



HAL
open science

Temporal dynamics in tree-based models and applications to lapse behaviour in life insurance

Mathias Valla

► **To cite this version:**

Mathias Valla. Temporal dynamics in tree-based models and applications to lapse behaviour in life insurance. General Finance [q-fin.GN]. Université Claude Bernard - Lyon I; Katholieke Universiteit Leuven (Brabant flamand, Belgique ; 1970-..), 2024. English. ⟨NNT : 2024LYO10032⟩. ⟨tel-05121356⟩

HAL Id: tel-05121356

<https://theses.hal.science/tel-05121356v1>

Submitted on 19 Jun 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization



THÈSE de DOCTORAT DE
L'UNIVERSITÉ CLAUDE BERNARD LYON 1
Ecole Doctorale n°486
Sciences Economiques et de Gestion
En cotutelle avec Katholieke Universiteit Leuven

Disciplines : Sciences de Gestion
Soutenue publiquement le 14/03/2024, par :

Mathias VALLA

**Dynamique temporelle dans les modèles par
arbres et applications aux comportements de
rachat en assurance vie**

Devant le jury composé de :

Olivier LOPEZ

Professeur à l'ENSAE, Institut Polytechnique de Paris

Mercè CLARAMUNT BIELSA

Professeure associée à l'Université de Barcelone

Caroline HILLAIRET

Professeure à l'ENSAE, Institut Polytechnique de Paris

Frédéric PLANCHET

Professeur des Universités à l'Université Claude Bernard Lyon 1

Christian-Yann ROBERT

Professeur des Universités à l'Université Claude Bernard Lyon 1

Katrien ANTONIO

Professeure à KU Leuven et à l'Université d'Amsterdam

Xavier MILHAUD

Maître de conférence à Aix-Marseille Université

Denys POMMERET

Professeur des Universités à Aix-Marseille Université

Rapporteur

Rapporteuse

Examinatrice

Président de jury

Directeur de thèse

Directrice de thèse

Invité (co-encadrant de la thèse)

Invité (ancien directeur de thèse)

KU LEUVEN

FACULTY OF ECONOMICS
AND BUSINESS

Temporal dynamics in tree-based models

and applications to lapse behaviour in life insurance.



Dissertation presented to
obtain the degree of Doctor in
Business Economics

by

Mathias VALLA

Number: 0864924

2023-2024

Since the theses in the series published by the Faculty of Economics and Business are the personal work of their authors, only the latter bear full responsibility.

Les dissertations émanant de la Faculté des sciences économiques et de gestion étant l'œuvre personnelle de leurs auteurs, ces derniers en sont seuls responsables.

Daar de proefschriften in de reeks van de Faculteit Economie en Bedrijfswetenschappen het persoonlijk werk zijn van hun auteurs, zijn alleen deze laatsten daarvoor verantwoordelijk.

Committee

Supervisors:

Prof. dr. Katrien ANTONIO

KU Leuven and University of Amsterdam

Prof. dr. Christian-Yann ROBERT

Université Claude Bernard Lyon 1

Prof. dr. Xavier MILHAUD

Aix-Marseille Université

Doctoral committee:

Prof. dr. Olivier LOPEZ

ENSAE, Institut Polytechnique de Paris

Prof. dr. Mercè CLARAMUNT BIELSA

Barcelona University

Prof. dr. Caroline HILLAIRET

ENSAE, Institut Polytechnique de Paris

Prof. dr. Frédéric PLANCHET

Université Claude Bernard Lyon 1

Prof. dr. Jan Dhaene

KU Leuven

Chairman:

Prof. dr. MMM

KU Leuven

Abstract (French)

Cette thèse, “*Dynamique temporelle dans les modèles par arbres et applications aux comportements de rachat en assurance vie.*”, explore l’application de nouveaux modèles statistiques et de stratégies innovantes pour analyser le comportement de rachat en assurance vie. Ce travail est une combinaison de deux parties introductives et de trois articles de recherche, chacun offrant une perspective singulière et diverses contributions à la gestion du rachat ainsi qu’à l’inclusion d’une dimension temporelle dans les modèles de Machine Learning dits *par arbres*. Le premier article, “*Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies*”, présente une stratégie de rétention qui va au-delà d’une simple prédiction du comportement assuré. Cette stratégie est basée sur une méthodologie de gestion des rachats intégrant les notions de valeur client et de rentabilité, en misant sur une individualisation des approches existantes. L’étude démontre comment les modèles de survie par arbres surpassent les approches paramétriques, contribuant ainsi à une gestion des campagnes de rétention plus efficace et plus informée, pour les assureurs vie. S’appuyant sur ces travaux, le deuxième article, “*A longitudinal framework for lapse management in life insurance*”, souligne l’importance d’une approche incluant la dimension temporelle des comportements assurés dans la gestion du rachat. L’article propose un cadre longitudinal pour la gestion des comportements de rachat, qui exploite l’ensemble des données historiques passées de chaque assuré, une ressource souvent négligée mais abondamment disponible dans les systèmes d’information des assureurs. Cette méthodologie affine davantage la précision du ciblage des assurés à retenir en portefeuille, améliorant ainsi la compréhension globale des assureurs quant au risque qu’ils portent. Le dernier article, “*Time penalized tree TpT: a new tree-based data mining algorithm for time-varying covariates*”, présente un nouvel algorithme de construction d’arbre de décision acceptant des variables évoluant avec le temps sous forme de données structurées longitudinalement. L’article explicite et remet en question les hypothèses classiques des modèles acceptant uniquement des variables statiques en proposant un algorithme qui permet le partitionnement récursif des espaces des covariables et du temps, conjointement. Cette méthode innovante aide à capturer les tendances historiques pertinentes pour l’analyse, permettant une étude plus précise et interprétable de phénomènes évoluant dans des environnements dynamiques.

Dans l’ensemble, cette thèse offre une approche holistique de l’analyse du comportement de rachat dans l’assurance vie en intégrant des modèles par arbres avancés et de l’analyse longitudinale. Elle souligne le potentiel de ces stratégies innovantes pour informer les décisions commerciales et stratégiques dans l’industrie de l’assurance tout en garantissant leur interprétabilité et leur explicabilité.

Abstract (English)

This thesis, “*Temporal dynamics in tree-based models and applications to lapse behaviour in life insurance*”, delves into the application of novel statistical models and strategies to address lapse behavior in life insurance. This work is an amalgamation of two introductory parts and three research articles, each offering a unique perspective and various contributions to the understanding of lapse management and the inclusion of a time dimension in tree-based Machine Learning models. The first article, “*Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies*”, presents a retention strategy that goes beyond a mere prediction of lapse. This strategy is grounded on a lapse management framework that integrates Customer Lifetime Value and profitability, with a focus on the individualization of existing approaches. The study demonstrates how survival tree-based models outperform parametric approaches, thereby leading to more efficient and informed management of retention campaigns, for life insurers. Building on this work, the second article, “*A longitudinal framework for lapse management in life insurance*”, emphasizes the importance of a time-informed approach in lapse management. The article proposes a longitudinal lapse management strategy framework that leverages the complete past trajectory of policyholders, a resource often overlooked yet abundantly available in insurers’ information systems. This methodology further refines the targeting precision, thereby enhancing the insurers’ understanding of their global portfolio risk. The final article, “*Time penalized tree (TpT): a new tree-based data mining algorithm for time-varying covariates*”, introduces a novel decision tree algorithm that accounts for time-varying covariates within longitudinally structured datasets. The article challenges the traditional static assumption of covariates by proposing an algorithm that allows recursive partitioning of the covariates space together with time. This innovative method helps to capture relevant historical trends for analysis, enabling a more accurate and interpretable study of phenomena evolving in dynamic environments.

Overall, this thesis provides a global approach to lapse behavior analysis in life insurance by integrating advanced tree-based models and longitudinal analysis. It underscores the potential of these innovative strategies in informing commercial and strategic decisions in the insurance industry while ensuring their interpretability and explainability.

Table of contents

iii | KU Leuven doctoral committee

vi | Table of contents



Introduction

2 | Chapter 1
General introduction

- 1.1 Thesis Context 3
- 1.2 Why AI in insurance ? 3
- 1.3 Why Tree-based models (TBMs) ? 4
- 1.4 Why focusing on temporal dynamic in actuarial applications ? 6

8 | Chapter 2
Research objectives and scope

- 2.1 What is churn? 8
- 2.2 Life insurance and lapse management strategies (LMS) 10
 - 2.2.1 Specificities about life insurance 10
 - 2.2.2 Application to lapse management strategy 11

13 | Chapter 3
Thesis structure, objectives, and contributions

- 3.1 Thesis structure and objectives 13

16 | Bibliography

Machine learning and tree-based models

22	Chapter 4 Methodology and methods
4.1	Machine learning methodology 22
4.1.1	Training 23
4.1.2	Tuning and evaluation 27
4.2	Models description 34
4.2.1	History of decision trees 34
4.2.2	Ensemble models 41
4.2.3	Theoretical guarantees 44
49	Chapter 5 Survival analysis
5.0.1	Survival notations 49
5.1	Models 51
5.1.1	Inverse probability of censorship weighted (IPCW) models 51
5.1.2	Survival trees 52
5.1.3	Random survival forests (RSF) 53
5.1.4	Gradient Boosting Survival Model (GBSM) 54
5.2	Survival performance metrics 55
5.2.1	Brier Score and variations 55
5.2.2	Concordance indices 56
5.2.3	Dynamic AUC 57
58	Bibliography



Lapse management strategy

65		Chapter 6 Contributions of Part III
67		Chapter 7 The Customer Lifetime Value and its insights for insurance
71		Chapter 8 Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies
		8.1 Introduction 71
		8.2 Data 74
		8.3 Framework 76
		8.4 Methodology 79
		8.4.1 Step 1: Modeling $r_{\text{acceptant}}$ and r_{lapser} 80
		8.4.2 Step 2: Classification tasks 83
		8.5 Real-life application 85
		8.5.1 Considered lapse management strategies 85
		8.5.2 Numerical results 86
		8.5.3 Comments 86
		8.6 Discussion 89
		8.6.1 General statements 89
		8.6.2 Marketing decision making 90
		8.6.3 Management rules decision making 91
		8.7 Conclusion and perspectives 93
		8.8 Suggestions for future work 94
		8.8.1 $z^{(i)}$ efficiency border 94
		8.8.2 Other improvements 97
98		Bibliography

Longitudinal setting

- 104** | Chapter 9
Contributions of Part IV
- 105** | Chapter 10
Introduction to longitudinal studies
- 108** | Chapter 11
Longitudinal models
- 11.1 (Semi)-parametric models 108
 - 11.1.1 Mixed Models (MMs) and extensions 108
 - 11.1.2 Cox model and time-varying covariates 108
 - 11.2 Overview of TBMs in a longitudinal setting 109
 - 11.2.1 Dynamic regression on a longitudinal outcome with tree-based models 109
 - 11.2.2 Longitudinal survival analysis with tree-based models 112
 - 11.2.3 Metrics 114
- 115** | Chapter 12
A longitudinal ML framework for lapse management in life insurance
- 12.1 Introduction 115
 - 12.2 Longitudinal framework 120
 - 12.2.1 Preliminaries on time-varying covariates and longitudinal notations 120
 - 12.2.2 LMS longitudinal framework 123
 - 12.3 Application 126
 - 12.3.1 Data 126
 - 12.3.2 Application: survival step 127
 - 12.3.3 Application: regression step 131
 - 12.4 Conclusion, limitations and future work 135
- 137** | Bibliography

Towards time-dependent trees

- 144** | Chapter 13
Contributions of Part V
- 145** | Chapter 14
Time-penalized trees: a new tree-based data mining algorithm for time-varying covariates
- 14.1 Introduction 145
 - 14.2 Preliminaries 146
 - 14.2.1 Classification and regression trees 146
 - 14.2.2 Longitudinal notations 150
 - 14.2.3 Existing longitudinal tree-based algorithms 151
 - 14.3 Time penalized trees 153
 - 14.3.1 TpT splitting criterion 155
 - 14.3.2 TpT pruning process 160
 - 14.3.3 Time-to-event 160
 - 14.4 Applications 160
 - 14.4.1 Properties of TpT for the maximal tree 161
 - 14.4.2 Use-case with minsplitt 164
 - 14.4.3 Pathways vizualizations 166
 - 14.5 Conclusion, limitations and future work 168
 - 14.5.1 Conclusion 168
 - 14.5.2 Limitations and future work 169
- 172** | Bibliography

Epilogue

- 176** | Chapter 15
Final words
- 178** | Bibliography



Appendices

180	Appendix A General Appendix
	A.0.1 Permutation test 180
	A.0.2 Pearson's chi-square test 180
	A.0.3 Bonferroni correction 181
	A.0.4 Generalized M-Fluctuation Tests 182
	A.0.5 Lifetime function estimate by Kaplan-Meier 182
	A.0.6 Log-rank test (Mantel-Cox) 183
	A.0.7 Cox-Model 184
	A.1 Competing risk framework 184
	A.1.1 Cause-specific approach 185
	A.1.2 Subdistribution approach 186
187	Appendix B Appendix of the first article (chap. 8)
	B.1 Survival analysis results 187
	B.1.1 Cox-model 187
	B.1.2 RSF 189
	B.1.3 XGSB 190
	B.1.4 Final survival model 191
	B.2 Other results 191
	B.3 Considering various statistical metrics 191
	B.4 Complete LMS numerical results 192
198	Appendix C Appendix of the second article (chap. 12)
	C.1 Note on parametric models 198
	C.2 Model selection methodology 198
	C.3 Estimation of π_* 200
201	Appendix D Appendix of the third article (chap. 14)
	D.1 About cost-complexity pruning 201
	D.2 Frechet trees 201
	D.3 More results 201
	D.3.1 Results without stopping criterion 201
	D.3.2 Results with <code>minsplit = 25</code> 211
	D.3.3 Results with <code>minsplit = 50</code> 213
216	LIST OF FIGURES

218 | LIST OF TABLES

219 | Appendix E
Doctoral booklet - Curriculum Vitae - Research Data Management

List of abbreviations and symbols

Following symbols are used in the present work:

Part I

<i>AI</i>	Artificial Intelligence
<i>ASTIN</i>	Actuarial Studies in Non-life Insurance
<i>CANN</i>	Combined Actuarial Neural Network
<i>CLV</i>	Customer Lifetime Value
<i>DIALog</i>	Digital Insurance and Long-term Risks
<i>DL</i>	Deep Learning
<i>DT</i>	Decision tree
<i>GBM</i>	Gradient boosting machine
<i>GLM</i>	Generalized Linear Model
<i>KU Leuven</i>	Katholieke Universiteit Leuven
<i>ML</i>	Machine Learning
<i>NLP</i>	Natural Language Processing
<i>NN</i>	Neural Network
<i>PH</i>	Policyholder
<i>RF</i>	Random Forest
<i>TBM</i>	Tree-based model
<i>TpT</i>	Time-penalized Tree
<i>UCBL1</i>	Université Claude Bernard Lyon 1

Part II

<i>AIC</i>	Akaike Information Criterion
<i>AID</i>	Automatic Interaction Detector
<i>AUC – PR</i>	Area Under the P-R Curve
<i>AUROC</i>	Area Under the ROC Curve
<i>BIC</i>	Bayesian Information Criterion
<i>BNH</i>	Between-nodes heterogeneity
<i>BS</i>	Brier Score
<i>BSS</i>	Brier Skill Score
<i>C – index</i>	Concordance index
<i>CART</i>	Classification and Regression Tree
<i>CHAID</i>	Chi-square Automatic Interaction Detector
<i>CIF</i>	Conditional Inference
<i>CTREE</i>	Conditional Inference Trees
<i>ELISEE</i>	Exploration of Links and Interactions through Segmentation of an Experimental Ensemble
<i>FACT</i>	Fast Algorithm for Classification Tree
<i>FPR</i>	false-positive rate

<i>FWER</i>	Family-wise error rate
<i>GBSM</i>	Gradient Boosting Survival Model
<i>GUIDE</i>	Generalized, Unbiased, Interaction Detection and Estimation
<i>H – CV</i>	Holdout cross-validation
<i>IBS</i>	Integrated Brier Score
<i>IBSS</i>	Integrated Brier Skill Score
<i>ID</i>	Iterative Dichotomizer
<i>IDEA</i>	Interactive Data Exploration and Analysis
<i>IPCW</i>	Inverse probability of censorship weight
<i>kF – CV</i>	k-fold cross-validation
<i>KM</i>	Kaplan-Meier
<i>LMS</i>	Lapse Management Strategy
<i>LOTUS</i>	Logistic regression tree
<i>LpO – CV</i>	Leave-p-out cross-validation
<i>MAE</i>	Mean Absolute error
<i>MAID</i>	Multivariate AID
<i>MAPE</i>	Mean Absolute Percentage Error
<i>MC – CV</i>	Monte-Carlo cross-validation
<i>MOB</i>	Model-Based recursive partitioning
<i>MSE</i>	Mean Squared Error
<i>PR</i>	Precision-Recall
<i>QUEST</i>	Quick, Unbiased and Efficient Statistical Tree
<i>R – CV</i>	Rolling cross-validation
<i>ROC</i>	Receiver Operating Characteristic
<i>RSF</i>	Random survival forests
<i>SUPPORT</i>	Smoothed and Unsmoothed Piecewise-Polynomial Regression Trees
<i>THAID</i>	Theta AID
<i>TPR</i>	true-positive rate
<i>WNH</i>	Within-node heterogeneity
<i>XGB</i>	eXtrem Gradient Boosting
<i>XGBoost</i>	eXtrem Gradient Boosting

Part III

<i>ALM</i>	Asset and Liabilities Management
<i>CATE</i>	Conditional Average Treatment Effect
<i>CIF</i>	Cumulative Incidence Function
<i>CPH</i>	Cox Proportional Hazard
<i>CPV</i>	Control Portfolio Value
<i>LMPV</i>	Lapse Managed Portfolio Value
<i>RG</i>	Retention Gain

Part IV

<i>CV</i>	Control Value
<i>EM</i>	Expectation-Maximization
<i>GDPR</i>	General Data Protection Regulation
<i>GEE</i>	Generalized Estimating Equations
<i>GLL</i>	Generalized Log-Likelihood
<i>GLMM</i>	Generalized Linear Mixed Model
<i>GMM</i>	Generalized Mixed Model

<i>LLMS</i>	Longitudinal Lapse Management Strategy
<i>LMM</i>	Linear Mixed Model
<i>LMV</i>	Lapse-managed Value
<i>LTRC</i>	Left-truncated and right-censored
<i>MASE</i>	Mean Absolute Scaled Error
<i>MELT</i>	Mixed Effects Longitudinal Tree
<i>MERF</i>	Mixed Effects Random Forest
<i>MERT</i>	Mixed Effects Regression Tree
<i>METBM</i>	Mixed Effect Tree-Based Model
<i>MM</i>	Mixed Model
<i>MODERN</i>	MODEl-basEd RaNdom effects
<i>RE – EM</i>	Random-Effects Expectation-Maximization
<i>td</i>	time-dependent
<i>TI</i>	time-invariant
<i>TV</i>	time-varying

Part V

<i>FA</i>	Face Amount
-----------	-------------

Mathematical formatting rules

x	x is an element of \mathbb{R}
\hat{x}	Predicted value of x
\mathbf{x}	x is a vector
\mathbf{X}	X is a matrix
x^* or x_*	True value of x
Fx	Future value of x
Px	Past value of x
x_{train}	Restiction of x to observations in the training set
x_{test}	Restiction of x to observations in the testing set
$x^{(i)}$	Restiction of x to observation or vector of observations for subject (i)
\mathcal{D}	Dataset
\mathcal{D}^{long}	Longitudinal dataset
$\mathcal{X}, \mathcal{Y}, \mathcal{T}$	Mathematical spaces of covariates, response variables, and follow-up times
\mathcal{T}	Single tree
example	R packages, Python libraries as well as functions and parameters (but not the algorithm names)

All other mathematical symbols are introduced and defined directly in the body of the manuscript

Part I

Introduction

Chapter 1 General introduction

- 1.1 Thesis Context 3
- 1.2 Why AI in insurance ? 3
- 1.3 Why Tree-based models (TBMs) ? 4
- 1.4 Why focusing on temporal dynamic in actuarial applications ? 6

Chapter 2 Research objectives and scope

- 2.1 What is churn? 8
- 2.2 Life insurance and lapse management strategies (LMS) 10
 - 2.2.1 Specificities about life insurance 10
 - 2.2.2 Application to lapse management strategy 11

Chapter 3 Thesis structure, objectives, and contributions

- 3.1 Thesis structure and objectives 13

Bibliography

1. General introduction

“ There is a story about two friends, who were classmates in high school, talking about their jobs. One of them became a statistician and was working on population trends. He showed a reprint to his former classmate. The reprint started, as usual, with the Gaussian distribution and the statistician explained to his former classmate the meaning of the symbols for the actual population, for the average population, and so on. His classmate was a bit incredulous and was not quite sure whether the statistician was pulling his leg. “How can you know that?” was his query. “And what is this symbol here?” “Oh,” said the statistician, “this is π .” “What is that?” “The ratio of the circumference of the circle to its diameter.” “Well, now you are pushing your joke too far,” said the classmate, “surely the population has nothing to do with the circumference of the circle.”¹

Eugene P. Wigner

“However, it does” is the natural answer for a statistician. This insightful short story can be found as an introduction to the remarkable article, *The unreasonable effectiveness of mathematics in the natural sciences*, from Eugene Wigner (Wigner 1960). I remember my first experience reading this story, and it fit remarkably well with my view of statistics and actuarial science. At that time, I was a student who identified with the lay student being taught statistical modeling in the dialog. Now, this thesis somehow brought me on the other side of this story, and I return to this anecdote as a reminder of the deep feeling of wonder that the joint product of statistical research and popularization can bring.

After these introductory words, I hope humbly to replace the statistician and present my thesis on the consideration of the time dimension of actuarial problems with tree-based machine learning (ML) models. The proliferation of ML has had remarkable impacts on business, economics, management, and the actuarial sciences, it has significantly enhanced risk assessment accuracy, improved fraud detection methods, and enabled more precise pricing models for instance. A last example of such impacts, that is of interest within the scope of this thesis is the development of more accurate and individualized prediction approaches by considering the temporal dynamics of data in decision-making processes. This has been further facilitated by sophisticated ML techniques allowing the manipulation of large datasets. Consequently, these predictive models are now considered reliable and useful. This individualization of predictions enables better adaptation to the specific needs of individuals and organizations, thereby improving the quality of decisions. In addition, by integrating the temporal dimension of data, ML enables the capturing of variations and evolutions over time, which is essential for informed and responsive decision-making. Thus, ML opens up exciting new perspectives in the field of management science, enabling predictions to be more individualized, and the temporal dynamics of data to be more meaningfully included in decision-making processes.

First, I will provide some context on the development of this work, and then I will explain what justifies this topic and how it can be beneficial to the field of actuarial science, why we use artificial intelligence (AI) to solve actuarial problems, why we restrict the analysis to tree-based models and eventually, what we mean by considering temporal dynamics to modeling. Having defined the terms of the subject, I will then introduce the rest of the thesis and the objectives it seeks to achieve.

1.1 Thesis Context

This thesis is part of a partnership between the Actuarial Science Laboratory at Université Claude Bernard Lyon 1 (UCBL1) and Katholieke Universiteit Leuven (KU Leuven). As such, it was organized by a joint doctoral agreement that governs the supervision and award of a double doctoral diploma. If defended and based on a favorable report from the examination committee, this collaborative work should be rewarded with a joint doctoral degree in Sciences de Gestion, Mathématiques appliquées—that is, management sciences and applied mathematics—from UCBL1 and in Business Economics from KU Leuven. This research work has also been conducted within the research chaire Digital Insurance and Long-term Risks (DIALog), whose objectives are to explore AI and ML tools and adapt them for the treatment of actuarial problems with massive data.

The design and construction of this thesis highlight the strong willingness of the author to fulfill and respect the expectations, as well as the regulations and guidelines, of all involved institutions in terms of research fields. It merges actuarial science, management science, applied mathematics, ML, business, and economics, and the proportion to which each domain is represented may vary across the chapters and sections.

1.2 Why AI in insurance ?

Progress in AI is swift and has the potential to transform strategies and efficiency in the realms of finance and insurance. An extensive study examining the influence of AI on the insurance industry, which uses a data compilation of 91 articles and 22 industry reports, can be found in the 2021 publication by Eling, Nuessle, and Staubli [2021](#). The researchers inferred that the incorporation of AI into the insurance sector could result in significant cost savings and new income opportunities. Hence, the conventional insurance business model, which predominantly focuses on loss compensation, is expected to shift emphasis toward loss prediction and prevention. ML algorithms offer the unique capability of identifying relationships and intricate non-linear interactions that might not be foreseen by humans or conventional statistical models (Lestavel [2017](#)). Specifically, these technological solutions enable insurance firms to improve the precision of loss probability forecasting. This development could potentially solve a fundamental problem in the industry: the challenge of information asymmetry. Using AI technologies, insurers can lessen the information imbalance between themselves and their policyholders, resulting in more accurate and personalized risk evaluations and enhanced decision-making processes.

Actuaries in the insurance sector are progressively utilizing AI for model improvement and quick updating. Numerous leading insurance conglomerates, such as Axa, Allianz, CNP, and Generali, are investing in AI by setting up data labs and using collaborative ML platforms and solutions. Conversely, online insurance companies are reaping the benefits of amplified intelligence, enabling nearly instantaneous pricing simulations that require extensive processing time. These advancements will inevitably improve customer satisfaction (see Eling, Nuessle, and Staubli [2021](#);

Eckert, Neunsinger, and Osterrieder 2022). Nonetheless, applying self-learning algorithms to traditional cross-sectional data may not result in a significant improvement in rate precision. To achieve higher accuracy scores, and thus more precise pricing, longitudinally structured data or even unstructured data from emails or customer letters can be examined using grammatical and semantic recognition algorithms. These avant-garde methods permit the analysis of time effects in data, capture price sensitivities, and adjust prices based on the anticipated individual behavior. This transition from traditional clustering or cohort methods towards individualization is regarded as promising for tangible applications by insurance companies. This evolution of the field can also be observed in research in asset management (Bartram, Branke, and Motahari 2020), life insurance (Chancel et al. 2022) or non-life reserving with dedicated research groups at the Actuarial Studies in Non-life Insurance (ASTIN), for instance.

The integration of AI provides remarkable opportunities but also carries inherent risks. The importance of maintaining control over the algorithms and critically analyzing their outputs instead of blindly relying on the tool cannot be overstated. Actuaries play an indispensable role in formulating algorithms, confirming data dependability, deciphering the answers generated by machines, and implementing them in competitive regulated settings. Moreover, they are instrumental in illustrating data and popularizing findings to convince stakeholders of the pertinence of the models. As AI persists in molding the insurance sector, ethical issues must be confronted, and responsibility must be guaranteed. Upholding human supervision and embedding ethical standards are crucial for preventing bias and unfair practices. AI should be regarded as a facilitator of decision-making, rather than a complete substitute for human judgment.

In conclusion, the finance and insurance sectors are experiencing significant transformations with the adoption of AI. The potential practical and theoretical benefits of AI in data analysis, claims processing, risk assessment, asset management, and wealth management consulting are vast.

1.3 Why Tree-based models (TBMs) ?

Tree-based algorithms in machine learning, like Decision Trees (DT), Random Forests (RF), and Gradient Boosting Machines (GBM), use a hierarchical structure of decision rules based on features to make predictions or decisions (the intricacies of TBMs are discussed in Chapter 4). While these methods are relatively simple (in the sense that they do not require a large number of (hyper-)parameters), one might assume that increasing model complexity, with deep learning (DL) approaches for instance, would enhance overall predictive capability. The rise of highly complex deep learning models that we are witnessing in all areas of applied AI, such as computer vision (Simonyan and Zisserman 2015; He et al. 2016), Natural Language Processing (NLP) (Devlin et al. 2018; OpenAI 2023), medical research (Esteva et al. 2017; Rajpurkar et al. 2017), engineering (Schulman et al. 2015; Gu et al. 2017) or environmental science (Jin et al. 2019; Xie, D. Zhang, and Wang 2019) could suggest so. Not surprisingly, research in actuarial sciences and finance has not been left behind by the recent proliferation of cutting-edge literature (See Wüthrich 2018; Frees and Valdez 2019; Q. Shi, Yang, and Li 2020; L. Zhang, Frees, and Valdez 2020; Chen, Y. Wu, and C. Wu 2020; Teixeira, Ferreira, and Pestana 2020; Belzile 2021; McDonnell et al. 2023 and Tsantekidis et al. 2018; L. Zhang and W. Wu 2019; Dehghani and Larijani 2023 respectively). These state-of-the-art articles provide insights into the applications of DL methods in actuarial science and their performance in areas such as mortality modeling, claim prediction, reservation, pricing, and valuation. This begs the question: why favor the use of TBMs over DL methods in this thesis?

Both DL models and TBMs exhibit desirable properties, such as the ability to capture nonlinear relationships between covariates and target variables, provide insights into the relative importance of different features, and handle missing values, outliers, and mixed data types. Ultimately, they are both scalable because they can handle large datasets efficiently and can be parallelized and distributed across multiple processors or clusters, making them suitable for big data scenarios. Thus, with very good predictive performance for specific tasks, one might ask why one would want to use less complex tree-based models, as was done in our study. The main reason is that, although neural networks (NNs) have exceptional capabilities on problems that require unstructured data, tree-based models still outperform DL models in most supervised learning tasks with tabular data (see Borisov et al. 2021)), as demonstrated using 11 various datasets by Shwartz-Ziv and Armon 2022 and another set of 45 benchmark tabular datasets from diverse fields, incorporating both categorical and numerical features (see Grinsztajn, Oyallon, and Varoquaux 2022), with a benchmarking methodology for both fitting models and identifying optimal hyperparameters. In the field of insurance research, a survey was recently conducted (see McDonnell et al. 2023) that revealed that the performance of tree-based models and DL methods can be very close, but using the latter may not be worth the time and computational needs. Recently, Wuthrich 2019 and Schelldorfer and Wuthrich 2019 suggested the idea of using NNs to correct a baseline prediction, established with a Generalized Linear Model (GLM) or TBMs, with a NN. That framework, denoted as *combined actuarial neural network* (CANN) proposes an innovative way for actuarial studies to benefit from NNs. A comparative study of such approaches can be found in Holvoet, Antonio, and Henckaerts 2023.

The broadest survey (Grinsztajn, Oyallon, and Varoquaux 2022) aimed to understand not if, but why tree-based models are still superior to NNs and revealed two main reasons. First, DL methods tend to create prediction functions that are very smooth, whereas the real relationship between the covariates and the target in tabular datasets is often irregular; NNs are biased toward overly smooth solutions. Second, NNs are highly sensitive to uninformative features because of their rotational invariance, whereas the TBMs show greater robustness.

Remark 1.1

Previous remarks of this section are to be nuanced: any real-world application on a single specific dataset has unique properties. The No Free Lunch theorem (Wolpert and Macready 1997) suggests that it is impossible to determine in advance which model will consistently yield optimal performance. This uncertainty arises from the fact that, on average, any two optimization algorithms are equivalent across all possible problems. In other words, there is no universal best algorithm that outperforms others across every problem domain.

That said, we argue that TBMs are a reasonable choice for actuarial purposes that rely on tabular data. In fact, they showed great performance in pricing tasks (see Henckaerts et al. 2021)), as well as in the prediction of cyber claims (Farkas, Lopez, and Thomas 2021) and tornado-induced claims (Maillart and Robert 2023). In addition, TBMs have inherent advantages that contribute to their interpretability, making them highly desirable in management domains, such as actuarial science, where understanding the decision-making process of the model is crucial. In terms of interpretability, some key features of DTs exist that justify our thesis orientation. First, the structure of a decision tree is straightforward and intuitive, and it mimics human or company decision-making by considering different features and making choices based on them. This re-

semblance to human or company decision processes enhances interpretability because the reasoning behind the model predictions aligns with human intuition. This advantage is exploited in the last parts of this thesis, as this does not hold for ensemble methods that are used earlier on.

Secondly, TBMs provide an intuitive measure of feature importance, enabling us to understand which features have the most significant impact on the model predictions. Features that appear higher in the tree structures and are involved in more splits are considered more important, which facilitates identification of the key drivers of the model decisions. Individual decision trees can then be visualized graphically, facilitating comprehension and communication of the model behaviors. Single trees enable tracing of the path from the root node to the leaf nodes and observing how decisions are made, which can be interpreted as an explanation for an individual prediction. This property makes DTs mechanisms easy to understand, even for non-experts. The use of familiar concepts, such as branches, nodes, and decision rules, contributes to the model's interpretability, making it accessible to a wider audience, including all actors in any decision-making process. The extension of those interpretability properties to ensemble TBM is an active area of research.

For now, these mechanisms may not be understood by the lay reader. The next chapter (Chapter 4) is dedicated to detailing tree-based algorithms and illustrating those properties.

1.4 Why focusing on temporal dynamic in actuarial applications ?

In the insurance domain, premiums are collected by insurers from policyholders in exchange for assuming the risk of potential future losses or damages. These premiums form the revenue stream for insurers, whereas the actual costs incurred, including claim payments, expenses, and reserves, typically materialize after the premiums have been collected. This fundamental characteristic of the insurance business model gives rise to a temporal disconnection between revenue collection and cost realization, resulting in a reverse or delayed production cycle. Consequently, time plays a critical role in actuarial modeling, enabling a comprehensive understanding and quantification of risk dynamics. Extensive literature exists in this domain, as exemplified by notable contributions (Bauer and Hommel 2007, Tan and Yow 2011, Gao and X. Wu 2013, Olivieri and Pitacco 2015, Zaks and Sherris 2018). This reverse production cycle in insurance inherently entails moral hazards (Holmström 1979). Abbring et al. 2003 argued that dynamic insurance data surpasses static data that has problems distinguishing between moral hazard and selection and dealing with dynamic features of actual insurance contracts. In the presence of moral hazards, experience ratings lead to negative occurrence dependence in which individual claim intensities decrease as the number of past claims increases. This study also establishes the possibility of testing for adverse selection even when based on asymmetric learning.

Insurance pricing necessitates the use of historical data and statistical methods to determine appropriate premium rates. Time is a fundamental component in evaluating the frequency and severity of past events and facilitating their projection into the future. For instance, Bolancé, Guillén, and Pinquet 2003 accounted for the seniority of claims by introducing dynamic random effects instead of static ones in the context of automobile insurance. P. Shi and Valdez 2014 explored the utility of copulas in modeling the number of insurance claims for individual policyholders within a longitudinal framework. Recently, Lee and P. Shi 2019 proposed a dependent modeling framework to examine jointly the frequency and severity components within a longitudinal context by employing a novel copula regression approach. These examples demonstrate

the integration of policy history into pricing, specifically in terms of the frequency or severity components.

Estimating liabilities and reserves is a crucial responsibility of insurance companies and pension funds to fulfil their obligations. Time plays a critical role in projecting future claim payments, policyholder behavior, and investment returns. Actuarial models account for the time value of money, discount future cash flows and incorporate the timing of expected payments.

Actuarial time-dependent models also play a pivotal role in long-term planning, prospective research, and decision-making. Actuaries analyze demographic trends, mortality rates, economic factors, and regulatory changes over time to assess the long-term sustainability of insurance products or pension plans. Noteworthy theoretical and practical applications of dynamic decision-making processes can be found in Guillén et al. 2012, where time-varying effects are exhibited in the analysis of customer loyalty as well as in the applications discussed in Chapters 8 and 12 of this thesis.

