0.595	0.666	**0.687**	0.575	0.657
EN_Diabetes	0.803	0.795	**0.804**	**0.807**	0.783	0.798
EN_HBP	0.582	**0.599**	0.582	0.580	0.596	**0.583**
EN_Heartattack	0.622	**0.658**	0.632	0.603	**0.642**	0.628
EN_Osteoporosis	**0.605**	0.571	0.589	**0.595**	0.586	0.585
EN_Parkinsons	0.560	0.551	**0.588**	**0.537**	0.501	0.526
EN_Stroke	**0.639**	0.624	0.626	**0.635**	0.606	**0.635**
EC_Angina	0.684	0.669	**0.693**	**0.706**	0.669	0.671
EC_Arthritis	**0.712**	0.676	0.696	**0.702**	0.668	0.686
EC_Cataract	0.589	0.638	**0.652**	0.586	0.637	**0.662**
EC_Dementia	0.742	0.728	**0.745**	0.744	**0.751**	0.748
EC_Diabetes	**0.666**	0.645	0.662	0.678	0.640	**0.680**
EC_HBP	**0.632**	0.598	0.609	**0.631**	0.595	0.606
EC_Heartattack	**0.667**	0.619	0.619	**0.648**	0.629	0.618
EC_Osteoporosis	**0.656**	0.622	0.624	**0.649**	0.592	0.611
EC_Parkinsons	0.622	**0.698**	0.661	0.711	0.640	**0.715**
EC_Stroke	0.646	0.653	**0.669**	**0.661**	0.649	0.645
TI_Angina	**0.701**	0.591	0.698	**0.708**	0.578	0.697
TI_Arthritis	0.596	0.590	**0.606**	0.597	0.564	**0.608**
TI_Cancer	0.521	**0.597**	0.521	**0.514**	0.535	0.508
TI_Cataract	**0.669**	0.648	0.653	**0.657**	0.602	0.650
TI_Diabetes	**0.768**	0.591	0.738	**0.755**	0.554	0.739
TI_HBP	**0.609**	0.537	**0.609**	**0.609**	0.535	0.608
TI_Heartattack	**0.727**	0.579	0.703	**0.736**	0.548	0.721
TI_Ministroke	**0.655**	0.583	0.617	**0.656**	0.585	0.650
TI_Osteoporosis	0.661	0.559	**0.665**	**0.658**	0.544	0.647
TI_Stroke	**0.622**	0.544	0.597	**0.641**	0.534	0.598
AvgRank E-Nurse	**1.6**	2.5	2.0	**1.6**	2.4	2.1
AvgRank E-Core	**1.8**	2.6	1.7	**1.6**	2.5	1.9
AvgRank TILDA	**1.4**	2.8	1.8	**1.2**	2.8	2.0
AvgRank Overall	**1.6**	2.6	1.8	**1.5**	2.6	2.0

Table C.15: DiffAgeMean Sensitivity and Specificity results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.617**	0.608	0.599	**0.574**	0.570	0.554
EN_Arthritis	0.565	**0.623**	0.571	**0.571**	0.507	0.551
EN_Cataract	0.603	**0.604**	0.589	0.588	**0.609**	0.587
EN_Dementia	0.679	**0.696**	0.681	0.642	**0.655**	0.642
EN_Diabetes	**0.810**	0.789	0.804	**0.804**	0.778	0.799
EN_HBP	0.586	**0.586**	0.580	**0.589**	0.588	0.578
EN_Heartattack	**0.655**	0.629	0.638	0.594	**0.631**	0.608
EN_Osteoporosis	**0.618**	0.614	0.607	**0.615**	0.602	0.604
EN_Parkinsons	0.633	**0.648**	0.628	0.485	**0.545**	0.485
EN_Stroke	0.634	0.633	**0.635**	0.591	**0.594**	**0.594**
EC_Angina	0.652	0.637	**0.672**	**0.740**	0.702	0.688
EC_Arthritis	**0.738**	0.704	0.722	**0.679**	0.667	0.656
EC_Cataract	0.646	**0.664**	0.657	**0.721**	0.652	0.672
EC_Dementia	0.750	**0.758**	0.744	**0.783**	0.776	0.764
EC_Diabetes	**0.662**	0.652	0.650	0.687	0.674	**0.700**
EC_HBP	**0.639**	0.601	0.618	**0.617**	0.596	0.601
EC_Heartattack	0.641	**0.644**	0.629	**0.673**	0.662	0.651
EC_Osteoporosis	**0.674**	0.658	0.665	**0.685**	0.634	0.645
EC_Parkinsons	**0.676**	0.651	0.655	**0.760**	0.707	**0.760**
EC_Stroke	**0.650**	0.631	0.650	**0.672**	**0.672**	0.670
TI_Angina	0.677	0.615	**0.694**	**0.744**	0.640	0.712
TI_Arthritis	**0.668**	0.605	0.600	0.554	0.536	**0.600**
TI_Cancer	0.511	**0.563**	0.517	0.493	0.503	**0.553**
TI_Cataract	0.609	**0.649**	0.610	0.596	**0.636**	0.609
TI_Diabetes	0.697	**0.709**	0.674	0.681	**0.717**	0.681
TI_HBP	**0.613**	0.551	0.609	**0.593**	0.519	0.592
TI_Heartattack	**0.728**	0.617	0.682	0.659	0.620	**0.688**
TI_Ministroke	**0.657**	0.588	0.614	**0.686**	0.667	0.657
TI_Osteoporosis	**0.621**	0.663	0.612	**0.663**	0.652	0.604
TI_Stroke	0.599	0.522	**0.612**	**0.569**	0.523	**0.569**
AvgRank E-Nurse	**1.8**	1.9	2.4	**1.8**	1.9	2.4
AvgRank E-Core	**1.6**	2.3	2.2	**1.2**	2.5	2.4
AvgRank TILDA	**1.8**	2.1	2.1	**1.8**	2.3	1.9
AvgRank Overall	**1.7**	2.1	2.2	**1.6**	2.2	2.2

Table C.16: DiffAgeMean Accuracy and GMean results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.616**	0.606	0.597	**0.595**	0.588	0.576
EN_Arthritis	0.568	**0.573**	0.563	**0.568**	0.562	0.561
EN_Cataract	0.598	**0.605**	0.588	0.595	**0.606**	0.588
EN_Dementia	0.678	**0.695**	0.680	0.660	**0.676**	0.661
EN_Diabetes	**0.810**	0.788	0.803	**0.807**	0.784	0.802
EN_HBP	**0.587**	0.587	0.580	**0.588**	0.587	0.579
EN_Heartattack	**0.652**	0.629	0.636	0.624	**0.630**	0.623
EN_Osteoporosis	**0.618**	0.613	0.607	**0.616**	0.608	0.606
EN_Parkinsons	0.632	**0.647**	0.627	0.554	**0.594**	0.552
EN_Stroke	0.631	0.631	**0.632**	0.612	0.613	**0.614**
EC_Angina	0.655	0.639	**0.673**	**0.695**	0.668	0.680
EC_Arthritis	**0.714**	0.689	0.696	**0.708**	0.685	0.688
EC_Cataract	**0.668**	0.660	0.661	**0.683**	0.658	0.664
EC_Dementia	0.750	**0.758**	0.745	0.766	**0.767**	0.754
EC_Diabetes	**0.665**	0.654	0.657	0.675	0.663	**0.675**
EC_HBP	**0.631**	0.599	0.612	**0.628**	0.598	0.610
EC_Heartattack	0.642	**0.645**	0.630	**0.657**	0.653	0.640
EC_Osteoporosis	**0.675**	0.656	0.663	**0.679**	0.646	0.655
EC_Parkinsons	**0.677**	0.652	0.656	**0.717**	0.678	0.706
EC_Stroke	**0.651**	0.633	0.651	**0.661**	0.651	0.660
TI_Angina	0.680	0.616	**0.695**	**0.710**	0.627	0.703
TI_Arthritis	**0.633**	0.583	0.600	**0.609**	0.569	0.600
TI_Cancer	0.510	**0.560**	0.519	0.502	0.532	**0.535**
TI_Cataract	0.608	**0.648**	0.609	0.603	**0.642**	0.609
TI_Diabetes	0.696	**0.710**	0.675	0.689	**0.713**	0.677
TI_HBP	**0.605**	0.539	0.603	**0.603**	0.535	0.601
TI_Heartattack	**0.725**	0.617	0.683	**0.692**	0.618	0.685
TI_Ministroke	**0.657**	0.590	0.615	**0.671**	0.626	0.635
TI_Osteoporosis	0.625	**0.662**	0.611	0.642	**0.657**	0.608
TI_Stroke	0.599	0.522	**0.612**	0.584	0.522	**0.590**
AvgRank E-Nurse	**1.7**	1.8	2.5	**1.7**	**1.7**	2.6
AvgRank E-Core	**1.4**	2.6	2.1	**1.2**	2.7	2.2
AvgRank TILDA	**1.8**	2.2	2.0	**1.7**	2.3	2.0
AvgRank Overall	**1.6**	2.2	2.2	**1.5**	2.2	2.3

Table C.17: AvgDiffAgeMean Sensitivity and Specificity results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	0.611	0.575	**0.621**	0.570	0.543	**0.589**
EN_Arthritis	0.558	**0.589**	0.564	0.555	0.561	**0.565**
EN_Cataract	0.594	0.627	**0.595**	0.573	**0.580**	0.570
EN_Dementia	**0.643**	0.627	0.638	**0.655**	0.588	0.642
EN_Diabetes	**0.812**	0.783	0.803	**0.792**	0.774	0.789
EN_HBP	0.624	0.601	**0.626**	**0.615**	0.608	0.600
EN_Heartattack	**0.643**	0.611	0.636	0.631	**0.658**	0.603
EN_Osteoporosis	0.598	0.613	**0.617**	**0.622**	0.572	0.570
EN_Parkinsons	**0.587**	0.550	0.569	0.515	0.515	**0.591**
EN_Stroke	**0.630**	0.621	0.620	**0.594**	0.558	0.584
EC_Angina	0.688	0.664	**0.690**	**0.660**	0.639	0.632
EC_Arthritis	**0.724**	0.704	0.712	0.679	0.676	**0.681**
EC_Cataract	0.599	0.600	**0.619**	0.571	0.625	**0.640**
EC_Dementia	0.684	0.726	**0.720**	**0.714**	0.689	0.671
EC_Diabetes	0.672	0.640	**0.673**	**0.684**	0.666	0.664
EC_HBP	0.621	**0.626**	0.612	**0.590**	0.581	0.572
EC_Heartattack	**0.636**	0.628	0.601	0.576	0.580	**0.642**
EC_Osteoporosis	**0.651**	0.613	0.632	0.569	**0.593**	0.587
EC_Parkinsons	0.642	0.613	**0.651**	0.573	0.640	**0.680**
EC_Stroke	**0.665**	0.650	0.661	0.600	0.614	**0.618**
TI_Angina	0.745	**1.000**	0.754	0.732	0.000	**0.760**
TI_Arthritis	0.603	**1.000**	0.619	0.614	0.000	**0.618**
TI_Cancer	0.533	**1.000**	0.528	0.543	0.000	**0.589**
TI_Cataract	0.691	**1.000**	0.666	**0.628**	0.000	**0.628**
TI_Diabetes	0.768	**1.000**	0.773	0.758	0.000	**0.771**
TI_HBP	0.652	**1.000**	0.649	**0.640**	0.000	0.621
TI_Heartattack	0.760	**1.000**	0.741	**0.751**	0.000	**0.751**
TI_Ministroke	0.684	**1.000**	0.686	0.676	0.000	**0.686**
TI_Osteoporosis	0.671	**1.000**	0.677	0.650	0.000	**0.670**
TI_Stroke	0.687	**1.000**	0.660	**0.646**	0.000	0.631
AvgRank E-Nurse	**1.8**	2.4	**1.8**	**1.7**	2.3	2.1
AvgRank E-Core	1.8	2.4	**1.8**	2.1	2.0	**1.9**
AvgRank TILDA	2.5	**1.0**	2.5	1.7	3.0	**1.3**
AvgRank Overall	2.0	**1.9**	2.0	1.8	2.4	**1.8**

Table C.18: AvgDiffAgeMean Accuracy and GMean results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	0.609	0.574	**0.620**	0.590	0.558	**0.605**
EN_Arthritis	0.557	**0.577**	0.564	0.557	**0.575**	0.565
EN_Cataract	0.587	**0.612**	0.587	0.583	**0.603**	0.582
EN_Dementia	**0.644**	0.627	0.638	**0.649**	0.607	0.640
EN_Diabetes	**0.809**	0.782	0.801	**0.802**	0.778	0.796
EN_HBP	**0.620**	0.604	0.616	**0.619**	0.605	0.613
EN_Heartattack	**0.643**	0.614	0.634	**0.637**	0.634	0.620
EN_Osteoporosis	0.601	0.609	**0.613**	**0.610**	0.592	0.593
EN_Parkinsons	**0.586**	0.549	0.570	0.550	0.532	**0.580**
EN_Stroke	**0.628**	0.617	0.618	**0.612**	0.589	0.602
EC_Angina	0.687	0.663	**0.688**	**0.674**	0.651	0.660
EC_Arthritis	**0.706**	0.693	0.700	**0.701**	0.690	0.696
EC_Cataract	0.591	0.608	**0.625**	0.585	0.613	**0.629**
EC_Dementia	0.684	**0.725**	0.719	0.699	**0.707**	0.695
EC_Diabetes	**0.673**	0.643	0.672	**0.678**	0.653	0.668
EC_HBP	**0.609**	0.608	0.597	**0.605**	0.603	0.592
EC_Heartattack	**0.633**	0.625	0.603	0.605	0.604	**0.621**
EC_Osteoporosis	**0.644**	0.611	0.628	0.608	0.603	**0.609**
EC_Parkinsons	0.641	0.614	**0.651**	0.606	0.627	**0.665**
EC_Stroke	**0.662**	0.648	0.659	0.632	0.631	**0.639**
TI_Angina	0.744	**0.956**	0.755	0.738	0.000	**0.757**
TI_Arthritis	0.606	**0.690**	0.619	0.608	0.000	**0.618**
TI_Cancer	0.533	**0.947**	0.532	0.538	0.000	**0.558**
TI_Cataract	0.686	**0.916**	0.662	**0.659**	0.000	0.646
TI_Diabetes	0.768	**0.933**	0.773	0.763	0.000	**0.772**
TI_HBP	**0.648**	0.620	0.638	**0.646**	0.000	0.635
TI_Heartattack	0.760	**0.964**	0.742	**0.756**	0.000	0.746
TI_Ministroke	0.684	**0.982**	0.686	0.680	0.000	**0.686**
TI_Osteoporosis	0.669	**0.905**	0.676	0.661	0.000	**0.673**
TI_Stroke	0.687	**0.989**	0.660	**0.666**	0.000	0.645
AvgRank E-Nurse	**1.7**	2.5	1.9	**1.5**	2.5	2.0
AvgRank E-Core	**1.6**	2.5	1.9	1.8	2.5	**1.7**
AvgRank TILDA	2.4	**1.2**	2.4	1.6	3.0	**1.4**
AvgRank Overall	**1.9**	2.1	2.1	1.6	2.7	1.7

Table C.19: AvgDiffAgeMean Sensitivity and Specificity results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	0.616	0.598	**0.620**	**0.593**	0.589	0.578
EN_Arthritis	0.570	**0.597**	0.554	0.555	0.549	**0.558**
EN_Cataract	**0.591**	0.588	0.588	0.584	**0.637**	0.592
EN_Dementia	**0.675**	0.671	0.673	**0.628**	0.601	0.622
EN_Diabetes	**0.809**	0.781	0.803	**0.797**	0.771	0.789
EN_HBP	0.624	0.604	**0.626**	**0.618**	0.596	0.606
EN_Heartattack	0.643	0.642	**0.644**	**0.633**	0.621	0.626
EN_Osteoporosis	**0.614**	0.593	0.612	0.593	**0.656**	0.609
EN_Parkinsons	**0.631**	0.602	0.621	0.500	0.545	**0.561**
EN_Stroke	**0.615**	0.604	0.615	0.591	**0.603**	0.568
EC_Angina	0.669	0.665	**0.693**	**0.681**	0.674	0.628
EC_Arthritis	**0.727**	0.710	0.716	0.677	**0.685**	0.674
EC_Cataract	**0.647**	**0.647**	0.640	0.653	**0.658**	0.639
EC_Dementia	**0.742**	0.727	0.729	0.758	**0.795**	0.789
EC_Diabetes	**0.678**	0.652	0.670	0.676	0.663	**0.686**
EC_HBP	0.621	0.623	**0.624**	0.582	0.583	**0.584**
EC_Heartattack	0.637	**0.657**	0.633	0.626	0.610	**0.673**
EC_Osteoporosis	**0.687**	0.654	0.674	0.606	**0.641**	0.631
EC_Parkinsons	**0.664**	0.631	0.650	0.547	0.640	**0.653**
EC_Stroke	0.638	**0.653**	0.649	0.616	0.631	**0.638**
TI_Angina	0.751	0.655	**0.752**	0.724	0.680	**0.776**
TI_Arthritis	0.614	**0.648**	0.623	0.598	0.534	**0.620**
TI_Cancer	**0.547**	0.447	0.536	0.572	**0.678**	0.569
TI_Cataract	0.617	**0.716**	0.600	0.619	**0.701**	0.590
TI_Diabetes	0.719	0.703	**0.725**	0.730	**0.823**	0.706
TI_HBP	0.653	0.600	**0.655**	**0.646**	0.564	0.645
TI_Heartattack	**0.742**	0.638	0.715	0.688	0.688	**0.751**
TI_Ministroke	**0.693**	0.671	0.682	**0.706**	0.520	0.598
TI_Osteoporosis	**0.667**	0.599	0.660	0.624	**0.801**	0.639
TI_Stroke	**0.682**	0.571	0.650	**0.631**	0.569	0.600
AvgRank E-Nurse	**1.5**	2.8	1.8	**1.8**	2.2	2.0
AvgRank E-Core	**1.7**	2.4	2.0	2.4	**1.8**	**1.8**
AvgRank TILDA	**1.6**	2.6	1.8	**1.9**	2.2	2.0
AvgRank Overall	**1.6**	2.6	1.9	2.0	2.1	**1.9**

Table C.20: AvgDiffAgeMean Accuracy and GMean results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	0.615	0.598	**0.619**	**0.604**	0.594	0.599
EN_Arthritis	0.563	0.577	**0.556**	0.562	**0.573**	0.556
EN_Cataract	0.589	0.604	**0.589**	0.587	**0.612**	0.590
EN_Dementia	**0.674**	0.670	0.672	**0.651**	0.635	0.647
EN_Diabetes	**0.808**	0.780	0.801	**0.803**	0.776	0.796
EN_HBP	**0.622**	0.601	0.618	**0.621**	0.600	0.616
EN_Heartattack	0.642	0.641	**0.643**	**0.638**	0.631	0.635
EN_Osteoporosis	**0.612**	0.599	0.612	0.604	**0.624**	0.610
EN_Parkinsons	**0.630**	0.602	0.621	0.562	0.573	**0.590**
EN_Stroke	**0.614**	0.604	0.612	0.603	**0.604**	0.591
EC_Angina	0.670	0.666	**0.691**	**0.675**	0.669	0.660
EC_Arthritis	**0.707**	0.700	0.699	**0.701**	0.698	0.695
EC_Cataract	0.649	**0.651**	0.640	0.650	**0.653**	0.639
EC_Dementia	**0.742**	0.729	0.730	0.750	**0.760**	0.758
EC_Diabetes	**0.677**	0.654	0.672	0.677	0.658	**0.678**
EC_HBP	0.606	0.608	**0.609**	0.601	0.603	**0.604**
EC_Heartattack	0.637	**0.654**	0.635	0.632	0.633	**0.653**
EC_Osteoporosis	**0.680**	0.652	0.671	0.645	0.647	**0.652**
EC_Parkinsons	**0.663**	0.631	0.650	0.603	0.635	**0.652**
EC_Stroke	0.637	**0.652**	0.649	0.627	0.642	**0.643**
TI_Angina	0.750	0.656	**0.753**	0.737	0.667	**0.764**
TI_Arthritis	0.609	0.613	**0.622**	0.606	0.588	**0.621**
TI_Cancer	**0.548**	0.460	0.538	**0.559**	0.551	0.553
TI_Cataract	0.617	**0.715**	0.599	0.618	**0.709**	0.595
TI_Diabetes	0.719	0.711	**0.724**	0.724	**0.761**	0.716
TI_HBP	**0.651**	0.587	**0.651**	**0.650**	0.582	**0.650**
TI_Heartattack	**0.740**	0.640	0.716	0.714	0.662	**0.733**
TI_Ministroke	**0.694**	0.668	0.680	**0.700**	0.590	0.639
TI_Osteoporosis	**0.663**	0.619	0.658	0.645	**0.693**	0.649
TI_Stroke	**0.682**	0.571	0.649	**0.656**	0.570	0.624
AvgRank E-Nurse	**1.5**	2.6	1.9	**1.8**	2.1	2.1
AvgRank E-Core	**1.7**	2.2	2.1	2.4	1.9	**1.7**
AvgRank TILDA	**1.6**	2.7	1.8	**1.8**	2.4	1.9
AvgRank Overall	**1.6**	2.5	1.9	2.0	2.1	**1.9**

Table C.21: Percentile Sensitivity and Specificity results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Sensitivity			Specificity		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.612**	0.591	0.603	0.550	0.589	**0.605**
EN_Arthritis	**0.627**	0.579	0.571	0.530	0.542	**0.550**
EN_Cataract	**0.607**	0.580	0.597	0.565	0.563	**0.572**
EN_Dementia	**0.678**	0.641	0.654	**0.696**	0.669	0.662
EN_Diabetes	0.801	0.796	**0.804**	**0.813**	0.790	0.799
EN_HBP	**0.589**	0.579	**0.589**	0.571	0.581	**0.589**
EN_Heartattack	**0.625**	0.623	0.624	0.581	**0.643**	0.603
EN_Osteoporosis	**0.607**	0.584	0.605	0.583	**0.601**	0.564
EN_Parkinsons	**0.561**	0.560	0.551	0.515	**0.621**	0.576
EN_Stroke	**0.640**	0.623	0.637	**0.629**	0.625	0.589
EC_Angina	**0.683**	0.662	0.673	**0.730**	0.688	0.684
EC_Arthritis	**0.745**	0.704	0.743	**0.662**	0.642	0.657
EC_Cataract	**0.594**	**0.594**	0.592	0.578	0.584	**0.625**
EC_Dementia	**0.742**	0.731	0.760	**0.745**	**0.745**	0.720
EC_Diabetes	0.662	**0.666**	0.654	**0.694**	0.642	0.691
EC_HBP	**0.637**	0.630	0.623	**0.625**	0.563	0.622
EC_Heartattack	**0.669**	0.617	0.635	**0.628**	0.615	0.662
EC_Osteoporosis	**0.657**	0.635	0.644	**0.641**	0.558	0.618
EC_Parkinsons	0.621	**0.689**	0.669	**0.813**	0.733	0.773
EC_Stroke	0.644	0.664	**0.665**	**0.679**	0.618	0.627
TI_Angina	0.700	0.660	**0.701**	0.716	**0.728**	0.724
TI_Arthritis	0.594	**0.610**	0.576	**0.600**	0.542	0.597
TI_Cancer	0.521	**0.543**	0.524	0.507	0.464	**0.539**
TI_Cataract	**0.671**	0.628	0.640	**0.644**	0.605	0.630
TI_Diabetes	**0.770**	0.665	0.741	0.740	0.660	**0.756**
TI_HBP	0.611	0.591	**0.615**	**0.606**	0.581	0.593
TI_Heartattack	**0.726**	0.677	0.719	**0.746**	0.702	0.654
TI_Ministroke	**0.655**	0.613	0.631	**0.657**	0.578	0.647
TI_Osteoporosis	**0.661**	0.560	0.656	**0.656**	0.541	0.645
TI_Stroke	**0.622**	0.598	0.585	0.662	**0.754**	0.708
AvgRank E-Nurse	**1.2**	2.8	2.1	2.2	1.9	**1.9**
AvgRank E-Core	**1.7**	2.3	2.1	**1.4**	2.7	2.0
AvgRank TILDA	**1.5**	2.5	2.0	**1.6**	2.5	1.9
AvgRank Overall	**1.4**	2.5	2.1	**1.7**	2.4	1.9

Table C.22: Percentile Accuracy and GMean results for the Scenario 1 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	Accuracy			GMean		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.609**	0.591	0.603	0.580	0.590	**0.604**
EN_Arthritis	**0.586**	0.563	0.562	**0.577**	0.560	0.560
EN_Cataract	**0.593**	0.575	0.589	**0.586**	0.572	0.584
EN_Dementia	**0.678**	0.641	0.655	**0.687**	0.655	0.658
EN_Diabetes	**0.803**	0.796	**0.803**	**0.807**	0.793	0.801
EN_HBP	0.582	0.580	**0.589**	0.580	0.580	**0.589**
EN_Heartattack	0.622	**0.624**	0.623	0.603	**0.633**	0.614
EN_Osteoporosis	**0.605**	0.586	0.602	**0.595**	0.592	0.584
EN_Parkinsons	**0.560**	**0.560**	0.552	0.537	**0.590**	0.563
EN_Stroke	**0.639**	0.623	0.634	**0.635**	0.624	0.612
EC_Angina	**0.684**	0.663	0.673	**0.706**	0.675	0.678
EC_Arthritis	**0.712**	0.679	0.709	**0.702**	0.672	0.698
EC_Cataract	0.589	0.591	**0.602**	0.586	0.589	**0.608**
EC_Dementia	0.742	0.731	**0.759**	**0.744**	0.738	0.740
EC_Diabetes	**0.666**	0.663	0.659	**0.678**	0.654	0.672
EC_HBP	**0.632**	0.604	0.623	**0.631**	0.596	0.623
EC_Heartattack	**0.667**	0.617	0.637	0.648	0.616	**0.649**
EC_Osteoporosis	**0.656**	0.628	0.642	**0.649**	0.595	0.631
EC_Parkinsons	0.622	**0.689**	0.670	0.711	0.711	**0.719**
EC_Stroke	0.646	0.661	**0.663**	**0.661**	0.640	0.646
TI_Angina	0.701	0.663	**0.702**	0.708	0.693	**0.712**
TI_Arthritis	**0.596**	0.589	0.582	**0.597**	0.575	0.586
TI_Cancer	0.521	**0.539**	0.524	0.514	0.502	**0.531**
TI_Cataract	**0.669**	0.626	0.639	**0.657**	0.616	0.635
TI_Diabetes	**0.768**	0.664	0.742	**0.755**	0.662	0.749
TI_HBP	**0.609**	0.587	0.607	**0.609**	0.586	0.604
TI_Heartattack	**0.727**	0.678	0.716	**0.736**	0.689	0.685
TI_Ministroke	**0.655**	0.612	0.631	**0.656**	0.595	0.639
TI_Osteoporosis	**0.661**	0.558	0.655	**0.658**	0.551	0.650
TI_Stroke	**0.622**	0.600	0.587	0.641	**0.671**	0.644
AvgRank E-Nurse	**1.4**	2.6	2.1	**1.8**	2.2	2.1
AvgRank E-Core	**1.7**	2.5	1.8	**1.5**	2.9	1.7
AvgRank TILDA	**1.3**	2.6	2.1	**1.4**	2.7	1.9
AvgRank Overall	**1.5**	2.6	2.0	**1.5**	2.6	1.9

Table C.23: Percentile Sensitivity and Specificity results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	SENSITIVITY			SPECIFICITY		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.617**	0.591	0.608	0.574	**0.578**	0.566
EN_Arthritis	**0.565**	0.553	0.544	**0.571**	0.551	0.558
EN_Cataract	**0.603**	0.598	0.602	0.588	**0.600**	0.587
EN_Dementia	**0.679**	0.662	0.669	0.642	0.628	**0.649**
EN_Diabetes	**0.810**	0.788	0.808	**0.804**	0.792	0.797
EN_HBP	0.586	0.594	**0.597**	**0.589**	0.573	0.584
EN_Heartattack	**0.655**	0.610	0.652	0.594	**0.633**	0.599
EN_Osteoporosis	**0.618**	0.609	0.608	**0.615**	0.613	0.609
EN_Parkinsons	**0.633**	0.613	0.607	0.485	**0.636**	0.500
EN_Stroke	**0.634**	0.612	0.627	0.591	**0.637**	0.603
EC_Angina	0.652	0.685	**0.686**	**0.740**	0.695	0.674
EC_Arthritis	**0.738**	0.700	0.707	**0.679**	0.664	0.676
EC_Cataract	0.646	**0.663**	0.659	**0.721**	0.654	0.690
EC_Dementia	**0.750**	0.747	0.738	0.783	0.783	**0.807**
EC_Diabetes	0.662	0.657	**0.666**	**0.687**	0.660	0.679
EC_HBP	**0.639**	0.620	0.639	**0.617**	0.580	0.597
EC_Heartattack	0.641	**0.648**	0.626	0.673	0.671	**0.692**
EC_Osteoporosis	**0.674**	0.669	0.665	**0.685**	0.654	0.659
EC_Parkinsons	**0.676**	0.670	0.676	**0.760**	0.707	0.747
EC_Stroke	**0.650**	0.638	0.637	**0.672**	0.666	0.633
TI_Angina	0.677	0.673	**0.701**	**0.744**	0.688	0.708
TI_Arthritis	**0.668**	0.570	0.592	0.554	**0.589**	0.579
TI_Cancer	0.511	**0.514**	0.499	0.493	0.490	**0.513**
TI_Cataract	0.609	**0.655**	0.605	0.596	**0.661**	0.617
TI_Diabetes	0.697	**0.745**	0.684	0.681	**0.717**	0.681
TI_HBP	**0.613**	0.596	0.601	0.593	0.601	**0.615**
TI_Heartattack	**0.728**	0.666	0.710	0.659	0.663	**0.732**
TI_Ministroke	**0.657**	0.579	0.639	**0.686**	0.618	0.588
TI_Osteoporosis	**0.621**	0.660	0.619	**0.663**	0.641	0.613
TI_Stroke	0.599	0.606	**0.619**	0.569	**0.677**	0.662
AvgRank E-Nurse	**1.2**	2.6	2.2	**1.9**	**1.9**	2.2
AvgRank E-Core	**1.7**	2.2	2.1	**1.3**	2.8	2.0
AvgRank TILDA	**1.7**	2.1	2.2	2.3	**1.8**	2.0
AvgRank Overall	**1.5**	2.3	2.2	**1.8**	2.2	2.1

Table C.24: Percentile Accuracy and GMean results for the Scenario 2 experiments with J48 Decision Tree classifiers. The best result for each row is boldfaced, and the average ranks per dataset and overall are presented in the last 4 rows of the Table.

	ACCURACY			GMEAN		
	Baseline	CTFs-only	BL+CTFs	Baseline	CTFs-only	BL+CTFs
EN_Angina	**0.616**	0.590	0.607	**0.595**	0.584	0.587
EN_Arthritis	**0.568**	0.552	0.550	**0.568**	0.552	0.551
EN_Cataract	0.598	**0.599**	0.597	0.595	0.599	**0.595**
EN_Dementia	**0.678**	0.662	0.669	**0.660**	0.645	0.659
EN_Diabetes	**0.810**	0.789	0.806	**0.807**	0.790	0.802
EN_HBP	0.587	0.586	**0.592**	0.588	0.584	**0.591**
EN_Heartattack	**0.652**	0.612	0.649	0.624	0.622	**0.625**
EN_Osteoporosis	**0.618**	0.610	0.608	**0.616**	0.611	0.608
EN_Parkinsons	**0.632**	0.613	0.606	0.554	**0.625**	0.551
EN_Stroke	**0.631**	0.613	0.626	0.612	**0.624**	0.615
EC_Angina	0.655	**0.686**	**0.686**	**0.695**	0.690	0.680
EC_Arthritis	**0.714**	0.686	0.695	**0.708**	0.682	0.691
EC_Cataract	0.668	0.660	**0.668**	**0.683**	0.658	0.674
EC_Dementia	**0.750**	0.748	0.740	0.766	0.765	**0.772**
EC_Diabetes	0.665	0.657	**0.667**	**0.675**	0.658	0.672
EC_HBP	**0.631**	0.604	0.623	**0.628**	0.599	0.617
EC_Heartattack	0.642	**0.649**	0.630	0.657	**0.660**	0.658
EC_Osteoporosis	**0.675**	0.668	0.665	**0.679**	0.661	0.662
EC_Parkinsons	**0.677**	0.671	0.676	**0.717**	0.688	0.710
EC_Stroke	**0.651**	0.639	0.637	**0.661**	0.652	0.635
TI_Angina	0.680	0.673	**0.701**	**0.710**	0.680	0.705
TI_Arthritis	**0.633**	0.576	0.588	**0.609**	0.579	0.585
TI_Cancer	0.510	**0.512**	0.500	0.502	0.502	**0.506**
TI_Cataract	0.608	**0.656**	0.606	0.603	**0.658**	0.611
TI_Diabetes	0.696	**0.743**	0.684	0.689	**0.731**	0.682
TI_HBP	0.605	0.598	**0.606**	0.603	0.599	**0.608**
TI_Heartattack	**0.725**	0.666	0.711	0.692	0.665	**0.721**
TI_Ministroke	**0.657**	0.580	0.638	**0.671**	0.598	0.613
TI_Osteoporosis	0.625	**0.658**	0.618	0.642	**0.650**	0.616
TI_Stroke	0.599	0.607	**0.619**	0.584	**0.640**	**0.640**
AvgRank E-Nurse	**1.2**	2.5	2.3	**1.7**	2.2	2.2
AvgRank E-Core	**1.5**	2.4	2.2	**1.3**	2.6	2.1
AvgRank TILDA	**1.8**	2.1	2.1	2.0	2.2	**1.9**
AvgRank Overall	**1.5**	2.3	2.2	**1.6**	2.3	2.0

Appendix D

Lexicographic Approach Experiments - Additional Tables

In this Appendix we first report the Sensitivity and Specificity tables for the automated threshold selection experiments with RF classifiers, in Section D.1.

Then, in Sections D.2 and D.3, we report on the same set of experiments ran in Chapter 6, but using the C4.5 decision tree classifier instead of Random Forests. For each of the two scenarios, we first compare the automated data-driven threshold selection to fixed threshold values, regarding how to set the tie threshold parameter in the proposed lexicographic split approach. Then we compare the No Lexic and Lexic configurations, using the standard split function vs the proposed bi-objective split.