Another aspect in which time is incorporated is the performance of financial projections used to evaluate the financial health and solvency of insurance companies, pension funds, and other financial institutions. These projections involve modeling future cash flows, asset values, liabilities, and capital requirements over time. By accounting for the time dimension, actuaries can assess the adequacy of reserves, capital buffers, and risk-mitigation strategies in the face of uncertain future events. Research contributions in this area include the examination of portfolio insurance and hedging strategies by Katsikis et al. 2020 and the work of Medvedeva et al. 2021.

Finally, actuarial models incorporate time to analyze the probabilities of future events, such as claims, lapses, deaths, and financial market fluctuations, and to estimate their associated financial consequences. Consequently, the use of survival analysis tools to predict time-to-event target variables is common and can benefit from a dynamic framework, as demonstrated in this thesis.

In summary, the time dimension holds significant importance in actuarial modeling, as it enables the analysis of risk, accurate pricing of policies, estimation of liabilities, long-term planning, and financial projections. Actuaries rely heavily on time-dependent models to comprehend and manage the inherent uncertainties within insurance and financial systems.

2. Research objectives and scope

We now delve into the essence of our research, focusing on the consideration of time-dependent features in tree-based models, especially their applications to lapse behavior in life insurance. Building on the broad motivations established in the previous chapter, where we provided a foreword on the research topic and outlined its significance, this chapter identifies specific research problems and establishes the objectives and scope of this thesis. We begin by defining the key concepts in the domains of management, insurance, and life insurance, including churn and lapse. We recognize the importance of thoroughly understanding these concepts as they serve as fundamental building blocks for subsequent analyses. We then clearly define the research problem this thesis addresses and outline the specific research applications discussed.

We recognize the importance of grounding our analysis in a real-world context, and for this purpose, we will detail compelling application examples in the domain of life insurance. These examples are all based on the analysis of a unique dataset and serve as illustrative case studies throughout Parts 3 to 5 of the thesis, enabling us to explore and examine the intricacies of lapse behavior in a practical setting. By meticulously studying temporal dynamics and leveraging the power of tree-based models, we aim to gain deeper insights into the underlying factors influencing lapses, thereby contributing to enhanced risk assessment and management in the life insurance industry.

2.1 What is churn?

In business management, churn refers to the phenomenon in which customers discontinue their relationships with a company or brand. It is a critical metric for businesses across various industries as it directly impacts customer loyalty, revenue, and long-term sustainability. The importance of understanding and addressing churn has been widely recognized in the literature. Ascarza et al. 2018 highlights the significance of enhanced customer retention management strategies to mitigate churn and emphasizes the need for businesses to manage customer relationships proactively. To manage churn effectively, accurate prediction of customer churn is essential.

Churn in various industries, including insurance, is often modeled using statistical and ML approaches. These models aim to capture the underlying patterns and factors that contribute to churn behaviors. For instance, Duchemin and Matheus 2021, for instance, compares the performance of statistical methods for churn prediction over various evaluation metrics. The commonly used techniques include survival analysis, causal inference, logistic regression, TBMs, NNs, and ensemble methods. Survival analyses, such as Cox proportional hazards models, are frequently employed to model the time to a churn event, considering the duration of the customer relationship before the churn occurs. Using causal inference methods, businesses can assess the impacts of different strategies or interventions on churn and make informed decisions regarding the actions that are the most effective in reducing churn rates. They can test the effectiveness of

targeted retention campaigns, pricing adjustments, and service enhancements in reducing churn. In automobile insurance, such an approach can be found in Verschuren 2022. Logistic regression is a parametric approach often used to predict binary churn outcomes based on the relevant predictors (see Loisel, Piette, and C.H.J. Tsai 2021, for instance). Conversely, decision trees, random forests, and gradient boosting provide non-linear approaches for identifying churn predictors and establishing decision rules. In addition, NNs offer flexible and powerful methods of capturing complex relationships and patterns in churn behaviors. Overall, ML techniques have been demonstrated as effective in this domain and references for approaches leveraging historical data related to customer characteristics, behaviors, transactional information, or interactions to estimate the churn likelihood are compiled in this chapter. For example, hybrid NNs have shown promise for predicting customer churn. C.F. Tsai and Lu 2009 propose a hybrid neural network approach for churn prediction, combining the strengths of different neural network architectures. Such hybrid models can capture complex nonlinear relationships and improve prediction performance. Recently, Bogaert and Delaere 2023 conduct a comprehensive analysis of ensemble methods for customer churn prediction. This analysis demonstrated the ability of ML algorithms to predict churn, showing that these algorithms surpassed traditional statistical approaches in terms of accuracy and predictive power.

ML algorithms can leverage vast amounts of customer data to identify patterns and indicators of potential churns. Geiler, Affeldt, and Nadif 2022 provides a comprehensive survey of ML methods for churn prediction, highlighting the diverse range of techniques used, including decision trees, NNs, and ensemble methods. By incorporating various data sources and considering multiple factors, such as customer demographics, purchase histories, and interaction patterns, ML models can capture complex relationships and deliver accurate churn predictions. In addition to traditional data sources, incorporating non-traditional data such as customer-company interaction emails and emotional cues can further improve churn predictions. Coussement and Van den Poel 2009 explored the integration of emotions from client-company interaction emails, while Coussement, Dries, and Van den Poel 2010 demonstrated the use of generalized additive models in marketing decision-making within a churn prediction context. These studies highlight the potential of leveraging additional data dimensions to enhance churn prediction accuracy.

Predicting churn enables businesses to retain customers proactively and implement targeted marketing strategies. Burez and Van den Poel 2007; Burez and Van den Poel 2009 highlighted the use of analytical models to reduce customer attrition through targeted marketing and identified various challenges that arise in such analyses. These studies illustrate how churn prediction can inform marketing decision making, enabling businesses to allocate resources effectively and tailor retention initiatives to at-risk customers.

Overall, the definition and understanding of churn coupled with the implementation of effective customer retention management strategies are crucial for businesses aiming to reduce customer attrition and enhance customer loyalty. ML techniques offer powerful tools for predicting churn, enabling businesses to address customer churn proactively. By leveraging these models, businesses can gain insights into customer churn dynamics, identify high-risk individuals, and devise effective retention strategies to mitigate churn and foster long-term customer relationships.

2.2 Life insurance and lapse management strategies (LMS)

2.2.1 Specificities about life insurance

Life insurance involves a contractual agreement between an individual (or policyholder (PH)) and an insurance company. It provides financial protection by offering a payout or death benefit to designated beneficiaries after the death of the policyholder. Life insurance policies often include provisions for savings or investment components, which enable policyholders to accumulate cash value over time. The unique nature of life insurance establishes long-term relationships between customers and insurance companies, typically spanning many years. In a life insurance customer-company relationship, policyholders pay regular premiums to insurance companies, ensuring continuous coverage and guaranteeing death benefits. In return, the insurance company commits to honoring the policy terms and providing agreement benefits to beneficiaries upon the death of the policyholder. This relationship relies on the understanding that policyholders maintain their policies and keep up with premium payments over the duration of the policy.

However, policyholder behaviors, known as lapses and surrenders, pose challenges to this relationship. A lapse occurs when a policyholder discontinues premium payments, enabling the policy to be terminated before the intended duration. This situation can occur for various reasons such as financial difficulties, changing priorities, or dissatisfaction with policies. Surrender, on the other hand, refers to the voluntary termination of the policy by the policyholder, often resulting in the withdrawal of the accumulated cash values. Throughout this thesis, we will refer to both behaviors using the word “lapse” without distinction. Modeling lapses is critical for insurers because of their financial and actuarial implications. When policyholders lapse or surrender their policies, insurers face financial consequences including the loss of future premium payments and potential surrender charges. Moreover, lapses disrupt the actuarial calculations and risk assessments necessary for insurers to price policies accurately and manage their financial stability. The similarities between lapses in life insurance and churns in any other industry lie in the fact that it describes the event of customers discontinuing their relationships with a company. However, significant differences can be noted. In life insurance, the term “lapse” refers specifically to the termination of an insurance policy, often involving the cessation of premium payments, and its timing and conditions are often contractually agreed. The critical distinction is that lapses in life insurance involve the discontinuation of coverage, potentially leaving policyholders without intended financial protection. By contrast, churn in other industries may not have such severe consequences in terms of the loss of essential services or benefits.

ML has revolutionized numerous industries, resulting in significant benefits and advancements. However, the life insurance industry has been relatively slow in adopting ML techniques compared with other sectors. Traditionally, statistical models have been the preferred choice for risk assessment in life insurance, leading insurers to question the value and effectiveness of AI in their domain. Although the traditional actuarial methodologies have been effective in survival modeling, incorporating ML techniques can enhance predictive accuracy and provide additional insights. By understanding the drivers and patterns of lapses, insurers can develop targeted retention strategies, adjust pricing structures, implement proactive measures to minimize lapses, optimize policyholder retention, and maintain stable customer portfolios.

2.2.2 Application to lapse management strategy

Throughout this thesis, the contributions are illustrated using real-world applications and case studies that are directly linked to the management of policy lapses in the life insurance industry. From the perspective of a life insurer, effective policyholder retention is crucial for maximizing profitability and managing risks. Attracting new customers in competitive markets is challenging and requires investments five to six times greater than the cost of preventing existing customers from churning (see Athanassopoulos 2000). Regardless of the context and specificities of the insurer, the design of a retention strategy is at least always justified because a high structural lapse rate automatically translates into *customer volatility* and leads to increased management and marketing costs, as new policyholders must be constantly sought. A retention strategy is a set of actions and initiatives developed and implemented by insurers to encourage policyholders to continue their policies and minimize lapse probabilities or lapse financial consequences. These strategies involve understanding the behaviors, preferences, and motivations of policyholders, enabling insurers to personalize their approaches and provide tailored incentives or services to retain policyholders.

However, not every policyholder who is likely to lapse should be retained because they may not generate future profits for the insurer. Despite having a high probability of lapsing, some policyholders may not contribute significantly to the profitability of the insurer because of low premiums or other factors. Therefore, insurers must differentiate between policyholders who are likely to lapse but still generate future profits and those who are less likely to contribute to profitability. In our research, we address this challenge by developing innovative frameworks that utilize the concept of customer lifetime value (CLV), a well-established marketing tool that estimates the potential future value generated by a customer over his or her entire relationship with a business to estimate the profitability of individual policies. By applying the CLV to life insurance, we can assess the profitability of each policy by considering factors such as future premium payments, potential policy upgrades, and the likelihood of policyholder lapses or death. Given a portfolio, this enables us to identify which policyholders are the most likely to lapse while generating substantial profits for the insurer. Consequently, we could prioritize retention efforts and allocate resources effectively, targeting policyholders with higher CLVs and higher probabilities of lapsing who still contribute significantly to the profitability of the insurer.

Importantly, the inclusion of CLV in the lapse management process ensures a **policyholder-centered** analysis. This thesis argues that this approach is to be preferred for several reasons. First, life insurance policies are distinct financial contracts with unique characteristics including premium amounts, coverage terms, and policyholder demographics. By focusing on individual policies rather than on aggregated data, customer-centered analysis enables a more granular understanding of policyholder behaviors and their effects on profitability. Second, individualized analysis enables insurers to tailor their retention strategies to the specific needs and circumstances of each policyholder. This approach acknowledges that policyholders have varying motivations for lapsing or surrendering their policies and that a personalized approach is more effective in addressing their concerns and retaining their policies.

Furthermore, if lapse management is individualized to account for variations in behavior across policyholders, it should also account for variations over time. We recognize that the likelihoods of policyholders to lapse or surrender can change over time and that considering the evolving nature of these behaviors is essential. Therefore, we explore highly individualized methodologies and considered the use of historical data to capture policyholder behavioral patterns and account for temporal dynamics. By analyzing past policyholder actions, such as premium pay-

ment history and changes in coverage, we can identify patterns that help predict future behaviors and tailor retention efforts accordingly. The incorporation of temporal dynamics into tree-based models is a central aspect of our study. By considering the time dimension, we can capture the dynamic nature of policyholder behavior and make more accurate predictions regarding lapses or surrenders. By understanding the evolving patterns and adapting our approaches over time, insurers can proactively address policyholder concerns, maximize policyholder retention, optimize profitability, and make informed decisions. These techniques provide a structured framework for analyzing and predicting the likelihood of policyholder lapses as well as the expected profitability at stake, considering multiple individualized factors such as policy features, policyholder demographics, and historical behaviors, thereby enabling insurers to make informed decisions regarding their retention strategies.

In summary, our research aims to address the challenge of optimizing policyholder retention in a whole-life life insurance portfolio. We developed different frameworks that utilize individualized future CLVs to estimate the profitability generated by any given policy, enabling insurers to identify and target the policyholders who are the most likely to lapse while still generating future profits. These applications demonstrate that our research enhances life insurance portfolio management and enables insurers to make data-driven decisions that maximize profitability and mitigate risks. Overall, through these applications, this thesis emphasizes the importance of a policy-centered analysis in life insurance portfolio management. By employing advanced tree-based ML methods, the applications show that our frameworks can provide insurers with individualized insights into policyholder behaviors, enabling the development of effective lapse management strategies that consider temporal dynamics and maximize profitability.

Lapse management strategies, when implemented effectively by life insurers, can also offer benefits to policyholders in several ways. Firstly, profitable PH might receive incentives or benefits to encourage them to maintain their policies, it can include premium discounts, bonuses, or additional coverage options. Secondly, encouraging PH to retain their policies ensures that they maintain the financial protection and coverage that the policy originally provided. This continuity in coverage can be crucial for the policyholder's financial security and protection of their beneficiaries. While acknowledging for those advantages from the PH's perspective, our work consistently adopts the perspective of the insurer and exploring the viewpoints of an individual, a broker, or a reinsurer, would be truly insightful but remains out of the scope of this thesis.

3. Thesis structure, objectives, and contributions

3.1 Thesis structure and objectives

This thesis comprises six main parts.

The first one, in which we currently are, is a general introduction. It serves as a contextualizing part by providing background information, relevant theories, concepts, and previous research on ML, tree-based algorithms, and insurance; defines our research problem; and evokes the life insurance application examples that are utilized throughout the following parts.

The second part focuses on introducing theoretical generalities about tree-based ML algorithms. It provides a historical overview of the classical tree-based algorithms and modeling methodologies at our disposal. It then describes the specificities of survival analysis by detailing the usual notations and introduces tree-based survival methods. Along with the introduction, this part describes background knowledge rather than presenting novel information. It is intended to enable an independent understanding of the remaining parts of the manuscript.

The third part is dedicated to defining a policyholder-centered lapse management framework for life insurers: an LMS. It considers tree-based models and involves a survival analysis with competing risks. After a brief overview of the use of the CLV in management and actuarial science, this section explores the various benefits of an enlightened, individualized, and profit-driven tree-based retention strategy based on survival - and time-dependent considerations. This part shows that time considerations can be beneficial to the field of actuarial lapse management. It is based on the article *Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategy* published by the European Actuarial Journal as a joint work with Xavier Milhaud and Anani Olympio.

The fourth part suggests that the LMS framework can be adapted to time-varying covariates. It introduces general notations, presents state-of-the-art longitudinal models, and discusses how such models could benefit the actuarial field. It defines an LMS longitudinal framework that provides a time-informed retention strategy. This new methodology constitutes a second temporal layer that can be included in the LMS framework. Its strengths, flaws, and further improvements are discussed in various settings. This application is based on the article entitled *A longitudinal ML framework for lapse management in life insurance* submitted to the Annals of Actuarial Science. We argue that it provides a valuable contribution to the field of lapse analysis for life insurers and highlights the importance of using the complete past trajectory of policyholders, which is often available in the information systems of insurers but is rarely exploited.

The fifth part is dedicated to the exploration of innovative longitudinal tree-based algorithms that can handle time-varying covariates. It introduces a new tree-based data mining algorithm: Time-penalized Tree (TpT), and an application to policyholder-centered lapse analysis is discussed.

The sixth and last section constitutes the general conclusion of this thesis.

Remark 3.1

Before diving into the following parts of the thesis, note that the sections based on published, submitted works or working papers include complete and unmodified parts of the original articles (Chapters 8, 12, and 14). This characteristic provides the advantage that every part of the manuscript can be read independently, with its corresponding appendix. However, it also means that **notations or contextual elements may be repeated several times along this thesis**. All passages directly taken from articles are referenced and contained within clearly identifiable chapters, with the abstracts and keywords of the original source included. Nevertheless, most general notations are homogenous throughout all chapters and all common mathematical symbols and their meanings can be found in the [list of abbreviations and symbols](#). This choice was made out of convenience for the future readers of this work, who may not need nor want to read it all at once.

List of contributions

Thesis parts III, IV, and V are based on the following publications, submissions, and working papers:

1. Mathias Valla, Xavier Milhaud, Anani Ayodélé Olympio. Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies. *European Actuarial Journal*, inPress. [\(hal-03903047v3\)](#) (Valla, Milhaud, and Olympio [2023](#))
2. Mathias Valla. A longitudinal ML framework for lapse management in life insurance. Working Paper. [\(hal-04178278\)](#) (Valla [2023a](#))
3. Mathias Valla. Time-penalized trees (TpT): a new tree-based data mining algorithm for time-varying covariates. Working Paper. [\(hal-04178282\)](#) (Valla [2023b](#))

The present thesis and the articles it relies on attempt to address multiple gaps in the actuarial literature. A list of the main contributions of this work to the field can be found below:

65		Individualized CLV with competing risks
65		Using RSF and GBSM for lapse analysis
66		Development of a new LMS framework
66		Business-oriented discussion
104		New longitudinal LMS framework
104		Use of longitudinal TBM in life insurance
144		Gaps within the longitudinal TBM literature
144		Introduction of TpT

Bibliography

- Wigner, E.P. (1960). “The unreasonable effectiveness of mathematics in the natural sciences. Richard Courant lecture in mathematical sciences delivered at New York University, May 11, 1959”. In: *Communications on Pure and Applied Mathematics* 13.1, pp. 1–14. DOI: <https://doi.org/10.1002/cpa.3160130102>.
- Eling, M., D. Nuesle, and J. Staubli (Feb. 2021). “The impact of artificial intelligence along the insurance value chain and on the insurability of risks”. In: *The Geneva Papers on Risk and Insurance - Issues and Practice* 47. DOI: [10.1057/s41288-020-00201-7](https://doi.org/10.1057/s41288-020-00201-7).
- Lestavel, T (Mar. 2017). In: *L'Actuariel* n°24, pp. 12–19. URL: <https://thomaslestavel.files.wordpress.com/2017/04/actuariel-24--dossieria.pdf%7D>.
- Eckert, C., C. Neunsinger, and K. Osterrieder (Feb. 2022). “Managing customer satisfaction: digital applications for insurance companies”. In: *The Geneva Papers on Risk and Insurance - Issues and Practice* 47. DOI: [10.1057/s41288-021-00257-z](https://doi.org/10.1057/s41288-021-00257-z).
- Bartram, S.M., J. Branke, and M. Motahari (Mar. 2020). *Artificial Intelligence in Asset Management*. URL: <https://ssrn.com/abstract=3560333>.
- Chancel, A. et al. (2022). *Applying Machine Learning to Life Insurance: some knowledge sharing to master it*. arXiv: [2209.02057](https://arxiv.org/abs/2209.02057) [stat.ML].
- Simonyan, K. and A. Zisserman (2015). “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *International Conference on Learning Representations*.
- He, K. et al. (2016). “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778. DOI: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90).
- Devlin, J. et al. (2018). “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *CoRR* abs/1810.04805. arXiv: [1810.04805](https://arxiv.org/abs/1810.04805). URL: <http://arxiv.org/abs/1810.04805>.
- OpenAI (2023). *GPT-4 Technical Report*. arXiv: [2303.08774](https://arxiv.org/abs/2303.08774) [cs.CL].
- Esteva, A. et al. (2017). “Dermatologist-level classification of skin cancer with deep neural networks”. In: *Nature* 542.7639, pp. 115–118.
- Rajpurkar, P. et al. (2017). “CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning”. In: *CoRR* abs/1711.05225. arXiv: [1711.05225](https://arxiv.org/abs/1711.05225). URL: <http://arxiv.org/abs/1711.05225>.
- Schulman, J. et al. (2015). “Trust Region Policy Optimization”. In: *CoRR* abs/1502.05477. arXiv: [1502.05477](https://arxiv.org/abs/1502.05477). URL: <http://arxiv.org/abs/1502.05477>.
- Gu, S. et al. (2017). “Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates”. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. Singapore, Singapore: IEEE Press, pp. 3389–3396. DOI: [10.1109/ICRA.2017.7989385](https://doi.org/10.1109/ICRA.2017.7989385).
- Jin, X. et al. (2019). “Deep learning for plant identification in natural environment”. In: *Computers and Electronics in Agriculture* 163, p. 104859.

- Xie, Y., D. Zhang, and G. Wang (2019). “A deep learning framework for crop classification using multi-temporal satellite SAR images”. In: *Remote Sensing of Environment* 232, p. 111320.
- Wüthrich, M.V. (2018). “Machine learning in individual claims reserving”. In: *Scandinavian Actuarial Journal* 2018.6, pp. 465–480. DOI: [10.1080/03461238.2018.1428681](https://doi.org/10.1080/03461238.2018.1428681).
- Frees, E.W. and E.A. Valdez (2019). “Deep learning in actuarial science: Generalized linear models with neural networks”. In: *Annals of Actuarial Science* 13.2, pp. 221–251.
- Shi, Q., J. Yang, and Z. Li (2020). “Deep learning in actuarial science: A review”. In: *Scandinavian Actuarial Journal* 2020.4, pp. 301–331.
- Zhang, L., E.W. Frees, and E.A. Valdez (2020). “Deep learning for insurance claim prediction”. In: *Risks* 8.1, p. 16.
- Chen, Y., Y. Wu, and C. Wu (2020). “A deep learning framework for mortality modeling”. In: *Insurance: Mathematics and Economics* 92, pp. 99–110.
- Teixeira, R., M.A. Ferreira, and D. Pestana (2020). “Deep learning for modeling insurance claims using satellite images”. In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 69.4, pp. 911–930.
- Belzile, L. (2021). “Deep learning with long short-term memory for pricing and valuation in insurance”. In: *Scandinavian Actuarial Journal* 2021.3, pp. 271–292.
- McDonnell, K. et al. (May 2023). “Deep learning in insurance: Accuracy and model interpretability using TabNet”. In: *Expert Systems with Applications* 217, p. 119543. DOI: [10.1016/j.eswa.2023.119543](https://doi.org/10.1016/j.eswa.2023.119543).
- Tsantekidis, A. et al. (2018). “Forecasting stock prices using an echo state network variant with trend bias correction”. In: *Neurocomputing* 275, pp. 1988–1998.
- Zhang, L. and W. Wu (2019). “Deep learning for sentiment analysis on financial news articles”. In: *Journal of Big Data* 6.1, pp. 1–23.
- Dehghani, F. and A. Larijani (2023). “Average Portfolio Optimization Using Multi-Layer Neural Networks With Risk Consideration”. In: Available at SSRN: <https://ssrn.com/abstract=4436648>. DOI: [10.2139/ssrn.4436648](https://doi.org/10.2139/ssrn.4436648). URL: <https://ssrn.com/abstract=4436648>.
- Borisov, V. et al. (2021). “Deep Neural Networks and Tabular Data: A Survey”. In: *IEEE transactions on neural networks and learning systems* PP.
- Shwartz-Ziv, R. and A. Armon (2022). “Tabular data: Deep learning is not all you need”. In: *Information Fusion* 81, pp. 84–90. ISSN: 1566-2535. DOI: <https://doi.org/10.1016/j.inffus.2021.11.011>. URL: <https://www.sciencedirect.com/science/article/pii/S1566253521002360>.
- Grinsztajn, L., E. Oyallon, and G. Varoquaux (2022). “Why do tree-based models still outperform deep learning on typical tabular data?” In: *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Wuthrich, M.V. (2019). “From Generalized Linear Models to Neural Networks, and Back”. In: This paper has been integrated into SSRN Manuscript 3822407. DOI: [10.2139/ssrn.3491790](https://doi.org/10.2139/ssrn.3491790). URL: <https://ssrn.com/abstract=3491790>.
- Schelldorfer, J. and M.V. Wuthrich (2019). “Nesting Classical Actuarial Models into Neural Networks”. In: This paper has been integrated into SSRN Manuscript 3320525. DOI: [10.2139/ssrn.3320525](https://doi.org/10.2139/ssrn.3320525). URL: <https://ssrn.com/abstract=3320525>.
- Holvoet, F., K. Antonio, and R. Henckaerts (Oct. 2023). *Neural networks for insurance pricing with frequency and severity data: a benchmark study from data preprocessing to technical tariff*. Tech. rep. arXiv.org. URL: <https://ideas.repec.org/p/arx/papers/2310.12671.html>.
- Wolpert, D.H. and W.G. Macready (1997). “No free lunch theorems for optimization”. In: *IEEE Transactions on Evolutionary Computation* 1.1, pp. 67–82. DOI: [10.1109/4235.585893](https://doi.org/10.1109/4235.585893).

- Henckaerts, R. et al. (2021). “Boosting Insights in Insurance Tariff Plans with Tree-Based Machine Learning Methods”. In: *North American Actuarial Journal* 25.2, pp. 255–285. DOI: [10.1080/10920277.2020.1745656](https://doi.org/10.1080/10920277.2020.1745656).
- Farkas, S., O. Lopez, and M. Thomas (2021). “Cyber claim analysis using Generalized Pareto regression trees with applications to insurance”. In: *Insurance: Mathematics and Economics* 98, pp. 92–105. ISSN: 0167-6687. DOI: <https://doi.org/10.1016/j.insmatheco.2021.02.009>. URL: <https://www.sciencedirect.com/science/article/pii/S0167668721000330>.
- Maillart, A. and C.Y. Robert (2023). “Tail index partition-based rules extraction with application to tornado damage insurance”. In: *ASTIN Bulletin: The Journal of the IAA* 53.2, pp. 258–284. DOI: [10.1017/asb.2023.1](https://doi.org/10.1017/asb.2023.1).
- Bauer, D. and U. Hommel (2007). “Time Trends in Insurance: A Survey”. In: *European Actuarial Journal* 7.1, pp. 21–45.
- Tan, K.S. and H.K. Yow (2011). “Risk Management and Time”. In: *ASTIN Bulletin: The Journal of the International Actuarial Association* 41.2, pp. 613–651.
- Gao, J. and X. Wu (2013). “The Impact of Time on Life Insurance Profitability”. In: *ASTIN Bulletin: The Journal of the International Actuarial Association* 43.1, pp. 1–25.
- Olivieri, A. and E. Pitacco (2015). “Modeling and Forecasting Mortality: A Smooth Transition Approach with Time-Varying Transition Probabilities”. In: *Scandinavian Actuarial Journal* 2015.5, pp. 377–405.
- Zaks, Y. and M. Sherris (2018). “Modeling Dependence Structures and Correlations with Time: Applications to Insurance”. In: *North American Actuarial Journal* 22.4, pp. 367–398.
- Holmström, B. (1979). “Moral Hazard and Observability”. In: *The Bell Journal of Economics* 10.1, pp. 74–91. ISSN: 0361915X, 23263032. URL: <http://www.jstor.org/stable/3003320> (visited on 05/25/2023).
- Abbring, J.H. et al. (May 2003). “Adverse Selection and Moral Hazard in Insurance: Can Dynamic Data Help to Distinguish?” In: *Journal of the European Economic Association* 1.2-3, pp. 512–521. ISSN: 1542-4766. DOI: [10.1162/154247603322391152](https://doi.org/10.1162/154247603322391152). eprint: <https://academic.oup.com/jeea/article-pdf/1/2-3/512/10313493/jeea0512.pdf>.
- Bolancé, C., M. Guillén, and J. Pinquet (2003). “Time-varying credibility for frequency risk models: estimation and tests for autoregressive specifications on the random effects”. In: *Insurance: Mathematics and Economics* 33.2. Papers presented at the 6th IME Conference, Lisbon, 15-17 July 2002, pp. 273–282. ISSN: 0167-6687. DOI: [https://doi.org/10.1016/S0167-6687\(03\)00139-2](https://doi.org/10.1016/S0167-6687(03)00139-2). URL: <https://www.sciencedirect.com/science/article/pii/S0167668703001392>.
- Shi, P. and E.A. Valdez (2014). “Longitudinal modeling of insurance claim counts using jitters”. In: *Scandinavian Actuarial Journal* 2014.2, pp. 159–179. DOI: [10.1080/03461238.2012.670611](https://doi.org/10.1080/03461238.2012.670611).
- Lee, G.Y. and P. Shi (2019). “A dependent frequency–severity approach to modeling longitudinal insurance claims”. In: *Insurance: Mathematics and Economics* 87, pp. 115–129. ISSN: 0167-6687. DOI: <https://doi.org/10.1016/j.insmatheco.2019.04.004>. URL: <https://www.sciencedirect.com/science/article/pii/S0167668718301951>.
- Guillén, M. et al. (2012). “Time-varying effects in the analysis of customer loyalty: A case study in insurance”. In: *Expert Systems with Applications* 39.3, pp. 3551–3558. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2011.09.045>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417411013546>.

- Katsikis, V.N. et al. (2020). "Time-varying minimum-cost portfolio insurance under transaction costs problem via Beetle Antennae Search Algorithm (BAS)". In: *Applied Mathematics and Computation* 385, p. 125453. ISSN: 0096-3003. DOI: <https://doi.org/10.1016/j.amc.2020.125453>. URL: <https://www.sciencedirect.com/science/article/pii/S0096300320304148>.
- Medvedeva, M.A. et al. (2021). "Randomized time-varying knapsack problems via binary beetle antennae search algorithm: Emphasis on applications in portfolio insurance". In: *Mathematical Methods in the Applied Sciences* 44.2, pp. 2002–2012. DOI: <https://doi.org/10.1002/mma.6904>.
- Ascarza, E. et al. (Mar. 2018). "In Pursuit of Enhanced Customer Retention Management: Review, Key Issues, and Future Directions". In: *Customer Needs and Solutions* 5. DOI: [10.1007/s40547-017-0080-0](https://doi.org/10.1007/s40547-017-0080-0).
- Duchemin, R. and R. Matheus (Dec. 2021). "Forecasting customer churn: Comparing the performance of statistical methods on more than just accuracy". en. In: *Journal of Supply Chain Management Science* 2.3-4, pp. 115–137.
- Verschuren, R.M. (2022). "Customer price sensitivities in competitive insurance markets". In: *Expert Systems with Applications* 202, p. 117133. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2022.117133>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417422005309>.
- Loisel, S., P. Piette, and C.H.J. Tsai (2021). "Applying economic measures to lapse risk management with Machine Learning approaches". In: *ASTIN Bulletin: The Journal of the IAA* 51.3, pp. 839–871. DOI: [10.1017/asb.2021.10](https://doi.org/10.1017/asb.2021.10).
- Tsai, C.F. and Y.H. Lu (2009). "Customer churn prediction by hybrid neural networks". In: *Expert Systems with Applications* 36.10, pp. 12547–12553. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2009.05.032>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417409004758>.
- Bogaert, M. and L. Delaere (2023). "Ensemble Methods in Customer Churn Prediction: A Comparative Analysis of the State-of-the-Art". In: *Mathematics* 11.5, pp. 1–28. URL: <https://ideas.repec.org/a/gam/jmathe/v11y2023i5p1137-d1079547.html>.
- Geiler, L., S. Affeldt, and M. Nadif (2022). "A survey on machine learning methods for churn prediction". In: *International Journal of Data Science and Analytics* 14.3, pp. 217–242. ISSN: 2364-4168. DOI: [10.1007/s41060-022-00312-5](https://doi.org/10.1007/s41060-022-00312-5).
- Coussement, K. and D. Van den Poel (2009). "Improving customer attrition prediction by integrating emotions from client/company interaction emails and evaluating multiple classifiers". In: *Expert Systems with Applications* 36.3, Part 2, pp. 6127–6134. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2008.07.021>. URL: <https://www.sciencedirect.com/science/article/pii/S095741740800479X>.
- Coussement, K., B.F. Dries, and D. Van den Poel (Mar. 2010). "Improved marketing decision making in a customer churn prediction context using generalized additive models". English. In: *Expert Systems with Applications* 37.3, pp. 2132–2143. ISSN: 0957-4174. DOI: [10.1016/j.eswa.2009.07.029](https://doi.org/10.1016/j.eswa.2009.07.029).
- Burez, J. and D. Van den Poel (2007). "CRM at a pay-TV company: Using analytical models to reduce customer attrition by targeted marketing for subscription services". In: *Expert Systems with Applications* 32.2, pp. 277–288. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2005.11.037>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417405003374>.
- (2009). "Handling class imbalance in customer churn prediction". In: *Expert Systems with Applications* 36.3, Part 1, pp. 4626–4636. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2009.07.029>.

- 10.1016/j.eswa.2008.05.027. URL: <https://www.sciencedirect.com/science/article/pii/S0957417408002121>.
- Athanassopoulos, A. (Mar. 2000). “Customer Satisfaction Cues To Support Market Segmentation and Explain Switching Behavior”. In: *Journal of Business Research* 47, pp. 191–207. DOI: 10.1016/S0148-2963(98)00060-5.
- Valla, M., X. Milhaud, and A.A. Olympio (Sept. 2023). “Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies”. In: *European Actuarial Journal*. DOI: 10.1007/s13385-023-00358-0. URL: <https://hal.science/hal-03903047> (HAL), <https://export.arxiv.org/pdf/2307.06651> (arxiv), https://link.springer.com/article/10.1007/s13385-023-00358-0?code=84d3a0d0-b866-48d5-bc60-5ed6832d144a&error=cookies_not_supported (journal).
- Valla, M. (July 2023a). “A longitudinal framework for lapse management in life insurance”. working paper or preprint. URL: <https://hal.science/hal-04178278>.
- (Aug. 2023b). “Time-penalized trees (TpT): a new tree-based data mining algorithm for time-varying covariates”. working paper or preprint. URL: <https://hal.science/hal-04178282> (HAL), <https://www.researchsquare.com/article/rs-3400744/v1> (research square).

Part II

Machine learning and tree-based models

Chapter 4 Methodology and methods

- 4.1 Machine learning methodology 22
 - 4.1.1 Training 23
 - 4.1.2 Tuning and evaluation 27
- 4.2 Models description 34
 - 4.2.1 History of decision trees 34
 - 4.2.2 Ensemble models 41
 - 4.2.3 Theoretical guarantees 44

Chapter 5 Survival analysis

- 5.0.1 Survival notations 49
- 5.1 Models 51
 - 5.1.1 Inverse probability of censorship weighted (IPCW) models 51
 - 5.1.2 Survival trees 52
 - 5.1.3 Random survival forests (RSF) 53
 - 5.1.4 Gradient Boosting Survival Model (GBSM) 54
- 5.2 Survival performance metrics 55
 - 5.2.1 Brier Score and variations 55
 - 5.2.2 Concordance indices 56
 - 5.2.3 Dynamic AUC 57

Bibliography

4. Methodology and methods

4.1 Machine learning methodology

This chapter broadens the extent of PART I by presenting the ML methodology employed to address our research questions and introducing classical tree-based models and evaluation metrics that are used in various applications throughout this thesis. This chapter serves as a comprehensive guide to the techniques, algorithms, and processes employed in our studies.