D.1 RF Threshold Selection Experiments - Sensitivity and Specificity Tables

D.2 Decision Tree Experiments - Baseline Datasets

For the first scenario, where the datasets have only the original features in the ELSA-nurse, ELSA-core and TILDA datasets, the results were as follows. In the fixed vs automated threshold comparisons, the proposed data-driven threshold approach did not achieve the smallest average rank in any case. However, the smallest ranks were divided among different fixed values, indicating no clear

Table D.1: Sensitivity results for threshold selection experiments in the Baseline datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**Data-Driven**
EN_Angina	0.677	0.682	0.681	0.681	0.685	0.685	0.684	0.683	0.687	0.683	0.684	0.693
EN_Arthritis	0.666	0.663	0.669	0.668	0.672	0.667	0.663	0.666	0.664	0.661	0.659	0.669
EN_Cataract	0.623	0.624	0.627	0.625	0.624	0.629	0.631	0.623	0.627	0.631	0.629	0.63
EN_Dementia	0.731	0.734	0.733	0.733	0.737	0.737	0.731	0.733	0.734	0.733	0.732	0.74
EN_Diabetes	0.844	0.845	0.844	0.844	0.845	0.844	0.845	0.843	0.845	0.844	0.843	0.843
EN_HBP	0.647	0.651	0.649	0.647	0.645	0.638	0.646	0.646	0.64	0.643	0.637	0.647
EN_HeartAttack	0.703	0.705	0.699	0.697	0.699	0.698	0.703	0.7	0.7	0.7	0.698	0.703
EN_Osteoporosis	0.652	0.653	0.657	0.652	0.653	0.647	0.648	0.648	0.647	0.652	0.654	0.654
EN_Parkinsons	0.633	0.633	0.632	0.632	0.634	0.636	0.634	0.646	0.641	0.645	0.643	0.634
EN_Stroke	0.67	0.671	0.67	0.671	0.67	0.679	0.677	0.676	0.675	0.677	0.681	0.677
EC_Angina	0.71	0.71	0.711	0.712	0.711	0.704	0.706	0.711	0.712	0.712	0.71	0.709
EC_Arthritis	0.746	0.75	0.749	0.75	0.75	0.752	0.751	0.76	0.757	0.752	0.752	0.752
EC_Cataract	0.616	0.613	0.614	0.62	0.617	0.626	0.626	0.629	0.628	0.628	0.626	0.625
EC_Dementia	0.765	0.764	0.766	0.767	0.767	0.769	0.767	0.768	0.77	0.772	0.772	0.768
EC_Diabetes	0.676	0.672	0.673	0.674	0.671	0.668	0.67	0.668	0.661	0.666	0.663	0.672
EC_HBP	0.626	0.636	0.639	0.637	0.637	0.64	0.631	0.631	0.635	0.637	0.633	0.633
EC_HeartAttack	0.679	0.678	0.681	0.68	0.686	0.684	0.684	0.682	0.682	0.683	0.68	0.683
EC_Osteoporosis	0.697	0.697	0.699	0.696	0.698	0.704	0.702	0.699	0.699	0.697	0.699	0.701
EC_Parkinsons	0.695	0.698	0.704	0.699	0.698	0.7	0.702	0.701	0.701	0.701	0.704	0.701
EC_Stroke	0.692	0.692	0.698	0.694	0.693	0.693	0.695	0.698	0.697	0.698	0.698	0.697
TI_Angina	0.747	0.749	0.748	0.748	0.749	0.747	0.745	0.743	0.741	0.74	0.738	0.749
TI_Arthritis	0.74	0.724	0.731	0.73	0.721	0.721	0.717	0.704	0.713	0.71	0.711	0.729
TI_Cancer	0.555	0.553	0.541	0.544	0.54	0.537	0.537	0.541	0.546	0.538	0.539	0.542
TI_Cataract	0.703	0.699	0.699	0.706	0.7	0.697	0.697	0.698	0.696	0.697	0.694	0.699
TI_Diabetes	0.776	0.778	0.776	0.78	0.777	0.775	0.775	0.774	0.768	0.767	0.766	0.771
TI_HBP	0.679	0.679	0.671	0.672	0.671	0.673	0.669	0.676	0.664	0.669	0.666	0.679
TI_HeartAttack	0.753	0.75	0.752	0.75	0.749	0.75	0.748	0.748	0.751	0.744	0.744	0.749
TI_Ministroke	0.709	0.707	0.711	0.705	0.705	0.704	0.707	0.71	0.705	0.7	0.7	0.706
TI_Osteoporosis	0.681	0.68	0.678	0.677	0.676	0.669	0.67	0.664	0.66	0.663	0.663	0.671
TI_Stroke	0.725	0.728	0.728	0.726	0.722	0.724	0.721	0.721	0.72	0.714	0.711	0.718
AvgRank Elsanurse	8.30	5.95	6.80	8.00	5.80	6.25	6.05	7.30	6.20	6.25	7.20	**3.90**
AvgRank Elsacore	9.85	9.55	5.95	7.15	7.00	5.60	6.30	5.15	5.15	**4.70**	5.75	5.85
AvgRank TILDA	**2.60**	3.10	3.75	3.95	5.65	7.30	8.10	7.45	8.70	10.60	11.15	5.65
AvgRank Overall	6.92	6.20	5.50	6.37	6.15	6.38	6.82	6.63	6.68	7.18	8.03	**5.13**

Table D.2: Specificity results for threshold selection experiments in the Baseline datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**Data-Driven**
EN_Angina	0.698	0.709	0.674	0.713	0.705	0.694	0.685	0.682	0.694	0.729	0.686	0.69
EN_Arthritis	0.589	0.595	0.591	0.588	0.581	0.586	0.63	0.592	0.592	0.592	0.581	0.589
EN_Cataract	0.72	0.717	0.72	0.727	0.724	0.72	0.657	0.711	0.714	0.722	0.712	0.72
EN_Dementia	0.736	0.716	0.73	0.73	0.736	0.723	0.731	0.723	0.723	0.716	0.723	0.723
EN_Diabetes	0.862	0.86	0.86	0.871	0.867	0.867	0.848	0.859	0.86	0.855	0.86	0.863
EN_HBP	0.747	0.745	0.748	0.744	0.733	0.749	0.687	0.743	0.75	0.733	0.734	0.749
EN_HeartAttack	0.713	0.716	0.738	0.728	0.728	0.721	0.703	0.723	0.721	0.723	0.723	0.741
EN_Osteoporosis	0.702	0.708	0.696	0.705	0.693	0.703	0.653	0.711	0.703	0.705	0.699	0.716
EN_Parkinsons	0.712	0.712	0.697	0.667	0.652	0.652	0.634	0.652	0.682	0.697	0.697	0.636
EN_Stroke	0.732	0.705	0.724	0.694	0.703	0.72	0.678	0.705	0.71	0.701	0.715	0.713
EC_Angina	0.726	0.74	0.758	0.747	0.772	0.744	0.754	0.782	0.775	0.772	0.768	0.737
EC_Arthritis	0.719	0.716	0.714	0.71	0.716	0.718	0.714	0.709	0.704	0.707	0.709	0.719
EC_Cataract	0.711	0.71	0.713	0.703	0.711	0.706	0.708	0.709	0.717	0.706	0.713	0.715
EC_Dementia	0.758	0.77	0.758	0.752	0.764	0.77	0.764	0.783	0.77	0.752	0.77	0.776
EC_Diabetes	0.758	0.75	0.75	0.761	0.755	0.755	0.75	0.755	0.75	0.75	0.758	0.764
EC_HBP	0.663	0.671	0.667	0.662	0.659	0.657	0.659	0.659	0.671	0.659	0.668	0.665
EC_HeartAttack	0.705	0.701	0.698	0.707	0.707	0.703	0.696	0.723	0.698	0.698	0.696	0.689
EC_Osteoporosis	0.679	0.668	0.669	0.658	0.659	0.672	0.665	0.668	0.679	0.677	0.673	0.677
EC_Parkinsons	0.68	0.707	0.693	0.693	0.72	0.707	0.733	0.733	0.72	0.733	0.707	0.733
EC_Stroke	0.707	0.716	0.716	0.718	0.714	0.716	0.731	0.721	0.712	0.723	0.723	0.729
TI_Angina	0.908	0.916	0.904	0.904	0.896	0.904	0.896	0.888	0.888	0.908	0.9	0.88
TI_Arthritis	0.647	0.653	0.65	0.662	0.649	0.653	0.656	0.646	0.654	0.66	0.657	0.654
TI_Cancer	0.599	0.612	0.589	0.566	0.556	0.586	0.599	0.602	0.618	0.582	0.569	0.546
TI_Cataract	0.728	0.728	0.743	0.726	0.743	0.747	0.736	0.726	0.713	0.741	0.72	0.715
TI_Diabetes	0.823	0.839	0.839	0.834	0.831	0.839	0.839	0.834	0.834	0.844	0.855	0.834
TI_HBP	0.76	0.758	0.764	0.765	0.763	0.757	0.762	0.764	0.763	0.753	0.759	0.764
TI_HeartAttack	0.863	0.873	0.878	0.868	0.878	0.898	0.893	0.898	0.878	0.883	0.888	0.873
TI_Ministroke	0.735	0.725	0.745	0.716	0.735	0.735	0.735	0.735	0.735	0.725	0.716	0.755
TI_Osteoporosis	0.808	0.799	0.81	0.81	0.803	0.81	0.816	0.814	0.818	0.808	0.816	0.799
TI_Stroke	0.738	0.723	0.723	0.708	0.708	0.723	0.738	0.738	0.738	0.723	0.708	0.708
AvgRank Elsanurse	5.10	6.00	5.75	**5.05**	6.45	6.20	9.80	7.65	6.30	6.55	7.80	5.35
AvgRank Elsacore	6.75	6.80	7.30	7.95	6.55	7.20	7.35	5.20	5.70	7.35	5.60	**4.25**
AvgRank TILDA	7.30	6.85	**5.15**	7.30	8.15	5.55	4.70	5.90	5.90	6.00	6.75	8.45
AvgRank Overall	6.38	6.55	6.07	6.77	7.05	6.32	7.28	6.25	5.97	6.63	6.72	**6.02**

Table D.3: Sensitivity results for threshold selection experiments in the Baseline+CTF datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**Data-Driven**
EN_Angina	0.682	0.684	0.688	0.688	0.686	0.684	0.686	0.682	0.677	0.677	0.679	0.683
EN_Arthritis	0.669	0.673	0.671	0.668	0.665	0.677	0.668	0.667	0.666	0.661	0.664	0.669
EN_Cataract	0.605	0.603	0.609	0.613	0.61	0.611	0.604	0.61	0.611	0.605	0.605	0.614
EN_Dementia	0.743	0.744	0.744	0.742	0.741	0.741	0.744	0.743	0.741	0.743	0.744	0.745
EN_Diabetes	0.843	0.842	0.844	0.843	0.843	0.842	0.845	0.845	0.845	0.845	0.844	0.846
EN_HBP	0.647	0.648	0.647	0.652	0.646	0.644	0.645	0.648	0.645	0.654	0.649	0.648
EN_HeartAttack	0.699	0.701	0.703	0.703	0.704	0.703	0.699	0.7	0.701	0.703	0.704	0.702
EN_Osteoporosis	0.645	0.647	0.646	0.645	0.645	0.642	0.64	0.641	0.642	0.642	0.641	0.643
EN_Parkinsons	0.63	0.631	0.634	0.633	0.638	0.639	0.642	0.643	0.643	0.645	0.648	0.644
EN_Stroke	0.675	0.679	0.678	0.678	0.677	0.679	0.675	0.679	0.676	0.68	0.675	0.681
EC_Angina	0.708	0.705	0.705	0.708	0.707	0.704	0.704	0.707	0.706	0.705	0.703	0.705
EC_Arthritis	0.752	0.749	0.747	0.748	0.751	0.747	0.75	0.748	0.756	0.752	0.75	0.751
EC_Cataract	0.616	0.616	0.615	0.616	0.619	0.616	0.615	0.62	0.616	0.615	0.622	0.627
EC_Dementia	0.766	0.768	0.768	0.766	0.768	0.771	0.766	0.767	0.768	0.77	0.77	0.772
EC_Diabetes	0.68	0.677	0.68	0.682	0.679	0.679	0.676	0.672	0.675	0.678	0.678	0.677
EC_HBP	0.639	0.628	0.637	0.632	0.643	0.648	0.638	0.638	0.643	0.638	0.643	0.632
EC_HeartAttack	0.679	0.674	0.677	0.679	0.674	0.672	0.672	0.676	0.674	0.675	0.676	0.677
EC_Osteoporosis	0.697	0.701	0.697	0.695	0.694	0.694	0.692	0.696	0.698	0.696	0.695	0.697
EC_Parkinsons	0.697	0.694	0.694	0.697	0.697	0.697	0.7	0.701	0.7	0.701	0.703	0.709
EC_Stroke	0.695	0.696	0.696	0.699	0.696	0.696	0.697	0.699	0.694	0.697	0.699	0.695
TI_Angina	0.742	0.742	0.744	0.745	0.742	0.742	0.742	0.746	0.744	0.74	0.742	0.739
TI_Arthritis	0.713	0.716	0.715	0.71	0.713	0.709	0.705	0.711	0.71	0.708	0.711	0.721
TI_Cancer	0.552	0.547	0.548	0.548	0.553	0.554	0.557	0.551	0.554	0.554	0.553	0.555
TI_Cataract	0.701	0.696	0.693	0.698	0.7	0.7	0.699	0.702	0.704	0.703	0.699	0.699
TI_Diabetes	0.757	0.756	0.755	0.751	0.752	0.754	0.756	0.752	0.75	0.75	0.75	0.755
TI_HBP	0.668	0.667	0.673	0.673	0.668	0.665	0.666	0.662	0.662	0.659	0.659	0.669
TI_HeartAttack	0.741	0.742	0.741	0.742	0.742	0.744	0.742	0.741	0.748	0.742	0.743	0.743
TI_Ministroke	0.709	0.706	0.702	0.702	0.703	0.702	0.702	0.699	0.701	0.698	0.699	0.706
TI_Osteoporosis	0.668	0.668	0.668	0.673	0.67	0.669	0.67	0.673	0.671	0.668	0.669	0.67
TI_Stroke	0.705	0.704	0.704	0.705	0.706	0.708	0.708	0.706	0.708	0.705	0.705	0.708
AvgRank Elsanurse	8.40	6.40	5.10	5.50	6.95	6.80	7.90	6.65	7.90	6.05	6.70	**3.65**
AvgRank Elsacore	5.40	7.85	7.45	6.10	6.15	7.45	8.95	6.15	6.20	5.95	**5.15**	5.20
AvgRank TILDA	6.05	7.25	7.45	6.50	5.75	5.95	5.90	6.50	5.30	8.85	8.10	**4.40**
AvgRank Overall	6.62	7.17	6.67	6.03	6.28	6.73	7.58	6.43	6.47	6.95	6.65	**4.42**

Table D.4: Specificity results for threshold selection experiments in the Baseline+CTF datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**Data-Driven**
EN_Angina	0.678	0.682	0.709	0.663	0.686	0.674	0.686	0.69	0.69	0.698	0.69	0.702
EN_Arthritis	0.592	0.604	0.594	0.604	0.602	0.603	0.594	0.596	0.592	0.606	0.605	0.594
EN_Cataract	0.74	0.732	0.733	0.746	0.726	0.733	0.731	0.729	0.727	0.725	0.734	0.734
EN_Dementia	0.723	0.723	0.703	0.709	0.716	0.709	0.696	0.689	0.696	0.716	0.709	0.736
EN_Diabetes	0.865	0.86	0.855	0.866	0.864	0.866	0.862	0.856	0.865	0.855	0.853	0.864
EN_HBP	0.748	0.748	0.739	0.746	0.73	0.74	0.738	0.729	0.741	0.741	0.741	0.742
EN_HeartAttack	0.711	0.711	0.718	0.721	0.723	0.726	0.728	0.731	0.736	0.718	0.736	0.726
EN_Osteoporosis	0.719	0.719	0.711	0.708	0.714	0.706	0.708	0.697	0.702	0.7	0.7	0.713
EN_Parkinsons	0.712	0.742	0.742	0.697	0.712	0.712	0.712	0.712	0.682	0.697	0.697	0.636
EN_Stroke	0.703	0.701	0.703	0.717	0.703	0.701	0.713	0.72	0.703	0.708	0.696	0.708
EC_Angina	0.751	0.733	0.747	0.747	0.737	0.737	0.747	0.747	0.775	0.779	0.772	0.772
EC_Arthritis	0.723	0.722	0.723	0.723	0.723	0.72	0.718	0.722	0.718	0.718	0.713	0.719
EC_Cataract	0.765	0.758	0.762	0.762	0.758	0.765	0.763	0.768	0.76	0.763	0.766	0.75
EC_Dementia	0.814	0.814	0.82	0.801	0.783	0.776	0.814	0.814	0.839	0.82	0.826	0.832
EC_Diabetes	0.766	0.76	0.746	0.757	0.756	0.776	0.771	0.765	0.757	0.763	0.761	0.734
EC_HBP	0.671	0.679	0.677	0.671	0.668	0.679	0.677	0.669	0.673	0.671	0.677	0.678
EC_HeartAttack	0.698	0.692	0.692	0.696	0.721	0.714	0.728	0.721	0.723	0.705	0.714	0.692
EC_Osteoporosis	0.665	0.665	0.666	0.687	0.669	0.663	0.67	0.665	0.658	0.676	0.662	0.659
EC_Parkinsons	0.72	0.72	0.733	0.72	0.747	0.733	0.72	0.707	0.707	0.72	0.707	0.72
EC_Stroke	0.747	0.742	0.742	0.749	0.731	0.736	0.749	0.736	0.742	0.731	0.729	0.751
TI_Angina	0.88	0.876	0.88	0.864	0.864	0.872	0.868	0.88	0.876	0.872	0.868	0.896
TI_Arthritis	0.648	0.657	0.648	0.663	0.655	0.656	0.655	0.662	0.657	0.651	0.655	0.642
TI_Cancer	0.579	0.576	0.563	0.553	0.546	0.569	0.566	0.539	0.563	0.563	0.543	0.599
TI_Cataract	0.741	0.743	0.734	0.745	0.73	0.749	0.728	0.73	0.738	0.73	0.726	0.741
TI_Diabetes	0.813	0.821	0.81	0.818	0.805	0.818	0.816	0.821	0.821	0.816	0.81	0.818
TI_HBP	0.746	0.75	0.751	0.753	0.752	0.753	0.756	0.756	0.75	0.753	0.747	0.754
TI_HeartAttack	0.859	0.844	0.839	0.849	0.839	0.844	0.854	0.844	0.854	0.859	0.854	0.849
TI_Ministroke	0.716	0.716	0.716	0.735	0.745	0.725	0.725	0.725	0.725	0.725	0.706	0.716
TI_Osteoporosis	0.757	0.772	0.77	0.777	0.783	0.775	0.775	0.775	0.777	0.761	0.757	0.759
TI_Stroke	0.692	0.708	0.708	0.723	0.723	0.723	0.723	0.738	0.738	0.738	0.738	0.754
AvgRank Elsanurse	5.65	5.65	6.75	5.35	6.80	6.70	7.05	7.50	7.55	7.05	6.85	**5.10**
AvgRank Elsacore	5.60	7.45	6.25	6.40	7.55	6.10	**5.05**	6.65	6.85	6.10	7.00	7.00
AvgRank TILDA	7.55	6.25	8.55	5.20	7.75	5.40	6.30	5.20	**4.85**	6.40	9.20	5.35
AvgRank Overall	6.27	6.45	7.18	**5.65**	7.37	6.07	6.13	6.45	6.42	6.52	7.68	5.82

answer as to what the best choice would be in each case.

Notably, smaller threshold values such as 0.0 and 0.005 have performed well, and we believe this is due to the fact that the decision tree calculates the information gain for every feature in each node. This makes ties more likely to happen, so we need to have a more restrictive tie threshold to avoid ill-advised changes to the chosen split feature. Nevertheless, the average ranks of the proposed data-driven automated threshold selection are still more reliable than a fixed value.

In the Lexic vs NoLexic comparison, the latter unexpectedly got its best results for the ELSA-core datasets. Overall, however, the Lexic approach was slightly superior, with smaller average ranks for Accuracy and GMean in all other cases. Even though a single decision tree seems to benefit less from the proposed lexicographic split approach than an ensemble method, these results encourage further experiments in this direction.

D.3 Decision Tree Experiments - Baseline+CTF Datasets

In the second scenario, we added constructed temporal features to the datasets, which are ignored by the lexicographic split approach. For these experiments, the automated vs fixed threshold results were unchanged, meaning the data-driven approach did not achieve the smallest average rank in any of the cases, but still had the most reliable results, with the smallest values varying between different fixed values. The smallest fixed values, 0.0 and 0.05, still performed better in general.

In the Lexic vs NoLexic comparison, we see much closer results between the two approaches. The GMean global metric of performance, arguably the most important of our metrics, had both methods tied overall and for ELSA-core datasets, with Lexic winning for ELSA-nurse and NoLexic winning for TILDA. For the other metrics, NoLexic models had better Sensitivity and (by consequence) Accuracy, and Lexic models had better Specificity, in general. Thus, in this case we cannot say the proposed lexicographic split increased predictive performance overall, although it did not reduce it either. As discussed in Chapter 6, the CTFs compose

Table D.5: C4.5 decision tree Sensitivity results for threshold selection experiments in the Baseline datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments. The last column, DD, refers to the data-driven approach, which uses internal cross-validation to select the threshold value for each decision tree.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**DD**
EN_Angina	0.603	0.609	0.593	0.608	0.608	0.615	0.61	0.612	0.608	0.61	0.61	0.617
EN_Arthritis	0.571	0.563	0.56	0.563	0.549	0.557	0.561	0.551	0.55	0.564	0.559	0.561
EN_Cataract	0.596	0.595	0.611	0.591	0.597	0.607	0.607	0.583	0.598	0.601	0.611	0.596
EN_Dementia	0.683	0.681	0.669	0.667	0.679	0.689	0.686	0.683	0.671	0.681	0.675	0.683
EN_Diabetes	0.813	0.815	0.81	0.819	0.821	0.819	0.815	0.819	0.813	0.807	0.812	0.808
EN_HBP	0.616	0.621	0.606	0.609	0.619	0.619	0.61	0.601	0.602	0.598	0.605	0.618
EN_HeartAttack	0.641	0.657	0.642	0.632	0.638	0.654	0.647	0.649	0.647	0.641	0.639	0.641
EN_Osteoporosis	0.612	0.606	0.607	0.626	0.626	0.616	0.596	0.613	0.609	0.604	0.611	0.614
EN_Parkinsons	0.609	0.62	0.643	0.638	0.64	0.634	0.649	0.639	0.645	0.643	0.651	0.646
EN_Stroke	0.631	0.621	0.629	0.637	0.634	0.633	0.616	0.616	0.628	0.622	0.64	0.629
EC_Angina	0.661	0.662	0.661	0.658	0.662	0.653	0.667	0.669	0.65	0.668	0.662	0.675
EC_Arthritis	0.72	0.71	0.712	0.741	0.733	0.716	0.717	0.715	0.705	0.733	0.722	0.724
EC_Cataract	0.66	0.658	0.656	0.651	0.627	0.636	0.659	0.663	0.655	0.668	0.655	0.646
EC_Dementia	0.715	0.716	0.714	0.713	0.717	0.707	0.704	0.697	0.701	0.703	0.712	0.715
EC_Diabetes	0.666	0.671	0.666	0.66	0.66	0.67	0.673	0.666	0.661	0.662	0.659	0.652
EC_HBP	0.621	0.616	0.631	0.605	0.63	0.599	0.612	0.605	0.613	0.629	0.623	0.621
EC_HeartAttack	0.629	0.627	0.63	0.623	0.615	0.649	0.647	0.662	0.644	0.633	0.635	0.636
EC_Osteoporosis	0.678	0.687	0.683	0.675	0.684	0.668	0.68	0.681	0.691	0.693	0.7	0.68
EC_Parkinsons	0.606	0.612	0.626	0.615	0.623	0.636	0.639	0.643	0.654	0.665	0.659	0.612
EC_Stroke	0.655	0.656	0.659	0.656	0.661	0.654	0.648	0.664	0.671	0.659	0.64	0.644
TI_Angina	0.732	0.735	0.741	0.729	0.718	0.719	0.723	0.719	0.726	0.727	0.725	0.726
TI_Arthritis	0.615	0.628	0.609	0.627	0.62	0.628	0.624	0.644	0.625	0.616	0.608	0.624
TI_Cancer	0.525	0.536	0.545	0.53	0.568	0.538	0.536	0.534	0.541	0.523	0.526	0.519
TI_Cataract	0.665	0.678	0.662	0.669	0.668	0.667	0.671	0.673	0.675	0.66	0.663	0.671
TI_Diabetes	0.766	0.767	0.772	0.781	0.773	0.777	0.772	0.762	0.753	0.754	0.759	0.767
TI_HBP	0.648	0.653	0.653	0.66	0.645	0.65	0.642	0.658	0.646	0.65	0.66	0.651
TI_HeartAttack	0.736	0.739	0.731	0.732	0.737	0.728	0.739	0.736	0.727	0.728	0.725	0.744
TI_Ministroke	0.666	0.668	0.693	0.668	0.677	0.674	0.678	0.658	0.688	0.689	0.673	0.666
TI_Osteoporosis	0.664	0.668	0.671	0.671	0.666	0.671	0.674	0.672	0.671	0.681	0.686	0.668
TI_Stroke	0.683	0.682	0.668	0.649	0.654	0.671	0.653	0.659	0.653	0.639	0.619	0.652
AvgRank Elsanurse	6.80	6.55	7.65	7.00	6.20	**4.00**	5.75	7.05	7.80	7.60	6.05	5.55
AvgRank Elsacore	7.05	6.40	6.10	8.35	6.10	8.20	6.05	5.45	6.95	**4.10**	6.15	7.10
AvgRank TILDA	7.80	**4.45**	5.35	5.65	6.80	5.95	5.90	6.00	6.65	8.00	8.25	7.20
AvgRank Overall	7.22	5.80	6.37	7.00	6.37	6.05	**5.90**	6.17	7.13	6.57	6.82	6.62

a large portion of the features in the dataset, so ignoring them reduces the effectiveness of the lexicographic split. We believe that extending the split to somehow include CTFs might improve these results.

Table D.6: C4.5 decision tree Specificity results for threshold selection experiments in the Baseline datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments. The last column, DD, refers to the data-driven approach, which uses internal cross-validation to select the threshold value for each decision tree.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**DD**
EN_Angina	0.589	0.632	0.624	0.601	0.597	0.585	0.61	0.574	0.578	0.593	0.554	0.574
EN_Arthritis	0.577	0.565	0.569	0.572	0.568	0.57	0.557	0.558	0.565	0.56	0.562	0.571
EN_Cataract	0.582	0.578	0.567	0.573	0.587	0.581	0.596	0.587	0.579	0.583	0.596	0.585
EN_Dementia	0.662	0.703	0.703	0.676	0.628	0.649	0.686	0.642	0.689	0.655	0.628	0.649
EN_Diabetes	0.795	0.803	0.781	0.797	0.804	0.814	0.815	0.799	0.801	0.803	0.803	0.797
EN_HBP	0.601	0.607	0.598	0.608	0.602	0.607	0.603	0.598	0.596	0.603	0.601	0.616
EN_HeartAttack	0.636	0.638	0.638	0.616	0.636	0.618	0.646	0.618	0.608	0.594	0.591	0.621
EN_Osteoporosis	0.599	0.619	0.644	0.596	0.61	0.619	0.599	0.616	0.59	0.583	0.586	0.612
EN_Parkinsons	0.652	0.561	0.576	0.576	0.591	0.561	0.648	0.606	0.621	0.652	0.561	0.636
EN_Stroke	0.658	0.653	0.61	0.61	0.622	0.599	0.614	0.591	0.57	0.606	0.594	0.57
EC_Angina	0.639	0.656	0.656	0.642	0.677	0.663	0.635	0.639	0.681	0.632	0.639	0.632
EC_Arthritis	0.667	0.669	0.668	0.654	0.655	0.659	0.648	0.678	0.675	0.647	0.66	0.665
EC_Cataract	0.636	0.651	0.639	0.652	0.666	0.681	0.669	0.638	0.644	0.657	0.642	0.652
EC_Dementia	0.814	0.776	0.795	0.776	0.758	0.783	0.783	0.795	0.807	0.795	0.795	0.789
EC_Diabetes	0.673	0.663	0.674	0.693	0.67	0.657	0.679	0.664	0.694	0.691	0.675	0.686
EC_HBP	0.577	0.593	0.559	0.59	0.59	0.614	0.593	0.623	0.606	0.594	0.557	0.598
EC_HeartAttack	0.594	0.621	0.61	0.587	0.635	0.676	0.657	0.66	0.678	0.633	0.685	0.659
EC_Osteoporosis	0.651	0.634	0.654	0.63	0.639	0.65	0.662	0.627	0.643	0.629	0.639	0.661
EC_Parkinsons	0.667	0.68	0.693	0.667	0.667	0.627	0.653	0.733	0.707	0.707	0.667	0.64
EC_Stroke	0.635	0.611	0.614	0.634	0.618	0.631	0.655	0.618	0.62	0.627	0.67	0.6
TI_Angina	0.764	0.748	0.748	0.76	0.776	0.776	0.74	0.74	0.756	0.74	0.744	0.772
TI_Arthritis	0.635	0.623	0.61	0.616	0.609	0.599	0.613	0.61	0.603	0.605	0.615	0.616
TI_Cancer	0.582	0.605	0.53	0.572	0.553	0.563	0.559	0.543	0.576	0.559	0.546	0.595
TI_Cataract	0.682	0.651	0.682	0.701	0.697	0.695	0.669	0.682	0.695	0.669	0.69	0.682
TI_Diabetes	0.774	0.756	0.761	0.73	0.73	0.717	0.714	0.735	0.743	0.73	0.743	0.699
TI_HBP	0.643	0.624	0.648	0.628	0.611	0.63	0.616	0.609	0.611	0.616	0.596	0.65
TI_HeartAttack	0.766	0.8	0.766	0.771	0.737	0.751	0.766	0.761	0.766	0.751	0.756	0.732
TI_Ministroke	0.716	0.716	0.637	0.745	0.735	0.725	0.755	0.765	0.696	0.706	0.696	0.696
TI_Osteoporosis	0.667	0.667	0.681	0.685	0.694	0.681	0.654	0.652	0.678	0.657	0.652	0.667
TI_Stroke	0.6	0.615	0.662	0.708	0.677	0.662	0.677	0.662	0.631	0.631	0.631	0.677
AvgRank Elsanurse	5.50	4.65	6.05	6.55	5.65	6.30	**4.25**	8.10	8.40	7.10	9.05	6.40
AvgRank Elsacore	6.95	7.65	6.80	7.60	7.40	5.95	6.20	6.10	**3.45**	6.75	6.35	6.80
AvgRank TILDA	**4.95**	5.70	6.45	3.55	5.80	6.10	7.30	7.95	6.70	9.00	8.40	6.10
AvgRank Overall	**5.80**	6.00	6.43	5.90	6.28	6.12	5.92	7.38	6.18	7.62	7.93	6.43

Table D.7: C4.5 decision tree Accuracy results for threshold selection experiments in the Baseline datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments. The last column, DD, refers to the data-driven approach, which uses internal cross-validation to select the threshold value for each decision tree.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**DD**
EN_Angina	0.603	0.609	0.594	0.608	0.607	0.614	0.607	0.61	0.607	0.61	0.608	0.615
EN_Arthritis	0.574	0.564	0.564	0.567	0.557	0.563	0.556	0.554	0.556	0.562	0.56	0.565
EN_Cataract	0.592	0.59	0.597	0.585	0.593	0.599	0.59	0.584	0.592	0.595	0.606	0.592
EN_Dementia	0.682	0.681	0.67	0.667	0.678	0.689	0.677	0.682	0.672	0.68	0.675	0.683
EN_Diabetes	0.81	0.813	0.806	0.816	0.819	0.818	0.814	0.816	0.811	0.806	0.811	0.806
EN_HBP	0.61	0.615	0.603	0.608	0.612	0.614	0.602	0.6	0.6	0.6	0.604	0.617
EN_HeartAttack	0.641	0.656	0.641	0.631	0.637	0.652	0.638	0.647	0.645	0.638	0.637	0.64
EN_Osteoporosis	0.61	0.607	0.61	0.623	0.624	0.616	0.613	0.613	0.607	0.602	0.609	0.613
EN_Parkinsons	0.609	0.62	0.642	0.637	0.64	0.634	0.611	0.639	0.645	0.643	0.65	0.646
EN_Stroke	0.633	0.623	0.628	0.635	0.634	0.631	0.604	0.614	0.625	0.621	0.637	0.626
EC_Angina	0.66	0.662	0.661	0.657	0.663	0.653	0.666	0.668	0.651	0.666	0.661	0.673
EC_Arthritis	0.699	0.694	0.695	0.706	0.702	0.693	0.689	0.7	0.693	0.699	0.697	0.7
EC_Cataract	0.653	0.656	0.651	0.651	0.639	0.649	0.662	0.656	0.652	0.665	0.651	0.647
EC_Dementia	0.717	0.717	0.716	0.714	0.718	0.708	0.705	0.699	0.703	0.705	0.714	0.717
EC_Diabetes	0.667	0.67	0.667	0.664	0.662	0.668	0.674	0.666	0.666	0.666	0.661	0.656
EC_HBP	0.605	0.607	0.603	0.6	0.614	0.605	0.605	0.612	0.61	0.615	0.597	0.612
EC_HeartAttack	0.627	0.627	0.629	0.621	0.616	0.651	0.647	0.662	0.646	0.633	0.638	0.637
EC_Osteoporosis	0.676	0.682	0.681	0.672	0.68	0.667	0.678	0.676	0.687	0.688	0.695	0.678
EC_Parkinsons	0.607	0.612	0.626	0.615	0.624	0.636	0.639	0.644	0.655	0.665	0.659	0.612
EC_Stroke	0.654	0.653	0.657	0.655	0.659	0.653	0.649	0.662	0.668	0.657	0.642	0.642
TI_Angina	0.733	0.736	0.741	0.731	0.721	0.722	0.723	0.72	0.727	0.728	0.726	0.728
TI_Arthritis	0.621	0.627	0.609	0.624	0.617	0.619	0.621	0.633	0.618	0.613	0.61	0.622
TI_Cancer	0.528	0.54	0.544	0.532	0.567	0.539	0.537	0.534	0.542	0.525	0.527	0.523
TI_Cataract	0.667	0.676	0.664	0.671	0.67	0.67	0.671	0.673	0.677	0.661	0.665	0.672
TI_Diabetes	0.767	0.767	0.771	0.777	0.77	0.773	0.768	0.761	0.752	0.752	0.758	0.763
TI_HBP	0.646	0.642	0.651	0.648	0.632	0.642	0.632	0.64	0.633	0.637	0.636	0.65
TI_HeartAttack	0.737	0.741	0.732	0.733	0.737	0.729	0.74	0.737	0.728	0.728	0.726	0.743
TI_Ministroke	0.667	0.669	0.692	0.669	0.678	0.675	0.68	0.659	0.688	0.689	0.673	0.666
TI_Osteoporosis	0.665	0.668	0.672	0.673	0.668	0.672	0.672	0.671	0.671	0.679	0.683	0.668
TI_Stroke	0.682	0.681	0.668	0.65	0.654	0.671	0.653	0.659	0.653	0.639	0.619	0.652
AvgRank Elsanurse	6.55	6.25	7.35	6.50	5.55	**3.50**	8.75	7.15	8.05	7.70	6.05	4.60
AvgRank Elsacore	7.30	6.20	6.75	8.25	6.10	7.90	6.15	4.80	6.25	**4.20**	7.15	6.95
AvgRank TILDA	6.50	**4.45**	4.80	5.30	7.00	6.00	6.20	6.85	6.90	8.35	9.00	6.65
AvgRank Overall	6.78	**5.63**	6.30	6.68	6.22	5.80	7.03	6.27	7.07	6.75	7.40	6.07