Statistical models are broadly categorized into two types based on the number of parameters they use: parametric models and non-parametric models. Parametric models assume a fixed form of the relationship between inputs and outputs with a predetermined number of parameters learned from training data, such as linear regression. On the other hand, non-parametric models, don't rely on fixed assumptions about the functional form of the underlying distribution of variables of interest and have a flexible number of parameters that grow with the data. Most ML approaches, including TBMs, are non-parametric, and their calibration and evaluation necessitate a specific methodology, detailed in this part.

Here, we provide a concise overview of the ML methodology, emphasizing its fundamental principles and underlying concepts. Understanding the core ideas of ML and tree-based models is crucial for comprehending the subsequent sections, in which we delve into the specific algorithms employed. We outline the steps of the ML pipeline, which encapsulates the entire process from data collection and preprocessing to model training and evaluation. Each step is discussed in detail, highlighting the choices made and justifying the decisions based on the research objectives and available resources. Moreover, we focus on model training and evaluation processes, shedding light on the selection of appropriate performance metrics, cross-validation techniques, and (hyper)parameter tuning. These critical steps are essential for optimizing the predictive capabilities and generalization power of the models.

Furthermore, we present a thorough exploration of the various tree-based algorithms used in our study with descriptions and illustrations. We cover both traditional algorithms, such as decision trees, and more advanced techniques, such as bagging and boosting ensemble methods. For each algorithm, we discuss the existing variations, availability, complexity, theoretical underpinnings, suitability in our research context, and potential limitations.

Through this chapter, readers will gain a comprehensive understanding of the ML methodology and properties of the tree-based models employed in our research, enabling them to evaluate our approach critically and interpret the results of the subsequent chapters with confidence.

4.1.1 Training

In this section, we explore the training methodology in detail, revealing the intricate steps, techniques, and considerations involved in the training process. By understanding the essence of the models and predictions, we can grasp the fundamental purpose and significance of the training methodology in ML. Subsequently, we introduce the *Train/Test/Validation* methodology and justify it by detailing the concepts of bias, variance, overfitting, and underfitting.

Model and predictions

Let us assume that we study a dataset, denoted \mathcal{D} , with N rows, or observations, each containing the values of p variables. The vector of covariates corresponding to the i -th observation is denoted $\mathbf{x}^{(i)} = (x_1^{(i)}, x_2^{(i)}, \dots, x_p^{(i)})$, $i \in [1, \dots, N]$. These variables are called covariates, inputs, features, or predictors and are shown in equation 4.1. Each row has a corresponding variable $y^{(i)}$ that can be denoted as a target, response, outcome, label, or output variable. For all inputs $\mathbf{x}^{(i)} \in \mathcal{X}$, the covariate space, a unique output $y^{(i)} \in \mathcal{Y}$, the output space, is provided. If the output - thus predictions - takes continuous values, this is a regression model; if it takes categorical or discrete values, this is a classification one. In any case, we denote

$$X = \begin{pmatrix} x_1^{(1)} & x_2^{(1)} & \dots & x_p^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \dots & x_p^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{(N)} & x_2^{(N)} & \dots & x_p^{(N)} \end{pmatrix} \quad Y = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(N)} \end{pmatrix}. \quad (4.1)$$

X and Y are the input and output matrices of \mathcal{D} , with $\mathcal{D} = \{X, Y\} = \{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^N$.

The typical supervised learning task then consists of predicting an output $y \in \mathcal{Y}$ from an input $\mathbf{x} \in \mathcal{X}$, where the pairs (\mathbf{x}, y) are taken from an unknown joint distribution, \mathcal{J} : it is the process of learning a mapping from \mathcal{X} to \mathcal{Y} . In other words, a learning algorithm seeks to estimate a function $g : \mathcal{X} \rightarrow \mathcal{Y}$ from a finite training dataset \mathcal{D}_{train} , consisting of N_{train} samples from \mathcal{J} . We refer to the learned function g as a hypothesis function, an element of a space of possible functions allowed by the model \mathcal{G} called the hypothesis space. Then, the learned function g and the learning algorithm $\mathcal{A} : (\mathcal{X} \times \mathcal{Y})^{N_{train}} \rightarrow \mathcal{G}$ can be formalized as $g \leftarrow \mathcal{A}(\mathcal{D}_{train})$. Ideally, a perfect learning model would yield $g = f$, where f denotes the hypothetical “true mapping” from \mathcal{X} to \mathcal{Y} . In practice, however, f cannot be found explicitly and thus needs to be estimated. This estimation is provided using a loss - or scoring, or evaluation - function $\ell : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ that assigns a score to any prediction made by a given model. We denote \mathcal{L} as the space of all possible loss functions and will discuss this specific topic in more depth in the next sections. The quality of a predictor g is then assessed by its expected loss (or risk) given by

$$\mathfrak{L}(g) = \mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}} [\ell(g(\mathbf{x}), y)].$$

The chosen model is then the function $g \in \mathcal{G}$ that produces outputs minimizing the expected loss: $g = \arg \min_{g \in \mathcal{G}} (\mathfrak{L}(g))$.

However in practice, $\mathfrak{L}(g)$ can only be estimated as we do not know what the underlying true joint distribution \mathcal{J} is. We can only estimate $\mathfrak{L}(g)$ from the empirical training error given by

$$\hat{\mathfrak{L}}(g) = \mathfrak{L}_{train}(g) = \mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}_{train}} [\ell(g(\mathbf{x}), y)].$$

The chosen model is then the function $g \in \mathcal{G}$ that produces outputs minimizing the expected empirical loss: it is learned by minimizing $\hat{\mathfrak{L}}(g)$ as a surrogate for $\mathfrak{L}(g)$. It is what we refer to as a *model* and it allows to make predictions of the target variables given certain features.

Training methodology, bias-variance and over/underfitting

Every ML model has its own training specificities (see section 4.2 for a detailed overview of classical tree-based algorithms); however, a common objective exists regarding the ability of supervised learning models for prediction tasks. An optimal model aims to minimize errors committed to predictions while avoiding overfitting. Overfitting is a significant concern in ML. It describes the phenomenon in which a model learns training data so well that it performs poorly on unseen or new data. This situation occurs when a model becomes excessively complex or highly specialized for the training dataset, effectively memorizing the noise or idiosyncrasies present in the training observations rather than capturing the underlying general patterns that can reliably be applied to unseen examples.

The common literature on the subject (see Geman, Bienenstock, and Doursat 1992) states that a model that is too simple will yield highly biased results and underfit the data, whereas a model that is too complex will yield predictions with high variance and overfit the data. Hence, the widely accepted concept of the bias-variance trade-off predicts a U-shaped test error curve (see Figure 4.1).

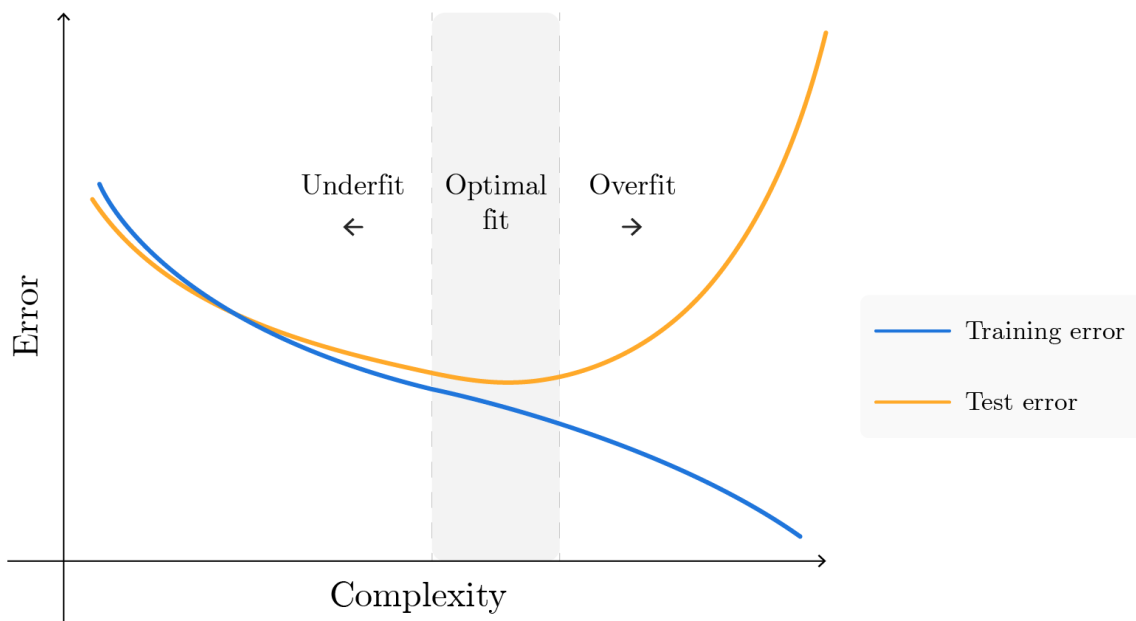


Figure 4.1: Bias-variance tradeoff U-shaped curve

The idea is that a model becomes more accurate as it increases in complexity until it overfits and loses its generalization ability. That is, the error committed to the training set $\mathcal{L}_{train}(g)$ is a decreasing function of the model complexity, whereas $\mathcal{L}_{test}(g)$ first decreases then increases with the complexity of g . Statistical learning theory (see Vapnik 1998) provides a strong theoretical background supporting the concept of trade-offs for several classic ML models. A concrete illustration of that idea for ML methods can be found in Geman, Bienenstock, and Doursat 1992.

Formally, the bias of a model is defined as a measure of how close its expected prediction is to the true underlying function f . For some $\mathbf{x} \sim \mathcal{X}$, the bias is given by:

$$\mathfrak{B}(g) = \mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}_{train}} [g(\mathbf{x})] - f(\mathbf{x}).$$

A model with $\mathfrak{B}(g) = 0$ is said to be unbiased.

However, the variance of a model is a measure of the variability or spread of its predictions, where deviations from their expected values result from different samplings of \mathcal{D}_{train} . For some $\mathbf{x} \sim \mathcal{X}$, the variance is given by¹

$$\text{var}(g) = \mathbb{E}_{(\mathbf{x},y) \sim \mathcal{D}_{train}} \left[(g(\mathbf{x}) - \mathbb{E}_{(\mathbf{x},y) \sim \mathcal{D}_{train}} [g(\mathbf{x})])^2 \right].$$

The intuition behind the concept of generalization is clear. A good generalization implies that the model learns similar functions when trained on different training sets.

In the specific setting, where the squared-loss function is considered as a choice of ℓ , the average loss that can be expected over different training sets can be derived and decomposed into bias and variance components as such

$$\mathfrak{L}_{N_{train}} = \varepsilon_{\text{bias}} + \varepsilon_{\text{variance}} + \varepsilon_{\text{noise}},$$

with

$$\mathbb{E}_{(\mathbf{x},y) \sim \mathcal{D}_{train}} [\varepsilon_{\text{noise}}] = 0, \text{var}(\varepsilon_{\text{noise}}) = \mathbb{E}_{(\mathbf{x},y) \sim \mathcal{D}_{train}} [\varepsilon_{\text{noise}}^2] = \sigma_{\varepsilon_{\text{noise}}}^2.$$

Sketch of proof.

For simplicity's sake, as all expected values are expectations over samples (\mathbf{x}, y) drawn from \mathcal{D}_{train} , we will not carry the identifiers in the following steps.

$$\begin{aligned} \mathbb{E} [(y - g(\mathbf{x}))^2] &= \mathbb{E} [(f(\mathbf{x}) + \varepsilon_{\text{noise}} - g(\mathbf{x}))^2] \\ &= \mathbb{E} [(f(\mathbf{x}) - g(\mathbf{x}))^2] + \mathbb{E} [\varepsilon_{\text{noise}}^2] + 2\mathbb{E}[(f(\mathbf{x}) - g(\mathbf{x}))\varepsilon_{\text{noise}}] \\ &= \mathbb{E} [(f(\mathbf{x}) - g(\mathbf{x}))^2] + \underbrace{\mathbb{E} [\varepsilon_{\text{noise}}^2]}_{=\sigma_{\varepsilon_{\text{noise}}}^2} + 2\mathbb{E}[(f(\mathbf{x}) - g(\mathbf{x}))] \underbrace{\mathbb{E}[\varepsilon_{\text{noise}}]}_{=0} \\ &= \mathbb{E} [(f(\mathbf{x}) - \hat{f}(\mathbf{x}))^2] + \sigma_{\varepsilon_{\text{noise}}}^2. \end{aligned}$$

Then it follows that,

$$\begin{aligned} \mathbb{E} [(f(\mathbf{x}) - g(\mathbf{x}))^2] &= \mathbb{E} [((f(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})]) - (g(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})]))^2] \\ &= \mathbb{E} [(\mathbb{E}[g(\mathbf{x})] - f(\mathbf{x}))^2] + \mathbb{E} [(g(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])^2] \\ &\quad - 2\mathbb{E}[(f(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])(g(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])] \\ &= \underbrace{(\mathbb{E}[g(\mathbf{x})] - f(\mathbf{x}))^2}_{=\mathfrak{B}[g(\mathbf{x})]} + \underbrace{\mathbb{E} [(g(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])^2]}_{=\text{var}(g(\mathbf{x}))} \\ &\quad - 2(f(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])\mathbb{E}[(g(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])] \\ &= \mathfrak{B}[g(\mathbf{x})]^2 + \text{var}(g(\mathbf{x})) \\ &\quad - 2(f(\mathbf{x}) - \mathbb{E}[g(\mathbf{x})])(\mathbb{E}[g(\mathbf{x})] - \mathbb{E}[g(\mathbf{x})]) \\ &= \mathfrak{B}[g(\mathbf{x})]^2 + \text{var}(g(\mathbf{x})). \end{aligned}$$

And finally,

$$\mathbb{E} [(y - g(\mathbf{x}))^2] = \mathbb{E} [\mathfrak{B}(g)^2] + \mathbb{E} [\text{var}(g)] + \sigma_{\varepsilon}^2.$$

□

¹We consider here the regression case where $\mathcal{Y} = \mathbb{R}$.

Remark 4.1

Similar proofs can be found and are detailed for ML methods in Geman, Bienenstock, and Doursat 1992 for example. Such decompositions show that the Mean Squared Error (MSE) of a model can be written as a sum of bias and variance terms. For a given constant MSE, an increase (decrease) in the bias component must be balanced with a decrease (increase) in the variance component. This insight strongly supports the idea of a bias-variance tradeoff in the statistical learning method. In all generality, recent works (see Neal et al. 2019) unveil a growing body of empirical evidence demonstrating that bias-variance trade-off and U-shaped error curves do not always hold true, and these concepts are to be nuanced. The emerging alternative hypothesis is that the test error curve can follow a “double descent” curve with empirical examples of that phenomenon in tree-based models (see Belkin et al. 2019).

Remark 4.2

In a classification context, usual loss functions do not allow for such additive bias-variance decomposition of the general error (see Geurts 2005; Bouckaert 2008). This situation does not prevent the application of the concept of tradeoff but hinders any theoretical insight in such settings.

Various methodologies and techniques exist to avoid overfitting and promote better generalization. First, the risk of overfitting can be avoided beforehand, and careful selection of relevant features can help eliminate irrelevant or noisy features that may introduce overfitting tendencies (see Hawkins 2004). Regularization techniques can then be used to control the complexity of a model and mitigate overfitting. By adding a regularization term to the objective function of the model, the model is encouraged to have simpler and smoother optimal parameter values, thereby reducing its sensitivity to noise and outliers in the training data (see Ying 2019).

In practice, the most commonly used methodology involves dividing the dataset into two separate subsets: a training set used to train the model and a test set used for evaluation purposes. Several methods for that purpose and references are discussed in Section 4.1.2. Evaluating the performance of a model on unseen test data can provide an estimate of how well the model is likely to perform in new instances. This evaluation helps detect overfitting by identifying a significant deterioration in performance on the test set compared with the training set. This general methodology is known as the train–test split, and is an efficient training methodology whenever the test set has a sufficient sample size to yield statistically significant outcomes, and is representative of the overall dataset. It is essential to avoid selecting a test set that possesses characteristics different from those of the training set.

In all generality, if a model performs almost equally well on the test data as it does on the training data, this characteristic indicates that overfitting was effectively avoided. Splitting the training datasets into several subsets is the core idea of optimal tuning strategies. Different training–testing split variations are discussed in detail in the following section.

4.1.2 Tuning and evaluation

Tuning and validation

As discussed in the previous section, any ML model requires a strong tuning and validation strategy. In this section, we briefly introduce the general training–testing split methodology, then detail and illustrate the mechanisms used in the different applications discussed in this thesis. We begin with the aforementioned training–testing split and show how it can be modified to build robust parameter tuning and model evaluation methodologies.

Holdout cross-validation (H-CV): H-CV, also known as simple cross-validation involves splitting the available dataset into two disjoint subsets: a training set \mathcal{D}_{train} and a validation set, or holdout set \mathcal{D}_{test} . An illustration of that process is depicted in Figure 4.2. A common practice is to allocate a high proportion of the data (usually between 60 and 80%) to the training set and the remaining part to the validation set. Splitting is typically performed randomly to ensure representativeness.

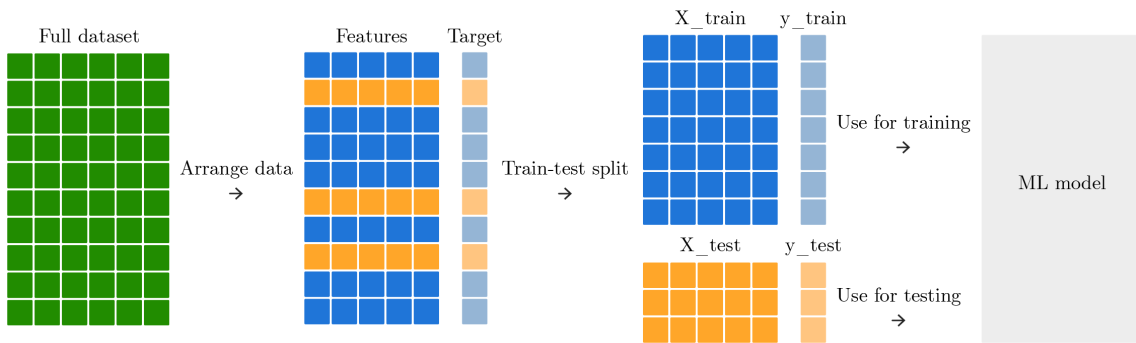


Figure 4.2: Train-test split procedure

First, the model is trained using only the training set. Subsequently, it is evaluated using a holdout set. The model makes predictions based on holdout set instances that are evaluated to assess its performance. These two steps can be performed on different models with various complexities, and the evaluation results obtained from the holdout set serve as estimates that can be compared to provide insight into how well the models generalize to new instances.

This approach offers the advantage of being a simple and straightforward means of estimating the performance of a model without involving complex resampling techniques. As only one split was required, the model was built only once and executed quickly. On the other hand, it has the serious limitation of being very sensitive to the specific instances chosen for the holdout set. Variations exist in this approach to mitigate this issue.

k -fold cross-validation (kf-CV): The kf-CV overcomes some of the limitations of simple H-CV. It provides a more reliable and robust estimate of model performance by leveraging multiple iterations of training and evaluation using different subsets of the dataset. In this approach, the entire dataset is partitioned into k equally sized subsets. In each iteration, one subset is used as the validation set, whereas the remaining $k - 1$ subsets are employed as the training set. The model is trained and evaluated using the training and validation sets, respectively. This process is repeated such that each subset is used as a validation set exactly once. Figure 4.3 illustrates this type of cross-validation.

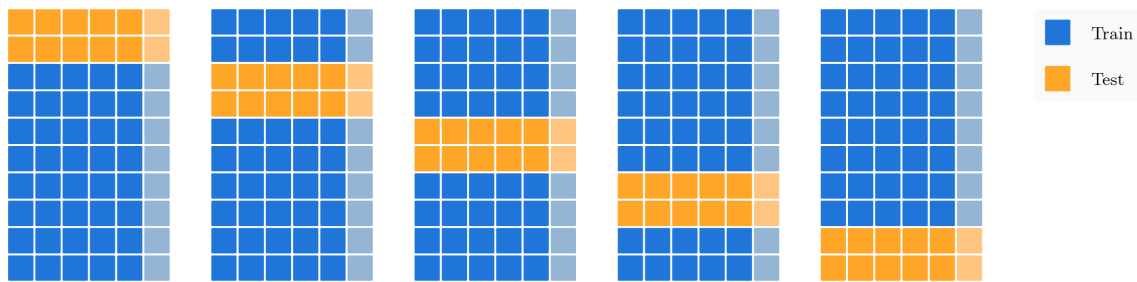


Figure 4.3: k -fold cross-validation

By averaging the results from multiple iterations, kf-CV provides a more reliable estimate of model performance. This reduces the dependency on a single validation set and helps mitigate the impact of variations in the dataset. Furthermore, a significant portion of the data was set aside as a holdout set in simple cross-validation, resulting in reduced training data for model development. In kf-CV, all data points are utilized $k - 1$ times for training and once for validation across multiple folds, maximizing the use of available data and improving the model’s learning capability. Furthermore, this type of cross-validation can be improved to tune and select models over highly imbalanced datasets by ensuring that the mean (or class proportions) of the target variable is constant among all sampled folds. This tuning and evaluation method is the most common one and is used in parts of the applications presented in Chapters 8 and 12.

Leave-p-out cross-validation (LpO-CV): LpO-CV is a variant of cross-validation where a predetermined number of samples, p , are left out as the validation set, and the model is trained on the remaining data. This process is repeated until all the observations are used in both a training and a testing set. In LpO-CV, all possible ways of drawing p samples from \mathcal{D} are considered (see Figure 4.4), and the model is trained and evaluated for each combination to assess the model performance comprehensively. This approach is more rigorous but computationally expensive than traditional k -fold cross-validation because it exhaustively considers all possible combinations of leaving p instances out.

This approach has been used and studied extensively in the actuarial literature, such as by Lin

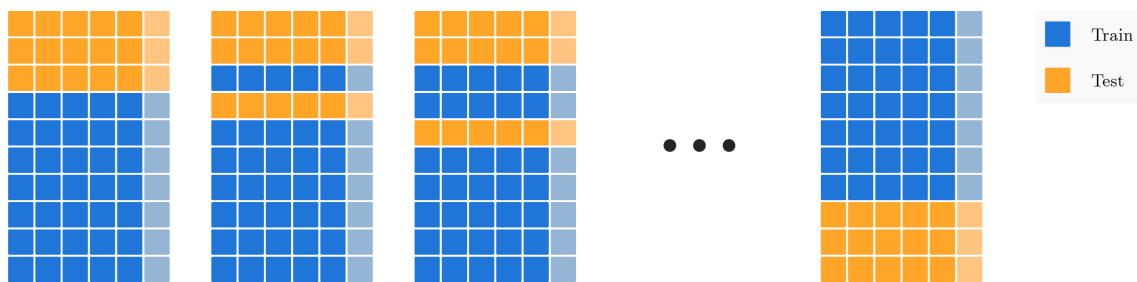


Figure 4.4: Leave-p-out cross-validation

et al. 2018), who discussed the application of LpO-CV to parameter estimation in severity and frequency models loss models, and by Y. Zhang and Siu 2019 who explored the use of LpO-CV for claim count models that involve excess zeros, providing insights into the evaluation of predictive performance in actuarial studies.

Monte-Carlo cross-validation (MC-CV): MC-CV is another evaluation method in which a predetermined number of iterated training–testing splits and evaluations are performed. At iteration m , the dataset is randomly subsampled into training data and test data. Similarly to other

cross-validation techniques, the common practice is to set the proportion of observations going in \mathcal{D}_{train}^m between 60% to 80% while the remaining instances go into \mathcal{D}_{test}^m . The considered model is trained on \mathcal{D}_{train}^m , then evaluated on \mathcal{D}_{test}^m . These steps are repeated at each iteration, as depicted in Figure 4.5.

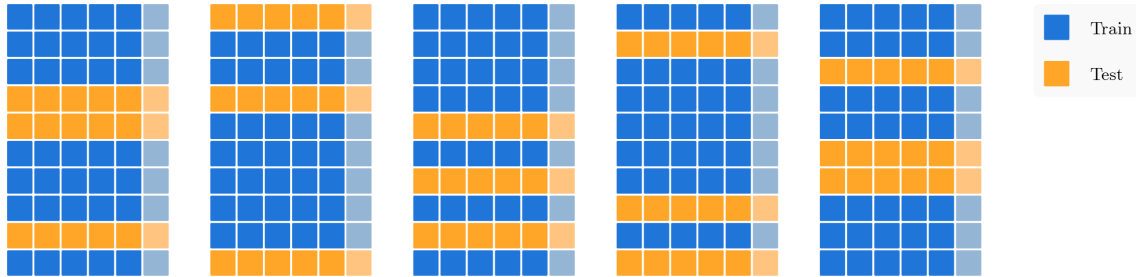


Figure 4.5: Monte-Carlo CV

All samplings are independent; thus, any given observation can be part of several test sets during the procedure or none. Finally, the average of all the test errors is evaluated.

This approach is also used in various actuarial applications. For instance, Bevilacqua, Braglia, and Montanari 2015 employ it for outlier detection in asset valuation, and Nigro and Veltri 2020 utilize it to evaluate risk in dynamic portfolio management modeling. It has also been used in the applications discussed in Chapter 12.

Rolling cross-validation (R-CV): R-CV, also known as rolling window cross-validation, is used for time-series or sequential data. Because the order of data is very important for time-series-related problems, it is not advisable to draw data instances randomly and assign them to \mathcal{D}_{train} or \mathcal{D}_{test} . The dataset is sequentially split into training and validation sets, which address the temporal nature of the data by simulating a real-world scenario in which the model is trained on historical data and tested on future data. A fixed-size window or time period is defined, and the data are divided into multiple overlapping segments or folds. The model is trained on data within the window and then evaluated on subsequent data points that fall outside the window at future time points (see Figure 4.6). The window is then shifted forward by a specified time or number of data points, and the process is repeated until the entire dataset has been used for training and testing.

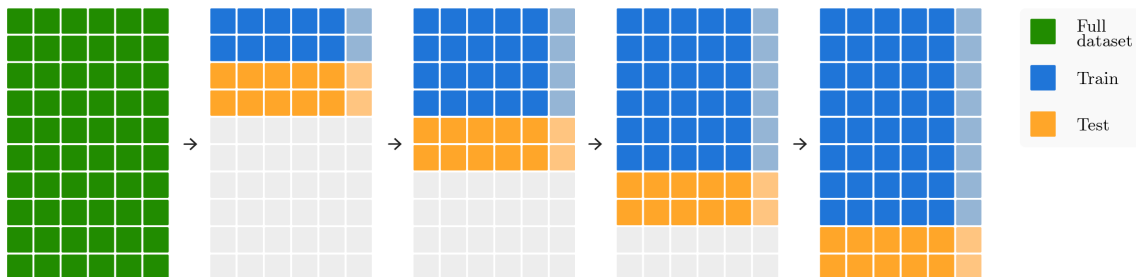


Figure 4.6: Time series rolling CV

R-CV offers a significant advantage in facilitating the evaluation of the performance of a model

on data points that mirror real-world scenarios. This technique effectively captures the temporal dynamics inherent in the data, enabling a reliable assessment of the capacity of the model to generalize and make predictions in a time-dependent manner. By simulating the behavior of the model over time, rolling cross-validation enhances the credibility of its predictive capabilities and contributes to a more robust evaluation of its overall performance. See Chen, Li, and Lin 2020; Liao et al. 2020; Kajikawa and Yamasaki 2019 for actuarial research references that uses R-CV to assess model performance.

These methodologies collectively serve as effective methods to mitigate overfitting. By utilizing these techniques, an optimal tradeoff between model complexity and predictive performance can be achieved, thereby enhancing the capacity of the model to generate accurate predictions using previously unseen data. It is important to note that all cross-validation methods require the selection of an appropriate evaluation metric to assess the performance of a model. This issue is discussed in the next section.

Metrics

The choice of the evaluation metric holds utmost significance as it plays a pivotal role in the comprehensive evaluation of the model’s predictive capabilities. The selection should be made meticulously, considering the specific research questions and desired objectives of the analysis. Various evaluation metrics can be considered depending on the nature of the modeling context (e.g., classification, regression, and survival analysis). A few of these issues are discussed in this section.

Classification: In simple terms, a classification prediction problem in Machine Learning is akin to the task of sorting different items into specific boxes based on their characteristics. For example, you could imagine a system that separates apples from oranges based on their colour and shape, in the context of this thesis, you could imagine finding a way to differentiate lapsers from non-lapsers. To answer a classification task, we create a model that can separate data into different categories based on their individual features. In all generality, with the notations introduced in Section 4.1.1, the supervised learning task is a classification task with J classes if $\mathcal{Y} = \{0, 1, \dots, J\}$.

For the simplicity of the explanations, we will illustrate different choices of ℓ for binary classification ($\mathcal{Y} = \{0, 1\}$), all of which are based on what’s called, the confusion matrix. All choices of ℓ detailed in the following paragraphs can be generalized to multiclass classification. With this in mind, in binary classification, an observation can be well classified or misclassified in only four ways: a 0 can be classified as such or misclassified as a 1, and a 1 can be classified as such or misclassified as a 0. A confusion matrix is essentially a table containing the number of observations $N(j, k)$, $(j, k) \in (0, 1)^2$ in the test set that falls in each of those possibilities:

$$N(j, k) = \sum_{i=1}^N \mathbf{1}(y_i = j, g(\mathbf{x}^i) = k).$$

Table 4.1 is called the confusion matrix and $N(1, 1)$, $N(1, 0)$, $N(0, 1)$, $N(0, 0)$ are respectively referred to as the number of true positives, false negatives, false positives and true negatives.

		Real outcome y		Total
		1	0	
Predicted outcome $g(\mathbf{x})$	1	$N(1, 1)$	$N(0, 1)$	$N(-, 1)$
	0	$N(1, 0)$	$N(0, 0)$	$N(-, 0)$
Total		$N(1, -)$	$N(-, 0)$	N

Table 4.1: Confusion matrix for binary classification

From this number, we can derive several widely used metrics:

$$\begin{aligned} \text{Accuracy}(g(\mathbf{x}), y) &= \frac{N(1, 1) + N(0, 0)}{N}, \\ \text{Precision}(g(\mathbf{x}), y) &= \frac{N(1, 1)}{N(1, 1) + N(0, 1)} = \frac{N(1, 1)}{N(-, 1)}, \\ \text{Recall}(g(\mathbf{x}), y) &= \frac{N(1, 1)}{N(1, 1) + N(1, 0)} = \frac{N(1, 1)}{N(1, -)}. \end{aligned}$$

$\text{Accuracy}(g(\mathbf{x}), y)$ is undoubtedly the most intuitive performance measure, and it is defined as the proportion of correctly predicted observations among all observations. It is widely used for binary classification and churn analysis; however, it appears to be a satisfactory performance measure only for balanced datasets.

$\text{Precision}(g(\mathbf{x}), y)$ measures the proportion of positive observations among the observations predicted as follows: the higher this metric, the lower the false positive rate.

$\text{Recall}(g(\mathbf{x}), y)$ measures the proportion of observations that are correctly predicted as 1 among all the positive observations. The higher this metric, the lower the false negative rate, which is of great interest for churn predictions.

More complex metrics can be derived from these basic metrics. For example, one can easily imagine a problem in which any misclassification comes at a cost depending on its nature. Both the false positive and false negative rates must be minimized and balanced according to their respective costs. In that case, it can be relevant to consider the F_β Score, based on the work of Rijsbergen 1979, and defined as

$$F_\beta\text{Score}(g(\mathbf{x}), y) = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}.$$

In this case, $F_\beta\text{Score}(g(\mathbf{x}), y)$ can be seen as the weighted harmonic mean of precision and recall, taking values between 1 (a perfect model) and 0 (the worst possible model). Parameter β represents the ratio of recall importance to precision importance. Very intuitively, values of $\beta < 1$ give more weight to precision, while values of $\beta > 1$ favor recall. For example, setting $\beta = 2$ makes recall twice as important as precision. And it follows that $\lim_{\beta \rightarrow 0} F_\beta\text{Score} = \text{Precision}$ and $\lim_{\beta \rightarrow +\infty} F_\beta\text{Score} = \text{Recall}$. It is usually more efficient than accuracy in presence of uneven class distribution. Accuracy seems to perform better if false positives and false negatives have similar costs. Other metrics such as the Jaccard index (see Jaccard 1912) also exhibit the property of weighting the misclassification costs but will not be used nor discussed throughout this thesis.

Also based on those basic metrics, the Precision-Recall (PR) and the Receiver Operating Characteristic (ROC) Curves (depicted in Figure 4.7) are widely used metrics for evaluating the performance of a classification model in machine learning. On the one hand, the ROC curve is

a graphical representation of how well a classification model can distinguish between classes. The curve is plotted using the true-positive rate (TPR) against the false-positive rate (FPR) at various threshold settings. The Area Under the ROC Curve (AUROC) measures the entire two-dimensional area underneath the entire ROC curve, and it provides an aggregate measure of performance across all possible classification thresholds. A model whose predictions are 100% correct has an AUC of 1 while a model whose predictions are 100% wrong has an AUC of 0.

On the other hand, the PR curve is a plot of precision (also known as the positive predictive value) against recall (also known as sensitivity) for different thresholds. A PR curve is more informative for unbalanced datasets than an ROC curve. The AUC-PR measures the area under the PR curve and provides a measure of the model performance.

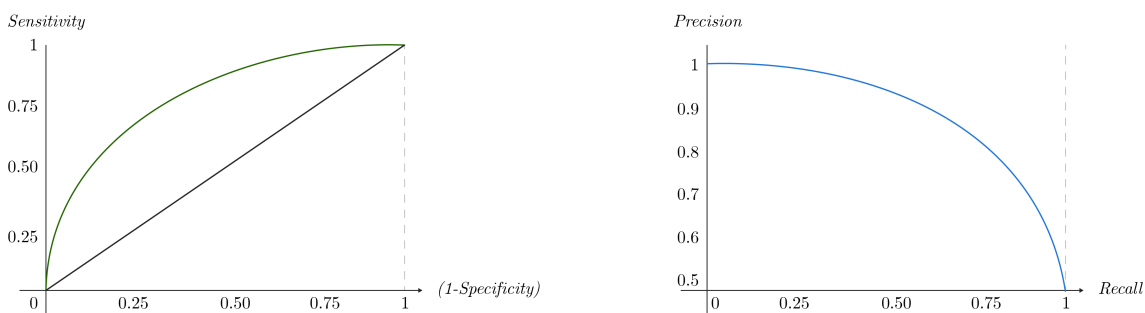


Figure 4.7: Illustration of the Areas under the ROC and PR curves

Eventually, we will introduce a last choice of ℓ for a loss function: the negative log loss function or cross-entropy, defined as

$$\text{Log-Loss} = -\frac{1}{N} \sum_{i=0}^N \left[y^{(i)} \log \left(g(\mathbf{x}^{(i)}) \right) + \left(1 - y^{(i)} \right) \log \left(1 - g(\mathbf{x}^{(i)}) \right) \right].$$

This quantity is not defined as a combination of measures based on the confusion matrix, but rather is derived from maximum likelihood optimization. It shows various desirable properties. First, it directly optimizes the predicted probabilities to match the actual class labels. This is beneficial in scenarios in which we care not only about the final class prediction, but also about how confident the model is in its prediction. Second, the logarithmic components of this function ensure that incorrect predictions are heavily penalized; the penalty grows exponentially, and the prediction becomes more confident. This encourages the model to make accurate predictions. The use of logarithms ensures prediction stability and prevents numerical underflow, which can occur when dealing with extremely small probabilities. Therefore, it is recommended for churn prediction problems (see Henckaerts and Antonio 2022, in an insurance context). Eventually, this function is smooth and differentiable everywhere, which is desirable as it allows the use of gradient-based optimization algorithms like eXtrem Gradient Boosting (XGBoost) (see Section 4.2.2).

Regression: Various regression loss functions exist (see Wang et al. 2022 for a comprehensive survey), among these, the loss functions below, to name a few, are prominent:

- The Mean Absolute error (MAE) :

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y^{(i)} - g(\mathbf{x}^{(i)})|.$$

- The Mean Absolute Percentage Error (MAPE):

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^N \frac{|y^{(i)} - g(\mathbf{x}^{(i)})|}{y^{(i)}} \cdot 100.$$

- The Mean Squared Error (MSE):

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N \left(y^{(i)} - g(\mathbf{x}^{(i)}) \right)^2.$$

- The Huber Loss:

$$L_{\delta}(y, g(\mathbf{x})) = \begin{cases} \frac{1}{2}(y - g(\mathbf{x}))^2 & \text{for } |y - g(\mathbf{x})| \leq \delta \\ \delta|y - g(\mathbf{x})| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases}.$$

- The Log Cosh Loss:

$$\log \cosh(t) = \sum_{i=1}^N \log \left(\cosh \left(g(\mathbf{x}^{(i)}) - y^{(i)} \right) \right).$$

- The Poisson Deviance:

$$D = 2 \sum_{i=1}^N \left[y^{(i)} \log \left(\frac{y^{(i)}}{g(\mathbf{x}^{(i)})} \right) - \left(y^{(i)} - g(\mathbf{x}^{(i)}) \right) \right].$$

- The Quantile Loss:

$$L_{\text{Quantile}} = \sum_{i|y^{(i)} < g(\mathbf{x}^{(i)})} (\gamma - 1) |y^{(i)} - g(\mathbf{x}^{(i)})| + \sum_{i|y^{(i)} \geq g(\mathbf{x}^{(i)})} (\gamma) |y^{(i)} - g(\mathbf{x}^{(i)})|.$$

Each of these functions has its domain of superiority; for instance, the MSE is sensitive to outliers but easy to compute and differentiate, whereas the absolute error is robust against outliers but lacks computational efficiency. Log Cosh Loss and Huber Loss are combinations, offering a balance between robustness against outliers and computational ease. Quantile Loss, on the other hand, is excellent for predictions involving quantiles. In this thesis, we primarily focus on the MSE when dealing with regression problems, owing to its simplicity and consistency in providing comparable results for our applications. However, let us not overlook the fact that the choice of loss function is highly dependent on the specific task, dataset, and requirements at hand (see Henckaerts, Côté, et al. 2021).

4.2 Models description

The subsequent literature review presents a comprehensive examination of tree-based ML models, which have garnered significant attention in data science owing to their interpretability, versatility, and robust performance across diverse datasets. The following descriptions delve into the underlying mechanics of the fundamental tree-based algorithms and elucidate the operational principles that drive their functionality. First, we retrace the historical steps that have led to the emergence of diverse tree-based models. This review elaborates the strengths and weaknesses of these algorithms, thereby providing a balanced evaluation of their efficacy. In addition, the complexities, variations, and unique characteristics of these models are expounded, offering a nuanced understanding of their potential applications. The objective of this review is to provide a historical overview of tree-based methods, underscoring their pivotal role in advancing ML and predictive analytics and their advantages in management science.

4.2.1 History of decision trees

Static tree structures

Classification and regression trees, commonly referred to as decision trees, are increasingly utilized in both predictive and exploratory roles. They are particularly valuable for identifying and managing non-linear impacts on the targeted variables and for uncovering complex interactions among predictors. This section's objective is to delve into the inner workings of the Classification and Regression tree algorithm (CART), as showcased in L. Breiman et al. 1984's work, to explore the evolution of tree methods that led to its development as well as the evolutions that ensued. This section should enable readers to understand the common concepts behind their functioning. For complete and comprehensive reviews, consider the early work of Fielding and O'Muircheartaigh 1977 or the more recent overviews of Ritschard 2013 and Loh 2014. We will begin with the general concepts and historical review.

In ML and predictive analytics, a decision tree is a powerful iterative model, that makes no assumptions about the underlying distribution of (x, y) . It operates based on the principles of recursive partitioning and is characterized by a tree-like structure. The model divides the feature space into a series of binary splits, creating a hierarchical structure of nodes representing decision points. Each split is based on a specific feature and threshold, enabling clear decision rules to be extracted from the tree. Every tree-based model mentioned throughout this thesis is built using decision trees; hence, the term TBM. Belson 1959 first proposed the concept of recursive partitioning. His focus was on the problem of predicting the value of a target variable for homogenous subgroups. He dichotomized the predictors and the outcome variable based on a growth criterion defined as the difference between the observed count and the expected number under the no-association assumption. Considering only binary predictors (x_1, \dots, x_p) and outcome y , at a given node g the best variable to split on, x^* is selected as

$$x^* = \operatorname{argmax}_{x \in (x_1, \dots, x_p)} \left\{ \frac{1}{\mathcal{N}(g)} \cdot \sum_{i \in g} \mathbb{1}_{x^i=1} \cdot \sum_{i \in g} \mathbb{1}_{y^i=0} - \sum_{i \in g} \mathbb{1}_{y^i=0, x^i=1} \right\},$$

with $\mathcal{N}(g)$ the number of observation in node g .

Other early ideas came from Morgan and Sonquist 1963 who introduced the Automatic Interaction Detector (AID) algorithm to build a binary regression tree, and Cellard, Labbe, and Savitsky 1967 who proposed *Exploration of Links and Interactions through Segmentation of an Experimental Ensemble* (ELISEE), a binary method for categorical dependent variables.

These early models enable to generate a graph called *decision tree*, as an output. Such a tree illustrates decisions and their possible consequences, providing a graphical view of the decision-making process (see Figure 4.8).

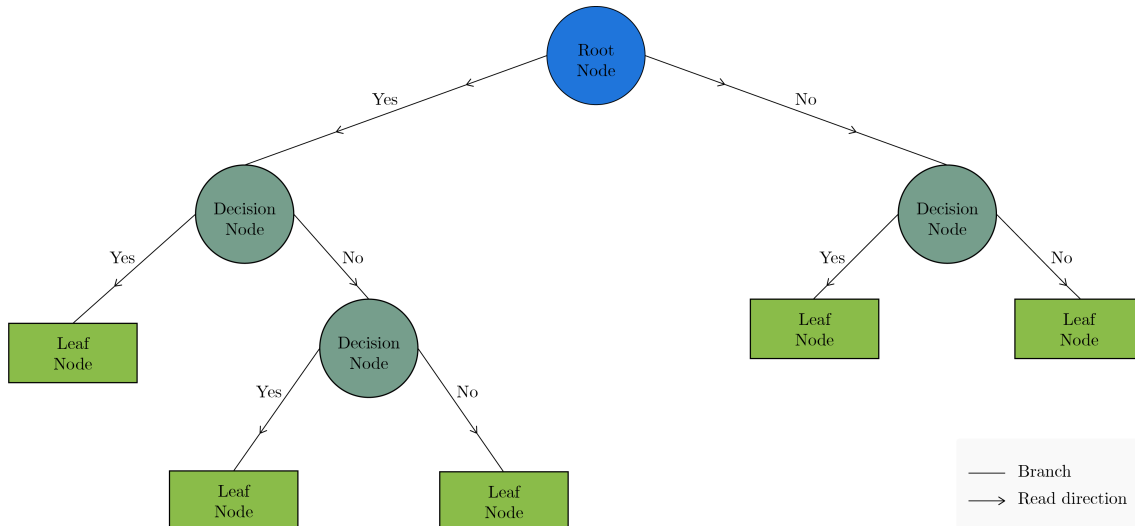


Figure 4.8: A simple decision tree

The nodes in this structure represent the attributes or features, while the branches correspond to the decision rules, leading to the leaf nodes that signify the outcome.

The AID method was popularized by Sonquist et al. 1971, with Sonquist 1969 demonstrating its usefulness alongside multiple correlation analysis², while Bouroche, J.M. and Tenenhaus, M. 1970 popularized ELISEE. Further extensions of AID then emerged, such as *Interactive Data Exploration and Analysis* (IDEA), a tree-growing algorithm that allows splits with more than two child nodes by Press, Rogers, and Shure 1969. Extensions to handle categorical outcomes with THAID were developed by Messenger and Mandell 1972 and Morgan and Messenger 1973, or to handle multivariate quantitative outcomes with MAID (Gillo 1972; Gillo and Shelly 1974). In parallel and coincidentally, Hunt, Marin, and Stone 1966 proposed a series of classification tree induction algorithms called Concept Learning Systems.

The primary goal of these initial methods is to understand the relationship and interactions between the target variable and factor covariates. Apart from Hunt, most authors (see Morgan and Sonquist 1963, Press, Rogers, and Shure 1969) seek alternatives to the limitations of the linear model, in which the impact of the explanatory variables is essentially additive. The main focus was to identify significant interactions, not necessarily to enhance prediction, but to deepen our understanding of how the target variable relates to covariates. Thus, these early methods aimed to divide data into groups that showed distinct distributions of the target variable. Therefore, these methods naturally use measures of correlation between the outcome and split variables as their splitting criteria. If the outcome is quantitative, as in AID (or MAID), the split that leads to the largest reduction in the residual sum of squares (or its multivariate generalization) is chosen. It is equivalent to the maximization of a modified version of R^2 , the proportion of explained vari-

²For a detailed comparison of AID and Belson’s method, refer to Thompson 2018.

ation. As an illustration, AID’s splitting criterion (as defined in the original work from Morgan and Sonquist 1963) finds the split that divides the node g into two child nodes g_r and g_l as the one that maximizes

$$\begin{aligned} \mathcal{N}(g_r)\bar{y}_r^2 + \mathcal{N}(g_l)\bar{y}_l^2 &= \mathcal{N}(g_r) \left(\frac{\sum y_r}{\mathcal{N}(g_r)} \right)^2 + (\mathcal{N}(g) - \mathcal{N}(g_r)) \left(\frac{\sum y_l}{\mathcal{N}(g) - \mathcal{N}(g_r)} \right)^2 \\ &= \frac{(\sum y_r)^2}{\mathcal{N}(g_r)} + \frac{(\sum y - \sum y_r)^2}{\mathcal{N}(g) - \mathcal{N}(g_r)}, \end{aligned}$$

with y_r and y_l the values of y in nodes g_r and g_l , and \bar{y}_r and \bar{y}_l their respective means.

A variance component can be introduced within that splitting criterion in order to estimate the explained sum of squares among the total population not merely on the sample provided.

When the target variable is quantitative, the IDEA algorithm (see Press, Rogers, and Shure 1969) is also based on a scaled R^2 . The notion of statistical significance was later introduced in these splitting procedures, by evaluating the p-value of the modified R^2 , first with a permutation test (see Kass 1975 and Appendix A.0.1), and then with its Chi-square approximation (Scott and Knott 1976). Whenever the target variable is categorical, all tree-based algorithms before 1976 rely on a splitting criterion, which is the p-value of Pearson’s chi-square independence test (see Appendix A.0.2). Most applications of these techniques appear in social science articles; however, we can also find early TBM works in management science and marketing (Cellard, Labbe, and Savitsky 1967; Armstrong and Andress 1970; Assael 1970). Relying on these strong foundations, *Chi-square Automatic Interaction Detector* (CHAID) was introduced by Kass 1980 as an extension of the AID and THAID. It can generate trees with nodes that can split into more than two child nodes and uses Bonferroni tests to identify the best splits. This approach is currently the most popular of these earlier statistical supervised tree-growing algorithms and is widely available; in R, the CHAID package can be used. Although it is rare to see any recent applications of AID, THAID, ELISEE, or IDEA in the actuarial or management science literature, CHAID remains competitive among tree algorithms. For example, it has been used for management and marketing applications in insurance by Onn and Mercer 1998, or for churn prediction in various applications in Almana 2014.

This historical review now comes to its tipping point as subsequent tree-growing techniques such as CART, GUIDE, Iterative Dichotomizer 3 (ID3), C4.5 and 5.0, M5, or conditional inference trees (CTREE), focus on classification and prediction rather than interaction analysis. Thus, they modified the existing splitting goal that consisted of minimizing some between-nodes heterogeneity in order to design models that maximize the within-node homogeneity. It changed the paradigm from minimizing variance measures to maximizing purity measures. The overview of subsequent tree-based algorithms can be presented both chronologically and by contributors, as Breiman, Quinlan, Loh then Hothorn have succeeded one another for the last 40 years.

Breiman’s trees:

Specifically, Breiman’s work, with CART, was pivotal in regenerating interest in the subject as it also brought new ideas such as pruning or surrogate splits. CART uses a similar top-down greedy approach as AID and THAID: it starts from an initial node - the root - containing all

observations in \mathcal{D} . Then it finds the covariate x_k and the threshold d^3 such that they optimize a splitting criterion. The root is then split into those two child nodes for which the same splitting process is repeated until a stopping criterion is triggered. Multiple ways of splitting a parent node g_p into the child nodes g_r and g_l exist. The classification splitting criteria to be maximized, first considered in the original work of Breiman are

1. Based on an impurity measure $I(\cdot)$:

$$I(g) + \left(\frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} I(g_l) + \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)} I(g_r) \right),$$

where Breiman considered the Gini impurity measure defined as $I(g) = -\sum_i p_{i,g} \log(p_{i,g})$, and the entropy defined as $I(g) = \frac{1}{2} \sum_i p_{i,g} (1 - p_{i,g})$,

2. the Twoing criterion:

$$\frac{\mathcal{N}(g_l)\mathcal{N}(g_r)}{4 \cdot \mathcal{N}^2} \left[\sum_i |p_{i,g_l} - p_{i,g_r}| \right]^2,$$

with $p_{i,g}$ the proportion of observation with class i in node g .

Some differences between splits obtained through these criteria are discussed in Section 4.2.3.

The regression splitting criterion originally considered for CART is

$$MSE(g) + \left(\frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} MSE(g_l) + \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)} MSE(g_r) \right),$$

with $MSE(g)$ the MSE of all observations (\mathbf{x}, y) contained in g . It is also to be maximized.

Whether it is used in classification or regression contexts, CART is capable of processing both categorical and numerical data. Once growing the tree is achieved, the *maximal tree* is obtained. It has been observed with AID for instance, that such a tree overfits the data, leading to predictions made on observations that were not used to grow the tree that are usually inaccurate. That is why a last refining step is required: the maximal tree is pruned to a subtree that has better generalization abilities. From an algorithmic perspective, growing a CART following the original Breiman's procedure is summarized in Algorithm 1⁴.

A tree is therefore defined by its splitting criterion (`SplittingCriterion`), its stopping rule(s) (`StoppingRules`) and its pruning process (`Prune`). No or weak stopping rules will generate a high-variance/low-bias over-fitted tree whereas constraining ones will lead to smaller, more interpretable low-variance/high-bias under-fitted trees. The idea of cost-complexity pruning developed by Breiman emerged from the need to find a compromise between the two extremes.

The main idea behind cost-complexity pruning is to consider sub-trees of the maximal tree and evaluate them with a cost function that increases as the error rate rises and decreases as the number of leaves drops. When a tree is pruned at a node, the weighted summed error of the leaves increases while the number of leaves reduces, thus a pruned sub-tree is selected only if the error gain is counter-balanced by the complexity loss.

³ d is a cutoff value if the covariate to split on is numerical and a set of classes if it is categorical.

⁴Please note that the specific implementations of all algorithms described in this section can vary depending upon the programming language and the specific variant of the algorithm one might use.

Algorithm 1 Grow algorithm for CART

```
1: Input: Training set  $\mathcal{D}$ , current node  $g$ 
2: Consider the current node  $g$ , if no current node exists, create a new tree  $\mathcal{T}$  with a single
   initial node  $g$ .
3:
4: if StoppingRules( $g$ ) = True then let  $g$  be a leaf with the prediction  $f_{\mathcal{T}}(g)$ .
5: else
6:   for all possible covariates and thresholds do
7:     Find the pair  $(x_k, d)$  that obtains the best SplittingCriterion( $\mathcal{D}, x_k, d$ ).
8:     Split the node  $g$  along covariate  $x_k$  at threshold  $d$  into two child nodes  $g_r$  and  $g_l$ .
9:     Grow( $\mathcal{D}(g_r), g_r$ ).
10:    Grow( $\mathcal{D}(g_l), g_l$ ).
11:   end for
12: end if
13: Output: Prune( $\mathcal{T}$ )
```

The cost of a tree \mathcal{T} is given by

$$C_{\alpha}(\mathcal{T}) = R(\mathcal{T}) + \alpha n_L(\mathcal{T}), \quad \alpha \geq 0, \quad (4.2)$$

with $R(\mathcal{T})$ is the sum of all error of the leaves of \mathcal{T} , weighted by the number of individuals they represent. The number of leaves of \mathcal{T} is denoted $n_L(\mathcal{T})$, and the penalty α is the complexity parameter: the higher it is, the smaller the pruned tree. The interest of α is that for a fixed complexity parameter value, there exists a unique smallest subtree \mathcal{T} of the maximal tree \mathcal{T}_{max} that minimizes $C_{\alpha}(\mathcal{T})$. Thus by decreasing α , we can construct a sequence of pruned optimal subtrees $[\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{max}]$ of different sizes. This tree sequence is such that \mathcal{T}_1 is the root node, \mathcal{T}_2 a sub-tree of \mathcal{T} with more leaves and accuracy than \mathcal{T}_1 and so on until \mathcal{T}_{max} , the unpruned maximal tree. With Breiman's notation, we have

$$\mathcal{T}_{max} \supseteq \dots \supseteq \mathcal{T}_2 \supseteq \mathcal{T}_1.$$

The optimal complexity parameter value, hence the best tree in the sequence is selected using cross-validation: this method effectively addresses the issues of underfitting and overfitting that are prevalent in AID and THAID, although it does require more computational power.

Another significant contribution of Breiman's 1984 work is the suggestion of "surrogate" splits to handle missing values in \mathcal{D} . A surrogate split is an alternative to the main split when the latter cannot be applied due to missing values. This is a split that replicates as best as possible, the binary partitioning at a given node with another covariate. Moreover, surrogate splits also serve to derive an importance score for each feature.

The `sklearn` library in Python includes an implementation of CART⁵, while in R, the `rpart` package (Therneau, Atkinson, and Ripley 1999) is available for use.

For more details and visualizations regarding the CART procedure, we refer the astute reader to Section 14.1.

⁵See functions `DecisionTreeClassifier` and `DecisionTreeRegressor`.

Quinlan's trees:

Two years after the work of L. Breiman et al. 1984 was published, Quinlan 1986 proposed another tree-based algorithm. ID3 uses entropy and information gain for splitting; it can handle categorical data but not numerical data and missing values. Quinlan 1993 subsequently introduced C4.5, an improvement on ID3. It employs an entropy-based measure of node impurity called the gain ratio for splitting and can handle both categorical and continuous data. When considering a split s that divides node g into g_r and g_l along covariate \mathbf{x} , the gain ratio is computed as such

$$\text{Gain Ratio}(g, \mathbf{x}) = \frac{\text{Entropy Gain}(g, \mathbf{x})}{\text{Split information}(g, \mathbf{x})} = \frac{E(g) - \mathcal{N}(g_r) \cdot E(g_r) - \mathcal{N}(g_l) \cdot E(g_l)}{-\sum_i p_i \cdot \log_2 p_i},$$

with $E(g)$, the entropy of node g .

It also handles missing data and pruning. If, at a given node, an observation is missing the value of a split variable, it is sent to every child node with weights proportional to the number of non-missing observations in those nodes. Various applications have demonstrated that C4.5 achieves excellent prediction performance with low computation time and produces trees that are often substantially larger than those of other methods (Lim, Loh, and Shih 2000, Loh 2009). The C4.5 package, which is compatible with the `scikit-learn` library, can be used in Python, and the RWeka package can be used in R. A later extension of C4.5, called C5.0, exists; however, almost no scientific literature is available on this topic.

Quinlan 1992 also introduced the M5 tree algorithm, which is a decision-tree learner for regression tasks with linear regression functions at terminal nodes that can predict continuous numerical attributes. The result of M5 is a decision tree with a linear regression that is fit at every leaf, as depicted in Figure 4.9.

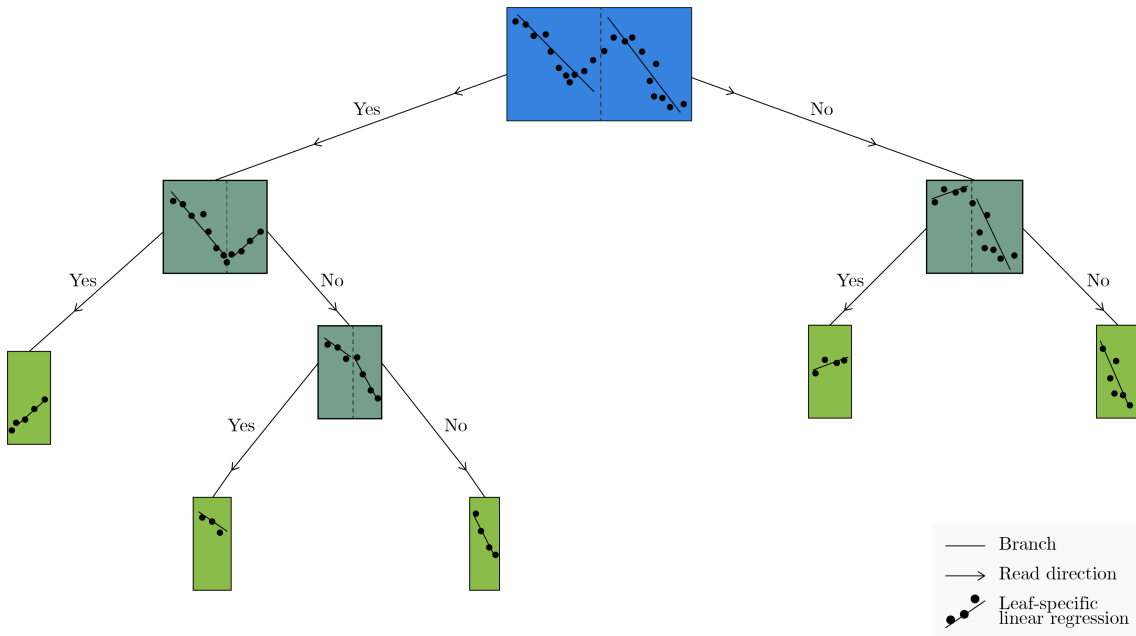


Figure 4.9: A M5 tree

The Rweka package in R and sklearn in Python can be used for M5.

Loh's trees:

The main drawback of the greedy search approach of all previous trees is the inclination to favor certain variables in the selection step, inducing an inherent bias. This flaw was initially pointed out by Breiman while discussing the CART algorithm. Later, White and Liu 1994 and Kononenko 1995 highlighted the extent of this bias in the C4.5 algorithm. Loh greatly contributed to the tree-based model literature by introducing a series of algorithms with an unbiased selection of splitting variables and cut points with statistical tests, that can handle missing values and detect interactions.

The *Fast Algorithm for Classification Tree* (FACT) offers unbiased variable selection when all covariates are ordered since it applies F-tests and linear discriminant analysis for feature selection. Nevertheless, it does show a bias towards categorical variables. The *Quick, Unbiased and Efficient Statistical Tree* (QUEST) proposed by Loh and Shih 1997 overcomes this bias. Notably, QUEST outperforms CART in computational efficiency, especially when dealing with categorical variables with numerous values.

CRUISE, an extension of QUEST, also allows linear splits using all variables and can fit linear discriminant models within each terminal node (see Kim and Loh 2003). Then Loh proposed a logistic regression tree (LOTUS) (see Chan and Loh 2004) for fitting models to data with a binary target variable, which also has a negligible bias in variable selection.

In addition to the selection bias, Loh 2001 found that CART tends to favor variables with more missing values when making splits, and prefers surrogate variables with fewer missing values. However, algorithms like CRUISE and QUEST do not show this bias. Building on the strengths of these algorithms, along with *Smoothed and Unsmoothed Piecewise-Polynomial Regression Trees* (SUPPORT) (see Chaudhuri et al. 1994), Loh introduced *Generalized, Unbiased, Interaction Detection and Estimation* (GUIDE) in 2009 (see Loh 2002). GUIDE improves these algorithms by rectifying their drawbacks. GUIDE constructs piecewise-constant, multiple linear, and simple polynomial tree models for least-squares, quantile, Poisson, and proportional hazards regression.

CART's pruning approach is used in all Loh's models. No R or Python implementations of those algorithms exist, but the original codes for CRUISE, GUIDE, and QUEST are freely available from [their author's webpage](#).

Hothorn's trees:

More recently, Hothorn, Hornik, and Zeileis 2006 proposed the implementation of another tree-growing procedure with unbiased variable selection, called conditional inference trees (CTREE). As in the algorithm of Loh, the selection of splitting variables at each node relies on statistical tests. By default, it uses p-values from a quadratic correlation test statistic and the resulting p-values are Bonferroni-corrected (see Appendix A.0.3) for multiple testing across the number of regressor covariates. A notable difference from previous procedures is that CTREE does not use pruning, but rather stops based on the Bonferroni-corrected splitting p-values to determine an optimal tree size. The `party` package in R was used to implement this algorithm.

Zeileis, Hothorn, and Hornik 2008 later suggested a new global framework for growing trees: model-based recursive partitioning (MOB). The idea of this framework is to fit a statistical model (linear regression, logistic regression, ridge or lasso regression, or generalized linear models

(GLM), for instance) at every node in the tree and find the split that will result in two child nodes with populations that are as different as possible in the model. Once again, statistical tests were performed at each node to select the optimal covariate to split, and then a threshold optimizing an objective function was selected for the chosen split covariate. Specifically, in MOB, a fluctuation test (see Appendix A.0.4) is performed to assess which covariate can lead to the highest parameter instability, and the split covariate is chosen based on its corresponding Bonferroni-corrected p-values.

The tree stops growing whenever there is no significant parameter instability and is eventually pruned based on the Akaike or the bayesian information criterion (AIC or BIC). Here, the idea is to partition the covariate space so as not to identify groups of individuals with similar values of the target variable but rather to identify groups of individuals with similar behaviors or association patterns. MobTree extends the scope of decision trees by facilitating the incorporation of various types of statistical models, thereby broadening their application spectrum and enhancing the extraction of meaningful insights from diverse and complex datasets. A clear representation of MOB with an illustration with GLM trees can be found in Dutang and Guibert 2021.

Loh and Hothorn introduced different unbiased tree algorithms that rely on the same strategy of selecting the variable to be split through statistical testing. The key difference is in the choice of the test that is used (association tests for CRUISE, QUEST, and GUIDE; conditional inference for CTREE; and parameter instability tests for MOB), and their relative advantages or disadvantages are, to the best of our knowledge, not well understood or studied (see Schlosser, Hothorn, and Zeileis 2019).

Dynamic tree structures

One of the aims of this thesis is to explore existing and innovative tree-based methods for dynamical data. Of course, when we think of time-dependent data, we can think of time-to-event data, and thus survival analysis. There exist TBMs specifically designed for such analysis, and as will prove themselves critical within our applications, a brief review of survival analysis and trees can be found in Chapter 5. We can also think of trees that can evolve or be updated with time as new independent observations are collected. Mondrian trees Lakshminarayanan 2016 or Dynamic regression trees (see Taddy, Gramacy, and Polson 2011) are examples of such time-evolving trees. It is to be noted that, as time goes by, these models do not allow the introduction of new observations from subjects already observed at previous times. The trees are dynamic, not the covariates and response variable, hence we will not detail here the mechanisms of such trees but refer to the publications mentioned in this paragraph.

4.2.2 Ensemble models

An ensemble tree-based model grows a large number of single “weak” tree-based models and aggregates them to enhance their predictive accuracy and decrease their individual variabilities. Each tree in the ensemble is built following the principle of recursive partitioning described in Section 4.2. Moving forward, this section delves into bagging and boosting—two quintessential ensemble techniques in ML—celebrating their abilities to improve model stability and accuracy, but not without drawbacks such as model complexity and potential overfitting to the training dataset. For an overview of tree-based ensemble techniques, we refer readers to Berk 2006; Strobl, Malley, and Tutz 2009.

Random Forest (RF) and Bagging

The RF algorithm, a form of an ensemble learning method, is a key tool in ML. It leverages the concept of “bagging”, or bootstrap aggregating, to generate multiple decision trees from randomly selected subsets of training data. Each tree, constructed in an unpruned manner, casts a unit prediction. The latter are then combined to reach a final prediction, as depicted in Figure 4.10.

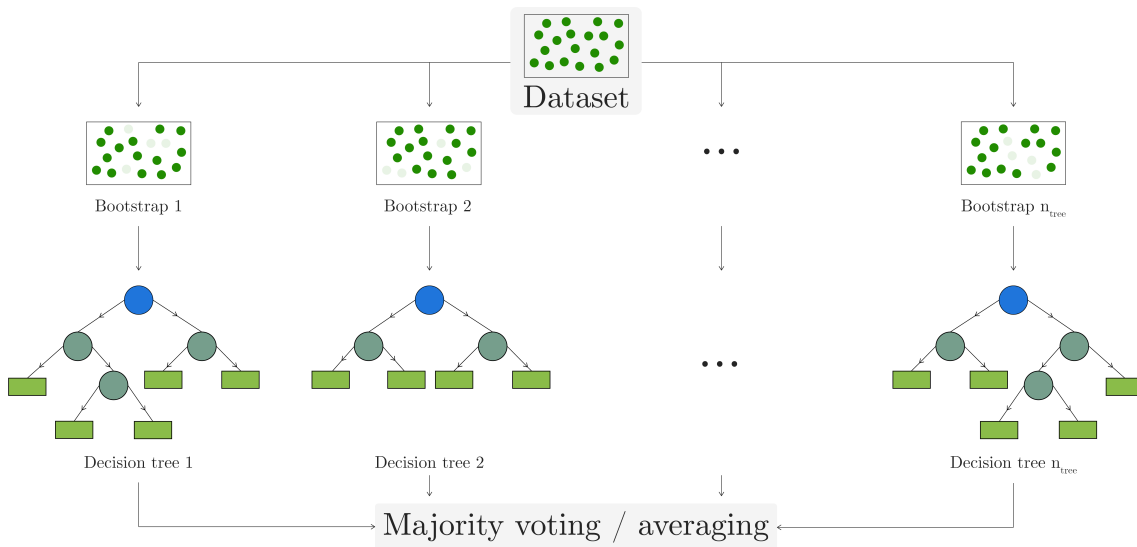


Figure 4.10: Mechanisms of a random forest

Algorithm 2 Random Forest Algorithm

- 1: **Input:** Training set \mathcal{D} , number of trees T , number of features F
 - 2: **for** $t = 1$ to T **do**
 - 3: Create a bootstrap sample \mathcal{D}_t of size $|\mathcal{D}|$ with replacement
 - 4: Build a decision tree \mathcal{T}_t on \mathcal{D}_t as follows:
 - 5: **for** each node of the tree **do**
 - 6: Randomly select F features without replacement
 - 7: Split the node using the feature that provides the best split according to the objective function, among the F features
 - 8: **end for**
 - 9: Add the tree \mathcal{T}_t to the set of trees \mathcal{F}
 - 10: **end for**
 - 11: **Output:** \mathcal{F}
-

The pseudo-code of Algorithm 2 follows the standard random forest algorithm introduced in L. Breiman 2001, where each tree is trained on a different bootstrap sample of the dataset, and the best split at each node is chosen among a subset of the features selected at random. Some variations of the generic algorithm exist and offer more hyperparameters or include more randomness in the growing process (for example, extremely randomized trees, also known as extra trees). In any case, the final model is an ensemble of all these individual trees. The prediction for a new sample is then made by aggregating the predictions of all the trees in the ensemble, typically by majority voting for classification or averaging for regression. The success of the algorithm

stems from its ability to diminish overfitting via reduction of variance, without increasing error due to bias. It achieves this objective by creating uncorrelated trees to maximize ensemble diversity, thereby producing a robust model that performs well on unseen data. Its inherent ability to handle large datasets with high dimensionality and missing values and its feature importance estimation ability make it a versatile tool for a range of prediction tasks.

Remark 4.3

Bagged-ensembles of any tree described in Section 4.2 can be built and implementations of such forests exist. We refer the astute reader to the works of L. Breiman 2001; Lee et al. 2018; Hothorn, Hornik, Strobl, et al. 2010; Garge, Bobashev, and Eggleston 2013 that discuss the implementations of forests of CART, C4.5, CTREE and MOB trees respectively.

XGBoost (XGB) and Boosting

Following bagging methods, other ensemble approaches have been proposed to reduce the sensitivity of individual trees. Boosting is an adaptive approach, formalized by Freund and Schapire 1996 and created with this issue in mind. This approach is also a tree aggregation method that, unlike random forests, does not aggregate models constructed in parallel and randomly on bootstrapped data copies, but rather aggregates models constructed iteratively, one after the other. Although this type of algorithm was initially designed to solve binary classification problems, it has now been adapted to a wider range of problems. This section describes the principles of boosting as they currently exist and provides the specificities of XGBoost, a tree-based boosting implementation. Tree boosting aims to reduce variance and bias in a single-tree model. To achieve this objective, boosting is based on the same idea as bagging: the construction of a large number of model trees, which are then aggregated by a weighted average of their forecasts. However, the tree-building stage is very different for boosting because it is conducted using an iterative procedure: a first tree is created, then a second that gives greater weight to observations poorly predicted by the first, and so on. In other words, each new tree of a boosting ensemble focuses its predictive efforts on the parts of the sample that were the most difficult for previous learners to predict. The weights are computed based on a gradient descent algorithm and depend on the hyperparameters of the boosting model. All learners built iteratively are eventually aggregated, generally by a weighted average, according to their goodness of fit. Generally speaking, the principle of boosting is illustrated in Figure 4.11.

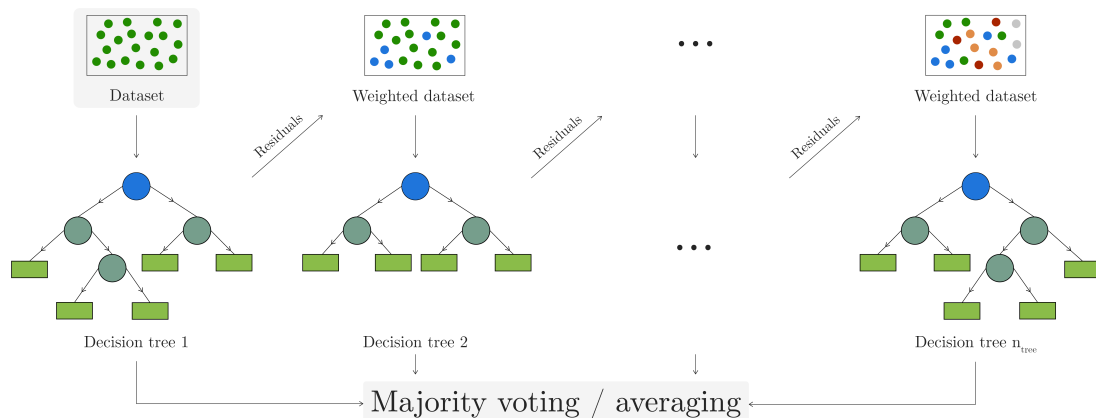


Figure 4.11: Mechanisms of tree boosting

XGBoost, short for extreme gradient boosting, is a specific tree-based boosting implementation that is used in some applications of this thesis. Let us provide some details about it, first by showing a general pseudo-algorithm for this approach as shown in Algorithm 3.