Table D.8: C4.5 decision tree GMean results for threshold selection experiments in the Baseline datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments. The last column, DD, refers to the data-driven approach, which uses internal cross-validation to select the threshold value for each decision tree.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**DD**
EN_Angina	0.596	0.62	0.609	0.604	0.602	0.6	0.607	0.592	0.593	0.602	0.582	0.595
EN_Arthritis	0.574	0.564	0.564	0.567	0.559	0.564	0.556	0.555	0.557	0.562	0.561	0.566
EN_Cataract	0.589	0.586	0.589	0.582	0.592	0.594	0.59	0.585	0.589	0.592	0.604	0.59
EN_Dementia	0.672	0.692	0.686	0.671	0.653	0.669	0.677	0.662	0.68	0.668	0.652	0.666
EN_Diabetes	0.804	0.809	0.795	0.808	0.813	0.816	0.814	0.809	0.807	0.805	0.808	0.802
EN_HBP	0.608	0.614	0.602	0.608	0.61	0.613	0.602	0.6	0.599	0.601	0.603	0.617
EN_HeartAttack	0.639	0.648	0.64	0.624	0.637	0.636	0.638	0.634	0.628	0.617	0.615	0.631
EN_Osteoporosis	0.605	0.613	0.625	0.611	0.618	0.618	0.613	0.615	0.599	0.593	0.598	0.613
EN_Parkinsons	0.63	0.59	0.608	0.606	0.615	0.596	0.611	0.622	0.633	0.647	0.604	0.641
EN_Stroke	0.644	0.637	0.62	0.623	0.628	0.615	0.604	0.604	0.599	0.614	0.616	0.599
EC_Angina	0.65	0.659	0.658	0.65	0.67	0.658	0.651	0.654	0.665	0.649	0.65	0.653
EC_Arthritis	0.693	0.69	0.69	0.696	0.693	0.687	0.682	0.696	0.69	0.689	0.69	0.694
EC_Cataract	0.648	0.654	0.647	0.652	0.646	0.658	0.664	0.651	0.65	0.663	0.649	0.649
EC_Dementia	0.763	0.746	0.753	0.744	0.737	0.744	0.742	0.744	0.752	0.748	0.753	0.751
EC_Diabetes	0.67	0.667	0.67	0.676	0.665	0.663	0.676	0.665	0.677	0.676	0.667	0.669
EC_HBP	0.599	0.605	0.594	0.598	0.609	0.606	0.603	0.614	0.61	0.611	0.589	0.609
EC_HeartAttack	0.611	0.624	0.62	0.605	0.625	0.663	0.652	0.661	0.661	0.633	0.659	0.647
EC_Osteoporosis	0.664	0.66	0.669	0.652	0.661	0.659	0.671	0.653	0.666	0.66	0.669	0.67
EC_Parkinsons	0.636	0.645	0.659	0.64	0.645	0.631	0.646	0.687	0.68	0.685	0.663	0.626
EC_Stroke	0.645	0.633	0.636	0.645	0.639	0.643	0.652	0.641	0.645	0.643	0.655	0.622
TI_Angina	0.748	0.742	0.744	0.745	0.747	0.747	0.731	0.73	0.741	0.733	0.735	0.749
TI_Arthritis	0.625	0.626	0.61	0.621	0.614	0.614	0.619	0.626	0.614	0.611	0.612	0.62
TI_Cancer	0.553	0.57	0.537	0.551	0.56	0.55	0.547	0.538	0.558	0.541	0.536	0.556
TI_Cataract	0.674	0.664	0.672	0.684	0.682	0.681	0.67	0.677	0.685	0.665	0.676	0.676
TI_Diabetes	0.77	0.762	0.766	0.755	0.751	0.746	0.743	0.749	0.748	0.742	0.751	0.732
TI_HBP	0.645	0.638	0.651	0.644	0.627	0.64	0.629	0.633	0.628	0.633	0.627	0.65
TI_HeartAttack	0.751	0.769	0.748	0.751	0.737	0.739	0.753	0.748	0.746	0.739	0.74	0.738
TI_Ministroke	0.691	0.691	0.664	0.705	0.705	0.699	0.716	0.709	0.692	0.697	0.684	0.681
TI_Osteoporosis	0.666	0.667	0.676	0.678	0.68	0.676	0.664	0.662	0.674	0.669	0.669	0.667
TI_Stroke	0.64	0.648	0.665	0.678	0.665	0.666	0.665	0.66	0.642	0.635	0.625	0.664
AvgRank Elsanurse	5.45	**4.45**	5.35	6.70	5.25	5.15	6.05	8.50	8.45	7.50	8.55	6.60
AvgRank Elsacore	7.10	7.30	7.05	7.55	7.60	7.40	5.70	5.65	**3.90**	5.90	6.15	6.70
AvgRank TILDA	5.40	5.65	6.60	**3.50**	5.40	5.65	7.20	6.95	6.60	9.25	9.10	6.70
AvgRank Overall	5.98	**5.80**	6.33	5.92	6.08	6.07	6.32	7.03	6.32	7.55	7.93	6.67

Table D.9: C4.5 decision tree Comparison of Lexic and NoLexic approaches for Baseline datasets.

Baseline Datasets	**SENSITIVITY**		**SPECIFICITY**		**ACCURACY**		**GMEAN**	
	Lexic	**NoLexic**	**Lexic**	**NoLexic**	**Lexic**	**NoLexic**	**Lexic**	**NoLexic**
EN_Angina	0.617	0.608	0.574	0.589	0.615	0.608	0.595	0.599
EN_Arthritis	0.561	0.57	0.571	0.56	0.565	0.566	0.566	0.565
EN_Cataract	0.596	0.598	0.585	0.578	0.592	0.592	0.59	0.588
EN_Dementia	0.683	0.679	0.649	0.622	0.683	0.678	0.666	0.65
EN_Diabetes	0.808	0.804	0.797	0.805	0.806	0.804	0.802	0.805
EN_HBP	0.618	0.587	0.616	0.603	0.617	0.594	0.617	0.595
EN_HeartAttack	0.641	0.646	0.621	0.638	0.64	0.646	0.631	0.642
EN_Osteoporosis	0.614	0.594	0.612	0.63	0.613	0.598	0.613	0.612
EN_Parkinsons	0.646	0.633	0.636	0.5	0.646	0.632	0.641	0.563
EN_Stroke	0.629	0.629	0.57	0.575	0.626	0.625	0.599	0.601
EC_Angina	0.675	0.674	0.632	0.635	0.673	0.672	0.653	0.654
EC_Arthritis	0.724	0.706	0.665	0.661	0.7	0.688	0.694	0.683
EC_Cataract	0.646	0.661	0.652	0.642	0.647	0.655	0.649	0.651
EC_Dementia	0.715	0.747	0.789	0.795	0.717	0.748	0.751	0.771
EC_Diabetes	0.652	0.673	0.686	0.689	0.656	0.675	0.669	0.681
EC_HBP	0.621	0.614	0.598	0.592	0.612	0.606	0.609	0.603
EC_HeartAttack	0.636	0.631	0.659	0.634	0.637	0.631	0.647	0.632
EC_Osteoporosis	0.68	0.684	0.661	0.638	0.678	0.68	0.67	0.661
EC_Parkinsons	0.612	0.657	0.64	0.676	0.612	0.658	0.626	0.667
EC_Stroke	0.644	0.649	0.6	0.666	0.642	0.65	0.622	0.658
TI_Angina	0.726	0.733	0.772	0.764	0.728	0.734	0.749	0.748
TI_Arthritis	0.624	0.626	0.616	0.596	0.622	0.616	0.62	0.611
TI_Cancer	0.519	0.522	0.595	0.53	0.523	0.522	0.556	0.526
TI_Cataract	0.671	0.677	0.682	0.642	0.672	0.674	0.676	0.659
TI_Diabetes	0.767	0.764	0.699	0.748	0.763	0.763	0.732	0.756
TI_HBP	0.651	0.638	0.65	0.608	0.65	0.627	0.65	0.623
TI_HeartAttack	0.744	0.74	0.732	0.756	0.743	0.741	0.738	0.748
TI_Ministroke	0.666	0.68	0.696	0.686	0.666	0.68	0.681	0.683
TI_Osteoporosis	0.668	0.677	0.667	0.657	0.668	0.675	0.667	0.667
TI_Stroke	0.652	0.639	0.677	0.615	0.652	0.639	0.664	0.627
AvgRank Elsanurse	**1.35**	1.65	**1.50**	**1.50**	**1.25**	1.75	**1.40**	1.60
AvgRank Elsacore	1.60	**1.40**	**1.50**	**1.50**	1.60	**1.40**	1.60	**1.40**
AvgRank TILDA	1.60	**1.40**	**1.20**	1.80	**1.45**	1.55	**1.35**	1.65
AvgRank Overall	1.52	**1.48**	**1.40**	1.60	**1.43**	1.57	**1.45**	1.55

Table D.10: C4.5 decision tree Sensitivity results for threshold selection experiments in the Baseline+CTF datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments. The last column, DD, refers to the data-driven approach, which uses internal cross-validation to select the threshold value for each decision tree.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**DD**
EN_Angina	0.579	0.595	0.595	0.604	0.605	0.598	0.617	0.605	0.601	0.588	0.594	0.598
EN_Arthritis	0.564	0.567	0.573	0.564	0.568	0.559	0.563	0.559	0.563	0.555	0.56	0.565
EN_Cataract	0.584	0.587	0.605	0.595	0.61	0.584	0.596	0.591	0.599	0.596	0.594	0.588
EN_Dementia	0.673	0.678	0.668	0.663	0.667	0.674	0.678	0.672	0.671	0.666	0.667	0.684
EN_Diabetes	0.794	0.802	0.802	0.799	0.796	0.792	0.8	0.796	0.795	0.794	0.789	0.796
EN_HBP	0.622	0.619	0.615	0.629	0.605	0.61	0.619	0.607	0.614	0.605	0.601	0.606
EN_HeartAttack	0.632	0.627	0.639	0.645	0.637	0.651	0.637	0.636	0.639	0.648	0.643	0.639
EN_Osteoporosis	0.603	0.6	0.6	0.606	0.612	0.597	0.592	0.602	0.608	0.606	0.601	0.602
EN_Parkinsons	0.584	0.6	0.598	0.611	0.605	0.627	0.627	0.637	0.636	0.636	0.615	0.604
EN_Stroke	0.626	0.617	0.628	0.627	0.632	0.614	0.633	0.627	0.627	0.635	0.638	0.624
EC_Angina	0.756	0.758	0.756	0.745	0.74	0.743	0.743	0.737	0.737	0.733	0.727	0.745
EC_Arthritis	0.611	0.613	0.62	0.62	0.606	0.606	0.611	0.615	0.607	0.612	0.61	0.616
EC_Cataract	0.541	0.531	0.547	0.509	0.527	0.522	0.517	0.52	0.524	0.534	0.519	0.524
EC_Dementia	0.686	0.684	0.653	0.669	0.671	0.666	0.659	0.66	0.657	0.65	0.652	0.656
EC_Diabetes	0.753	0.744	0.747	0.745	0.738	0.745	0.741	0.742	0.747	0.754	0.748	0.748
EC_HBP	0.647	0.657	0.651	0.647	0.631	0.638	0.641	0.631	0.637	0.642	0.634	0.645
EC_HeartAttack	0.73	0.732	0.735	0.73	0.745	0.741	0.719	0.719	0.722	0.722	0.718	0.723
EC_Osteoporosis	0.685	0.689	0.672	0.67	0.649	0.659	0.667	0.668	0.661	0.653	0.656	0.671
EC_Parkinsons	0.66	0.653	0.662	0.662	0.654	0.672	0.667	0.667	0.673	0.666	0.667	0.664
EC_Stroke	0.663	0.678	0.67	0.665	0.664	0.632	0.641	0.633	0.613	0.612	0.613	0.653
TI_Angina	0.756	0.758	0.756	0.745	0.74	0.743	0.743	0.737	0.737	0.733	0.727	0.725
TI_Arthritis	0.611	0.613	0.62	0.62	0.606	0.606	0.611	0.615	0.607	0.612	0.61	0.589
TI_Cancer	0.541	0.531	0.547	0.509	0.527	0.522	0.517	0.52	0.524	0.534	0.519	0.52
TI_Cataract	0.686	0.684	0.653	0.669	0.671	0.666	0.659	0.66	0.657	0.65	0.652	0.641
TI_Diabetes	0.753	0.744	0.747	0.745	0.738	0.745	0.741	0.742	0.747	0.754	0.748	0.752
TI_HBP	0.647	0.657	0.651	0.647	0.631	0.638	0.641	0.631	0.637	0.642	0.634	0.65
TI_HeartAttack	0.73	0.732	0.735	0.73	0.745	0.741	0.719	0.719	0.722	0.722	0.718	0.7
TI_Ministroke	0.685	0.689	0.672	0.67	0.649	0.659	0.667	0.668	0.661	0.653	0.656	0.665
TI_Osteoporosis	0.66	0.653	0.662	0.662	0.654	0.672	0.667	0.667	0.673	0.666	0.667	0.661
TI_Stroke	0.663	0.678	0.67	0.665	0.664	0.632	0.641	0.633	0.613	0.612	0.613	0.597
AvgRank Elsanurse	8.25	7.15	5.75	5.30	5.30	7.90	**5.00**	6.55	5.40	6.85	7.85	6.70
AvgRank Elsacore	4.10	4.00	**3.90**	5.50	7.90	6.85	7.85	7.85	7.65	7.65	9.20	5.55
AvgRank TILDA	4.10	3.90	**3.70**	5.35	7.70	6.45	7.35	7.30	7.20	7.15	8.75	9.05
AvgRank Overall	5.48	5.02	**4.45**	5.38	6.97	7.07	6.73	7.23	6.75	7.22	8.60	7.10

Table D.11: C4.5 decision tree Specificity results for threshold selection experiments in the Baseline+CTF datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments. The last column, DD, refers to the data-driven approach, which uses internal cross-validation to select the threshold value for each decision tree.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**DD**
EN_Angina	0.597	0.593	0.605	0.581	0.612	0.605	0.597	0.601	0.581	0.554	0.593	0.581
EN_Arthritis	0.564	0.567	0.553	0.564	0.559	0.553	0.555	0.569	0.565	0.562	0.56	0.555
EN_Cataract	0.575	0.593	0.566	0.584	0.588	0.574	0.578	0.579	0.575	0.587	0.58	0.578
EN_Dementia	0.635	0.655	0.682	0.655	0.703	0.676	0.669	0.662	0.676	0.669	0.676	0.635
EN_Diabetes	0.81	0.807	0.796	0.8	0.789	0.789	0.79	0.795	0.797	0.794	0.799	0.796
EN_HBP	0.607	0.592	0.608	0.612	0.593	0.59	0.594	0.6	0.591	0.587	0.589	0.599
EN_HeartAttack	0.636	0.613	0.636	0.633	0.594	0.589	0.581	0.576	0.591	0.586	0.566	0.618
EN_Osteoporosis	0.607	0.59	0.59	0.583	0.584	0.596	0.589	0.61	0.584	0.589	0.602	0.587
EN_Parkinsons	0.591	0.545	0.561	0.53	0.515	0.515	0.545	0.5	0.515	0.515	0.561	0.606
EN_Stroke	0.639	0.618	0.637	0.606	0.596	0.641	0.639	0.637	0.61	0.62	0.641	0.622
EC_Angina	0.7	0.76	0.728	0.708	0.724	0.744	0.728	0.704	0.712	0.748	0.752	0.72
EC_Arthritis	0.611	0.603	0.611	0.589	0.615	0.617	0.607	0.621	0.598	0.61	0.612	0.59
EC_Cataract	0.51	0.503	0.477	0.556	0.533	0.566	0.52	0.53	0.523	0.52	0.497	0.549
EC_Dementia	0.64	0.632	0.663	0.623	0.642	0.649	0.628	0.636	0.619	0.626	0.628	0.634
EC_Diabetes	0.74	0.732	0.727	0.725	0.738	0.748	0.756	0.738	0.769	0.766	0.758	0.743
EC_HBP	0.624	0.629	0.617	0.64	0.632	0.623	0.638	0.616	0.618	0.629	0.618	0.625
EC_HeartAttack	0.698	0.717	0.693	0.702	0.683	0.717	0.766	0.79	0.766	0.766	0.751	0.741
EC_Osteoporosis	0.647	0.647	0.647	0.657	0.647	0.627	0.618	0.637	0.676	0.637	0.627	0.686
EC_Parkinsons	0.637	0.687	0.685	0.657	0.654	0.667	0.617	0.665	0.654	0.656	0.654	0.656
EC_Stroke	0.662	0.569	0.554	0.554	0.585	0.569	0.554	0.538	0.538	0.554	0.554	0.6
TI_Angina	0.7	0.76	0.728	0.708	0.724	0.744	0.728	0.704	0.712	0.748	0.752	0.764
TI_Arthritis	0.611	0.603	0.611	0.589	0.615	0.617	0.607	0.621	0.598	0.61	0.612	0.611
TI_Cancer	0.51	0.503	0.477	0.556	0.533	0.566	0.52	0.53	0.523	0.52	0.497	0.563
TI_Cataract	0.64	0.632	0.663	0.623	0.642	0.649	0.628	0.636	0.619	0.626	0.628	0.665
TI_Diabetes	0.74	0.732	0.727	0.725	0.738	0.748	0.756	0.738	0.769	0.766	0.758	0.714
TI_HBP	0.624	0.629	0.617	0.64	0.632	0.623	0.638	0.616	0.618	0.629	0.618	0.601
TI_HeartAttack	0.698	0.717	0.693	0.702	0.683	0.717	0.766	0.79	0.766	0.766	0.751	0.756
TI_Ministroke	0.647	0.647	0.647	0.657	0.647	0.627	0.618	0.637	0.676	0.637	0.627	0.657
TI_Osteoporosis	0.637	0.687	0.685	0.657	0.654	0.667	0.617	0.665	0.654	0.656	0.654	0.661
TI_Stroke	0.662	0.569	0.554	0.554	0.585	0.569	0.554	0.538	0.538	0.554	0.554	0.692
AvgRank Elsanurse	**4.40**	5.50	5.25	6.50	6.85	7.35	7.15	6.05	7.75	8.25	5.85	7.10
AvgRank Elsacore	7.20	6.00	7.25	7.30	5.80	**4.75**	7.05	6.75	7.30	6.00	7.05	5.55
AvgRank TILDA	7.25	6.10	7.30	7.35	5.90	4.85	7.25	6.65	7.20	6.25	7.15	**4.75**
AvgRank Overall	6.28	5.87	6.60	7.05	6.18	**5.65**	7.15	6.48	7.42	6.83	6.68	5.80