Algorithm 3 XGBoost Pseudo-Algorithm

- 1: **Input:** Training set $\mathcal{D}_{train} = \{(x_i, y_i)\}_{i=1}^N$, a twice differentiable loss function $\ell(y, F)$ and a learning rate α
 - 2: **Initialize with a constant value for ρ :** $F_0(\mathbf{x}) = \arg \min_{\rho} \sum_{i=1}^N \ell(y^{(i)}, \rho)$
 - 3: **for** $k = 1$ to n_{tree} **do**
 - 4: **for** $i = 1$ to N **do**
 - 5: Compute $g_k^{(i)} = \frac{\partial \ell(y^{(i)}, F_{k-1}(\mathbf{x}^{(i)}))}{\partial F(\mathbf{x}^{(i)})}$ and $h_k^{(i)} = \frac{\partial^2 \ell(y^{(i)}, F_{k-1}(\mathbf{x}^{(i)}))}{\partial F(\mathbf{x})^2}$
 - 6: **end for**
 - 7: Fit a regression tree to the targets $-\frac{g_k^{(i)}}{h_k^{(i)}}$ giving terminal regions $R_j, j = 1, \dots, J$
 - 8: **for** $j = 1$ to J **do**
 - 9: Compute $\rho_k = \arg \min_{\rho} \sum_{\mathbf{x}^{(i)} \in R_j} [g_k^{(i)} + \rho h_k^{(i)}]$
 - 10: **end for**
 - 11: Update $F_k(\mathbf{x}) = F_{k-1}(\mathbf{x}) + \alpha \sum_{j=1}^J \rho_k I(\mathbf{x} \in R_j)$
 - 12: **end for**
 - 13: **Output:** $F_{n_{tree}}(\mathbf{x})$
-

XGBoost works similarly to Newton-Raphson’s algorithm unlike general gradient boosting, which works as a gradient descent in the function space. This connection with the Newton-Raphson method clearly appears when the second-order Taylor approximation is used in the loss function. In addition to its specificity, it offers a wide range of hyper-parameters. For instance, it includes a regularization term in the loss function, which controls the complexity of the model, thus preventing overfitting. It also implements the possibility of performing cross-validation at each iteration of the boosting process. Owing to this diversity of parameters, total control over the implementation of gradient boosting is possible. For any observation with a missing value on a covariate used for a split, the algorithm sends the observation to both child nodes and learns the path that reduces the loss the most. Eventually, the algorithm is designed to be highly efficient, flexible, and portable; implements parallel processing; and is notably fast and optimized. Therefore, solutions based on XGBoost are frequently winning in Kaggle competitions.

4.2.3 Theoretical guarantees

Minimizing within-node heterogeneity or maximizing between-node heterogeneity

As seen in the previous sections, there are two ways to design a splitting procedure for a given node. We either want to produce child nodes such that each of them is very homogeneous regarding the response variable, or we want to select a split that divides the dataset into populations with responses as different as possible. In other terms, we either want to minimize the within-node heterogeneity (WNH) or maximize the between-node heterogeneity (BNH). In most simple settings, with most TBM, we can prove that both approaches are equivalent and even precise how they relate. This is demonstrated with the derivation of Equation 4.3, in the case of a regression setting, with MSE as a measure of heterogeneity, as in CART for instance.

We denote $\langle x, y \rangle$, the scalar product between x and $y \in \mathbb{R}^2$, $\|x\|^2 = \langle x, x \rangle$ and $H(x, y) =$

$\|x - y\|^2$. $H(x, y)$ can be seen as a measure of the heterogeneity between x and y . For a given node g , we denote

$$\bar{y}(g) = \frac{1}{\mathcal{N}(g)} \sum_{\text{observations } i \in g} y^{(i)}.$$

The heterogeneity between two child nodes g_l and g_r , split from a parent node g_p is then given by

$$H(g_l, g_r) = H(\bar{y}(g_l), \bar{y}(g_r)),$$

which allows us to define the WNH and the BNH. The within-node g heterogeneity is immediately given by

$$WNH(g) = \frac{1}{\mathcal{N}(g)} \sum_{\text{observations } i \in g} H(\bar{y}(g), y^{(i)}),$$

in other words, the MSE of node g . And the between-nodes g_l and g_r heterogeneity is given by

$$BNH(g_l, g_r) = \frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} H(\bar{y}(g_l), \bar{y}(g_l \cup g_r)),$$

in other words, the mean of child nodes' MSEs weighted by the number of observations they represent.

Remark 4.4

The notion of BNH can be extended to more than 2 nodes. With k nodes, we have:

$$BNH(g_1, \dots, g_k) = \sum_{i=1}^k \frac{\mathcal{N}(g_i)}{\mathcal{N}(\bigcup_{j=1}^k g_j)} H\left(\bar{y}(g_i), \bar{y}\left(\bigcup_{j=1}^k g_j\right)\right).$$

Those notions being properly defined, we immediately deduce from Koenig-Huygens theorem the following result:

$$WNH(g_l \cup g_r) = \frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} WNH(g_l) + \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)} WNH(g_r) + BNH(g_l, g_r). \quad (4.3)$$

And because we consider a disjoint partitioning of g_p into g_l and g_r , we have $WNH(g_l \cup g_r) = WNH(g_p)$. Thus, the maximization of the BNH is equivalent to the minimization of the mean of WNHs in that regression setting, with MSE as a heterogeneity measure.

Remark 4.5

Equation 4.3 can also be extended if g_p is divided in k child nodes g_1, \dots, g_k forming a disjoint partition, with

$$WNH\left(\bigcup_{i=1}^k g_i\right) = WNH(g_p) \sum_{i=1}^k \frac{\mathcal{N}(g_i)}{\mathcal{N}(g_p)} WNH(g_i) + BNH(g_1, \dots, g_k).$$

Remark 4.6

This specific result is not generalizable to all tree-growing procedures. Notably, for all trees that produce splits based on statistical tests (which will also be relevant for survival trees of Chapter 5), such an equivalence between minimizing the mean WNH and maximizing BNH does not exist. To the best of our knowledge, the difference between those two approaches has not been studied yet in such cases.

Regarding the choice of the impurity function in CART

The last section demonstrated a result for CART in a regression setting where MSE originally was the intuitive choice of heterogeneity measure. For classification purposes with J classes, we rather talk about impurity function rather than heterogeneity measure. Conceptually, this is the same thing but in practice, there is no obvious choice for measuring classification impurity. As stated in Section 4.2.1, L. Breiman et al. 1984 originally considered the entropy, Gini, and Twoing criteria and the practical and theoretical differences between them have been well studied in the literature. In this section, we will mention some of the results derived in Breiman 1996 or Shih 1999.

Let there be J classes numbered $1, \dots, J$, and denote the proportions of the classes in node g by $\mathbf{p}(g) = p_{1,g}, \dots, p_{J,g}$. Let $I(\cdot)$ be an impurity function, defined and twice differentiable for $\mathbf{x} \in [0, 1]^J$. Assume that $I(\mathbf{x})$ is convex and let the impurity of node g be $I(g)$ and the goodness-of-split be the decrease in impurity from the parent node g_p when split into child nodes g_l and g_r with split rule s is given by

$$GoS(s, g_p) = I(g_p) - \frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} I(g_l) - \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)} I(g_r).$$

Breiman derives the following result (See Theorem 1 from Breiman 1996):

Let α be the vector of the proportions of each class sent to g_l such as $p_{j,g_l} = \alpha_j p_j / \mathcal{N}(g_l)$ and thus $p_{j,g_r} = (1 - \alpha_j) p_j / \mathcal{N}(g_r)$. Then the maximum impurity decrease over $\alpha \in [0, 1]^J$ is achieved at a vertex of $[0, 1]^J$.

With this result, we can now study which vertex of $[0, 1]^J$ corresponds to the optimal α , thus the optimal split for the different impurity measures considered in CART.

Using the entropy, the best split of g_p is chosen by maximizing

$$\frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} \sum_j p_{j,g_l} \log p_{j,g_l} + \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)} \sum_j p_{j,g_r} \log p_{j,g_r}. \quad (4.4)$$

Let $P_l = \frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)}$ and $P_r = \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)}$, for a given vertex, let $\mathcal{C}_0 = \{j; \alpha_j = 0\}$, $\mathcal{C}_1 = \{j; \alpha_j = 1\}$.

Equation 4.4 becomes

$$\begin{aligned} & P_l \sum_{j \in \mathcal{C}_1} \left(\frac{p_j}{P_l} \right) \log \left(\frac{p_j}{P_l} \right) + P_r \sum_{j \in \mathcal{C}_0} \left(\frac{p_j}{P_r} \right) \log \left(\frac{p_j}{P_r} \right) \\ &= \sum_j p_j \log p_j - P_l \log P_l - P_r \log P_r. \end{aligned}$$

Thus, the vertex at which the optimal split occurs maximizes

$$-P_l \log P_l - P_r \log P_r,$$

and thus, at the best vertex $|P_l - \frac{1}{2}|$ is minimized. The same result holds for using Twoing criterion as it can be seen as applying the entropy criterion after assembling all classes into two “superclasses”. As a result, entropy and Twoing criteria tend to produce splits that balance the sizes at the two children nodes.

If we now turn to the Gini impurity measure, we see that the vertex that produces the best split must minimize

$$P_l \sum_{j \in \mathcal{C}_1} p_{j,g_l} (1 - p_{j,g_l}) + P_r \sum_{j \in \mathcal{C}_0} p_{j,g_r} (1 - p_{j,g_r}) = \\ P_l \sum_{j \in \mathcal{C}_1} \left(\frac{p_j}{P_l} \right) \left(1 - \frac{p_j}{P_l} \right) + P_r \sum_{j \in \mathcal{C}_0} \left(\frac{p_j}{P_r} \right) \left(1 - \left(\frac{p_j}{P_r} \right) \right),$$

which corresponds to maximizing

$$\frac{1}{P_l} \sum_{j \in \mathcal{C}_1} p_j^2 + \frac{1}{P_r} \sum_{j \in \mathcal{C}_0} p_j^2.$$

We denote $p_m = \max_j (p_j)$, the proportion of the most represented of all classes. Then the optimal vertex, according to the Gini criterion tends to send all of class m to g_l and all other observations to g_r : the largest class into one pure node, and all others into the other.

Such theoretical results have practical consequences on the trees produced for problems with a great number of classes. In such cases, entropy or Twoing-based splitting criteria are likely to produce highly unstable first splits. Indeed, the number of vertices for which $P_l \simeq \frac{1}{2}$ grows with the number of classes, as such there may not be a unique way of choosing the early splits of a tree. Conversely, a gini-based criterion is likely to produce unbalanced splits. All those theoretical observations have been confirmed by practical simulations and applications (see Shih 1999 for instance).

Consistency results

A model is said to be Bayes consistent when it converges to the Bayes decision rule as the number of observations in the training set increases (see T. Zhang 2004). CART has been proven to be Bayes consistent under conditions (as early as in L. Breiman et al. 1984 or Devroye, Györfi, and Lugosi 1996), thus it can approximate any decision boundary arbitrarily well, given a sufficiently large training dataset.

Recently, an innovative approach by Klusowski and Tian 2023 demonstrated consistency conditions for CART and C4.5 trees both in regression and classification contexts, even when the number of predictor variables grows exponentially with the sample size. The result is then extended to random forests of CARTs and C4.5 trees.

With the notations defined in Section 4.1.1 and considering the additive function class

$$\mathcal{G}^1 := \{g(\mathbf{x}) := g_1(x_1) + g_2(x_2) + \dots + g_p(x_p)\},$$

where $g_1(x_1), g_2(x_2), \dots, g_p(x_p)$ is a set of p univariate and Borel measurable functions. The consistency results of Klusowski and Tian 2023 are obtained in the generalized additive modeling

framework, where \mathcal{G}^1 is the hypothesis space, i.e the TBM aims at finding a $g(\cdot) \in \mathcal{G}^1$ such that the true model is approximated by $g(\mathbf{x})$.

In that setting, a major result of Klusowski and Tian 2023 (Lemma 4.1 and Theorem 4.2) gives an empirical risk bound for CART and C4.5, for squared error loss and logistic loss:

Let \mathcal{T}_K be a depth $K \geq 1$ decision tree. Denoting $\widehat{\mathcal{L}}(g) := \frac{1}{N} \sum_{i=1}^N \ell(g(\mathbf{x}^{(i)}), y^{(i)})$ the empirical risk and \mathcal{L} the true risk of a model (see notations in Section 4.1.1), we have

$$\widehat{\mathcal{L}}(\hat{g}(T_K)) \leq \inf_{g(\cdot) \in \mathcal{G}^1} \left\{ \widehat{\mathcal{L}}(g) + \frac{V^2(g)}{K+3} \right\},$$

where $V(g)$ is a constant, different for CART or C4.5, specified in Klusowski and Tian 2023, Lemma 4.1. The theorem presented above asserts that when considering a decision tree with depth K , CART and C4.5 minimize the empirical risk among the additive function class. This bound tends to shrink toward the true risk of the model, with a convergence rate of $\mathcal{O}(\frac{1}{K})$.

This result is then extended to a consistency result in Corollary 4.4, stating that considering a sequence of prediction problems with true models $\{g_N^*(\cdot)\}_{N=1}^\infty$.

Assume that $g_N^*(\mathbf{x}) = \sum_{j=1}^{p_N} g_j(x_j) \in \mathcal{G}^1$ and $\sup_N \|g_N^*\|_\infty < \infty$. Suppose that $K_N \rightarrow \infty$, that the aggregated total variation of the individual component functions g_j s is $o(\sqrt{K_N})$, and $(2^{K_N} \log^2(N) \log(N p_N)) / N \rightarrow 0$ as $N \rightarrow \infty$. Eventually, assuming a sub-gaussian noise, regression trees are consistent, and classification trees are consistent.

We refer the astute reader to this work for further details and references to other results on TBM consistency, especially ensemble models, under various conditions (the many works of Biau (Biau, Devroye, and Lugosi 2008; Biau 2012; Biau and Cadre 2017) or Scornet (Scornet, Biau, and Vert 2015; Scornet 2016) for instance).

5. Survival analysis

The most natural encounter of a temporal dimension in the data is when the target variable is itself of a temporal nature. In actuarial science, one can study the time-to-death or lapse in life insurance, the time to the next claim or its settlement in non-life insurance, disease, recovery, credit failure, termination of contracts and so on. Such analysis requires the study of the occurrence of an event - or events - through time and it requires a specific modeling approach called survival analysis. Time is measured from the beginning of the follow-up of an individual until the hypothetical occurrence of the event and is referred to as survival time. Some subjects will have experienced the event of interest during the follow-up period and some will not. The latter observations are described as right-censored and this censorship is the main underlying specificity in the data structure for survival analysis.

5.0.1 Survival notations

For the analysis of survival data, we will focus on studying the individual survival time - or failure, or event time - until the occurrence of an event, T . This time T can usually be defined as the time since the individual was born, the time since the individual entered the study, or the time since a fixed date common to all individuals.

In survival studies, the duration of the follow-up is limited in time and some subjects may be right-censored. This can happen when an individual leaves the study prematurely, either voluntarily or involuntarily - a so-called *dropout* - or if an individual is still under study but did not experience an event at the end of the follow-up period.

When the censorship is independent of the event, in other words when the individual's survival probability is not linked to the fact that the subject's data is observable, the survival outcome is defined using the classical survival notations. Subject i will eventually experience the event at time $\overset{\star}{T}^{(i)}$ but she is no longer observed after a censoring time $C^{(i)}$. We let $T^{(i)}$ denote the observed event time for subject i , defined as $T^{(i)} = \min \left(\overset{\star}{T}^{(i)}, C^{(i)} \right)$. Eventually, we introduce the event indicators:

$$\Delta^{(i)} = \mathbb{1} \left\{ \overset{\star}{T}^{(i)} \leq C^{(i)} \right\}, \quad (5.1)$$

and,

$$\delta^{(i)}(t) = \mathbb{1} \left\{ \overset{\star}{T}^{(i)} \leq t \right\}. \quad (5.2)$$

In other words, we have:

$$\Delta^{(i)} = \begin{cases} 1 & \text{if the event is observed i.e. } T^{(i)} = \overset{\star}{T}^{(i)} \\ 0 & \text{if the subject is right-censored i.e. } T^{(i)} = C^{(i)} \end{cases}, \quad (5.3)$$

which indicates whether an individual experienced the event or was right-censored. The event indicator $\delta^{(i)}(t)$ indicates whether an individual experienced the event at time t .

An illustration of survival data is shown in Figures 5.1 and 5.2, where the event is not observed for individuals (1) and (2) at censoring time C , in contrast to individuals (3) and (4) where the event occurred before time C .

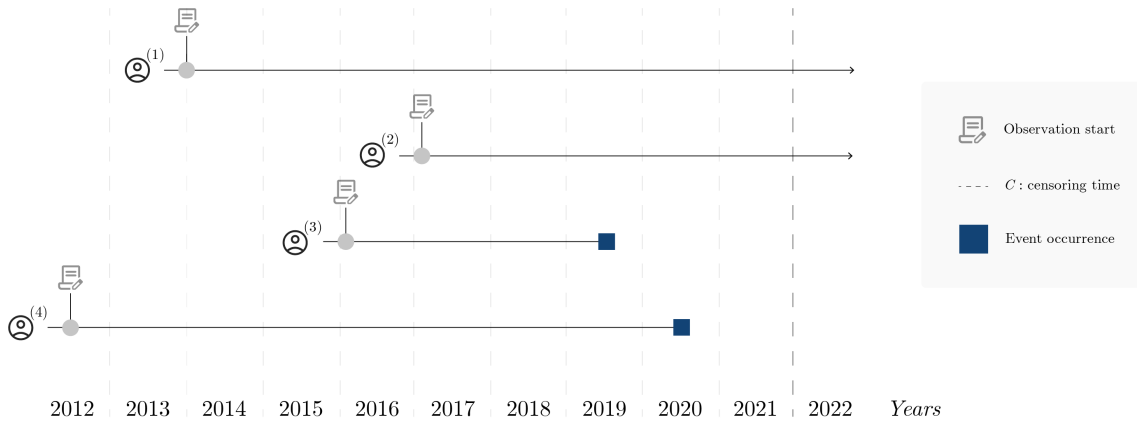


Figure 5.1: Illustration of survival analysis and censorship

Figure 5.1 can also be represented by aligning all subjects' observation starting times.

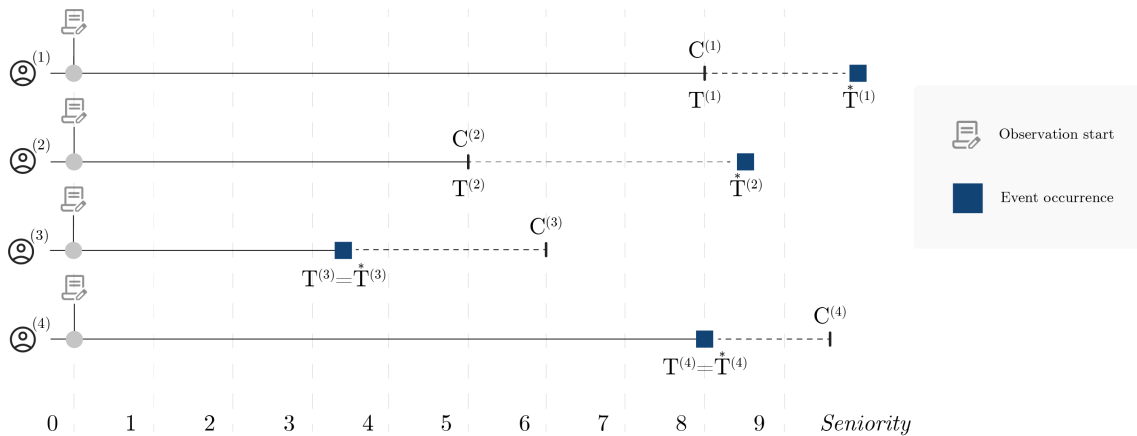


Figure 5.2: Illustration of survival analysis and censorship - Starting times aligned

The survival time T^* is commonly studied through the analysis of its related survival function $S(t)$. It represents the probability of not experiencing the event before censorship time and is defined by

$$S(t) = \Pr[T^* > t] = 1 - \Pr[T^* \leq t] = e^{-\Lambda(t)} = e^{-\int_0^t \lambda(u) du}, \quad (5.4)$$

where $\Lambda(t)$ is the cumulative hazard function and $\lambda(u)$ is the instantaneous hazard rate of the event. The survival function is defined as a probability function, with values between 0 and 1,

where $S(0) = 1$ and $S(t) \xrightarrow{+\infty} 0$ and $t_1 \leq t_2 \Rightarrow S(t_1) \geq S(t_2)$. The distribution function $F(t)$ can also be derived from the survivor function by

$$F(t) = 1 - S(t) = P(T \leq t) = \int_0^t f(u)du. \quad (5.5)$$

where $f(t)$ is the density function defined as

$$f(t) = \lim_{\Delta_t \rightarrow 0} \frac{P(t \leq T < t + \Delta_t)}{\Delta_t}. \quad (5.6)$$

where Δ_t is a small time interval. From that last equation, we can derive that $\lambda(u) = \frac{f(u)}{S(u)}$.

We refer the astute reader to the introduction chapters of Devaux 2022 for more details regarding survival analysis notations and generalities.

5.1 Models

Survival models are required to analyze time-to-event outcomes in the presence of censored observations. Various tree-based models have been devised for this purpose, providing a non-parametric approach to this problem. These models offer great flexibility, making them popular alternatives to their parametric counterparts (Bou-Hamad, Larocque, and Ben-Ameur 2011). One of their key advantages is their ability to detect intricate relationships and interactions within data automatically, which could be beyond the reach of parametric models (see Bertsimas et al. 2022). Their tree structure also facilitates the derivation of risk groups. However, they are not without drawbacks: these models can potentially overfit the data, especially if not properly tuned, and their interpretability may not be as straightforward as that of parametric models.

Survival tree-based models extend the scope of regular tree-based models by modifying the splitting criterion to account for censored data. This tailoring enables the models to handle the unique challenges introduced by survival analysis more effectively, such as the need to consider not only whether an event occurred but also when it occurred. For more details about surviving trees, we refer the reader to the complete review of Bou-Hamad, Larocque, and Ben-Ameur 2011, up to 2011, from which this section is inspired. Thus, these models are important tools in the ML researcher's arsenal, offering a nuanced and robust approach to survival analysis.

5.1.1 Inverse probability of censorship weighted (IPCW) models

Molinaro, Dudoit, and van der Laan 2004 proposed a method for building trees with censored data by modifying the split criterion of a tree built for uncensored observations. Their approach is based on weighting the impurity function without censoring by the IPCW¹ in order to adapt it to censoring. They used an IPC-weighted MSE splitting criterion and showed that training a regression model with this weighted criterion with all the observations considered as fully observed is equivalent to training a time-to-event model accounting for censorship. They demonstrated the consistency of their method with fixed covariates and time-to-event outcome (see Vock et al. 2016). With this approach, time-to-event trees can directly take the survival time as the outcome, which is an alternative to the usual tree-based survival methodology that consists of indirectly studying hazard functions at each considered split. Unlike the methods detailed in the following

¹More details about IPCW can be found in Section 5.2

sections, this does not describe a specific model, but rather a framework in which any regression model can be adapted to account for censorship, via its loss function.

5.1.2 Survival trees

To perform survival analysis, the usual decision-tree algorithm, specifically its splitting procedure, must be modified. Most survival tree algorithms in the literature use survival similarity/dissimilarity measures, e.g. two-sample test statistics such as the log-rank (see Appendix A), for splitting. Such statistics inherently use the IPCW to account for censorship. At each node, the idea is to design a split function that selects the split that maximizes the separation in terms of the survival profile between the two child nodes. It depends on the likelihood of the hazard functions, as the splits are accomplished by maximizing the log-rank statistic (see Mantel 1966 for insightful details on the statistic, see LeBlanc and Crowley 1992 for a survival tree based on it) or any other survival distribution distance (such as the Wasserstein metric in Gordon and Olshen 1985 or deviance measure in LeBlanc and Crowley 1992). Similarly, Ciampi, Hogg, and Kates 1986 proposed a general formulation using the likelihood ratio statistic to measure the dissimilarity between the two child nodes. In any case, the larger the statistic, the more dissimilar the two child nodes which is why the splits chosen at each node are those that maximize the statistic.

Algorithm 4 shows a pseudo-algorithm detailing how a log-rank-based survival tree is grown.

Algorithm 4 Survival Tree Pseudo-Algorithm

```
1: Initialize the root node, which includes the entire dataset  $\mathcal{D}$ .
2: while stopping criteria not met do
3:   for each terminal node do
4:     for each variable do
5:       for each possible split point of the variable do
6:         Calculate the log-rank test statistic between the two resulting nodes.
7:       end for
8:       Select the variable and split point with the maximum log-rank test statistic.
9:     end for
10:    Split the terminal node at the selected variable and split point.
11:   end for
12: end while
13: Assign survival function to each terminal node using Kaplan-Meier estimates.
```

The tree starts with a root node that includes the entire dataset. The algorithm then iteratively splits terminal nodes to maximize the separation of survival times. The splitting criterion is the log-rank test statistic, a measure of the difference in survival between two groups. Once a stopping criterion is met (e.g., a maximum tree depth is reached or the log-rank test statistic for all possible splits is below a threshold), the algorithm stops. The final step is to assign the survival function to each terminal node using Kaplan-Meier estimates, a non-parametric statistic used to estimate the survival function from lifetime data. Thus, the prediction given by each leaf is not only a number, as in a regression tree but a whole survival function as illustrated in Figure 5.3.

To make a prediction for a given subject, their data are passed down the tree from the root to a leaf node. The path is determined by the rules at each split, which compare the features of the

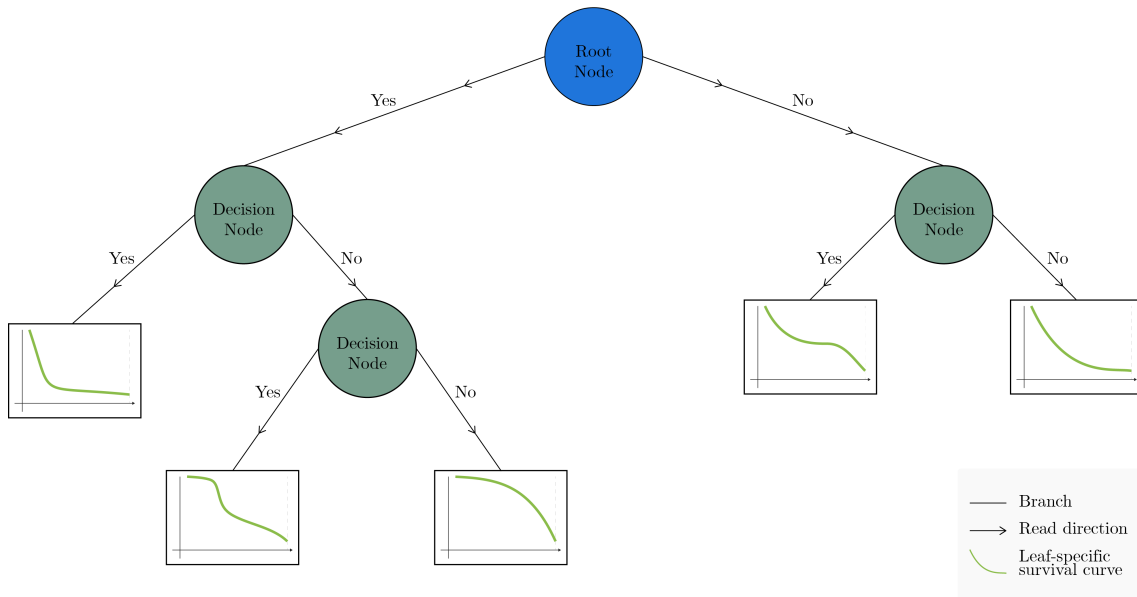


Figure 5.3: Illustration of a survival tree

subject to the thresholds determined during training. The leaf node where the subject ends up contains the survival times and event indicators (whether the event of interest occurred) for the training subjects that ended up in the same leaf node. The survival function is then estimated using these data.

In such models, the study of the time-to-event outcome is achieved by considering test statistics, where the null hypothesis is that the two groups have identical hazard functions. Therefore, the indirect study of the time-to-event outcome through its hazard function is how most models handle the time-dependent outcome in survival analysis. Single survival trees do have some drawbacks, most notably their instability. Small changes in the data can lead to significant changes in the tree structure, affecting the predictive accuracy of the trees. They can also be prone to overfitting, where the model learns the training data too well and performs poorly on unseen data.

5.1.3 Random survival forests (RSF)

An RSF, an ensemble method for combining multiple trees, can overcome the aforementioned limitations of single-survival trees. RSFs generally offer increased prediction accuracy and stability by averaging the predictions of a multitude of different trees (see Ishwaran et al. 2008). These characteristics make RSFs popular choices for survival analysis, especially for complex data structures, because the RSF approach is a flexible continuous-time method that is not constrained by strong assumptions on the survival function. A generic pseudo-algorithm exists for building an RSF, as shown in Algorithm 5.

As described in Section 5.1.2, each tree in a forest generates a survival function for a given subject. The final survival function for the subject is obtained by averaging the survival functions of all the trees in the forest. As in any bagging procedure, this averaging process helps make the prediction more robust and less prone to overfitting via variance reduction.

Algorithm 5 Random Survival Forest Pseudo-Algorithm

```
1: procedure RANDOM SURVIVAL FOREST( $\mathcal{D}, n_{tree}, n_{cov}$ )
2:   Initialize an empty set of trees  $Forest$ 
3:   for  $k = 1$  to  $n_{tree}$  do
4:     Draw a bootstrap sample  $\mathcal{D}_k$  from the original data
5:     Grow a survival tree  $\mathcal{T}_k$  on  $\mathcal{D}_k$  as follows:
6:       Initialize  $\mathcal{T}_k$  with a single node containing all observations
7:       Repeat:
8:         Select  $n_{cov}$  covariates at random from all variables
9:         Choose the best variable/split-point among the  $n_{cov}$ 
10:        Split the node into two child nodes using the log-rank split statistic
11:       Until: Minimum node size is reached
12:     Add  $\mathcal{T}_k$  to  $Forest$ 
13:   end for
14:   return  $Forest$ 
15: end procedure
```

The formula for the survival function $S(t)$ at time t for a subject with features $\mathbf{x}^{(i)}$ is:

$$S(t|\mathbf{x}^{(i)}) = \frac{1}{n_{tree}} \sum_{k=1}^{n_{tree}} S_k(t|\mathbf{x}^{(i)}). \quad (5.7)$$

Note that similarly to a single survival tree, this prediction does not provide a single time point, but rather a function that provides the probability of survival over time. This aspect is one of the key strengths of survival analysis models as it provides a more detailed picture of survival probabilities than binary classification or regression models.

5.1.4 Gradient Boosting Survival Model (GBSM)

GBSM is also an ensemble learning method that aims to predict the survival probabilities of an event by combining the predictions of multiple decision trees. In the same way tree boosting algorithms work, each individual tree is built sequentially and each tree attempts to correct the errors made by its predecessor. The algorithm uses a loss function suitable for survival data and applies the gradient descent method to minimize it. The pseudo-algorithm for building a GBSM is provided in Algorithm 6.

In Algorithm 6, ν is the learning rate, n_{tree} is the number of boosting iterations, J_k is the number of terminal nodes of the k -th tree, and γ_{jk} are the optimal terminal node predictions.

Please note that the loss function $\ell(y_i, F(x_i))$ and the way to compute the optimal terminal node predictions γ_{jk} depend on the specific survival model used. The choice of ℓ is usually the partial likelihood loss of Cox's proportional hazards model (see Section A.0.7 of Appendix A, and γ_{jk} can be estimated by solving a scoring equation. The objective is thus to maximize the log partial likelihood function, modified by replacing the traditional linear part - or the predicted risk - of the Cox model $(\mathbf{x}^{(i)\top} \cdot \beta^{(i)})$ with the additive model $F_{GBSM}(\mathbf{x}^{(i)})$, thus selecting the optimal model g as

$$g = \arg \min_{F_{GBSM}} \sum_{i=1}^N \Delta^{(i)} \left[F_{GBSM}(\mathbf{x}^{(i)}) - \log \left(\sum_{j \in \mathcal{R}^{(i)}} \exp(F_{GBSM}(\mathbf{x}^{(j)})) \right) \right]. \quad (5.8)$$

Algorithm 6 Gradient Boosting Survival Model Pseudo-Algorithm

- 1: Initialize model with a constant value: $F_0(\mathbf{x}) = \arg \min_c \sum \ell(y_i, c)$, with ℓ an arbitrary differentiable loss function.
 - 2: **for** $k = 0$ to n_{tree} **do**
 - 3: Compute so-called pseudo-residuals: $r_{ik} = - \left[\frac{\partial \ell(y_i, F_k(\mathbf{x}^{(i)}))}{\partial F(\mathbf{x}^{(i)})} \right]$ for $i = 1, \dots, N$.
 - 4: Fit a tree to the pseudo-residuals, i.e., train a tree to predict r_{ik} using $\mathbf{x}^{(i)}$, resulting in leaf regions R_{jk} , for $j = 1, \dots, J_k$.
 - 5: **for** $j = 1, \dots, J_k$ **do**
 - 6: Compute $\gamma_{jk} = \arg \min_\gamma \sum \ell(y_i, F_{k-1}(\mathbf{x}^{(i)}) + \gamma)$ for $\mathbf{x}^{(i)}$ in R_{jk} .
 - 7: **end for**
 - 8: Update the model: $F_k(\mathbf{x}) = F_{k-1}(\mathbf{x}) + \nu \sum \gamma_{jk} I(\mathbf{x} \in R_{jk})$.
 - 9: **end for**
 - 10: Output the boosted model: $F_{GBSM}(\mathbf{x})$
-

5.2 Survival performance metrics

5.2.1 Brier Score and variations

The Brier Score (BS) (see Brier 1950) is an extension of the mean squared error to right-censored data, providing a holistic measure of prediction accuracy for survival models.

With a given dataset \mathcal{D} and assuming that we are interested in the occurrence of only one event, any survival model yields $\hat{S}(t)$ the predicted survival probability function at any time t . Let $\hat{G}(t) = P[C > t]$ be the Kaplan-Meier (KM) estimate of the censoring distribution (see Appendix A.0.5 for details regarding KM estimation) and $\widehat{W}^{(i)}(t)$ the corresponding IPCW, the BS is given by:

$$\widehat{\text{BS}}(t, \hat{S}; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{i \in \mathcal{D}} \widehat{W}^{(i)}(t) \left[\delta^{(i)}(t) - \hat{S}(t) \right]^2.$$

With the notations introduced in Section 5.0.1, the IPCW are being computed as follows

$$\widehat{W}^{(i)}(t) = \frac{(1 - \delta^{(i)}(t)) \Delta^{(i)}}{\hat{G}(T^{(i)})} + \frac{\delta^{(i)}(t)}{\hat{G}(t)}.$$

The obtained BS is a vector of scores computed at different time points. In order to get a more concise evaluation metric, we can also define the integrated Brier Score (IBS), defined as

$$\widehat{\text{IBS}}(\hat{S}; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{i \in \mathcal{D}} \frac{1}{T^{(i)}} \int_0^{T^{(i)}} \widehat{W}^{(i)}(t) \left[\delta^{(i)}(t) - \hat{S}(t) \right]^2 dt.$$

The BS and IBS can be easily derived into the Brier Skill Score (BSS) and the integrated Brier Skill Score (IBSS) respectively. There are modified versions of BS and IBS that contrast the prediction accuracy of a model to a reference model. They are defined as

$$\widehat{\text{BSS}}(t, \hat{S}; \mathcal{D}) = 1 - \frac{\widehat{\text{BS}}(t, \hat{S}; \mathcal{D})}{\widehat{\text{BS}}(t, \widehat{S}_{ref}; \mathcal{D})},$$

$$\widehat{\text{IBSS}}(\hat{S}; \mathcal{D}) = 1 - \frac{\widehat{\text{IBS}}(\hat{S}; \mathcal{D})}{\widehat{\text{IBS}}(\widehat{S}_{ref}; \mathcal{D})}.$$

BSS measures the BS improvement of the considered model over some reference one that yields a survival function \widehat{S}_{ref} . We see that it takes positive (or negative) values whenever the $\widehat{BS}(t, \widehat{S}; \mathcal{D})$ - respectively $\widehat{IBS}(\widehat{S}; \mathcal{D})$ - is inferior (or superior) to $\widehat{BS}(t, \widehat{S}_{ref}; \mathcal{D})$ - respectively $\widehat{IBS}(\widehat{S}_{ref}; \mathcal{D})$. In definitive, the BSS and IBSS represent the improvement in terms of Brier Score over the naïve model: the higher, the better.

5.2.2 Concordance indices

Harrel's c-index

The c-index, introduced by Harrell et al. 1982, is one of the most commonly used survival model evaluation metrics that assess the correlation between predicted risks and actual event times. A higher c-index indicates better discrimination between instances with higher risks leading to earlier events, and those with lower risks leading to later events. This metric condenses three distinct aspects of predictions - risk level, event occurrence, and time - into a single figure, making it easier to distinguish effective models from those that perform almost randomly. But this succinct nature of the c-index also makes its practical interpretation more challenging compared to classification and ranking metrics. Moreover, not immediately evident in its standard definition, the c-index inherently depends on time. This often-overlooked aspect is essential for extracting significant insights about the model's performance. It is computed as

$$C_H = \frac{\sum_{i \neq j} I(T^{(i)} < T^{(j)})I(R^{(i)} < R^{(j)}) + 0.5 \sum_{i \neq j} I(T^{(i)} = T^{(j)})I(R^{(i)} < R^{(j)})}{\sum_{i \neq j} I(T^{(i)} < T^{(j)}) + \sum_{i \neq j} I(T^{(i)} = T^{(j)})}$$

where $\hat{R}^{(i)}$ and $\hat{R}^{(j)}$ are the predicted risks ² for instances i and j . If we already mentioned that for a Cox model, it corresponds to the linear part within the exponential, we voluntarily avoid detailing how the predicted risk is defined for other specific models as more insights can be found in Uno et al. 2011 or in Hartman et al. 2023 for instance.

The numerator counts the number of pairs (i, j) for which the subject with the shorter survival time also has a higher predicted risk, plus half the number of pairs with equal survival times. The denominator counts all the comparable pairs. The resulting c-index is a proportion that ranges from 0.5 (random prediction) to 1 (perfect prediction). Despite being extensively used, it has some limitations, especially in cases of high data censoring or when a specific time range is the primary focus.

Uno's index

Uno's estimator is an alternative to Harrell's c-index in survival analysis. It behaves better than Harrell's c-index when the amount of censoring in the test data is high, making it a more reliable measure in such situations. The disposal of pairs of censored observations ($\Delta_j = 0$) results in an upward bias in the estimation of the concordance probability. Therefore, Uno et al. 2011 proposed a variation of Harrell's c-index that includes the inverse probability of censoring weighting. In

²also referred to as the prognostic marker in the biomedical field

brief, IPCWs are introduced in Harrel's c-index, leading to the following performance metric:

$$\frac{\sum_{i \neq j} (I(T^{(i)} < T^{(j)})I(\hat{R}^{(i)} > \hat{R}^{(j)})W^{(i)}W^{(j)} + \frac{1}{2} \sum_{i \neq j} I(T^{(i)} = T^{(j)})I(\hat{R}^{(i)} = \hat{R}^{(j)})W^{(i)}W^{(j)})}{\sum_{i \neq j} I(T^{(i)} \neq T^{(j)})W^{(i)}W^{(j)}}.$$

In summary, Uno's c is preferable to Harrell's c-index in the presence of higher amount of censoring and uses the Kaplan-Meier estimator for computing the IPCW. This implies that the censoring is assumed to be independent of the variables.

5.2.3 Dynamic AUC

Eventually, the AUROC defined in Section 4.1.2 has also been adapted to survival analysis. It has been extended (see Lambert and Chevret 2016) to censored survival times: given a time point t , it estimates how well a predictive model can distinguish subjects who will experience an event by time t (sensitivity) from those who will not (specificity). The dynamic AUC represents the probability that, given two randomly selected subjects one having experienced the event of interest before time t and the other after, the predicted risks are correctly ranked.

$$AUC(t) = P\left(R^{(i)} > R^{(j)} \mid T^{(i)} \leq t, T^{(j)} > t\right) \quad (5.9)$$

In simple terms, it is the probability that the risk of occurrence of the event is greater for the subjects who have already experienced it compared with those who have not yet.

In Part I and II, we have explored the historical evolution and theoretical underpinnings of tree-based machine learning algorithms, specifically focusing on their adaptation to survival analysis. Understanding the intricacies and nuances of these algorithms sets the stage for the next phase of this thesis, where we pivot our attention toward the practical application of these methodologies within the domain of life insurance. Part III delves into the formulation of an LMS framework, leveraging economic measures such as CLV in tandem with tree-based models and survival analysis. The forthcoming chapters bridge theory with pragmatic implementation, and propose an insightful and adaptive approach to lapse management.

Bibliography

- Geman, S., E. Bienenstock, and R. Doursat (Jan. 1992). “Neural Networks and the Bias/Variance Dilemma”. In: *Neural Computation* 4, pp. 1–58. DOI: [10.1162/neco.1992.4.1.1](https://doi.org/10.1162/neco.1992.4.1.1).
- Vapnik, V.N. (1998). *Statistical Learning Theory*. Wiley-Interscience.
- Neal, B. et al. (2019). *A Modern Take on the Bias-Variance Tradeoff in Neural Networks*. arXiv: [1810.08591 \[cs.LG\]](https://arxiv.org/abs/1810.08591).
- Belkin, M. et al. (2019). “Reconciling modern machine-learning practice and the classical bias–variance trade-off”. In: *Proceedings of the National Academy of Sciences* 116.32, pp. 15849–15854. DOI: [10.1073/pnas.1903070116](https://doi.org/10.1073/pnas.1903070116).
- Geurts, P. (2005). “Bias vs Variance Decomposition for Regression and Classification”. In: *Data Mining and Knowledge Discovery Handbook*. Ed. by Oded Maimon and Lior Rokach. Boston, MA: Springer US, pp. 749–763. ISBN: 978-0-387-25465-4. DOI: [10.1007/0-387-25465-X_34](https://doi.org/10.1007/0-387-25465-X_34).
- Bouckaert, R.R. (2008). “Practical Bias Variance Decomposition”. In: *AI 2008: Advances in Artificial Intelligence*. Ed. by Wayne Wobcke and Mengjie Zhang. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 247–257. ISBN: 978-3-540-89378-3.
- Hawkins, Douglas M (2004). “The problem of overfitting”. In: *Journal of chemical information and computer sciences* 44.1, pp. 1–12.
- Ying, X. (2019). “An overview of overfitting and its solutions”. In: *Journal of physics: Conference series*. Vol. 1168. IOP Publishing, p. 022022.
- Lin, T. et al. (2018). “Leave-p-out cross-validation for parameter estimation of insurance loss models”. In: *Risks* 6.4, p. 121.
- Zhang, Y. and T.K. Siu (2019). “Leave-p-out cross-validation for claim count models with excess zeros”. In: *Insurance: Mathematics and Economics* 86, pp. 144–153.
- Bevilacqua, M., M. Braglia, and R. Montanari (2015). “The Monte Carlo cross-validation method for outlier detection in asset valuation”. In: *Journal of Applied Statistics* 42.2, pp. 225–240.
- Nigro, V. and G.A. Veltri (2020). “Monte Carlo cross-validation for evaluating risk in dynamic models”. In: *Journal of Banking and Finance* 118, p. 105972.
- Chen, Y., Q. Li, and T. Lin (2020). “Rolling-window approach to cross-validation in predictive modeling”. In: *Risks* 8.1, p. 20.
- Liao, W. et al. (2020). “Rolling-window cross-validation for credit scoring models”. In: *Applied Sciences* 10.12, p. 4204.
- Kajikawa, Y. and K. Yamasaki (2019). “Rolling cross-validation for hyperparameter tuning in credit scoring models”. In: *Journal of Risk and Financial Management* 12.1, p. 27.
- Rijsbergen, C. J. Van (1979). *Information Retrieval*. 2nd. Butterworth-Heinemann.
- Jaccard, P. (1912). “The distribution of the flora in the alpine zone”. In: *New Phytologist* 11.2, pp. 37–50. DOI: <https://doi.org/10.1111/j.1469-8137.1912.tb05611.x>.
- Henckaerts, R. and K. Antonio (July 2022). “The added value of dynamically updating motor insurance prices with telematics collected driving behavior data”. In: *Insurance: Mathematics*

- and Economics. DOI: [10.1016/j.insmatheco.2022.03.011](https://doi.org/10.1016/j.insmatheco.2022.03.011). URL: <https://hal.science/hal-04015750>.
- Wang, Q. et al. (2022). “A Comprehensive Survey of Loss Functions in Machine Learning”. In: *Annals of Data Science* 9.2, pp. 187–212. ISSN: 2198-5812. DOI: [10.1007/s40745-020-00253-5](https://doi.org/10.1007/s40745-020-00253-5).
- Henckaerts, R., M.P. Côté, et al. (2021). “Boosting Insights in Insurance Tariff Plans with Tree-Based Machine Learning Methods”. In: *North American Actuarial Journal* 25.2, pp. 255–285. DOI: [10.1080/10920277.2020.1745656](https://doi.org/10.1080/10920277.2020.1745656).
- Breiman, L. et al. (1984). *Classification and Regression Trees*. Chapman and Hall.
- Fielding, A. and C.A. O’Muircheartaigh (Mar. 1977). “Binary segmentation in survey analysis with particular reference to AID”. In: *Statistician* 26.1, p. 17.
- Ritschard, G. (2013). “CHAID and Earlier Supervised Tree Methods”. In: *Contemporary Issues in Exploratory Data Mining in Behavioral Sciences*, pp. 48–74.
- Loh, W.Y. (2014). “Fifty Years of Classification and Regression Trees”. In: *International Statistical Review* 82.3, pp. 329–348. DOI: <https://doi.org/10.1111/insr.12016>.
- Belson, W.A. (1959). “Matching and Prediction on the Principle of Biological Classification”. In: *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 8.2, pp. 65–75. ISSN: 00359254, 14679876. URL: <http://www.jstor.org/stable/2985543> (visited on 10/31/2023).
- Morgan, J.N. and J.A. Sonquist (1963). “Problems in the Analysis of Survey Data, and a Proposal”. In: *Journal of the American Statistical Association* 58.302, pp. 415–434. ISSN: 01621459. URL: <http://www.jstor.org/stable/2283276> (visited on 10/31/2023).
- Cellard, J.C., B. Labbe, and G. Savitsky (1967). “Le programme Elisee, presentation et application”. In: *METRA*.
- Sonquist, J.A. et al. (1971). *Searching for structure (alias-AID-III) : an approach to analysis of substantial bodies of micro-data and documentation for a computer program (successor to the Automatic Interaction Detector Program)*. Institute for Social Research, University of Michigan. URL: <https://cir.nii.ac.jp/crid/1130282272171994240>.
- Sonquist, J.A. (Jan. 1969). “Finding variables that work”. In: *Public Opinion Quarterly* 33.1, pp. 83–95. ISSN: 0033-362X. DOI: [10.1086/267669](https://doi.org/10.1086/267669). eprint: <https://academic.oup.com/poq/article-pdf/33/1/83/5270575/33-1-83.pdf>.
- Thompson, V.R. (Dec. 2018). “Sequential Dichotomisation: Two Techniques”. In: *Journal of the Royal Statistical Society Series D: The Statistician* 21.3, pp. 181–194. ISSN: 2515-7884. DOI: [10.2307/2986681](https://doi.org/10.2307/2986681).
- Bouroche, J.M. and Tenenhaus, M. (1970). “Quelques méthodes de segmentation”. In: *R.I.R.O.* 4, pp. 29–42. DOI: [10.1051/ro/197004V200291](https://doi.org/10.1051/ro/197004V200291).
- Press, L.I., M.S. Rogers, and G.H. Shure (1969). “An interactive technique for the analysis of multivariate data”. In: *Behavioral Science* 14.5, pp. 364–370. DOI: <https://doi.org/10.1002/bs.3830140504>.
- Messenger, R.C. and L. Mandell (1972). “A Modal Search Technique for Predictive Nominal Scale Multivariate Analysis”. In: *Journal of the American Statistical Association* 67.340, pp. 768–772. ISSN: 01621459. URL: <http://www.jstor.org/stable/2284634> (visited on 10/31/2023).
- Morgan, J.N. and R.C. Messenger (1973). *THAID, a Sequential Analysis Program for the Analysis of Nominal Scale Dependent Variables*. (Publications of the Institute for Social Research, University of Michigan. Survey Research Center series.) Survey Research Center, Institute for Social Research, University of Michigan. ISBN: 9780879441371. URL: <https://books.google.fr/books?id=rTtMAAAAMAAJ>.
- Gillo, M.W. (1972). *MAID, a Honeywell 600 program for an automatized survey analysis*.

- Gillo, M.W. and M.W. Shelly (1974). "Predictive modeling of multivariable and multivariate data". In: *Journal of the American Statistical Association* 69.347, pp. 646–653.
- Hunt, E.B., J. Marin, and P. J. Stone (1966). *Experiments in induction*. Academic Press.
- Kass, G.V. (1975). "Significance Testing in Automatic Interaction Detection (A.I.D.)" In: *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 24.2, pp. 178–189. ISSN: 00359254, 14679876. URL: <http://www.jstor.org/stable/2346565> (visited on 10/31/2023).
- Scott, A.J. and M. Knott (1976). "An Approximate Test for Use with AID". In: *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 25.2, pp. 103–106. ISSN: 00359254, 14679876. URL: <http://www.jstor.org/stable/2346677> (visited on 10/31/2023).
- Armstrong, J.S. and J.G. Address (1970). "Exploratory Analysis of Marketing Data: Trees vs. Regression". In: *Journal of Marketing Research* 7.4, pp. 487–492. DOI: [10.1177/002224377000700408](https://doi.org/10.1177/002224377000700408).
- Assael, H (1970). "Segmenting Markets by Group Purchasing Behavior: An Application of the AID Technique". In: *Journal of Marketing Research* 7.2, pp. 153–158. DOI: [10.1177/002224377000700201](https://doi.org/10.1177/002224377000700201).
- Kass, G.V. (1980). "An Exploratory Technique for Investigating Large Quantities of Categorical Data". In: *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 29.2, pp. 119–127. ISSN: 00359254, 14679876. URL: <http://www.jstor.org/stable/2986296> (visited on 10/31/2023).
- Onn, K.P. and A. Mercer (1998). "The direct marketing of insurance". In: *European Journal of Operational Research* 109.3, pp. 541–549. ISSN: 0377-2217. DOI: [https://doi.org/10.1016/S0377-2217\(98\)00024-1](https://doi.org/10.1016/S0377-2217(98)00024-1). URL: <https://www.sciencedirect.com/science/article/pii/S0377221798000241>.
- Almana, A.M. (2014). "A Survey On Data Mining Techniques In Customer Churn Analysis For Telecom Industry". In: *Int. Journal of Engineering Research and Applications* 4.5, pp. 165–171. ISSN: 2248-9622.
- Therneau, T., B. Atkinson, and B. Ripley (1999). *rpart: Recursive Partitioning and Regression Trees*. Recursive partitioning for classification, regression and survival trees. An implementation of most of the functionality of the 1984 book by Breiman, Friedman, Olshen and Stone.
- Quinlan, J.R. (1986). "Induction of decision trees". In: *Machine Learning* 1.1, pp. 81–106. ISSN: 1573-0565. DOI: [10.1007/BF00116251](https://doi.org/10.1007/BF00116251).
- (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann series in machine learning. Elsevier Science. ISBN: 9781558602380. URL: <https://books.google.fr/books?id=HExnpcjbyroC>.
- Lim, T.S., W.Y. Loh, and Y.S. Shih (2000). "A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-Three Old and New Classification Algorithms". In: *Machine Learning* 40, pp. 203–228. URL: <https://api.semanticscholar.org/CorpusID:17030953>.
- Loh, W.Y. (2009). "Improving the precision of classification trees". In: *The Annals of Applied Statistics* 3.4. DOI: [10.1214/09-aos260](https://doi.org/10.1214/09-aos260).
- Quinlan, J.R. (1992). "Learning With Continuous Classes". In: URL: <https://api.semanticscholar.org/CorpusID:1056674>.
- White, A.P. and W.Z. Liu (1994). "Technical Note: Bias in Information-Based Measures in Decision Tree Induction". In: *Machine Learning* 15, pp. 321–329. URL: <https://api.semanticscholar.org/CorpusID:20532511>.
- Kononenko, I. (1995). "On Biases in Estimating Multi-Valued Attributes". In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*. IJCAI'95. Montreal, Quebec, Canada: Morgan Kaufmann Publishers Inc., pp. 1034–1040. ISBN: 1558603638.
- Loh, W.Y. and Y.S. Shih (1997). "Split selection methods for classification trees". In: *Statistica Sinica* 7.4, pp. 815–840. ISSN: 10170405, 19968507. URL: <http://www.jstor.org/stable/24306157> (visited on 10/31/2023).

- Kim, H. and W.Y. Loh (2003). “Classification Trees With Bivariate Linear Discriminant Node Models”. In: *Journal of Computational and Graphical Statistics* 12, pp. 512–530. URL: <https://api.semanticscholar.org/CorpusID:1915650>.
- Chan, K.Y. and W.Y. Loh (2004). “LOTUS: An Algorithm for Building Accurate and Comprehensible Logistic Regression Trees”. In: *Journal of Computational and Graphical Statistics* 13.4, pp. 826–852. DOI: [10.1198/106186004X13064](https://doi.org/10.1198/106186004X13064).
- Loh, W.Y. (Feb. 2001). “Classification Trees With Unbiased Multiway Splits”. In: *Journal of the American Statistical Association* 96, pp. 589–604. DOI: [10.1198/016214501753168271](https://doi.org/10.1198/016214501753168271).
- Chaudhuri, P. et al. (1994). “Piecewise-polynomial regression trees”. In: *Statistica Sinica* 4.1, pp. 143–167. ISSN: 10170405, 19968507. URL: <http://www.jstor.org/stable/24305278> (visited on 10/31/2023).
- Loh, W.Y. (Apr. 2002). “Regression Trees With Unbiased Variable Selection and Interaction Detection”. In: *Statistica Sinica* 12, pp. 361–386.
- Hothorn, T., K. Hornik, and A. Zeileis (2006). “Unbiased Recursive Partitioning: A Conditional Inference Framework”. In: *Journal of Computational and Graphical Statistics* 15.3, pp. 651–674. DOI: [10.1198/106186006X133933](https://doi.org/10.1198/106186006X133933).
- Zeileis, A., T. Hothorn, and K. Hornik (2008). “Model-Based Recursive Partitioning”. In: *Journal of Computational and Graphical Statistics* 17.2, pp. 492–514. DOI: [10.1198/106186008X319331](https://doi.org/10.1198/106186008X319331).
- Dutang, C. and Q. Guibert (2021). “An explicit split point procedure in model-based trees allowing for a quick fitting of GLM trees and GLM forests”. In: *Statistics and Computing* 32.1, p. 6. ISSN: 1573-1375. DOI: [10.1007/s11222-021-10059-x](https://doi.org/10.1007/s11222-021-10059-x).
- Schlosser, L., T. Hothorn, and A. Zeileis (2019). *The Power of Unbiased Recursive Partitioning: A Unifying View of CTree, MOB, and GUIDE*. arXiv: [1906.10179 \[stat.ME\]](https://arxiv.org/abs/1906.10179).
- Lakshminarayanan, B. (2016). “Decision Trees and Forests: A Probabilistic Perspective”. PhD thesis. University College London.
- Taddy, M.A., R.B. Gramacy, and N.G. Polson (2011). “Dynamic Trees for Learning and Design”. In: *Journal of the American Statistical Association* 106.493, pp. 109–123. DOI: [10.1198/jasa.2011.ap09769](https://doi.org/10.1198/jasa.2011.ap09769).
- Berk, R.A. (2006). “An introduction to ensemble methods for data analysis”. In: *Sociological methods & research* 34.3, pp. 263–295.
- Strobl, C., J. Malley, and G. Tutz (Dec. 2009). “An Introduction to Recursive Partitioning: Rationale, Application, and Characteristics of Classification and Regression Trees, Bagging, and Random Forests”. In: *Psychological methods* 14, pp. 323–48. DOI: [10.1037/a0016973](https://doi.org/10.1037/a0016973).
- Breiman, L. (2001). “Random Forests”. English. In: *Machine Learning* 45.1, pp. 5–32. ISSN: 0885-6125. DOI: [10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324).
- Lee, S.J. et al. (2018). “A novel bagging C4.5 algorithm based on wrapper feature selection for supporting wise clinical decision making”. In: *Journal of Biomedical Informatics* 78, pp. 144–155. ISSN: 1532-0464. DOI: <https://doi.org/10.1016/j.jbi.2017.11.005>. URL: <https://www.sciencedirect.com/science/article/pii/S1532046417302423>.
- Hothorn, T., K. Hornik, C. Strobl, et al. (2010). *Party: A laboratory for recursive partytioning*.
- Garge, N., G. Bobashev, and B. Eggleston (Apr. 2013). “Random forest methodology for model-based recursive partitioning: The mobForest package for R”. In: *BMC bioinformatics* 14, p. 125. DOI: [10.1186/1471-2105-14-125](https://doi.org/10.1186/1471-2105-14-125).
- Freund, Y. and R.E. Schapire (1996). “Experiments with a New Boosting Algorithm”. In: *International Conference on Machine Learning*, pp. 148–156. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.6252>.
- Breiman, L (1996). “Technical Note: Some Properties of Splitting Criteria”. In: *Machine Learning* 24.1, pp. 41–47. ISSN: 1573-0565. DOI: [10.1023/A:1018094028462](https://doi.org/10.1023/A:1018094028462).

- Shih, Y.S. (1999). “Families of splitting criteria for classification trees”. In: *Statistics and Computing* 9.4, pp. 309–315. ISSN: 1573-1375. DOI: [10.1023/A:1008920224518](https://doi.org/10.1023/A:1008920224518).
- Zhang, T. (2004). “Statistical Behavior and Consistency of Classification Methods Based on Convex Risk Minimization”. In: *The Annals of Statistics* 32.1, pp. 56–85. ISSN: 00905364. URL: <http://www.jstor.org/stable/3448494> (visited on 07/10/2023).
- Devroye, L., L. Györfi, and G. Lugosi (1996). *A Probabilistic Theory of Pattern Recognition*. Vol. 31. Stochastic Modelling and Applied Probability. Springer, pp. 1–638. ISBN: 978-1-4612-0711-5.
- Klusowski, Jason M. and Peter M. Tian (2023). *Large Scale Prediction with Decision Trees*. arXiv: [2104.13881](https://arxiv.org/abs/2104.13881) [stat.ML].
- Biau, G., L. Devroye, and G. Lugosi (2008). “Consistency of Random Forests and Other Averaging Classifiers”. In: *J. Mach. Learn. Res.* 9, pp. 2015–2033. ISSN: 1532-4435.
- Biau, G. (2012). “Analysis of a Random Forests Model”. In: *J. Mach. Learn. Res.* 13, pp. 1063–1095. ISSN: 1532-4435.
- Biau, G. and B. Cadre (2017). *Optimization by gradient boosting*. arXiv: [1707.05023](https://arxiv.org/abs/1707.05023) [math.ST].
- Scornet, E., G. Biau, and J.P. Vert (2015). “Consistency of random forests”. In: *The Annals of Statistics* 43.4, pp. 1716–1741. DOI: [10.1214/15-AOS1321](https://doi.org/10.1214/15-AOS1321).
- Scornet, E. (2016). “On the Asymptotics of Random Forests”. In: *J. Multivar. Anal.* 146.C, pp. 72–83. ISSN: 0047-259X. DOI: [10.1016/j.jmva.2015.06.009](https://doi.org/10.1016/j.jmva.2015.06.009).
- Devaux, A. (Nov. 2022). “Modélisation et prédiction dynamique individuelle d’événements de santé à partir de données longitudinales multivariées”. Theses. Université de Bordeaux. URL: <https://theses.hal.science/tel-03909257>.
- Bou-Hamad, I., D. Larocque, and H. Ben-Ameur (2011). “A review of survival trees”. In: *Statistics Surveys* 5, pp. 44–71. DOI: [10.1214/09-SS047](https://doi.org/10.1214/09-SS047).
- Bertsimas, D. et al. (2022). “Optimal Survival Trees”. In: *Machine Learning* abs/2012.04284.
- Molinaro, A.M., S. Dudoit, and M.J. van der Laan (2004). “Tree-based multivariate regression and density estimation with right-censored data”. In: *Journal of Multivariate Analysis* 90, pp. 154–177.
- Vock, D.M. et al. (June 2016). “Adapting machine learning techniques to censored time-to-event health record data: A general-purpose approach using inverse probability of censoring weighting”. In: *J. Biomed. Inform.* 61, pp. 119–131.
- Mantel, N. (1966). “Evaluation of survival data and two new rank order statistics arising in its consideration”. In: *Cancer Chemotherapy Reports. Part 1* 50, pp. 163–170.
- LeBlanc, M. and J. Crowley (1992). “Relative Risk Trees for Censored Survival Data”. In: *Biometrics* 48.2, pp. 411–425. ISSN: 0006341X, 15410420. URL: <http://www.jstor.org/stable/2532300> (visited on 11/01/2023).
- Gordon, L. and R.A. Olshen (1985). “Tree-structured survival analysis.” In: *Cancer treatment reports* 69 10, pp. 1065–9. URL: <https://api.semanticscholar.org/CorpusID:38624647>.
- Ciampi, A., S.A. Hogg, and L. Kates (1986). “Regression analysis of censored survival data with the generalized F family—an alternative to the proportional hazards model”. In: *Statistics in Medicine* 5.1, pp. 85–96. DOI: <https://doi.org/10.1002/sim.4780050111>.
- Ishwaran, H. et al. (2008). “Random survival forests”. In: *The Annals of Applied Statistics* 2.3, pp. 841–860. DOI: [10.1214/08-AOAS169](https://doi.org/10.1214/08-AOAS169).
- Brier, G.W. (1950). “Verification of forecasts expressed in terms of probability”. In: *Monthly Weather Review* 78.1, pp. 1–3. DOI: [https://doi.org/10.1175/1520-0493\(1950\)078<0001:VOFEIT>2.0.CO;2](https://doi.org/10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2). URL: https://journals.ametsoc.org/view/journals/mwre/78/1/1520-0493_1950_078_0001_vofeit_2_0_co_2.xml.

- Harrell Jr, F E et al. (May 1982). "Evaluating the yield of medical tests". en. In: *JAMA* 247.18, pp. 2543–2546.
- Uno, H. et al. (2011). "On the C-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data". In: *Statistics in Medicine* 30.10, pp. 1105–1117. doi: <https://doi.org/10.1002/sim.4154>.
- Hartman, N. et al. (2023). "Pitfalls of the concordance index for survival outcomes". In: *Statistics in Medicine* 42.13, pp. 2179–2190. doi: <https://doi.org/10.1002/sim.9717>.
- Lambert, J. and S. Chevret (2016). "Summary measure of discrimination in survival models based on cumulative/dynamic time-dependent ROC curves". In: *Statistical Methods in Medical Research* 25.5. PMID: 24395866, pp. 2088–2102. doi: [10.1177/0962280213515571](https://doi.org/10.1177/0962280213515571).

Part III

Lapse management strategy

Chapter 6 Contributions of Part III

Chapter 7 The Customer Lifetime Value and its insights for insurance

Chapter 8 Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies

- 8.1 Introduction 71
- 8.2 Data 74
- 8.3 Framework 76
- 8.4 Methodology 79
 - 8.4.1 Step 1: Modeling $r_{\text{acceptant}}$ and r_{lapser} 80
 - 8.4.2 Step 2: Classification tasks 83
- 8.5 Real-life application 85
 - 8.5.1 Considered lapse management strategies 85
 - 8.5.2 Numerical results 86
 - 8.5.3 Comments 86
- 8.6 Discussion 89
 - 8.6.1 General statements 89
 - 8.6.2 Marketing decision making 90
 - 8.6.3 Management rules decision making 91
- 8.7 Conclusion and perspectives 93
- 8.8 Suggestions for future work 94
 - 8.8.1 $z^{(i)}$ efficiency border 94
 - 8.8.2 Other improvements 97

Bibliography

6. Contributions of Part III

This part is mainly based on the article “ Including Customer Lifetime Value in tree-based lapse management strategy ”, written in collaboration with Xavier Milhaud and Anani Olympio and published in the European Actuarial Journal¹. This work represents a contribution to the interwoven domains of actuarial science, management science, and business economics. If Parts I and II were quite encyclopedic - yet necessary to understand and contextualize the rest of the thesis -, Part III is more of an applied work. This research sheds light on critical aspects of lapse risk assessment and strategic decision-making. This part unveils novel methodologies and empirical evidence that offer insightful implications for both academia and industry. A list of various contributions can be found below:

Contributions 1: Individualized CLV with competing risks

Evaluating existing theories or models and proposing improvements or alternatives

The first significant contribution of the article is the development of a new model for the individualized future Customer Lifetime Value. This model takes into account the risks of lapse and death with a new survival approach for which they are treated as mutually exclusive competing risks. It considers both parametric approaches like Cox cause-specific and subdistribution models, and tree-based survival models like Random Survival Forest and Gradient boosting survival analysis. This is central for a customer-centered and profit-driven decision-making process.

Contributions 2: Using RSF and GBSM for lapse analysis

Application of existing theories or methods in a new context

Another contribution is the novel application of complex tree-based survival models, including random survival forest (RSF) and gradient boosting survival model (GBSM). This innovative use of these models in an actuarial context opens up new avenues for research and testing in this field, providing new perspectives and insights. In the context of this Part's study, applying the gradient boosting survival model within an actuarial context for the first time, allowed for the development of more accurate and individualized customer lifetime value predictions.

¹See Valla, Milhaud, and Olympio 2023.

Contributions 3: Development of a new LMS framework

Development of a new theoretical framework

A third contribution is the establishment of a new lapse management strategy framework. This part details a two-step lapse management modeling approach: we fit parametric and tree-based competing risk individual survival models to estimate individualized future CLVs that are part of an evaluation metric for tree-based lapse management models. This framework not only predicts lapses but also focuses on maximizing profitability and customer lifetime value, offering a more comprehensive approach to lapse management.

Contributions 4: Business-oriented discussion

Discussion of new empirical results

The final contribution is the business-oriented discussion of the new empirical results achieved by this framework. This discussion adds a practical dimension usually missing in similar research, showing the real-world benefits of the model in terms of commercial and strategic decision-making for life insurers

7. The Customer Lifetime Value and its insights for insurance

“Ultimately, marketing is the art of attracting and keeping profitable customers (Kotler 1996). A company should not try to pursue and satisfy every customer. Kotler and Armstrong define a profitable customer as “a [subject] whose revenues over time exceed, [...], the company costs of attracting, selling, and servicing that customer.” This excess is called customer lifetime value (CLV).” Berger and Nasr 1998

CLV is a notion that reflects the net present value of a customer. It serves as an indicator of the total revenue a business can reasonably expect to be generated by an individual customer, considering the difference between the revenue the company can earn from a customer and the company’s predicted expenses for acquiring and servicing that customer, over the lifetime of the business relationship. CLV is an essential metric as it enables businesses to understand the economic value, or profit, generated by customers over their lifetime. It allows companies to identify high-value customers, optimize customer acquisition costs, and enhance the effectiveness of cross-selling strategies. From a strategic marketing perspective, understanding Customer Lifetime Value is crucial for companies operating in various sectors. CLV is a concept rooted in marketing and management science, which became a crucial metric for the insurance industry, where customer relationships are long-term, and customer retention is of paramount importance. This brief literature review will delve into the essence of CLV, its emergence in the field of marketing and management science, and its subsequent use in the insurance sector, particularly regarding retention strategy.

The concept of CLV emerged in the field of marketing and has been substantially studied in the late 1980s (Dwyer 1989) and 1990s (Wang and Splegel 1994; Keane and Wang 1995). It was introduced as a metric to measure the profitability of customers over their entire relationship with a company. In management science, the use of CLV became more sophisticated with the advent of advanced analytical techniques. Various models were proposed to predict CLV, including regression models, duration models, and models incorporating customer-specific retention probabilities. These models allowed for a more nuanced understanding of CLV, considering factors such as cross-selling potential, customer satisfaction, and the likelihood of customer defection.

In all generality, the CLV of a subject i aims at capturing the expected profit or loss that will be generated over the duration $T^{(i)}$ of her relationship and is expressed as follows, in the general time-continuous case:

$$CLV^{(i)} = \int_{\tau=0}^{T^{(i)}} \frac{(Revenues^{(i)}(\tau) - Expenses^{(i)}(\tau))}{e^{d(\tau) \cdot \tau}} d\tau, \quad (7.1)$$

with $d(\tau)$, the discount rate. Usually, several simplifications are made to ensure this formula is tractable. First, its temporal dynamic is discretized, considering for instance yearly-cumulated revenues and expenses. Then, the duration part of the equation $T^{(i)}$ is replaced by a fixed time horizon T , and the annual probability that subject i is still a customer at every year τ . Thus, Equation 7.1 becomes

$$CLV^{(i)} = \sum_{\tau=0}^T \frac{(R_{\tau}^{(i)} - E_{\tau}^{(i)}) \cdot r_{\tau}^{(i)}}{(1 + d_{\tau})^{\tau}}, \quad (7.2)$$

with R and E the yearly revenues and expenses, $r_{\tau}^{(i)}$ and d_{τ} the yearly retention probability and discount rate, respectively. Obviously, the CLV can be split into two distinct parts: the past (or observed) CLV (${}^PCLV^{(i)}$) and the future CLV (${}^FCLV^{(i)}$).