Table D.12: C4.5 decision tree Accuracy results for threshold selection experiments in the Baseline+CTF datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments. The last column, DD, refers to the data-driven approach, which uses internal cross-validation to select the threshold value for each decision tree.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**DD**
EN_Angina	0.58	0.595	0.595	0.603	0.605	0.598	0.617	0.605	0.6	0.587	0.594	0.598
EN_Arthritis	0.564	0.567	0.564	0.564	0.564	0.557	0.56	0.563	0.564	0.558	0.56	0.561
EN_Cataract	0.581	0.589	0.593	0.591	0.603	0.581	0.591	0.587	0.591	0.593	0.589	0.585
EN_Dementia	0.672	0.677	0.668	0.663	0.667	0.674	0.678	0.672	0.671	0.666	0.667	0.683
EN_Diabetes	0.796	0.803	0.801	0.799	0.795	0.792	0.799	0.796	0.795	0.794	0.79	0.796
EN_HBP	0.616	0.608	0.612	0.622	0.6	0.602	0.609	0.604	0.605	0.598	0.596	0.603
EN_HeartAttack	0.632	0.626	0.639	0.644	0.635	0.647	0.634	0.633	0.637	0.645	0.638	0.637
EN_Osteoporosis	0.603	0.6	0.599	0.604	0.609	0.597	0.592	0.603	0.606	0.605	0.601	0.601
EN_Parkinsons	0.584	0.6	0.598	0.61	0.604	0.626	0.626	0.636	0.635	0.634	0.614	0.604
EN_Stroke	0.627	0.617	0.628	0.626	0.63	0.615	0.633	0.628	0.626	0.634	0.638	0.624
EC_Angina	0.753	0.758	0.755	0.743	0.74	0.743	0.743	0.736	0.736	0.734	0.728	0.744
EC_Arthritis	0.611	0.61	0.617	0.611	0.608	0.61	0.609	0.617	0.604	0.612	0.61	0.608
EC_Cataract	0.54	0.53	0.543	0.511	0.527	0.524	0.518	0.521	0.524	0.533	0.518	0.525
EC_Dementia	0.682	0.68	0.654	0.665	0.669	0.664	0.657	0.658	0.654	0.648	0.65	0.654
EC_Diabetes	0.752	0.743	0.746	0.744	0.738	0.746	0.742	0.742	0.748	0.755	0.749	0.748
EC_HBP	0.638	0.646	0.638	0.644	0.631	0.633	0.64	0.626	0.63	0.637	0.628	0.637
EC_HeartAttack	0.729	0.732	0.733	0.729	0.743	0.741	0.721	0.722	0.723	0.723	0.719	0.724
EC_Osteoporosis	0.684	0.688	0.672	0.669	0.649	0.658	0.666	0.667	0.661	0.652	0.655	0.671
EC_Parkinsons	0.658	0.656	0.664	0.662	0.654	0.671	0.662	0.667	0.671	0.665	0.665	0.664
EC_Stroke	0.663	0.677	0.668	0.664	0.663	0.631	0.64	0.631	0.612	0.612	0.612	0.652
TI_Angina	0.753	0.758	0.755	0.743	0.74	0.743	0.743	0.736	0.736	0.734	0.728	0.727
TI_Arthritis	0.611	0.61	0.617	0.611	0.608	0.61	0.609	0.617	0.604	0.612	0.61	0.595
TI_Cancer	0.54	0.53	0.543	0.511	0.527	0.524	0.518	0.521	0.524	0.533	0.518	0.522
TI_Cataract	0.682	0.68	0.654	0.665	0.669	0.664	0.657	0.658	0.654	0.648	0.65	0.643
TI_Diabetes	0.752	0.743	0.746	0.744	0.738	0.746	0.742	0.742	0.748	0.755	0.749	0.749
TI_HBP	0.638	0.646	0.638	0.644	0.631	0.633	0.64	0.626	0.63	0.637	0.628	0.632
TI_HeartAttack	0.729	0.732	0.733	0.729	0.743	0.741	0.721	0.722	0.723	0.723	0.719	0.702
TI_Ministroke	0.684	0.688	0.672	0.669	0.649	0.658	0.666	0.667	0.661	0.652	0.655	0.665
TI_Osteoporosis	0.658	0.656	0.664	0.662	0.654	0.671	0.662	0.667	0.671	0.665	0.665	0.661
TI_Stroke	0.663	0.677	0.668	0.664	0.663	0.631	0.64	0.631	0.612	0.612	0.612	0.598
AvgRank Elsanurse	7.65	6.80	5.85	5.20	5.70	8.25	**5.35**	5.90	5.45	6.65	8.00	7.20
AvgRank Elsacore	**3.90**	4.10	3.90	5.85	7.70	6.10	7.95	7.60	8.15	7.15	9.20	6.40
AvgRank TILDA	3.90	4.10	**3.80**	5.55	7.55	5.70	7.45	7.20	7.65	6.70	8.85	9.55
AvgRank Overall	5.15	5.00	**4.52**	5.53	6.98	6.68	6.92	6.90	7.08	6.83	8.68	7.72

Table D.13: C4.5 decision tree GMean results for threshold selection experiments in the Baseline+CTF datasets, varying threshold values from 0.0 to 0.05, in 0.005 increments. The last column, DD, refers to the data-driven approach, which uses internal cross-validation to select the threshold value for each decision tree.

Threshold	**0.0**	**0.005**	**0.01**	**0.015**	**0.02**	**0.025**	**0.03**	**0.035**	**0.04**	**0.045**	**0.05**	**DD**
EN_Angina	0.588	0.594	0.6	0.593	0.609	0.601	0.607	0.603	0.591	0.571	0.593	0.59
EN_Arthritis	0.564	0.567	0.563	0.564	0.564	0.556	0.559	0.564	0.564	0.558	0.56	0.56
EN_Cataract	0.579	0.59	0.585	0.59	0.599	0.579	0.587	0.585	0.587	0.592	0.587	0.583
EN_Dementia	0.654	0.667	0.675	0.659	0.684	0.675	0.673	0.667	0.673	0.667	0.671	0.659
EN_Diabetes	0.802	0.804	0.799	0.8	0.793	0.79	0.795	0.795	0.796	0.794	0.794	0.796
EN_HBP	0.615	0.605	0.612	0.62	0.599	0.6	0.606	0.603	0.602	0.596	0.595	0.603
EN_HeartAttack	0.634	0.62	0.638	0.639	0.615	0.619	0.608	0.605	0.615	0.616	0.603	0.628
EN_Osteoporosis	0.605	0.595	0.595	0.594	0.598	0.597	0.591	0.606	0.596	0.597	0.602	0.595
EN_Parkinsons	0.588	0.572	0.579	0.569	0.558	0.568	0.585	0.564	0.572	0.572	0.587	0.605
EN_Stroke	0.633	0.617	0.632	0.616	0.614	0.627	0.636	0.632	0.619	0.627	0.64	0.623
EC_Angina	0.727	0.759	0.742	0.726	0.732	0.743	0.736	0.72	0.725	0.74	0.74	0.732
EC_Arthritis	0.611	0.608	0.615	0.605	0.61	0.612	0.609	0.618	0.603	0.611	0.611	0.603
EC_Cataract	0.525	0.517	0.511	0.532	0.53	0.543	0.519	0.525	0.524	0.527	0.508	0.536
EC_Dementia	0.663	0.657	0.658	0.646	0.657	0.657	0.643	0.648	0.638	0.637	0.64	0.645
EC_Diabetes	0.746	0.738	0.737	0.735	0.738	0.747	0.748	0.74	0.758	0.76	0.753	0.745
EC_HBP	0.635	0.643	0.634	0.643	0.632	0.631	0.639	0.624	0.628	0.635	0.626	0.635
EC_HeartAttack	0.713	0.725	0.713	0.716	0.713	0.729	0.742	0.754	0.743	0.743	0.734	0.732
EC_Osteoporosis	0.666	0.668	0.66	0.663	0.648	0.643	0.642	0.652	0.669	0.645	0.641	0.678
EC_Parkinsons	0.648	0.67	0.673	0.66	0.654	0.669	0.642	0.666	0.663	0.661	0.66	0.66
EC_Stroke	0.662	0.621	0.609	0.607	0.623	0.6	0.596	0.584	0.574	0.582	0.582	0.626
TI_Angina	0.727	0.759	0.742	0.726	0.732	0.743	0.736	0.72	0.725	0.74	0.74	0.744
TI_Arthritis	0.611	0.608	0.615	0.605	0.61	0.612	0.609	0.618	0.603	0.611	0.611	0.6
TI_Cancer	0.525	0.517	0.511	0.532	0.53	0.543	0.519	0.525	0.524	0.527	0.508	0.541
TI_Cataract	0.663	0.657	0.658	0.646	0.657	0.657	0.643	0.648	0.638	0.637	0.64	0.653
TI_Diabetes	0.746	0.738	0.737	0.735	0.738	0.747	0.748	0.74	0.758	0.76	0.753	0.733
TI_HBP	0.635	0.643	0.634	0.643	0.632	0.631	0.639	0.624	0.628	0.635	0.626	0.625
TI_HeartAttack	0.713	0.725	0.713	0.716	0.713	0.729	0.742	0.754	0.743	0.743	0.734	0.727
TI_Ministroke	0.666	0.668	0.66	0.663	0.648	0.643	0.642	0.652	0.669	0.645	0.641	0.661
TI_Osteoporosis	0.648	0.67	0.673	0.66	0.654	0.669	0.642	0.666	0.663	0.661	0.66	0.661
TI_Stroke	0.662	0.621	0.609	0.607	0.623	0.6	0.596	0.584	0.574	0.582	0.582	0.643
AvgRank Elsanurse	5.25	5.55	**5.05**	6.15	6.45	7.90	6.20	6.50	6.85	7.95	6.85	7.30
AvgRank Elsacore	5.95	5.20	5.90	7.15	7.20	**5.10**	7.40	6.65	7.50	6.05	8.10	5.80
AvgRank TILDA	5.80	**5.00**	5.80	7.10	7.05	**5.00**	7.50	6.65	7.25	6.15	8.15	6.55
AvgRank Overall	5.67	**5.25**	5.58	6.80	6.90	6.00	7.03	6.60	7.20	6.72	7.70	6.55

Table D.14: C4.5 decision tree Comparison of Lexic and NoLexic approaches for Baseline+CTF datasets.

Baseline+CTFs Datasets	**SENSITIVITY**		**SPECIFICITY**		**ACCURACY**		**GMEAN**	
	Lexic	**NoLexic**	**Lexic**	**NoLexic**	**Lexic**	**NoLexic**	**Lexic**	**NoLexic**
EN_Angina	0.598	0.608	0.581	0.554	0.598	0.606	0.59	0.581
EN_Arthritis	0.565	0.557	0.555	0.561	0.561	0.559	0.56	0.559
EN_Cataract	0.588	0.604	0.578	0.579	0.585	0.596	0.583	0.591
EN_Dementia	0.684	0.658	0.635	0.628	0.683	0.657	0.659	0.643
EN_Diabetes	0.796	0.8	0.796	0.805	0.796	0.801	0.796	0.803
EN_HBP	0.606	0.615	0.599	0.589	0.603	0.604	0.603	0.602
EN_HeartAttack	0.639	0.633	0.618	0.608	0.637	0.632	0.628	0.621
EN_Osteoporosis	0.602	0.603	0.587	0.638	0.601	0.606	0.595	0.62
EN_Parkinsons	0.604	0.612	0.606	0.652	0.604	0.613	0.605	0.632
EN_Stroke	0.624	0.632	0.622	0.613	0.624	0.631	0.623	0.622
EC_Angina	0.745	0.725	0.72	0.764	0.744	0.727	0.732	0.744
EC_Arthritis	0.616	0.589	0.59	0.611	0.608	0.595	0.603	0.6
EC_Cataract	0.524	0.52	0.549	0.563	0.525	0.522	0.536	0.541
EC_Dementia	0.656	0.641	0.634	0.665	0.654	0.643	0.645	0.653
EC_Diabetes	0.748	0.752	0.743	0.714	0.748	0.749	0.745	0.733
EC_HBP	0.645	0.65	0.625	0.601	0.637	0.632	0.635	0.625
EC_HeartAttack	0.723	0.7	0.741	0.756	0.724	0.702	0.732	0.727
EC_Osteoporosis	0.671	0.665	0.686	0.657	0.671	0.665	0.678	0.661
EC_Parkinsons	0.664	0.661	0.656	0.661	0.664	0.661	0.66	0.661
EC_Stroke	0.653	0.597	0.6	0.692	0.652	0.598	0.626	0.643
TI_Angina	0.725	0.756	0.764	0.7	0.727	0.753	0.744	0.727
TI_Arthritis	0.589	0.611	0.611	0.611	0.595	0.611	0.6	0.611
TI_Cancer	0.52	0.541	0.563	0.51	0.522	0.54	0.541	0.525
TI_Cataract	0.641	0.686	0.665	0.64	0.643	0.682	0.653	0.663
TI_Diabetes	0.752	0.753	0.714	0.74	0.749	0.752	0.733	0.746
TI_HBP	0.65	0.647	0.601	0.624	0.632	0.638	0.625	0.635
TI_HeartAttack	0.7	0.73	0.756	0.698	0.702	0.729	0.727	0.713
TI_Ministroke	0.665	0.685	0.657	0.647	0.665	0.684	0.661	0.666
TI_Osteoporosis	0.661	0.66	0.661	0.637	0.661	0.658	0.661	0.648
TI_Stroke	0.597	0.663	0.692	0.662	0.598	0.663	0.643	0.662
AvgRank Elsanurse	1.70	**1.30**	**1.50**	**1.50**	1.70	**1.30**	**1.40**	1.60
AvgRank Elsacore	**1.20**	1.80	1.70	**1.30**	**1.10**	1.90	**1.50**	**1.50**
AvgRank TILDA	1.80	**1.20**	**1.25**	1.75	1.90	**1.10**	1.60	**1.40**
AvgRank Overall	1.57	**1.43**	**1.48**	1.52	1.57	**1.43**	**1.50**	**1.50**

Appendix E

Feature Importance Analysis of Random Forest Models Trained with the Lexicographic Split Approach

In this Appendix we present the 10 best-ranked features, i.e., those with highest average impurity decrease values, for all RF models which we did not discuss in Section 6.5. All models used for this analysis were trained using the entire datasets (no training and test data division), undersampled to a 1:1 ratio using the Balanced Random Forest method, with missing values replaced using the data-driven MVR approach (Chapter 4) and the lexicographic split function (Chapter 6). The number of trees in these Random Forests was increased from the default 100 to 1000, to get more precise feature importance results.

Tables E.1 to E.9 show the models created from ELSA-nurse datasets, from the least imbalanced class to the most imbalanced. Tables E.10 to E.18 show the models created from ELSA-core datasets, and the models created with TILDA datasets are presented in Tables E.19 to E.27, all ordered from the least imbalanced class to the most imbalanced. In all Tables we present the 10 top-ranked features, their description and average impurity decrease (AID) value.

Table E.1: The 10 top-ranked features for the ELSA-nurse RF model, class: Arthritis (Imbalance Ratio: 1.35).

Feature	Description	AID
chestin_w2	Lung function: Whether had any respiratory infection in last 3 weeks	0.47
hastro_w2	Whether been admitted to hospital with a heart complaint in the past month	0.46
mmlsre_w2	Leg raise (eyes shut): Outcome	0.44
eyesurg_w4	Whether have a detached retina or had eye or ear surgery in the past 3 months	0.43
mmcrre_w2	Single chair rise outcome	0.43
apoe_w2	Blood apolipoprotein E (apoE) level (mmol/l)	0.42
cfib_w4	Blood Fibrinogen level (g/l)	0.42
fglu_w4	Blood glucose level while fasting (mmol/L)	0.42
hdl_w2	Blood high-density lipoprotein level (mmol/l)	0.41
hba1c_w2	Blood glycated haemoglobin level (mmol/mol)	0.41

Table E.2: The 10 top-ranked features for the ELSA-nurse RF model, class: High Blood Pressure (Imbalance Ratio: 1.49).

Feature	Description	AID
sex	Sex of the participant (male/female)	0.54
chestin_w6	Lung function: Whether had any respiratory infection in last 3 weeks	0.46
eyesurg_w4	Whether have a detached retina or had eye or ear surgery in the past 3 months	0.44
indager_w8	Age of the participant at a given wave	0.43
clotb_w4	Blood sample: whether has clotting disorder	0.42
cfib_w8	Blood Fibrinogen level (g/l)	0.42
clotb_w6	Blood sample: whether has clotting disorder	0.42
hastro_w2	Whether been admitted to hospital with a heart complaint in the past month	0.42
diaval_w8	Mean diastolic blood pressure	0.42
igf1_w8	Blood insulin-like growth factor (IGF-1) level (nmol/l)	0.41

Table E.3: The 10 top-ranked features for the ELSA-nurse RF model, class: Cataract (Imbalance Ratio: 2.06).

Feature	Description	AID
eyesurg_w4	Whether have a detached retina or had eye or ear surgery in the past 3 months	0.56
clotb_w4	Blood sample: whether has clotting disorder	0.42
bmiobe_w6	Body mass index grouped according to WHO definitions	0.41
bmiobe_w4	Body mass index grouped according to WHO definitions	0.4
cfib_w6	Blood Fibrinogen level (g/l)	0.4
htpf_w2	LUNG: Highest technically satisfactory value for Peak Flow	0.4
chol_w4	Blood total cholesterol level (mmol/l)	0.39
chol_w6	Blood total cholesterol level (mmol/l)	0.39
htfev_w6	LUNG: Highest technically satisfactory value for Forced Expiratory Volume	0.39
trig_w2	Blood triglyceride level (mmol/l)	0.39

Table E.4: The 10 top-ranked features for the ELSA-nurse RF model, class: Osteoporosis (Imbalance Ratio: 9.85).

Feature	Description	AID
hasurg_w4	Whether had abdominal or chest surgery in the past 3 months	1
clotb_w4	Blood sample: whether has clotting disorder	0.92
mmssre_w2	Outcome of semi-tandem stand	0.92
mmcrre_w4	Single chair rise outcome	0.54
hastro_w2	Whether been admitted to hospital with a heart complaint in the past month	0.54
clotb_w6	Blood sample: whether has clotting disorder	0.51
chestin_w6	Lung function: Whether had any respiratory infection in last 3 weeks	0.49
hastro_w4	Whether been admitted to hospital with a heart complaint in the past month	0.47
bmiobe_w6	Body mass index grouped according to WHO definitions	0.46
chol_w8	Blood total cholesterol level (mmol/l)	0.44

Table E.5: The 10 top-ranked features for the ELSA-nurse RF model, class: Stroke (Imbalance Ratio: 15.86).

Feature	Description	AID
hastro_w4	Whether been admitted to hospital with a heart complaint in the past month	0.65
mmssre_w4	Outcome of semi-tandem stand	0.57
eyesurg_w2	Whether have a detached retina or had eye or ear surgery in the past 3 months	0.56
sex	Sex of the participant (male/female)	0.46
mmssre_w2	Outcome of semi-tandem stand	0.45
hdl_w6	Blood High-density lipoprotein (HDL) level (mmol/l)	0.45
htval_w6	Height (cm)	0.43
hastro_w2	Whether been admitted to hospital with a heart complaint in the past month	0.43
pulval_w2	Pulse pressure	0.43
clotb_w2	Blood sample: whether has clotting disorder	0.43

Table E.6: The 10 top-ranked features for the ELSA-nurse RF model, class: Heart Attack (Imbalance Ratio: 16.7).