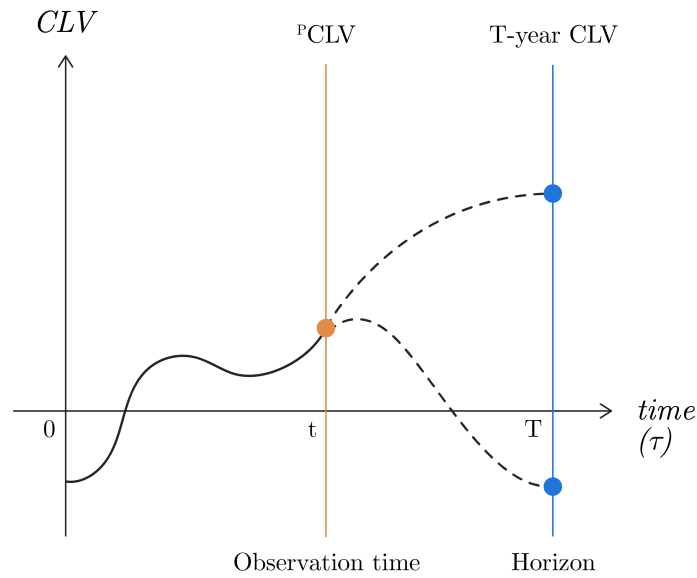


Figure 7.1: Past and future CLV

At any evaluation time t , the past part already happened and is fully determined by the financial flows that were observed during the first t years of the relationship. Thus, we can divide the formula for CLV as such:

$$CLV^{(i)}(t) = \underbrace{\sum_{\tau=0}^t \frac{(R_{\tau}^{(i)} - E_{\tau}^{(i)})}{(1 + d_{\tau})^{\tau}}}_{{}^PCLV^{(i)}} + \underbrace{\sum_{\tau=t+1}^T \frac{(\mathbb{E}[R_{\tau}^{(i)}] - \mathbb{E}[E_{\tau}^{(i)}]) \cdot r_{\tau}^{(i)}}{(1 + \hat{d}_{\tau})^{\tau}}}_{{}^FCLV^{(i)}}. \quad (7.3)$$

Evaluating the future part is usually more difficult as it implies, depending on the type of business considered, projection models for the revenue, expenses, and retention probabilities (see Fader, Hardie, and Lee 2005). The insurance sector, with its focus on long-term customer relationships and the significant costs associated with customer acquisition and servicing, is an ideal context for the application of CLV. Despite its advantages, implementing CLV in the insurance sector is not without challenges. These include the complexity of predicting future customer behavior and the need for extensive data on customer interactions. The difficulty of estimating the CLV of a policyholder is threefold.

First, the revenues are the premiums for the insurance coverage, which may vary for various reasons. The policyholder (PH) can choose to increase or decrease her level of coverage (for life insurance for instance), change the object of the insurance (for auto or home insurance for instance), or add beneficiaries. Other elements such as new regulations, market fluctuations, the probability of upselling or cross-selling in a multi-product company, or even results and internal decisions from the insurer (via profit-sharing or fees for instance) can also influence the amount of the expected future revenues. For life insurance, the revenues are constituted of the sum of payments and profit-sharings realized in one's policy.

Then, the expenses are, on the one hand, the acquisition and management costs and all the activity-based costs that can be anticipated, and on the other hand, the claims made. For the typical insurance product, the annual claim amount (the cost of health expenses, car reparation, theft, natural catastrophes, depending on the type of insurance) is unknown and must be modeled. For life insurance, however, the amount of the claims when ending the policy can be anticipated as it is equal to the face amount of the policy. However, it is the amount of the face amount that needs to be modeled. The PH can voluntarily decrease the amount of her coverage, which is known as a partial lapse. By estimating the occurrences and amounts of both payments and partial lapses, we derive the expected individual face amounts. And by predicting the difference between the guaranteed rate and the insurer's profitability rate, we can estimate the difference between the expected revenues and the expected expenses over time.

Eventually, the retention probabilities are the only remaining source of uncertainty for predicting ${}^FCLV^{(i)}$ once the revenues and expenses can be predicted. In all generality, $r_t^{(i)}$ represents the probability for subject i , to still have an active policy at year t . For most insurance products, it would correspond to the probability that the PH has not churned at year t . In a life insurance context, one's policy can only end with the death of the PH or the complete lapse of the policy. Thus $r_t^{(i)}$ corresponds to the probability that policyholder i has not died or lapsed her policy before year t .

The temporal dynamics of each of these elements must be studied in order to predict accurately the future CLV. In the application of this part of the thesis, we will focus on the temporal analysis of the retention probabilities, estimated with survival models.

Insurance companies have recognized the relevance of CLV in shaping their customer acquisition and retention strategies (Donkers, Verhoef, and Jong 2007; Loisel, Piette, and C.H.J. Tsai 2021). By identifying high-value customers and understanding their behavior, insurance companies can develop strategies to increase their retention rates. This involves offering tailored products and services, improving customer care, and increasing efforts to recover high-value customers who may be at risk of defection. All the potential impacts that CLV-driven management strategies can bring to the insurance industry have been reported in the work of Seyerle 2001, and are depicted in Figure 7.2.

Advanced models for predicting CLV in the insurance sector include probabilistic models, duration models, and multivariate models. These models consider factors such as customer retention, cross-selling, and service usage. They provide a more comprehensive and accurate prediction of CLV, allowing insurance companies to make informed decisions about customer acquisition and retention. High-value customers can be targeted with personalized offers and superior customer service to increase their loyalty and reduce the likelihood of defection. CLV can also help insur-

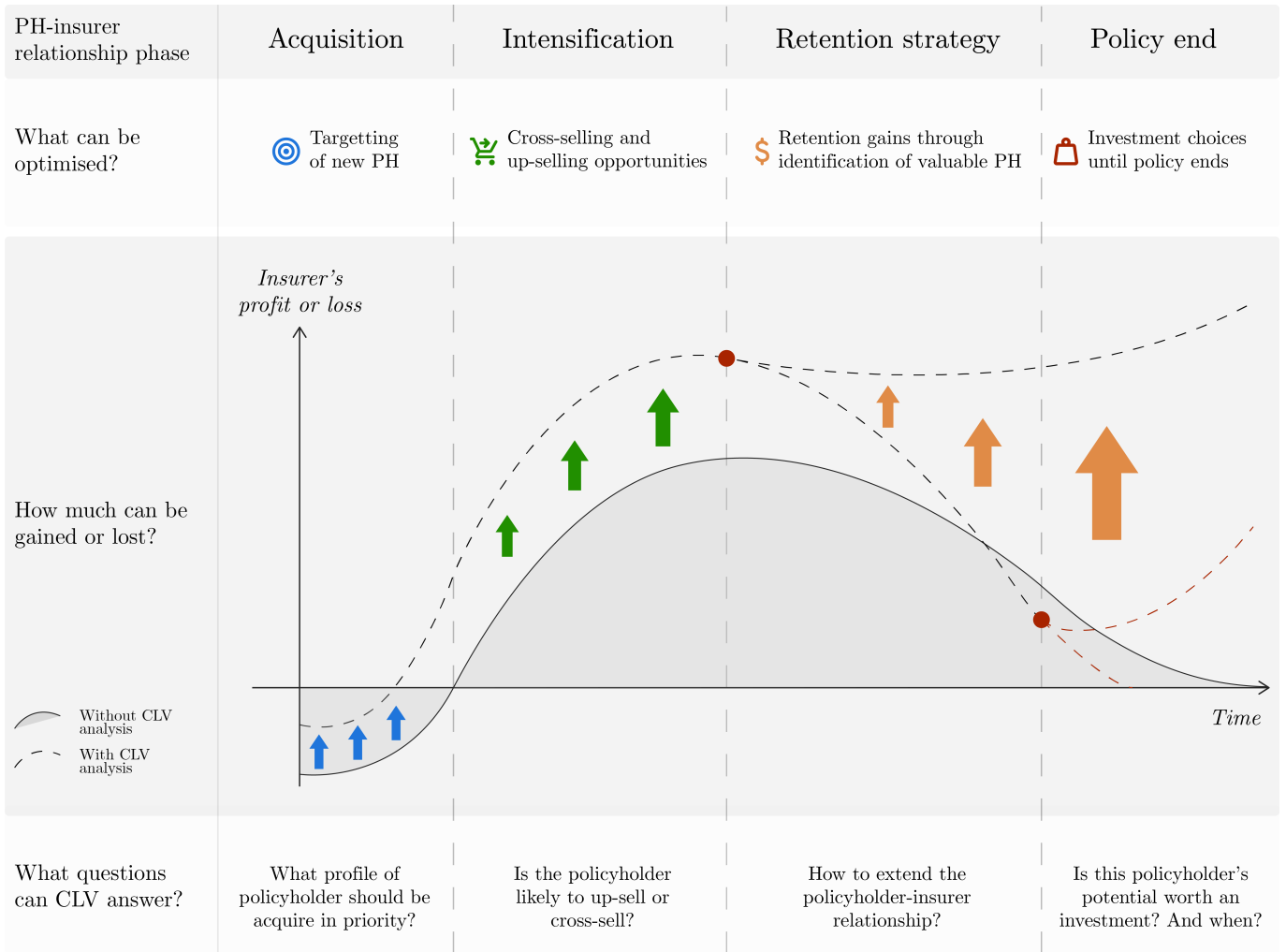


Figure 7.2: The potential of CLV in the insurance industry

ance companies optimize their cross-selling opportunities (Desirena et al. 2019). By understanding the cross-selling potential of their customers, they can offer additional products and services that are likely to appeal to these customers, thereby increasing their lifetime value.

In regard to the marketing literature about CLV, we found that the future CLV should be central in the design of any lapse management methodology in life insurance. More precisely, we suggest in this part of the thesis, the inclusion of an individual Customer Lifetime Value within a customer-centered and profit-driven lapse management framework. We pay special attention to the time-to-event modelling part of this framework.

8. Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies

Abstract. *A retention strategy based on an enlightened lapse model is a powerful profitability lever for a life insurer. Some machine learning models are excellent at predicting lapse, but from the insurer’s perspective, predicting which policyholder is likely to lapse is not enough to design a retention strategy. In our paper, we define a lapse management framework with an appropriate validation metric based on Customer Lifetime Value and profitability.*

We include the risk of death in the study through competing risks considerations in parametric and tree-based models and show that further individualization of the existing approaches leads to increased performance. We show that survival tree-based models outperform parametric approaches and that the actuarial literature can significantly benefit from them. Then, we compare, on real data, how this framework leads to increased predicted gains for a life insurer and discuss the benefits of our model in terms of commercial and strategic decision-making.

Key words: *Machine Learning, Life insurance, Customer lifetime value, Lapse, Lapse management strategy, Competing risks, Tree-based models*

8.1 Introduction

In life insurance, “lapse risk” or “persistency risk” is the risk that the policyholder will cancel the contract at a time other than when the issuer expected when pricing the contract (KPMG 2020). A life insurance policy can lapse if the policyholder stops paying the premiums required to keep the policy in force. This can happen if the policyholder becomes unable or unwilling to make the premium payments or if the policyholder chooses to surrender the policy for its cash value. When a policy lapses, the coverage and benefits the policy provides are no longer in effect, and the policyholder will not receive any payout if they pass away after the policy has lapsed. This risk is not considered an insurance risk because the payment to the policyholder “is not contingent on an uncertain future event that adversely affects the policyholder”. However, lapse management is still undoubtedly a primary concern for life insurers. Lapses may substantially affect a company’s solvency, its future profits and cash flows (Buchardt 2014; Buchardt, Moller, and Schmidt 2015) or its Asset and Liabilities Management (ALM) (Kim 2005; Gatzert and Schmeiser 2008; Eling and Kochanski 2013; Eling and Kiesenbauer 2014). The importance of measuring lapse and churn behaviours is global; it goes from yielding individual estimations of the Customer Lifetime Values (CLV) to being an estimator of a firm’s profitability (Gupta and

Lehmann 2006; Gupta 2009) or strength (Ascarza et al. 2018). Therefore, this paper focuses on developing strategies to prevent lapses before they occur: for a life insurer, an enlightened and proactive lapse management strategy (LMS) is critical for successful monitoring and steering. This paper is about defining a framework for a life insurer to measure and optimise the future loss or profit to be expected when applying such an LMS.

Part of the literature on lapse management adopts an economic-centred point of view (Dar and Dodds 1989; Kuo, C. Tsai, and W. Chen 2003; Kagraoka 2005; S. Cox and Y. Lin 2006; Kiesenbauer 2012; Russell et al. 2013; Sirak 2015; Vasudev, Bajaj, and Escolano 2016; Nolte and Schneider 2017; Poufinas and Michaelide 2018; Yu, Cheng, and T. Lin 2019); we refer the reader to the complete bibliometric analysis on this topic by Shamsuddin, Noriszura, and Roslan 2022 for a summarised view of all these references. This economic-centred research aims to determine lapse factors like interest rates, gross domestic product, or unemployment rates. They are driven by economic hypotheses such as the emergency fund hypothesis (lapsing is a way of constituting an emergency fund), the policy replacement hypothesis (lapsing will occur when one changes its policy) or the interest rate hypothesis (lapsing depends strongly on rate change and arbitration).

A large part of the literature, however, investigates the individual determinants of lapse with policyholder-centred approaches. Micro-oriented features such as policyholder's personal information or the policy characteristics have shown to give valuable insights into lapse behavior (Renshaw and Haberman 1986; Milhaud, Loisel, and Maume-Deschamps 2011; Eling and Kiesenbauer 2014; Hwang, Chan, and J. Tsai 2022). Ćurak, Podrug, and Poposki 2015 as well as Gemmo and Gotz 2016's works indicate that policyholders' features such as age and the number of beneficiaries are significant lapse factors, whereas Sirak 2015 dismissed those results. A recent work from Loisel, Piette, and C.H.J. Tsai 2021 proposes a comparison of lapse management strategies based on an innovative evaluation metric derived from the Customer Lifetime Value (CLV). Hu et al. 2021 investigates the benefits of incorporating spatial analysis in lapse modeling, and Azzone et al. 2022 shows, with an approach based on random forests, that microeconomic features such as the company's commercial approach are determining in the lapse decision. In contrast, macro-economical features only have a limited effect. This variety of results – sometimes contradicting each other – demonstrates the active interest in this research problem.

This paper focuses on lapse management strategy and retention targeting and will contribute to the existing literature on the relationship between retention strategy and lapse prediction: as in Ascarza et al. 2018 and Loisel, Piette, and C.H.J. Tsai 2021, our goal is not only to model the lapse behavior but rather to select policyholders that are expected to generate future profit, if targeted by a retention strategy. This work shows that a well-chosen strategy, based on individualized CLV and directed towards a well-chosen target, increases the insurer's expected profitability. A critical concept that motivates many CLV-driven decisions is that customers should be judged as assets based on their future profitability for the insurer. Thus, since retention often serves as the basis for CLV models (Gupta, Hanssens, et al. 2006; Donkers, Verhoef, and Jong 2007; Lemmens and Gupta 2020 - sometimes specifically designed for targeting tasks (von Mutius and Huchzermeier 2021) - and since CLV considerations should drive retention management, it seems natural to extend the existing life insurance applications linking those topics together. We make decision-making a central concern of our work and suggest proactive lapse management tools allowing the insurer to undertake actions to prevent the causes of lapse; that is opposed to a reactive management approach where decisions are taken after lapses and aim at recapturing lost policyholders.

The goal of this paper is to create an individualized CLV model that will be used to enhance classical binary churn models. We will then have a model for lapse management strategy and retention targeting that we further improve with tree-based survival analysis and competing risks considerations. The global framework is directly inspired by Loisel, Piette, and C.H.J. Tsai [2021](#). We try in this paper to build from that existing work and extend it. We model an individual future CLV with a new survival approach for which the risks of death and lapse are treated as mutually exclusive competing risks. For this purpose, we introduce parametric approaches - Cox cause-specific and subdistribution models - as well as tree-based survival models - Random Survival Forest (RSF) and Gradient boosting survival analysis. In a second step, we use the individual CLV to design a binary outcome representing whether investing in retaining each subject is profitable or not. For that purpose, we also focus here on tree-based models as they are often considered state-of-the-art models (Grinsztajn, Oyallon, and Varoquaux [2022](#)). Thus we introduce tree-based machine learning algorithms for binary prediction, including classification and regression tree (CART), random forests (RF), and extreme gradient boosting (XGBoost) to lapse behavior modeling. CART and XGBoost (Milhaud, Loisel, and Maume-Deschamps [2011](#); Loisel, Piette, and C.H.J. Tsai [2021](#)) were used in the literature for lapse modeling but have yet to be applied to predicting life insurance lapses in a competing risk setting. To our knowledge, while random survival forest has been used for churn prediction recently (Routh, Roy, and Meyer [2021](#)), both RSF and gradient boosting survival analysis have never been used for that purpose before in an actuarial context.

Our contribution to the actuarial literature is twofold. First, we detail a two-step lapse management modeling approach: we fit parametric and tree-based competing risk individual survival models to estimate individualized future CLVs that are part of an evaluation metric for tree-based lapse management models. Second, this work includes a business-oriented discussion of the results achieved by this framework, which is missing from existing similar approaches.

The results and discussions show that a CLV-based lapse management strategy very often outperforms a more classical binary classification approach, even with competing risks and individualized considerations. When the latter yields profitable retention gain, the former can produce higher profits, up to more than 60%. If a loss-inducing retention strategy is considered, our methodology limits the loss considerably, often setting 0 as a floor limit or even turning it into a profit-inducing retention strategy. Sensitivity analysis explores the influence of conjectural and structural parameters.

The rest of this paper is structured as follows. We briefly outline the data used in our study in [Section 8.2](#). In [Section 8.3](#), we then introduce the binary classification models we selected and detail our study's methodology, describing the classical and CLV-based performance measures and discussing substantial parametrization improvements over existing approaches. Then, [Section 8.4](#) details our two-step methodology, with the parametric and non-parametric modelings of individual survival predictions, in a competing risks framework and then their implementation in the tree-based classification approaches considered. [Section 8.5](#) presents the real-life application we considered and the different results it produces. Those results are analyzed and discussed in [Section 8.6](#) with commercial and strategical decision-making orientations. Eventually, [Section 8.7](#) concludes this paper.

8.2 Data

We apply our framework to a real-world insurance portfolio. For privacy reasons, all the data, statistics, product names and perimeters presented in this paper have been either anonymized or modified. All analyses, discussions and conclusions remain unchanged.

We illustrate our methodology with a life insurance portfolio from a French insurer contracted between 1997 and 2018. Each record in the data set represents a unique policy for a unique policyholder. In the following sections, we will often refer to a unique pair of policy and policyholder by the term “subject”. The dataset contains 251,325 rows with 248,737 unique policies and 235,076 unique policyholders. It means that some policies are shared between several policyholders and that one individual can detain several insurance policies. The dataset contains 43 covariates described in Table 8.1.

Table 8.1: Data set description

Covariates (Numerical, Categorical, Date)		Description
ID	CDI_ID_PERSONNE	Policyholder (PH) unique ID
	CDI_ID_CONTRAT	Policy unique ID
PH-level information	CDI_DT_NAISSANCE	PH's birth date (main PH when several policyholders owns one policy)
	Age_souscription	PH's age at subscription
	Nb_Contrats	Number of different policies owned by the policyholder
	CDI_CD_SEXE	PH's gender (1=Female; 2=Male; other=Non precised)
	CDI_DESTINATAIRE_COURRIER	Anonymised PH's name
	CDI_NUM_ET_NOM_VOIE	Anonymised PH's address
	CDI_CD_POSTAL	Anonymised PH's postcode
	CDI_COMMUNE	Anonymised PH's city of residence
	CDI_TOP_ASSURE	Binary: 1 if PH is the main PH on the policy, 0 otherwise
	Policy-level informations	CDI_TYPE_PRODUIT
CDI_NOM_PRODUIT		Name of life insurance product (“Product 1”, “Product 2” or “Product 3”)
CDI_PARTENAIRE		Name of the insurance distributor
CDI_DATE_DEB_CONTRAT		Policy's start date
CDI_DATE_FIN_CONTRAT		Policy's end date
START_YEAR		Policy's start year
END_YEAR		Policy's end year
SENIORITY		Policy's seniority (final seniority if the policy is ended, current seniority otherwise)
STATE		Policy's state (“Active”, “Lapsed”, or “Death” if the policy ended following PH's death)
YEAR		Last year of observation
External data	DISCOUNT RATE	Discount rate corresponding to the last year of observation
	TOTAL_PREMIUM_AMOUNT	Total face amount of the policy
Policy's cumulated financial flows	TOTAL_EURO_PREMIUM_AMOUNT	Face amount of the policy in euros
	TOTAL_UC_PREMIUM_AMOUNT	Face amount of the policy in units of account
	ARBITRATION_EURO	Cumulated arbitration amount of the policy in euros
	ARBITRATION_UC	Cumulated arbitration amount of the policy in units of account
	FEES_EURO	Cumulated fees amount of the policy in euros
	FEES_UC	Cumulated fees amount of the policy in units of account
	OTHER_EURO	Cumulated other parts of the face amount of the policy in euros
	OTHER_UC	Cumulated other parts of the face amount of the policy in units of account
	PREMIUM_EURO	Cumulated payments amount of the policy in euros
	PREMIUM_UC	Cumulated payments amount of the policy in units of account
	PROFIT_SHARING_EURO	Cumulated profit sharing amount of the policy in euros
	PROFIT_SHARING_UC	Cumulated profit sharing amount of the policy in units of account
	CLAIM_EURO	Cumulated partial or total lapsed amount of the policy in euros
	CLAIM_UC	Cumulated partial or total lapsed amount of the policy in units of account
	Covariates derived from financial flows	%TOTAL_UC_PREMIUM_AMOUNT
%TOTAL_EURO_PREMIUM_AMOUNT		Percentage of the face amount in euros
%CLAIM_UC		Percentage of the face amount in units of account that was lapsed
%CLAIM_EURO		Percentage of the face amount in euros that was lapsed
Target covariate	%CLAIM	Percentage of the total face amount that was lapsed
	EVENT	Policy's state (0=Active, 1=Lapsed, 2 ended following PH's death)

The data set represents policies that are majority owned by men (57.4%) for a mean censored seniority time of 13.4 years. Three products are present in the dataset. Product one was chosen by 72% of policyholders, product 2 by 25% and product 3 by 3%.

Regarding their state, 61% of the policies are still active, 22% lapsed, and 17% ended after the PH's death. We chose here to present the distribution of the variable *SENIORITY* as it is the response variable in our survival models. Its modeling has a critical influence on CLV, thus, on our lapse

management strategy framework. We also chose to show the distribution of the variable *TOTAL PREMIUM AMOUNT* representing the most recent observed face amount for every subject, as it is a known determinant of lapse behavior. We are aware that this covariate is time-varying as its value is updated at every payment, total or partial lapse, profit sharing, arbitration, or even fee movement on a policy, and only considering its most recent value ignores a large part of the insights it can provide. Without any better option to account for its whole time-varying trajectory, we can only use *TOTAL PREMIUM AMOUNT* as it is and defer any dynamic considerations for future work.

The seniorities and most recent face amount recorded before the potential end of the policy are distributed as in Figure 8.1:

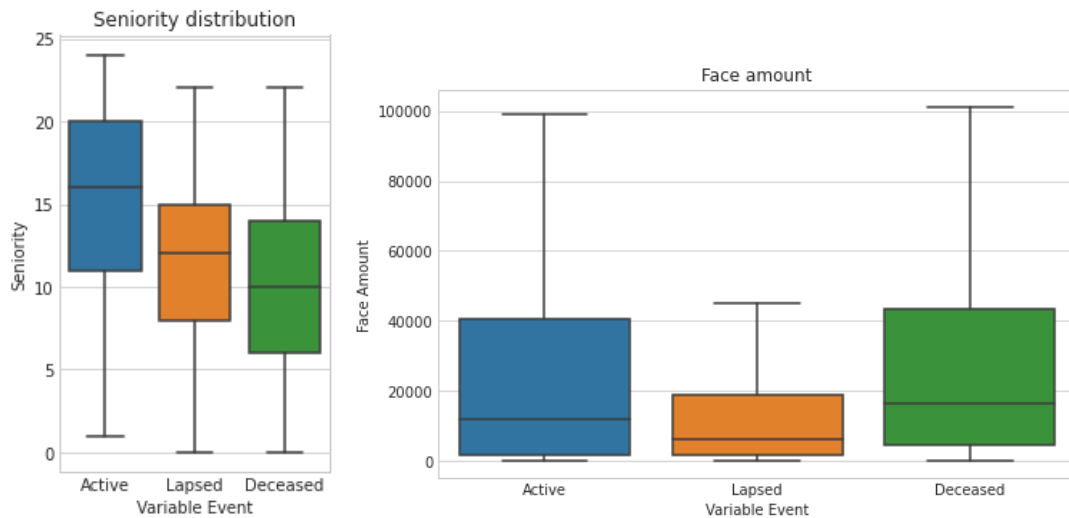


Figure 8.1: Seniorities and face amounts distributions

Without further analyzing the data, we can note several things. First, we can see that the mean censored seniority of 13.4 years is not equally distributed among our subjects. Active contracts tend to be older than lapsed ones, themselves older than policies that ended with the policyholder's death. That emphasizes the importance of several contributions, and the apparent difference in seniority regarding the cause of the policy's termination encourages a competing risks approach to analyze survival. Moreover, if we suspect lapse and death to be highly dependent on individual characteristics - such as the policyholder's age - this also supports an individualized survival analysis. Eventually, we can see that the last face amount observed is significantly lower for lapsed policies. It confirms our first intuitions and the face amount will be included in our model.

Among the covariates introduced in Table 8.1, several play a central part in our two-step modeling approach. First, the competing risks survival analysis step where *SENIORITY* will be the response variable, and all other covariates, including individual data and financial flows, are potential explanatory variables. The binary classification second step aims at predicting the *EVENT* outcome with minor transformations explained in Section 8.3 below. It is equivalent to using *STATE* as a target variable, as they are entirely similar. As a result, not all covariates are utilized and our predictions are solely based on the covariates underlined in Table 8.1 as they appeared to be of interest to insurers.

8.3 Framework

This section describes a modeling approach that follows Loisel, Piette, and C.H.J. Tsai 2021’s work. Our contributions place our work in a framework that differs from it by being only future-oriented, by a precise and individualized analysis of retention probabilities and by choosing a classification framework instead of regression. We chose to use a majority of their existing notations here.

Usual lapse management models based on classification aim to predict whether a policyholder will lapse. They may perform very well at that specific task, but it only reflects some aspects of this economic problem. Indeed, the literature is clear (Ascarza et al. 2018), and many policyholders may be predicted as “lapsers” but may not be profitable to the insurance company if targeted. In that case, keeping such policyholders would be irrational, and an efficient model should not predict them as targets. Targeting policyholders is an economic problem that requires an economic measure to assess. We propose to consider a measure based on the discounted expected profit of all the policies, in other words, the sum of all (CLVs). Optimizing a lapse, churn, or other prediction tasks with business-related measures is not new. However, to our knowledge, none of the existing approaches uses individualized future CLVs and models the profit of retention strategy by accounting for competing risks or using survival tree-based models.

CLV is a well-studied subject in marketing and business economics and has also been modeled in an insurance context. For a given subject i , her future CLV over horizon T can be modeled as

$${}^F CLV^{(i)}(\mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}^{(i)}, \mathbf{d}, T) = \sum_{t=0}^T \frac{p_t^{(i)} F_t^{(i)} r_t^{(i)}}{(1 + d_t^{(i)})^t}, \quad (8.1)$$

with t in years, $t = 0$ represents the last historical observation point for subject i . The quantity $p_t^{(i)}$ is her profitability ratio as a proportion of $F_t^{(i)}$, representing her face amount observed at time t . The quantity $r_t^{(i)}$ is the i -th subject’s probability of still being active at time t , and naturally, $d_t^{(i)}$ is the discount rate at time t , for subject i . We argue that both the profitability ratio and the discount rate should be as individualized as possible - either at the product or policy level - as ${}^F CLV^{(i)}$ reflects the individualized risk of policyholder i to the insurer. It is also worth mentioning that evaluating discount rates is well beyond the scope of this paper as it is complex and subject to significant judgment; for further details, we refer the astute reader to a variety of papers on the subject (Burrows and Lang 1997, Oh et al. 2018, Blum and Therond 2019).

It is worth pointing out that ${}^F CLV^{(i)}$ does not represent the global profit generated by subject i from her policy’s first year until time T as in Loisel, Piette, and C.H.J. Tsai 2021; it rather represents the future T years of profit. ${}^F CLV^{(i)}$ is not to be compared with the cash surrender value but rather with the fair market value (FMV) of the outstanding liabilities. The only difference with the latter is that ${}^F CLV^{(i)}$ is based on the insurer’s knowledge of its portfolio, thus computed with its own profitability and discount parameters rather than with market-consistent considerations. In our framework, the life insurer is more interested in maximizing its own realistic profitability rather than a sum of individual market values.

We suggest a model for the insurer’s estimated profit - or loss - resulting from a lapse management strategy (LMS). In order to do that, we will compare the expected value of the portfolio before and after applying a given strategy. We are aware that there could be infinite ways to

design a retention campaign: offering a punctual incentive, recurrent services or more profit sharing, for instance. Here, we define what we will consider an LMS.

Definition 1 (Lapse management strategy). *A T -years lapse management strategy is modeled by the offer of an incentive $\delta^{(i)}$ to subject i if she is targeted. The incentive is expressed as a percentage of her face amount and should not exceed the profitability ratio $p_t^{(i)}$, at any time point t . Contacting the targeted policyholder has a fixed cost c and after contact, the incentive is accepted with probability $\gamma^{(i)}$. A targeted subject who accepts the incentive will be considered as an “acceptant” who will never lapse. In our dataset, any subject that has never been observed to lapse is considered as an “acceptant” and her probability of being active at year $t \in [0, T]$ is denoted $r_{\text{acceptant}}^{(i)}(t)$. Conversely, a subject who refuses the incentive and prefers to lapse will be considered as a “lapser”. In our dataset, a subject is labeled as a lapser whenever she has been observed to lapse at year $t = 0$, and her probability of being active at year t is denoted $r_{\text{lapser}}^{(i)}(t)$. A lapse management strategy is uniquely defined by the parameters $(\mathbf{p}, \delta, \gamma, c, T)$*

It is to be noted that even if the framework involves a time dimension, it is still a static approach: the insurer would run all analyses on its portfolio at one given time and apply an appropriate LMS immediately.

Even if this definition is already a simplification of any real-life insurance retention strategy, various constraints and the data and tools at the insurer’s disposal do not always allow to conduct such a study. In the following section, we consider a simplified version of this framework by assuming that $p_t^{(i)}$, $F_t^{(i)}$, and $d_t^{(i)}$ remain constant across time, and denoted $p^{(i)}$, $F^{(i)}$ and $d^{(i)}$ hereafter, with $F^{(i)}$ being the most recent face amount observed for subject i . Moreover, we set $\gamma^{(i)}$ and $\delta^{(i)}$ to be the same for all subjects and denoted as γ and δ hereafter. The constraint that $\delta < \min(p^{(i)})$ detailed in Definition 1 still holds. Finally, the last observed state of subject i is denoted $y^{(i)}$, with $y^{(i)} = 1$ if the policy is lapsed, $y^{(i)} = 0$ otherwise.

With those considerations, we can then define the control portfolio’s future value as

$$\begin{aligned}
 {}^F CPV(\mathbf{p}, \delta, \gamma, c, T) = & \\
 & \sum_{i=1}^n {}^F CLV^{(i)} \left(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T \right) \cdot \mathbf{1} \left(y^{(i)} = 0 \right) \\
 & + \sum_{i=1}^n {}^F CLV^{(i)} \left(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapser}}^{(i)}, d^{(i)}, T \right) \cdot \mathbf{1} \left(y^{(i)} = 1 \right).
 \end{aligned} \tag{8.2}$$

It represents the hypothetical value of the portfolio, considering that:

- every subject that did not lapse up to her last observation point - $y^{(i)} = 0$ at $t = 0$ - has a vector of retention probabilities of $\mathbf{r}_{\text{acceptant}}^{(i)}$;
- every subject that has been observed to lapse - $y^{(i)} = 1$ at $t = 0$ - has a vector of retention probabilities of $\mathbf{r}_{\text{lapser}}^{(i)}$.

Remark 8.1

It is important to note that this does not reflect the actual future value of the portfolio - as the future CLV of lapsers should be 0 - but rather its hypothetical expected future value given the nature (lapse or not) of every subject but not their actual states (actually lapsed or not). It represents this hypothetical future CLV of all subjects if no customer relationship management about lapses is carried out.

A classification algorithm would take the lapse indicator $y^{(i)}$ as a target variable and yield predictions $\hat{y}^{(i)}$. Given a lapse management strategy and such a classification algorithm, we define the lapse managed portfolio future value by

$$\begin{aligned}
{}^F LMPV(\mathbf{p}, \delta, \gamma, c, T) = & \\
& \sum_{i=1}^n {}^F CLV^{(i)} \left(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T \right) \cdot \mathbf{1} \left(y^{(i)} = 0, \hat{y}^{(i)} = 0 \right) \\
& + \sum_{i=1}^n {}^F CLV^{(i)} \left(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapser}}^{(i)}, d^{(i)}, T \right) \cdot \mathbf{1} \left(y^{(i)} = 1, \hat{y}^{(i)} = 0 \right) \\
& + \sum_{i=1}^n {}^F CLV^{(i)} \left(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T \right) \cdot \mathbf{1} \left(y^{(i)} = 0, \hat{y}^{(i)} = 1 \right) \\
& + \gamma \cdot \sum_{i=1}^n {}^F CLV^{(i)} \left(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T \right) \cdot \mathbf{1} \left(y^{(i)} = 1, \hat{y}^{(i)} = 1 \right) \\
& + (1 - \gamma) \cdot \sum_{i=1}^n {}^F CLV^{(i)} \left(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapser}}^{(i)}, d^{(i)}, T \right) \cdot \mathbf{1} \left(y^{(i)} = 1, \hat{y}^{(i)} = 1 \right) \\
& - \sum_{i=1}^n c \cdot \mathbf{1} \left(\hat{y}^{(i)} = 1 \right).
\end{aligned} \tag{8.3}$$

Clearly, the sums appearing in the formulas above could be grouped to make them more concise. We chose not to do so for the sake of visualization: we can distinctly see each possible case in each summand.