Feature	Description	AID
clotb_w4	Blood sample: whether has clotting disorder	0.76
eyesurg_w6	Whether have a detached retina or had eye or ear surgery in the past 3 months	0.45
htval_w2	Height (cm)	0.45
trig_w2	Blood triglyceride level (mmol/l)	0.44
mmlsre_w2	Leg raise (eyes shut): Outcome	0.43
diaval_w2	Mean diastolic blood pressure	0.42
hipval_w4	Mean hip (cm)	0.42
fglu_w6	Blood glucose level while fasting (mmol/L)	0.42
clotb_w6	Blood sample: whether has clotting disorder	0.42
bmiobe_w4	Body mass index grouped according to WHO definitions	0.42

Table E.7: The 10 top-ranked features for the ELSA-nurse RF model, class: Angina (Imbalance Ratio: 26.51).

Feature	Description	AID
chestin_w4	Lung function: Whether had any respiratory infection in last 3 weeks	0.6
chestin_w2	Lung function: Whether had any respiratory infection in last 3 weeks	0.5
rtin_w2	Blood ferritin level (ng/ml)	0.5
mapval_w2	Mean arterial pressure	0.48
hasurg_w4	Whether had abdominal or chest surgery in the past 3 months	0.47
hba1c_w2	Blood glycated haemoglobin level (mmol/mol)	0.47
hscrp_w2	Blood C-reactive protein (CRP) level (mg/l)	0.46
hipval_w2	Mean hip (cm)	0.46
mmgsdavg_w2	Mean grip strenght with dominant hand	0.46
bmiobe_w2	Body mass index grouped according to WHO definitions	0.45

Table E.8: The 10 top-ranked features for the ELSA-nurse RF model, class: Dementia (Imbalance Ratio: 59.96).

Feature	Description	AID
hastro_w6	Whether been admitted to hospital with a heart complaint in the past month	1
chestin_w2	Lung function: Whether had any respiratory infection in last 3 weeks	0.92
mmstre_w2	Outcome of semi-tandem stand	0.92
hastro_w4	Whether been admitted to hospital with a heart complaint in the past month	0.72
hastro_w2	Whether been admitted to hospital with a heart complaint in the past month	0.67
trig_w4	Blood triglyceride level (mmol/l)	0.54
bmiobe_w6	Body mass index grouped according to WHO definitions	0.53
igf1_w8	Blood insulin-like growth factor (IGF-1) level (nmol/l)	0.52
mapval_w2	Mean arterial pressure	0.52
wtval_w4	Weight (Kg)	0.52

Table E.9: The 10 top-ranked features for the ELSA-nurse RF model, class: Parkinsons (Imbalance Ratio: 160.3).

Feature	Description	AID
hdl_w4	Blood high-density lipoprotein level (mmol/l)	0.63
wtval_w2	Weight (Kg)	0.62
hgb_w6	Blood haemoglobin level (g/dl)	0.61
cfib_w4	Blood Fibrinogen level (g/l)	0.61
hba1c_w2	Blood glycated haemoglobin level (mmol/mol)	0.59
bmiobe_w2	Body mass index grouped according to WHO definitions	0.52
pulval_w4	Pulse pressure	0.51
pulval_w2	Pulse pressure	0.51
chestin_w2	Lung function: Whether had any respiratory infection in last 3 weeks	0.5
mmcrre_w4	Single chair rise outcome	0.5

Table E.10: The 10 top-ranked features for the ELSA-core RF model, class: Arthritis (Imbalance Ratio: 2.52).

Feature	Description	AID
cesd_w7	Depression questionnaire score	0.56
indager_w8	Age of the participant at a given wave	0.56
dicdnm_w7	Cause of death of mother of respondent	0.56
cesd_w5	Depression questionnaire score	0.56
heiadlX-of-9_w6	Reported IADL difficulties (count)	0.54
cesd_w6	Depression questionnaire score	0.54
cesd_w2	Depression questionnaire score	0.54
cfmetper_w4	Perception of memory compared to 2 years ago	0.54
cfmetm_w7	Self-rated memory	0.54
cfmetper_w7	Perception of memory compared to 2 years ago	0.54

Table E.11: The 10 top-ranked features for the ELSA-core RF model, class: High Blood Pressure (Imbalance Ratio: 2.58).

Feature	Description	AID
indager_w8	Age of the participant at a given wave	0.57
cesd_w4	Depression questionnaire score	0.56
hepain_w2	Whether often troubled with pain	0.55
cesd_w7	Depression questionnaire score	0.55
cesd_w1	Depression questionnaire score	0.55
cfmetper_w4	Perception of memory compared to 2 years ago	0.55
cesd_w5	Depression questionnaire score	0.55
cesd_w2	Depression questionnaire score	0.55
cesd_w6	Depression questionnaire score	0.54
headlno_w5	Reported difficulty with ADL or IADL	0.54

Table E.12: The 10 top-ranked features for the ELSA-core RF model, class: Cataract (Imbalance Ratio: 3.38).

Feature	Description	AID
indager_w8	Age of the participant at a given wave	0.58
cesd_w7	Depression questionnaire score	0.57
dicdnm_w7	Cause of death of mother of respondent	0.57
cfmetper_w4	Perception of memory compared to 2 years ago	0.55
hemobno_w6	Reported difficulties with mobility (binary)	0.55
hemobX-of-10_w7	Reported mobility issues (count)	0.54
hemobX-of-10_w6	Reported mobility issues (count)	0.54
cesd_w6	Depression questionnaire score	0.54
cesd_w5	Depression questionnaire score	0.54
heacta_w6	Frequency does vigorous sports or activities	0.54

Table E.13: The 10 top-ranked features for the ELSA-core RF model, class: Diabetes (Imbalance Ratio: 7.80).

Feature	Description	AID
hefrac_w1	Whether has fractured hip	0.64
helng_w4	Whether taking medication for lung condition	0.63
heji_w3	Whether had joint replacement	0.63
heam_w2	Whether taking medication for asthma	0.63
indager_w7	Age of the participant at a given wave	0.6
cesd_w7	Depression questionnaire score	0.59
cesd_w6	Depression questionnaire score	0.58
hemobX-of-10_w7	Reported mobility issues (count)	0.58
cfmetper_w7	Perception of memory compared to 2 years ago	0.58
headlno_w6	Reported difficulty with ADL or IADL	0.58

Table E.14: The 10 top-ranked features for the ELSA-core RF model, class: Osteoporosis (Imbalance Ratio: 11.84).

Feature	Description	AID
hecanb_w1	Cancer: whether received treatment in last 2 years	0.67
heji_w1	Whether had joint replacement	0.63
cfmetper_w4	Perception of memory compared to 2 years ago	0.62
hechm_w6	Cholesterol: whether taking cholesterol medication	0.61
heyrc_w5	Experienced psychiatric problems in last 2 years	0.61
cesd_w7	Depression questionnaire score	0.6
indager_w7	Age of the participant at a given wave	0.6
hepain_w4	Whether often troubled with pain	0.6
heji_w2	Whether had joint replacement	0.59
cfmetper_w7	Perception of memory compared to 2 years ago	0.59

Table E.15: The 10 top-ranked features for the ELSA-core RF model, class: Stroke (Imbalance Ratio: 18.35).

Feature	Description	AID
hefrac_w5	Whether has fractured hip	0.88
hefrac_w1	Whether has fractured hip	0.74
hefrac_w4	Whether has fractured hip	0.68
heyrc_w1	Experienced psychiatric problems in last 2 years	0.66
heill_w5	Whether has self-reported long-standing illness	0.61
hepawX-of-7_w6	Pain reported (count)	0.61
hepain_w5	Whether often troubled with pain	0.61
hecanb_w3	Cancer: whether received treatment in last 2 years	0.6
hefrac_w6	Whether has fractured hip	0.6
hemobno_w3	Reported difficulties with mobility	0.6

Table E.16: The 10 top-ranked features for the ELSA-core RF model, class: Heart Attack (Imbalance Ratio: 19.06).

Feature	Description	AID
hefrac_w1	Whether has fractured hip	0.92
helng_w4	Whether taking medication for lung condition	0.88
hecanb_w6	Cancer: whether received treatment in last 2 years	0.73
hepawX-of-7_w7	Pain reported (count)	0.66
hecanb_w1	Cancer: whether received treatment in last 2 years	0.65
heiadlX-of-9_w7	Reported IADL difficulties (count)	0.65
hefrac_w3	Whether has fractured hip	0.65
heam_w6	Whether taking medication for asthma	0.54
cesd_w7	Depression questionnaire score	0.63
cesd_w6	Depression questionnaire score	0.62

Table E.17: The 10 top-ranked features for the ELSA-core RF model, class: Angina (Imbalance Ratio: 29.49).

Feature	Description	AID
heji_w3	Whether had joint replacement	0.92
hefrac_w3	Whether has fractured hip	0.92
hecanb_w1	Cancer: whether received treatment in last 2 years	0.92
helng_w3	Whether taking medication for lung condition	0.68
cfmetper_w3	Perception of memory compared to 2 years ago	0.68
hefrac_w5	Whether has fractured hip	0.65
cfmetper_w7	Perception of memory compared to 2 years ago	0.64
heacta_w1	Frequency does vigorous sports or activities	0.64
heill_w1	Whether has self-reported long-standing illness	0.62
cfmetper_w4	Perception of memory compared to 2 years ago	0.61

Table E.18: The 10 top-ranked features for the ELSA-core RF model, class: Parkinsons (Imbalance Ratio: 112.07).

Feature	Description	AID
heji_w5	Whether had joint replacement	1
heyrc_w4	Experienced psychiatric problems in last 2 years	0.95
helng_w4	Whether taking medication for lung condition	0.81
hepawX-of-7_w5	Pain reported (count)	0.76
hecanb_w4	Cancer: whether received treatment in last 2 years	0.74
hepsyX-of-9_w5	Psychiatric problems reported (count)	0.74
heji_w4	Whether had joint replacement	0.72
heiadlX-of-9_w5	Reported IADL difficulties (count)	0.71
cfmetper_w7	Perception of memory compared to 2 years ago	0.7
hechm_w3	Cholesterol: whether taking cholesterol medication	0.7

Table E.19: The 10 top-ranked features for the TILDA RF model, class: High Blood Pressure (Imbalance Ratio: 2.38).

Feature	Description	AID
sex	Sex of the participant (male/female)	0.54
ph505_w4	Takes pain medication to control pain	0.53
ph601_w4	Did you lose any urine beyond your control in the last year	0.49
indager_w4	Age of the participant at a given wave	0.49
bh107_w4	Hours spent sitting in a typical day	0.48
ph415_w4	Had any joint replacements	0.46
ipaqmetminutes_w4	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	0.46
behcage_w4	Count of CAGE questionnaire responses (measures alcoholism)	0.46
ph008_w4	Have you lost at least 4.5kg without trying in the past year	0.46
ph505_w3	Takes pain medication to control pain	0.46

Table E.20: The 10 top-ranked features for the TILDA RF model, class: Arthritis (Imbalance Ratio: 2.92).

Feature	Description	AID
sex	Sex of the participant (male/female)	0.52
bh107_w4	Hours spent sitting in a typical day	0.5
mdmeds_excl_supps_w4	Number of medications reported by respondent (excluding supplements)	0.49
ph601_w4	Did you lose any urine beyond your control in the last year	0.48
behcage_w4	Count of CAGE questionnaire responses (measures alcoholism)	0.48
ipaqmetminutes_w4	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	0.47
indager_w4	Age of the participant at a given wave	0.46
ph008_w4	Have you lost at least 4.5kg without trying in the past year	0.46
bphypertension_w3	Objective measured hypertension	0.45
mdmeds_excl_supps_w3	Number of medications reported by respondent (excluding supplements)	0.43

Table E.21: The 10 top-ranked features for the TILDA RF model, class: Osteoporosis (Imbalance Ratio: 9.53).

Feature	Description	AID
ph601_w4	Did you lose any urine beyond your control in the last year	0.53
ph008_w4	Have you lost at least 4.5kg without trying in the past year	0.5
bh107_w4	Hours spent sitting in a typical day	0.49
ph415_w4	Had any joint replacements	0.48
mdmeds_excl_supps_w4	Number of medications reported by respondent (excluding supplements)	0.47
ph505_w4	Takes pain medication to control pain	0.47
behcage_w4	Count of CAGE questionnaire responses (measures alcoholism)	0.47
bphypertension_w3	Objective measured hypertension	0.47
indager_w4	Age of the participant at a given wave	0.46
ipaqmetminutes_w4	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	0.46

Table E.22: The 10 top-ranked features for the TILDA RF model, class: Cataract (Imbalance Ratio: 10.83).

Feature	Description	AID
sex	Sex of the participant (male/female)	0.51
bphypertension_w3	Objective measured hypertension	0.48
ph601_w4	Did you lose any urine beyond your control in the last year	0.48
ph505_w3	Takes pain medication to control pain	0.48
ph505_w4	Takes pain medication to control pain	0.47
bh107_w4	Hours spent sitting in a typical day	0.46
bh107_w3	Hours spent sitting in a typical day	0.45
ph415_w4	Had any joint replacements	0.44
ipaqmetminutes_w4	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	0.43
indager_w4	Age of the participant at a given wave	0.43

Table E.23: The 10 top-ranked features for the TILDA RF model, class: Cancer (Imbalance Ratio: 17.02).

Feature	Description	AID
sex	Sex of the participant (male/female)	0.56
ph505_w4	Takes pain medication to control pain	0.54
bphypertension_w3	Objective measured hypertension	0.51
ph415_w4	Had any joint replacements	0.5
ph008_w4	Have you lost at least 4.5kg without trying in the past year	0.5
behcage_w4	Count of CAGE questionnaire responses (measures alcoholism)	0.49
ph601_w4	Did you lose any urine beyond your control in the last year	0.49
indager_w4	Age of the participant at a given wave	0.48
bh107_w4	Hours spent sitting in a typical day	0.45
ipaqmetminutes_w4	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	0.44

Table E.24: The 10 top-ranked features for the TILDA RF model, class: Angina (Imbalance Ratio: 20.70).

Feature	Description	AID
ph415_w4	Had any joint replacements	0.67
bphypertension_w3	Objective measured hypertension	0.58
sex	Sex of the participant (male/female)	0.53
ph601_w4	Did you lose any urine beyond your control in the last year	0.51
ph505_w4	Takes pain medication to control pain	0.49
ph601_w3	Did you lose any urine beyond your control in the last year	0.48
behcage_w2	Count of CAGE questionnaire responses (measures alcoholism)	0.48
mdmeds_excl_supps_w4	Number of medications reported by respondent (excluding supplements)	0.48
behcage_w4	Count of CAGE questionnaire responses (measures alcoholism)	0.47
bh107_w4	Hours spent sitting in a typical day	0.47

Table E.25: The 10 top-ranked features for the TILDA RF model, class: Heart Attack (Imbalance Ratio: 25.24).

Feature	Description	AID
ph505_w3	Takes pain medication to control pain	0.53
bphypertension_w3	Objective measured hypertension	0.51
behcage_w4	Count of CAGE questionnaire responses (measures alcoholism)	0.47
ph505_w4	Takes pain medication to control pain	0.47
hba1c_w1	Blood glycated haemoglobin level (mmol/mol)	0.47
behalc_freq_week_w1	Average amount of time respondent drinks a week	0.46
ph601_w4	Did you lose any urine beyond your control in the last year	0.46
bh107_w3	Hours spent sitting in a typical day	0.45
ph415_w1	Had any joint replacements	0.45
bh202_w1	How often do you have trouble sleeping	0.45

Table E.26: The 10 top-ranked features for the TILDA RF model, class: Mini-stroke (Imbalance Ratio: 50.74).

Feature	Description	AID
behcage_w1	Count of CAGE questionnaire responses (measures alcoholism)	0.59
ph415_w1	Had any joint replacements	0.57
ph505_w2	Takes pain medication to control pain	0.51
bphypertension_w1	Objective measured hypertension	0.51
behalc_freq_week_w1	Average amount of time respondent drinks a week	0.51
ph601_w1	Did you lose any urine beyond your control in the last year	0.5
ph402_w3	How many times have you fallen in this last year	0.48
ipaqmetminutes_w2	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	0.48
ph505_w3	Takes pain medication to control pain	0.48
bh202_w2	How often do you have trouble sleeping	0.47

Table E.27: The 10 top-ranked features for the TILDA RF model, class: Stroke (Imbalance Ratio: 79.62).

Feature	Description	AID
disadl_w2	Count of ADLs the respondent reported difficulty with (top coded at 5)	0.64
sex	Sex of the participant (male/female)	0.6
ph406_w1	How many times have you fainted in this last year	0.6
ph415_w2	Had any joint replacements	0.59
bphypertension_w3	Objective measured hypertension	0.57
ph601_w1	Did you lose any urine beyond your control in the last year	0.55
ph008_w2	Have you lost at least 4.5kg without trying in the past year	0.54
ipaqmetminutes_w4	Total met (metabolic equivalent) minutes spent on physical activities in last 7 days	0.54
disimpairments_w3	Physical impairments count (activities the respondent can't do)	0.51
ph601_w3	Did you lose any urine beyond your control in the last year	0.51

Bibliography

Abell, J. et al. (2018). *The Dynamics of ageing: Evidence from the English Longitudinal Study of Ageing 2002-2016 (Wave 8)*. London: Institute for Fiscal Studies.

Adhikari, S. et al. (2019). High-dimensional longitudinal classification with the multinomial fused lasso. *Statistics in medicine*, 38(12), pp. 2184–2205.

Aghili, M. et al. (2018). Predictive modeling of longitudinal data for alzheimers disease diagnosis using rnns. In *International Workshop on PRedictive Intelligence In MEdicine*, Springer, pp. 112–119.

Al-Otaibi, R., Flach, P. and Kull, M. (2014). Multi-label classification: A comparative study on threshold selection methods. In *First International Workshop on Learning over Multiple Contexts (LMCE) at ECML-PKDD 2014*, p. 8.

Al-Otaibi, R. et al. (2015). Versatile decision trees for learning over multiple contexts. In *Proc. European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2015)*, Springer, pp. 184–199.

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

Ali, K. and Pazzani, M. (1995). On the link between error correlation and error reduction in decision tree ensembles (Technical Report ICSTR-95-38). *Dept of Information and Computer Science, UCI, USA*.

Amaldi, E. and Kann, V. (1998). On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209(1-2), pp. 237–260.

Anagnostou, P. et al. (2021). Enhancing the human health status prediction: the athlos project. *Applied Artificial Intelligence*, pp. 1–23.

Banks, J. et al. (2019). English longitudinal study of ageing: Waves 0–8, 1998–2017.[data collection].

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

Belger, M. et al. (2016). How to deal with missing longitudinal data in cost of illness analysis in alzheimers diseasesuggestions from the geras observational study. *BMC Medical Research Methodology*, 16(1), p. 83.

Bennett, S. and Thomas, A. J. (2014). Depression and dementia: cause, consequence or coincidence? *Maturitas*, 79(2), pp. 184–190.

Beyer, K. et al. (1999). When is nearest neighbor meaningful? In *International conference on database theory*, Springer, pp. 217–235.

Bhagwat, N. et al. (2018). Modeling and prediction of clinical symptom trajectories in alzheimers disease using longitudinal data. *PLoS computational biology*, 14(9), p. 25.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Singapore: Springer Science+Business Media.

Bologna, G. and Hayashi, Y. (2018). A comparison study on rule extraction from neural network ensembles, boosted shallow trees, and svms. *Applied Computational Intelligence and Soft Computing*, volume 2018.

Bolón-Canedo, V., Sánchez-Maroño, N. and Alonso-Betanzos, A. (2015). *Feature selection for high-dimensional data*. Switzerland: Springer International Publishing.

Börsch-Supan, A. et al. (2013). Data resource profile: the survey of health, ageing and retirement in europe (share). *International journal of epidemiology*, 42(4), pp. 992–1001.

Bramer, M. (2007). *Principles of data mining*. London: Springer-Verlag Ltd.

Brazdil, P. et al. (2008). *Metalearning: Applications to data mining*. Berlin: Springer.

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

Buizza, G. et al. (2018). Early tumor response prediction for lung cancer patients using novel longitudinal pattern features from sequential pet/ct image scans. *Physica Medica*, 54, pp. 21–29.

Caruana, R. et al. (2015). Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1721–1730.

Cavagnolli, G. et al. (2011). Hba1c measurement for the diagnosis of diabetes: is it enough? *Diabetic medicine*, 28(1), pp. 31–35.

Centor, R. M. (1985). Receiver operating characteristic (roc) curve analysis using microcomputer spreadsheets. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*, American Medical Informatics Association, p. 207.

Chandrashekar, G. and Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1), pp. 16–28.

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

Chen, S. and DuBois Bowman, F. (2011). A novel support vector classifier for longitudinal high-dimensional data and its application to neuroimaging data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 4(6), pp. 604–611.

Cheng, X. et al. (2020). Population ageing and mortality during 1990–2017: A global decomposition analysis. *PLoS medicine*, 17(6), p. 17.

Cieslak, D. A. et al. (2012). Hellinger distance decision trees are robust and skew-insensitive. *Data Mining and Knowledge Discovery*, 24(1), pp. 136–158.

Condorcet, M. d. (1785). Essay on the application of analysis to the probability of majority decisions. *Paris: Imprimerie Royale*.

Cui, R. et al. (2019). Rnn-based longitudinal analysis for diagnosis of alzheimers disease. *Computerized Medical Imaging and Graphics*, 73, pp. 1–10.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan), pp. 1–30.

Deng, H. et al. (2013). A time series forest for classification and feature extraction. *Information Sciences*, 239, pp. 142–153.

Desai, A. K., Grossberg, G. T. and Sheth, D. N. (2004). Activities of daily living in patients with dementia. *CNS drugs*, 18(13), pp. 853–875.

Diggle, P. et al. (2013). *Analysis of Longitudinal Data - 2nd Edition*. UK: Oxford Univerisity Press.

DPPRG, D. P. P. R. G. (2009). Alcohol consumption and diabetes risk in the diabetes prevention program. *The American journal of clinical nutrition*, 90(3), pp. 595–601.

Drummond, C., Holte, R. C. et al. (2003). C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on learning from imbalanced datasets II*, vol. 11, Citeseer, pp. 1–8.

Du, W. et al. (2015). A longitudinal support vector regression for prediction of als score. In *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*, IEEE, pp. 1586–1590.

Effendy, V., Baizal, Z. A. et al. (2014). Handling imbalanced data in customer churn prediction using combined sampling and weighted random forest. In *2014 2nd International Conference on Information and Communication Technology (ICoICT)*, IEEE, pp. 325–330.

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

Epifanio, I. (2017). Intervention in prediction measure: a new approach to assessing variable importance for random forests. *BMC bioinformatics*, 18(1), p. 230.

Fabris, F., de Magalhães, J. P. and Freitas, A. A. (2017). A review of supervised machine learning applied to ageing research. *Biogerontology*, 18(2), pp. 171–188.

Farbstein, D. and Levy, A. P. (2012). Hdl dysfunction in diabetes: causes and possible treatments. *Expert review of cardiovascular therapy*, 10(3), pp. 353–361.

Fawagreh, K., Gaber, M. M. and Elyan, E. (2014). Random forests: from early developments to recent advancements. *Systems Science & Control Engineering: An Open Access Journal*, 2(1), pp. 602–609.

Fernández-Delgado, M. et al. (2014). Do we need hundreds of classifiers to solve real world classification problems. *Journal of Machine Learning Research*, 15(1), pp. 3133–3181.

Flach, P. A. (2003). The geometry of roc space: understanding machine learning metrics through roc isometrics. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pp. 194–201.

Flach, P. A. (2016). Roc analysis. In *Encyclopedia of Machine Learning and Data Mining*, Springer, pp. 1–8.

Foos, P. W. and Clark, M. C. (2016). *Human aging*. New York: Routledge.

Freitas, A. A. (2004). A critical review of multi-objective optimization in data mining: a position paper. *ACM SIGKDD Explorations Newsletter*, 6(2), pp. 77–86.

Freitas, A. A. (2014). Comprehensible classification models: a position paper. *ACM SIGKDD explorations newsletter*, 15(1), pp. 1–10.

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

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

Gale, E. A. and Gillespie, K. M. (2001). Diabetes and gender. *Diabetologia*, 44(1), pp. 3–15.

García, S., Luengo, J. and Herrera, F. (2015). *Data preprocessing in data mining*, vol. 72. Switzerland: Springer International Publishing.

Gill, J. M. and Cooper, A. R. (2008). Physical activity and prevention of type 2 diabetes mellitus. *Sports Medicine*, 38(10), pp. 807–824.

Gu, D. et al. (2020). The chinese longitudinal healthy longevity survey. *Encyclopedia of Gerontology and Population Aging*, pp. 469–482.

Haixiang, G. et al. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, 73, pp. 220–239.

Hall, M. A. (1999). *Correlation-based feature selection for machine learning*. Ph.D. thesis, University of Waikato Hamilton, Hamilton, New Zealand.

Hapfelmeier, A. et al. (2014). A new variable importance measure for random forests with missing data. *Statistics and Computing*, 24(1), pp. 21–34.

Harris, E. (2002). Information gain versus gain ratio: A study of split method biases. In *International Symposium on Artificial Intelligence and Mathematics*, pp. 1–20.

Hielscher, T. et al. (2014). Mining longitudinal epidemiological data to understand a reversible disorder. In *International Symposium on Intelligent Data Analysis*, Springer, pp. 120–130.

Higgins, J. J. (2004). *Introduction to modern nonparametric statistics*. Pacific Grove, CA: Brooks/Cole, 1st edn.

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

Hosni, M. et al. (2019). Reviewing ensemble classification methods in breast cancer. *Computer methods and programs in biomedicine*, 177, pp. 89–112.

Hu, Z. et al. (2017). Strategies for handling missing clinical data for automated surgical site infection detection from the electronic health record. *Journal of Biomedical Informatics*, 68, pp. 112–120.

Huang, L. et al. (2016). Longitudinal clinical score prediction in alzheimer's disease with soft-split sparse regression based random forest. *Neurobiology of aging*, 46, pp. 180–191.

Hyafil, L. and Rivest, R. L. (1976). Constructing optimal binary decision trees is np-complete. *Information Processing letters*, 5(1), pp. 15–17.

Janiszewski, P. M., Janssen, I. and Ross, R. (2007). Does waist circumference predict diabetes and cardiovascular disease beyond commonly evaluated cardiometabolic risk factors? *Diabetes care*, 30(12), pp. 3105–3109.

Japkowicz, N. and Shah, M. (2011). *Evaluating learning algorithms: a classification perspective*. UK: Cambridge University Press.

Jiang, L., Li, C. and Cai, Z. (2009). Decision tree with better class probability estimation. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(04), pp. 745–763.

Jiang, T., Gradus, J. L. and Rosellini, A. J. (2020). Supervised machine learning: a brief primer. *Behavior Therapy*, 51(5), pp. 675–687.

Jie, B. et al. (2017). Temporally constrained group sparse learning for longitudinal data analysis in alzheimer's disease. *IEEE Transactions on Biomedical Engineering*, 64(1), pp. 238–249.

Kaiser, A. (2013). A review of longitudinal datasets on ageing. *Journal of Population Ageing*, 6(1-2), pp. 5–27.

Karjalainen, M. et al. (2018). Frequent pain in older people with and without diabetes–finnish community based study. *BMC geriatrics*, 18(1), pp. 1–8.

Kaur, H., Pannu, H. S. and Malhi, A. K. (2019). A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Computing Surveys (CSUR)*, 52(4), pp. 1–36.

Kenny, R. A. et al. (2010). The design of the irish longitudinal study on ageing. *TILDA: The Irish Longitudinal Study on Ageing*.

Kim, D. et al. (2018). Association of diabetes diagnosis with dietary changes and weight reduction. *Expert review of pharmacoeconomics & outcomes research*, 18(5), pp. 543–550.

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

Lash, M. T. and Street, W. N. (2020). Personalized cardiovascular disease risk mitigation via longitudinal inverse classification. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, pp. 2610–2617.

Li, X. et al. (2020). Building auto-encoder intrusion detection system based on random forest feature selection. *Computers & Security*, 95, p. 15.

Lifford, K. L. et al. (2005). Type 2 diabetes mellitus and risk of developing urinary incontinence. *Journal of the American Geriatrics Society*, 53(11), pp. 1851–1857.

Little, R. J. and Rubin, D. B. (2019). *Statistical analysis with missing data*, vol. 793. USA, New Jersey: John Wiley & Sons.

Liu, W. et al. (2010). A robust decision tree algorithm for imbalanced data sets. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, SIAM, pp. 766–777.

Lönnrot, M. et al. (2017). Respiratory infections are temporally associated with initiation of type 1 diabetes autoimmunity: the teddy study. *Diabetologia*, 60(10), pp. 1931–1940.

López, V. et al. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*, 250, pp. 113–141.

Luo, J. et al. (2020). Hitanet: Hierarchical time-aware attention networks for risk prediction on electronic health records. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 647–656.

Lutz, W., Sanderson, W. and Scherbov, S. (2008). The coming acceleration of global population ageing. *Nature*, 451(7179), pp. 716–719.

Mainous III, A. G. et al. (2015). Grip strength as a marker of hypertension and diabetes in healthy weight adults. *American journal of preventive medicine*, 49(6), pp. 850–858.

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

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

McAuliffe, L., Brown, D. and Fetherstonhaugh, D. (2012). Pain and dementia: an overview of the literature. *International Journal of Older People Nursing*, 7(3), pp. 219–226.

Minhas, S. et al. (2015). Early alzheimers disease prediction in machine learning setup: Empirical analysis with missing value computation. In *International Conference on Intelligent Data Engineering and Automated Learning*, Springer, pp. 424–432.

Mo, J. et al. (2013). Classification of alzheimer diagnosis from adni plasma biomarker data. In *Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics*, ACM, pp. 569–574.

Morid, M. A. et al. (2020). Temporal pattern detection to predict adverse events in critical care: Case study with acute kidney injury. *JMIR medical informatics*, 8(3), p. 14.

Mueller, R. O. and Hancock, G. R. (2018). *Structural equation modeling*. Routledge.

Mulyar, A. and Krawczyk, B. (2018). Addressing local class imbalance in balanced datasets with dynamic impurity decision trees. In *International Conference on Discovery Science*, Springer, pp. 3–17.

Nemenyi, P. (1962). Distribution-free multiple comparisons. In *Biometrics*, vol. 18.2, International Biometric Soc 1441 I ST, NW, SUITE 700, WASHINGTON, DC 20005-2210, p. 263.

Niemann, U. et al. (2015). Can we classify the participants of a longitudinal epidemiological study from their previous evolution? In *Computer-Based Medical Systems (CBMS), 2015 IEEE 28th International Symposium on*, IEEE, pp. 121–126.

Pomsuwan, T. (2017). *Feature selection for the classification of longitudinal human ageing data*. Master's thesis, School of Computing – University of Kent, United Kingdom.

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

Quinlan, J. R. (1987a). Generating production rules from decision trees. In *IJCAI'87 – Proceedings of the 10th International Joint Conference on Artificial Intelligence*, vol. 1, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 304–307.

Quinlan, J. R. (1987b). Simplifying decision trees. *International Journal of Man-Machine Studies*, 27(3), pp. 221–234.

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

Radovic, M. et al. (2017). Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. *BMC bioinformatics*, 18(1), p. 9.

Rahman, M. M. and Davis, D. N. (2013). Machine learning-based missing value imputation method for clinical datasets. In *IAENG transactions on engineering technologies*, Springer, pp. 245–257.

Ribeiro, C. and Freitas, A. A. (2019a). 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*, p. 5.

Ribeiro, C. and Freitas, A. A. (2019b). 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*, p. 5.

Ribeiro, C. and Freitas, A. A. (2020). A new random forest method for longitudinal data classification using a lexicographic bi-objective approach. In *Proceedings of the 2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, pp. 806–813.

Ribeiro, C. and Freitas, A. A. (2021a). Constructed temporal features for longitudinal classification of human ageing data. In *ICHI21: IEEE International Conference on Healthcare Informatics*, IEEE, p. 7.

Ribeiro, C. and Freitas, A. A. (2021b). A data-driven missing value imputation approach for longitudinal datasets. *Artificial Intelligence Review (online, https://doiorg/101007/s10462-021-09963-5)*, p. 30.

Ribeiro, C. et al. (2017). A revision and analysis of the comprehensiveness of the main longitudinal studies of human aging for data mining research. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 7(3), p. 15.

Rokach, L. (2016). Decision forest: Twenty years of research. *Information Fusion*, 27, pp. 111–125.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5), pp. 206–215.

Ruitenberg, A. et al. (2001). Incidence of dementia: does gender make a difference? *Neurobiology of aging*, 22(4), pp. 575–580.

Sagi, O. and Rokach, L. (2018). Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4), p. e1249.

Saiepour, N. et al. (2019). Does attrition affect estimates of association: a longitudinal study. *Journal of psychiatric research*, 110, pp. 127–142.

Santos, M. S. et al. (2017). Influence of data distribution in missing data imputation. In A. ten Teije, C. Popow, J. H. Holmes and L. Sacchi, eds., *Artificial Intelligence in Medicine*, Cham: Springer International Publishing, pp. 285–294.

Schaffer, C. (1993). Overfitting avoidance as bias. *Machine learning*, 10(2), pp. 153–178.

Scornet, E. et al. (2015). Consistency of random forests. *The Annals of Statistics*, 43(4), pp. 1716–1741.

Sewell, W. H. et al. (2003). As we age: A review of the wisconsin longitudinal study, 1957–2001. *Research in social stratification and mobility*, 20, pp. 3–111.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B (Methodological)*, pp. 267–288.

Tibshirani, R. et al. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1), pp. 91–108.

Touw, W. G. et al. (2012). Data mining in the life sciences with random forest: a walk in the park or lost in the jungle? *Briefings in Bioinformatics*, 14(3), pp. 315–326.

Tsagris, M., Lagani, V. and Tsamardinos, I. (2018). Feature selection for high-dimensional temporal data. *BMC bioinformatics*, 19(1), pp. 1–14.

Tsoumakas, G., Katakis, I. and Vlahavas, I. (2009). Mining multi-label data. In *Data mining and knowledge discovery handbook*, Springer, pp. 667–685.

United Nations, D. o. E. and Social Affairs, P. D. U. D. D. (2019). World population prospects 2019: Ten key findings. *WHO World Population Prospects*.

Verleysen, M. and François, D. (2005). The curse of dimensionality in data mining and time series prediction. In *International work-conference on artificial neural networks*, Springer, pp. 758–770.

Vezhnevets, A. and Vezhnevets, V. (2005). Modest adaboost-teaching adaboost to generalize better. In *Graphicon*, vol. 12.5, pp. 987–997.

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

Wilcoxon, F. (1992). Individual comparisons by ranking methods. In *Breakthroughs in statistics*, Springer, pp. 196–202.

Yan, W. and Goebel, K. F. (2004). Designing classifier ensembles with constrained performance requirements. In *Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2004*, vol. 5434, International Society for Optics and Photonics, pp. 59–69.

Yap, B. W. et al. (2014). An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets. In *Proceedings of the first international conference on advanced data and information engineering (DaEng-2013)*, Springer, pp. 13–22.

Zhang, Y. et al. (2016). Study on prediction of activities of daily living of the aged people based on longitudinal data. *Procedia Computer Science*, 91, pp. 470–477.

Zhao, J. et al. (2019). Learning from longitudinal data in electronic health record and genetic data to improve cardiovascular event prediction. *Scientific reports*, 9(1), pp. 1–10.

Zhou, Z., Wang, P. and Fang, Y. (2018). Loneliness and the risk of dementia among older chinese adults: gender differences. *Aging & Mental Health*, 22(4), pp. 519–525.

Zhou, Z.-H. (2012). *Ensemble methods: foundations and algorithms*. USA New York: CRC press.

Zhu, X. (2014). Comparison of four methods for handing missing data in longitudinal data analysis through a simulation study. *Open Journal of Statistics*, 4(11), pp. 933–944.