Then, we define the economic metric of the algorithm as the retention gain, the future profit generated by the retention management strategy over T years as

$$RG(\mathbf{p}, \delta, \gamma, c, T) = {}^F LMPV(\mathbf{p}, \delta, \gamma, c, T) - {}^F CPV(\mathbf{p}, \delta, \gamma, c, T), \tag{8.4}$$

which can be simplified as

$$\begin{aligned}
RG(\mathbf{p}, \delta, \gamma, c, T) = & \sum_{i=1}^n \left[\gamma \left[{}^F CLV^{(i)} \left(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T \right) \right. \right. \\
& \left. \left. - {}^F CLV^{(i)} \left(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapser}}^{(i)}, d^{(i)}, T \right) \right] \cdot \mathbf{1} \left(y^{(i)} = 1, \hat{y}^{(i)} = 1 \right) \right. \\
& \left. - {}^F CLV^{(i)} \left(\delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T \right) \cdot \mathbf{1} \left(y^{(i)} = 0, \hat{y}^{(i)} = 1 \right) \right] \\
& - \sum_{i=1}^n c \cdot \mathbf{1} \left(\hat{y}^{(i)} = 1 \right).
\end{aligned} \tag{8.5}$$

This evaluation metric can now be derived into an individual retention gain measure. More specifically, we define $z^{(i)}$ as

$$z^{(i)} = \begin{cases} -{}^F CLV^{(i)}(\delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T) - c & \text{if } y^{(i)} = 0 \\ \gamma \cdot \left[{}^F CLV^{(i)}(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T) - {}^F CLV^{(i)}(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, d^{(i)}, T) \right] - c. & \text{if } y^{(i)} = 1 \end{cases} \quad (8.6)$$

That last equation can seem obscure at first glance. It simply assigns to each individual the expected profit or loss that would result from targeting her with a given lapse management strategy. A positive amount for subject i means that targeting her would generate profit, whereas a negative one would lead to a loss for the insurer. We can take the example of a hypothetical scenario where $p^{(i)} = 3\%$, $\delta = 0.05\%$, $\gamma = 10\%$ and $c = 10$ euros. It would generate $z^{(i)}$ s taking values from $-234,614\text{€}$ to $53,066\text{€}$ with a mean of -218€ and a median of -55€ . Different scenarios would result in very different distributions for the $z^{(i)}$'s.

Eventually, we define $\tilde{y}^{(i)}$ as a binary target variable indicating for policyholder i if the individual expected retention gain resulting from a given retention strategy is a profit or a loss. More specifically, we define $\tilde{y}^{(i)}$ as

$$\tilde{y}^{(i)} = \begin{cases} 1 & \text{if } z^{(i)} > 0 \\ 0 & \text{if } z^{(i)} \leq 0 \end{cases}. \quad (8.7)$$

Remark 8.2

A subject in the dataset for which $y^{(i)} = 0$ would produce $\tilde{y}^{(i)} = 0$, whereas one for which $y^{(i)} = 1$ could produce $\tilde{y}^{(i)} = 0$ or $\tilde{y}^{(i)} = 1$. In other words, it is never profitable for the insurer to offer an incentive to a subject that would not have lapsed. Conversely, offering that same incentive to a lapsers can be profitable. However, depending on the subject's features and the lapse management strategy parameters, it can also lead to a loss.

We can now include $\tilde{y}^{(i)}$ as a new binary target variable in our models and directly consider RG as the global evaluation metric in the tree-based models we consider.

We can now compare two models: the classical one with $y^{(i)}$ as a target variable and accuracy as the evaluation metric; and the CLV-augmented one with $\tilde{y}^{(i)}$ as a target variable and RG as the evaluation metric.

Intuitively, the former tries to predict whether a policyholder will lapse and tune its parameters by minimizing the misclassification rate. On the other hand, the latter aims at predicting whether applying a given retention strategy to the i -th individual will be profitable for the insurer and tune its parameters by maximizing the global expected retention gain.

8.4 Methodology

In Section 8.3, we described a business-oriented framework, augmenting lapse management strategy with an evaluation metric based on the future CLV of every subject. Evaluating this metric

requires computing $r_{\text{acceptant}}$ and r_{lapsers} , the matrices of size $(n, T + 1)$ containing for every subject, survival probabilities that we detail below. This individual survival analysis differs from Loisel, Piette, and C.H.J. Tsai 2021’s work where r_{lapsers} is estimated globally and takes the same value for every policyholder regardless of their characteristics and where $r_{\text{acceptant}} = 1$ for any subject and at any time, ignoring the fact that an “acceptant” ’s policy can end with the policyholder’s death.

Given this framework, we propose a two-step methodology: firstly, we detail how this survival analysis is carried out to model those retention parameters, and secondly, we explain how we use them for training tree-based classification models.

8.4.1 Step 1: Modeling $r_{\text{acceptant}}$ and r_{lapsers}

We recall that a given subject’s policy can end with lapse or death, and the policy is considered active if competing events are yet to occur. Furthermore, while a lapsers’s policy can end with lapse or death, whatever comes first, an acceptant one can only end with death.

$r_{\text{lapsers}}^{(i)}(t)$ represents the probability that the policy of subject i is still active at time t , given that the subject is labeled as a lapsers - $\text{EVENT} = 1$ - at $t = 0$. Predicting these overall conditional survival probabilities with competing risks can be achieved by creating a combined outcome: the policy ends with death or lapse, whichever comes first. To compute r_{lapsers} in practice, we recode the competing events as a combined event. This cause-specific approach has the advantage of being achievable with any survival analysis method.

Conversely, $r_{\text{acceptant}}^{(i)}(t)$ represents the probability that the policy of subject i is still active at time t , given that the subject is not labeled as a lapsers - $\text{EVENT} = 0$ or 2 - at $t = 0$. This estimation is more complex as we must dissociate the risks of lapse and death. These causes being mutually exclusive, a competing risks methodology is well-suited to estimate $r_{\text{acceptant}}^{(i)}$ Laurent, Norberg, and Planchet 2016.

It is also important to note that here, $r_{\text{lapsers}}^{(i)}$ is modeled on subjects that have lapsed in the past - they may have been offered an incentive in the past, this is unknown - and not on subjects that have been offered an incentive that they declined. Our framework makes the implicit hypothesis that both behaviors are alike. It is more intuitive for $r_{\text{acceptant}}^{(i)}$ as a subject that has not lapsed in the past would have accepted any incentive if offered.

Competing risks frameworks

We are aware that improvements of our model over Loisel, Piette, and C.H.J. Tsai 2021’s approach, require the analysis of both the risks of lapse and death, thus a competing risk setting. As detailed in Appendix A.1, several regression models exist to estimate the global hazard and the hazard of one risk in such settings: cause-specific and subdistribution models. They account for competing risks differently, obtaining different hazard functions and thus have distinct advantages, drawbacks and interpretations. These differences are discussed in Milhaud and Dutang 2018, where the authors also considered a competing risk framework for lapse prediction.

After discussions detailed in Appendix A.1, the simplicity of a cause-specific approach and the fact that it can be adapted to any survival method, including tree-based ones, oriented our choice towards it. We then computed $r_{\text{acceptant}}^{(i)}$ and $r_{\text{lapsers}}^{(i)}$ with three different methods - Cox model,

random survival forest, and gradient boosting survival model - and retained the best one. These methods are briefly described in the following sections.

Cox proportional hazard model

One of the most common survival models is the Cox proportional hazard (CPH) model (D. Cox 1972). It postulates that the hazard function can be modeled as the product of a time-dependent and a covariate-dependent functions. The hazard function at time t for subject i with covariate vector $\mathbf{X}^{(i)}$, under Cox proportionnal hazard model can be expressed as

$$\underbrace{\lambda(t|X_1^{(i)}, X_2^{(i)}, \dots)}_{\text{hazard function}} = \lambda(t|\mathbf{X}^{(i)}) = \underbrace{\lambda_0(t)}_{\text{baseline hazard}} \underbrace{e^{\left(\mathbf{X}^{(i)} \cdot \beta^{(i)}\right)}}_{\text{log-partial hazard}}.$$

It is crucial to note that in this model, the hazard function is the product of the baseline hazard, which only varies with time, and the partial hazard, which only varies depending on the covariates. The parameters of this model are the β , and they can be estimated with a maximum likelihood approach. Their estimation can be carried out without having to model $\lambda_0(t)$ - which is why CPH is considered semi-parametric.

We use Python and lifelines (Davidson-Pilon 2019) to implement it. We specify a spline estimation for the baseline hazard function. We select the covariates and model parameters using AIC (Akaike 1973) and use the concordance index (Harrell et al. 1982) to compare CPH to other models. The concordance index - or Harrell's c-index or simply c-index - is a metric to evaluate the predictions made by a survival model. It can be interpreted as a generalization of the area under a receiver operating characteristic (ROC) curve (Hanley and McNeil 1982) - or AUC - in a survival setting with censored data.

Random Survival Forest

Survival trees have been extensively studied for a long time, and a complete review of such existing methods up to 2011 can be found in Bou-Hamad, Larocque, and Ben-Ameur 2011. The most important thing to understand is that a survival tree can be created by modifying the splitting criterion of a regular tree. Most survival tree algorithms are designed with a split function that aims to maximize the separation of the resulting child nodes in terms of survival profiles. This separation between nodes is estimated by maximizing the log-rank statistic (Mantel 1966; LeBlanc and Crowley 1993). Each terminal node of a survival tree contains a survival profile from which we can derive the survival and cumulative hazard function.

An RSF is the counterpart of a random forest (see Appendix B.1.2) for such survival trees. It has been developed in Ishwaran, Kogalur, et al. 2008 and extended for competing risks a few years after (Ishwaran, Gerds, et al. 2014). A prediction with RSF for a given subject is made by getting his/her survival profile in each tree in the forest. His/her corresponding survival and cumulative hazard function are estimated in each tree with Kaplan-Meier and Nelson-Aalen estimators, respectively. Eventually, the aggregation of those single-tree estimates constitutes the RSF's prediction.

We use Python and `skSurv` (Polsterl 2020) to implement RSF, and we tune and evaluate our model using the concordance index.

Remark 8.3

Sksurv allows us to use RSF with a cause-specific consideration of the competing risks. To this day, `sksurv` does not have a subdistribution competing risks model, whereas its R implementation `randomForestSRC` does (Ishwaran and Kogalur 2007). Moreover, a severe limitation of that approach is that predictions can only be made at time points observed in the training set. Concretely, this prevents us from using RSF to extrapolate survival and hazard functions to unobserved time points.

Gradient Boosting Survival Model

In the same way random forest has a survival counterpart, this is also true for gradient boosting approaches. An essential distinction between classical boosting algorithms (see Appendix B.1.3) and gradient boosting survival model (GBSM) lies in its loss function. The loss function that we use with GBSM is the partial likelihood loss of a CPH model, and the optimization in such a model is achieved by maximizing a slightly modified log-partial likelihood function,

$$\arg \min_f \sum_{i=1}^n \delta^{(i)} \left[f(\mathbf{X}^{(i)}) - \log \left(\sum_{j \in g^{(i)}} e^{f(\mathbf{X}^{(j)})} \right) \right],$$

where $\delta^{(i)}$ is the event indicator and $f(\mathbf{X}^{(i)})$ is GBSM's prediction for the i -th subject, with a covariate vector $\mathbf{X}^{(i)}$; $g^{(i)}$ is the tree leaf including subject i .

Similarly to RSF, we use Python and `sksurv` (Polsterl 2020) to implement GBSM. We tune and evaluate our model using the concordance index. Remark 8.3 also applies here.

Final modeling choice

Our analysis, using train-test split validation based on concordance index, shows that RSF and GBSM both outperformed a semiparametric Cox model in our study case. Regarding interpretability, we note that the feature importance analysis is very similar between the three models. All the details about the final concordance index scores, covariates importance and various plots for further analysis are available in Appendix B.1.

In the following sections, we decide to retain **GBSM** for the modeling of $r_{\text{acceptant}}^{(i)}$ and $r_{\text{lapseser}}^{(i)}$ as it has the best concordance index.

Remark 8.4

As this study aims to be business-oriented and favor real-life decision-making, it is crucial to note that the computation times for fitting these different models are very different and could potentially be a huge constraint for real operational deployment. Specific computation times differ greatly depending on various factors, such as the number of subjects or features considered, the computation power or parallelization ability at disposal, for instance. However, we can still give here an order of magnitude for those differences. If the tuning and fitting process for CPH can last a few tens of seconds, it lasts hours for RSF and tens of hours for GBSM.

8.4.2 Step 2: Classification tasks

Our work focuses on lapse management with tree-based models. The final classification question we want to answer is the following: which policyholders would be worth targeting with a lapse management strategy to maximize the expected T -year profit for the insurer? We will consider a single tree built with Breiman’s CART algorithm, Random Forest, XGBoost, and RE-EM trees. The following sections detail how those different approaches work. Those models will be compared on two different classification tasks; and tuned with two different evaluation metrics, a statistical metric and a business-related one.

On $y^{(i)}$ First, we will use a classical lapse prediction framework to model the policyholder’s behavior. Each policyholder in the historical dataset will be labeled as a lapsers or a non-lapsers with a binary outcome $y^{(i)}$. Our first batch of models will be trained with $y^{(i)}$ as a response variable and produce predictions $\hat{y}^{(i)}$. Accuracy(y, \hat{y}), which is undoubtedly the most intuitive performance measure for binary classification, is defined as the proportion of correctly predicted observations over all observations. It is widely used for churn analysis and appears to be a satisfying performance measure for relatively balanced outcomes - 22% of all observed subjects in our dataset being lapsers - in binary classification problems. We will use it as an evaluation metric in a 10-fold cross-validation step for tuning our models.

We know that more advanced evaluation metrics are available for binary classification, including the recall, the F_β score family (Chinchor 1992), the AUC under the ROC or under the Precision-Recall curve, the Brier Score (Brier 1950) and lift curve. They are standard evaluation metrics in classification and provide valuable insights into the model’s performance, they are also frequently used in the applied binary classification literature, especially in the presence of a significant imbalance in the data (He and Garcia 2009). However, in this paper, the mildness of the imbalance of $y^{(i)}$ and our will to compare a customer-centered framework to representative real-world practices encourages us to use accuracy as a comparison. One of the goals of this article is to demonstrate that some of the current practices in real-world applications, based on statistical metrics such as accuracy can be significantly improved by considering a profit-driven target variable and evaluation metric. We are aware that accuracy may not be an optimal choice of evaluation metric for binary prediction in general and churn or lapse analysis specifically, still, it seems representative of what practitioners use (see Table 2 from Duchemin and Matheus 2021 for example), as it is suggested in Loisel, Piette, and C.H.J. Tsai 2021. We do not aim at comparing our framework against the best existing methods but rather against the most representative. Nevertheless, the numerical results of Table 8.3 have also been obtained with recall, F1-score, and AUC for tuning and cross-validation and some are available in Appendix B.3: the conclusions obtained with such measures are similar to those obtained with accuracy. Thus, in this article and as in Loisel, Piette, and C.H.J. Tsai 2021, we will only select, evaluate and discuss the models in the light of accuracy.

On $\tilde{y}^{(i)}$ Secondly, we will use the CLV-Augmented lapse prediction framework, detailed in Section 8.3. Each policyholder will be labeled as a targeted lapsers or a non-targeted policyholder with the binary outcome $\tilde{y}^{(i)}$ and prediction for that outcome are denoted $\hat{y}^{(i)}$.

Remark 8.5

Note that whenever $y^{(i)} = 0$, we also have $\tilde{y}^{(i)} = 0$. In other words, if subject i does not intend to lapse, it is never worth proposing her an incentive: the subject will accept it with probability 1 and would not have lapsed.

On the other hand, when $y^{(i)} = 1$, it corresponds to either $\tilde{y}^{(i)} = 1$ or $\tilde{y}^{(i)} = 0$. In other words, if subject i is labeled as a lapser, it does not necessarily mean it is worth targeting her. From the insurer's point of view, some policies are better off lapsed. $\tilde{y}^{(i)}$ can be seen as a more detailed version of $y^{(i)}$ as it carries not only behavioral information regarding lapse but also a profitability one.

We thus train a second batch of models with $\tilde{y}^{(i)}$ as a response variable. We use RG as an evaluation metric in a 10-fold cross-validation step for tuning these models.

Summary of our methodology: First, we train a CART, RF and XGBoost models with $y^{(i)}$ as a binary target variable and accuracy as a tuning evaluation metric. Then we train them with $\tilde{y}^{(i)}$ as a binary target variable and RG as a tuning evaluation metric. Finally, we train and test all six models on different random samples of our dataset and keep track of the model's classification performance on all of them and for various retention strategies for comparison's sake.

The sections below briefly introduce the tree-based model we selected before displaying how they performed in various lapse management scenarios.

CART

CART (*Classification And Regression Trees*) is an algorithm developed by Breiman et al. 1984 that consists of recursively partitioning the covariate space. It is a widespread, intuitive and flexible algorithm that handles regression and classification problems.

Random forest

A natural idea to correct CART's instability and enhance its prediction accuracy is the aggregation of a significant number of single trees, each grown on different subsamples of the dataset. A random forest (RF by Breiman 2001) is a tree-based bagging procedure where each tree is grown on randomly drawn observations and contains splits considering only randomly drawn covariates.

XGBoost

Other tree-based approaches have been designed to reduce the instability of a single-tree model. Model boosting is an adaptive technique, first developed by Freund et Shapire (Freund and Schapire 1996), that does not rely on the aggregation of independent weaker models but rather on the aggregation of weak models built sequentially, one after the other. XGBoost (T. Chen and Guestrin 2016) is a widespread and performant tree-boosting model that relies on a gradient-boosting step and provides a very optimized parallelized procedure. It is considered a state-of-the-art library for various prediction problems.

The interested reader can find more detailed explanations about CART, RF and XGBoost mechanisms in the aforementioned references. For these modeling approaches, we used Python and sklearn (Pedregosa et al. 2011).

8.5 Real-life application

Based on the real life-insurance dataset at our disposal (described in Section 8.2), we use the survival model we selected and estimate $r_{\text{acceptant}}^{(i)}$ and $r_{\text{lapses}}^{(i)}$ for every individual. This allows us to compute the individual CLVs, RGs, $z^{(i)}$'s and $\tilde{y}^{(i)}$. We have already defined what a strategy is (see Definition 1), and we can thus apply our classification methodology to various retention strategies.

8.5.1 Considered lapse management strategies

The strategies considered are based on several criteria. First, we selected realistic strategy parameters and time horizons based on actual retention campaigns led by life insurers. Moreover, we chose to present strategies that illustrate the exhaustive list of conclusions and discussions that are carried out in the next section. Finally, we also incorporated strategies that are “obviously bad” in the sense that such strategies would necessarily lead to a loss for the insurer. Such extreme scenarios will supplement our discussions. In any case, we consider $p^{(i)}$ and $d^{(i)}$ to be constant in our application, as both those parameters were not estimated at the individual level by the life insurer that provided us with the dataset.

Results related to the 64 considered LMS are given in Appendix B.4. Our analysis showed that all considered LMS results can be split into 5 categories depending on how applying our framework impacted their expected retention gain over a naïve targeting. We have realistic profitable strategies that are improved by our framework, but also highly loss-inducing, moderate loss-inducing, highly profitable and unrealistically highly profitable strategies. We refer to the LMS displayed in Table 8.2 as representative strategies as they all belong to one of those categories. Numerical results regarding the most representative strategies can be found in Section 8.5.2 and related comments on how to read these tables are given in Section 8.5.3.

Scenarios	p	δ	γ	c	d	T
A-1	2.50%	0.04%	25%	10	1.50%	5
A-5	2.50%	0.04%	5%	10	1.50%	5
A-25	5.00%	0.10%	25%	10	1.50%	5
B-6	2.50%	0.08%	10%	10	1.50%	20
B-27	5.00%	0.20%	20%	100	1.50%	5

Table 8.2: Insightful LMS

8.5.2 Numerical results

N°	time (s)	Model	% target diff (% of 1's)	Accuracy		Retention gain		RG/target		Improvement
				$y^{(i)}$	$\tilde{y}^{(i)}$	$y^{(i)}$	$\tilde{y}^{(i)}$	$y^{(i)}$	$\tilde{y}^{(i)}$	
A-1	4949	CART	62.6% (8.2%)	92.3%	85.3%	114 661	219 655	4.48	38.20	91.6%
		RF		92.9%	85.4%	232 314	287 884	9.82	56.65	23.90%
		XGB		93.4%	85.8%	243 365	324 952	9.61	54.64	33.50%
A-5	4753	CART	86.7% (2.9%)	92.3%	83.6%	- 514 477	- 112 372	- 20.08	- 86.48	78.20%
		RF		92.9%	83.4%	- 323 544	- 3 937	- 13.65	- 28.28	98.80%
		XGB		93.4%	83.3%	- 383 004	0	- 15.14	0	100.00%
A-25	5379	CART	31.0% (15.2%)	92.3%	89.2%	4 160 423	3 882 623	162.44	241.06	-6.70%
		RF		92.9%	89.5%	4 018 432	3 666 219	169.65	249.54	-8.80%
		XGB		93.4%	90.0%	4 455 108	4 410 629	176.09	267.87	-1.00%
B-6	5906	CART	36.6% (13.9%)	92.3%	88.8%	705 721	922 490	27.69	60.21	30.70%
		RF		92.9%	88.9%	1 352 182	1 269 349	57.11	97.63	-6.10%
		XGB		93.4%	89.6%	1 342 882	1 428 722	53.09	96.76	6.40%
B-27	4811	CART	73.9% (5.7%)	92.3%	84.3%	- 694 436	751 404	- 27.16	226.99	208.20%
		RF		92.9%	84.4%	- 13 512	1 018 369	- 0.51	356.48	7637.00%
		XGB		93.4%	84.7%	- 38 050	1 253 252	- 1.55	345.94	3393.70%

Table 8.3: Means of the results obtained on considered LMS

8.5.3 Comments

Several terms in the two previous tables need to be explained. “% target diff” represents how different y and \tilde{y} are. It is the percentage of subjects for which $y^{(i)} = 1$ and $\tilde{y}^{(i)} = 0$: in other words, the proportion of lapsers not worth targeting with a given strategy. The quantity “% of 1’s” represents the proportion of ones in \tilde{y} the target variable. It is to be compared with the 22% of ones in y : the proposed framework’s imbalance increases with “% target diff”.

Then the table shows the 10-fold cross-validated mean accuracies, retention gains and RG/target with two methodologies: the columns denoted $y^{(i)}$ represent the metrics obtained by a model with $y^{(i)}$ as a response variable and accuracy as an evaluation metric, and the columns denoted $\tilde{y}^{(i)}$ represent the metric obtained by a model with $\tilde{y}^{(i)}$ as a response variable and RG as an evaluation metric.

RG/target represents the achieved retention gain for every targeted individual, for $y^{(i)}$, it is $RG / \sum_i y^{(i)}$, for $\tilde{y}^{(i)}$ it is $RG / \sum_i \tilde{y}^{(i)}$. Eventually, “Improvement” represents the percentage of improvement between the RG obtained with a classification on $y^{(i)}$ and the gain obtained with a classification on $\tilde{y}^{(i)}$. As the reported financial information was distorted for confidentiality reasons (see Section 8.2), relative measures such as “Improvement” are certainly more informative than absolute ones such as RG .

Some LMS are worth focusing on. For every strategy, we display its 10-fold cross-validated results: 10% of the dataset acting as an out-of-sample validation set at every fold. Every model is tuned by cross-validation within every fold. The boxplots below summarize some typical key results illustrated by several strategies. Those results will be discussed in Section 8.6.

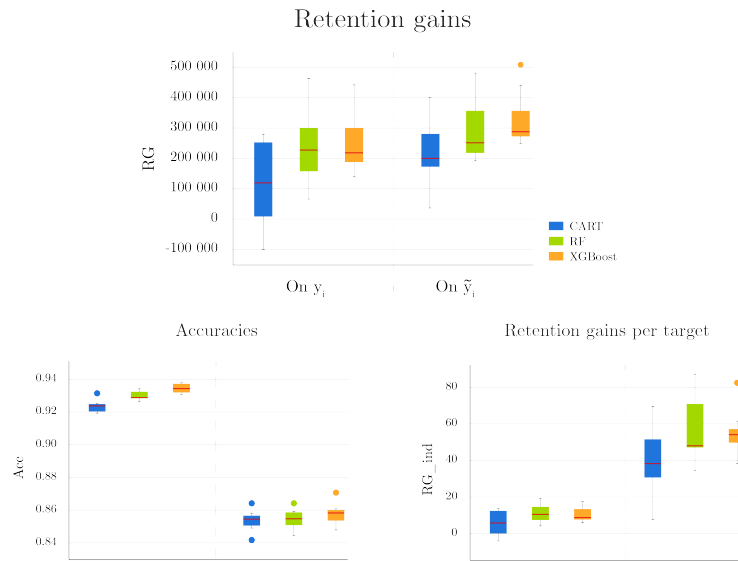


Figure 8.2: Strategy n°A-1: (Positive result on $y^{(i)}$) and an improved result on $\tilde{y}^{(i)}$.)

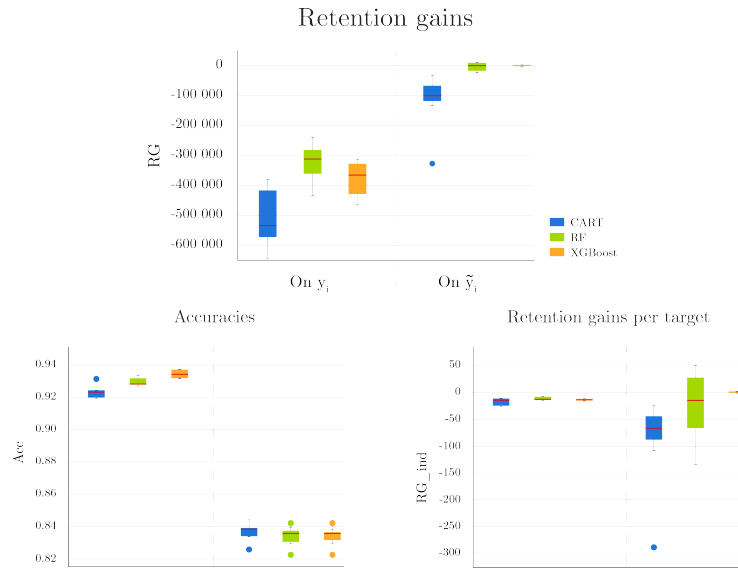


Figure 8.3: Strategy n°A-5: (Very negative result on $y^{(i)}$) and a loss-limiting result on $\tilde{y}^{(i)}$.)

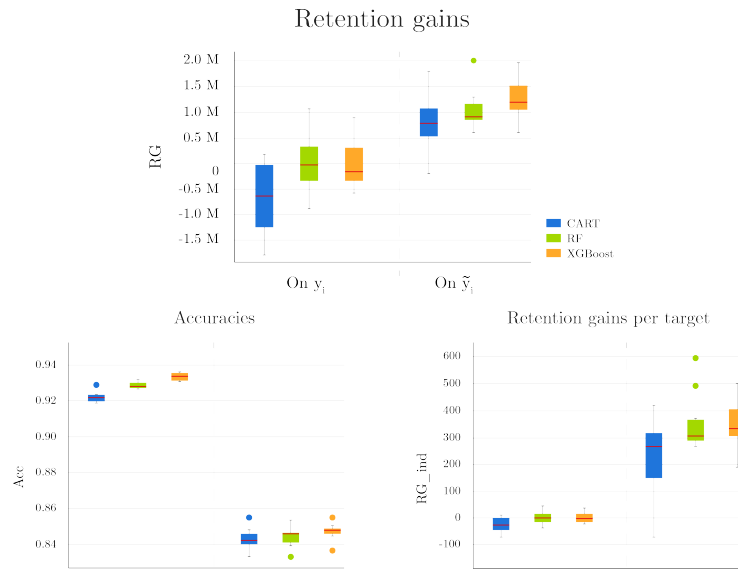


Figure 8.4: Strategy n°B-27: (Negative result on $y^{(i)}$ and positive one on $\tilde{y}^{(i)}$)



Figure 8.5: Strategy n°B-6: (High positive result on $y^{(i)}$ slightly improved with $\tilde{y}^{(i)}$.)

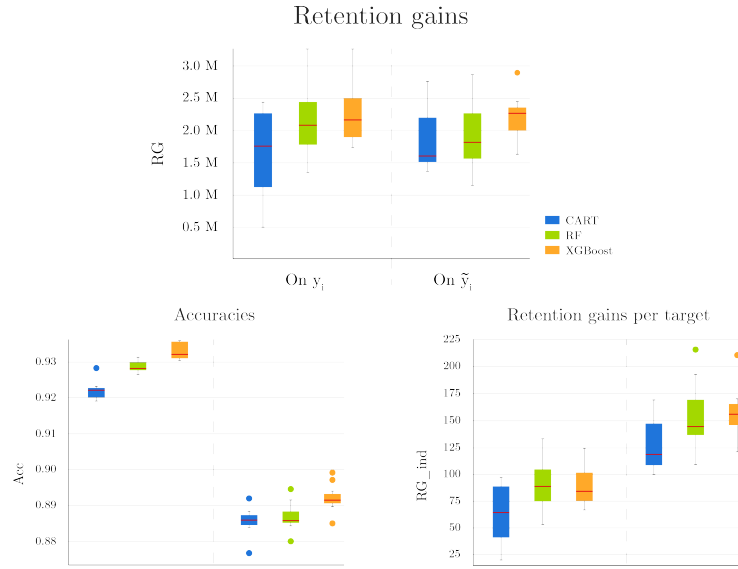


Figure 8.6: Strategy n°A-25: (Results on $y^{(i)}$ better than results on $\tilde{y}^{(i)}$.)

Remark 8.6

With considerable computation power and great parallelization, the results for all strategies - see other strategies in Appendix B.4 - were obtained with a wall time of less than 4 days and a CPU time of more than 100 days.

8.6 Discussion

8.6.1 General statements

As expected and shown in the actuarial literature, RF and XGBoost perform globally better than CART regarding mean accuracy and RG. It is true for all LMS considered in Table B.7. Globally, XGBoost is more consistent in performance and is the best model in most scenarios, both with and without the CLV-based measure. It is only outperformed by RF in strategies n°A-7, A-11, A-14, A-29, B-7, B-14 and B-31.

As expected, by design, the vast majority of strategies, including all the realistic ones, show that a classification on $\tilde{y}^{(i)}$ produces a targeting that yields better RG than a classification on $y^{(i)}$. Conversely, a classification on $y^{(i)}$ produces a targeting that delivers better accuracies regarding whether a policyholder will churn than a classification on $\tilde{y}^{(i)}$. These results were expected because of the models' respective objectives. Even if it is not surprising, it once again shows that for an insurer, lapse prediction and lapse management strategy are two very different prediction problems, often treated as similar ones.

Our CLV-augmented model shows different behavior depending on the strategy considered. As highlighted by Figure 8.2, a model on $y^{(i)}$ is greatly improved by our framework regarding RG and RG/target. Conversely, its accuracy in lapse prediction is not optimal.

An attractive property of our framework can be observed in Figure 8.3: it yields loss-limiting targeting. When the LMS considered is too aggressive, it will usually prefer to predict that an

LMS should not be applied at all ($\forall i, \hat{y}^{(i)} = 0$), thus generating a RG around 0€. This is made evident in some extreme strategies (LMS n°A-5, A-15, B-11, B-13 and B-16) and explains the presence of 0's in Table 8.3.

On less extreme strategies, it shows to yield substantial improvement when classification on $y^{(i)}$ gives negative RG . That observation confirms what was already pointed out by Loisel, Piette, and C.H.J. Tsai 2021: it can even turn a negative RG into a positive one (see LMS n°A-8, B-8, B-12, B-23 and B-27 (Figure 8.4)).

Our framework also improves a strategy where a classification on $y^{(i)}$ gives high RG . However, the improvement decreases as the difference between the total number of lapsers and the number of lapsers that would be profitable if retained is sizeable. An example of that is shown in Figure 8.5.

Finally, we can generate LMS for which our framework does not improve the expected RG . It is the case in LMS n°A-13, A-18 or A-27 (See e.g Figure 8.6). In LMS n°A-13, we can see that the mean of the RG is not improved, but the median is. In all those cases, the RG per target produced by the CLV-augmented model is greatly improved, indicating that a CLV-augmented strategy prefers to target fewer policyholders but only those who would generate high future profits. This last observation explains why a CLV-augmented LMS generates higher RG s when the cost of contact c is considerable. Indeed, the more costly a contact is, the more precise and specific a targeting strategy should be.

Generally, we can collate the results of various LMS - excluding LMS n°B-27 that has a very high improvement ratio - to obtain a mean performance of our framework.

The average observed RG improvement of a CLV-augmented framework over the classical lapse one is 57,9%^a. If we weigh these results by the expected RG s, the average improvement is still 31,7%. As a comparative result, it is reported in Section 6.2 of Loisel et al.'s work (Loisel, Piette, and C.H.J. Tsai 2021) that they obtain improvements over that same classical framework between 18% and 26%, depending on the considered strategies. This emphasizes that by extending their work, we seem to improve on their results. Obviously, as we were not able to compare our results on the same data and strategies, and because our definitions of RG differ, such a conclusion is to be treated cautiously.

^aUsing XGBoost

8.6.2 Marketing decision making

We already pointed out that the improvement of a lapse management strategy including CLV grows with the proportion of lapsers with a negative CLV (see Appendix B.2). Models resulting from our framework do not consider them as good targets. In fact, there is a Pearson correlation coefficient of 77% between RG improvement and the proportion of target differences among the LMS detailed in Table B.7. Of course, as the improvement ratio has no clear interpretation in some cases, this analysis should be carried out in more depth, separating the cases where both RG - with and without the inclusion of CLV - are positive from the cases where one of them is negative. By doing so, we observe that the Pearson correlation coefficient for LMS yielding positive RG regardless of the inclusion of CLV is even higher: 83%.

In terms of targeting, it seems crucial to understand what differentiates a subject for which $y^{(i)} = 1$ and $\tilde{y}^{(i)} = 0$ from the others. An investigation of such policyholder profiles can be carried out for every lapse management strategy. We take the example of LMS n°A-1, where 62,6% of policyholders were in that case (see Section 8.5.2). With that strategy, the profile of non-targeted lapsers indicates that

- 57.2% of them are men, similar to the entire dataset,
- 76.4% of them contracted product n°1 whereas 72% of all policyholders chose it,
- the mean seniority of their policy is 10.4 years compared to the 13.4 years for the complete dataset,
- the mean face amount of such policies is 12,156, whereas the average face amount for all considered policies is 40,263.

In that strategy, our framework indicates that marketing efforts on low seniority policyholders with low face amount policies are inefficient. Of course, this conclusion is only valid for the considered LMS; however, our framework allows us to conduct such analysis for any LMS and interpret the results at an individualized level.

8.6.3 Management rules decision making

Sensitivity analysis of those results can highly benefit management rules decision-making. This framework serves as a tool that compares future hypothetical lapse management strategies in order to choose the best one - among realistic scenarios -. It can also be used to tune a given strategy by answering questions like:

- For which incentive δ does the retention strategy become profitable ?
- For which acceptance probability γ does the retention strategy become profitable ?
- With a given budget, what is the optimal list of policies that should be targeted?
- At which horizon T , does the retention strategy become profitable ? In other words, when can the insurer expect a return on investment?

Answering these questions constitutes a 1-parameter sensitivity analysis. In our framework, six parameters influence the expected retention gain $(p, \delta, \gamma, c, d, T)$.

We can argue that among them are three structural parameters that are insurer's dependent and not linked to the external state of the world: δ , γ and c . Among them, the contact cost c is more or less fixed and can not be easily changed by the insurer. Conversely, δ and γ are to be chosen by the insurer. Moreover, they also are correlated with management and commercial efficiency - an efficient campaign impacts the final γ - and correlated together: the higher the incentive δ , the higher the probability of acceptance γ .

By fixing all other parameters and trying various combinations of δ and γ we obtain the fol-

lowing 3D surfaces.

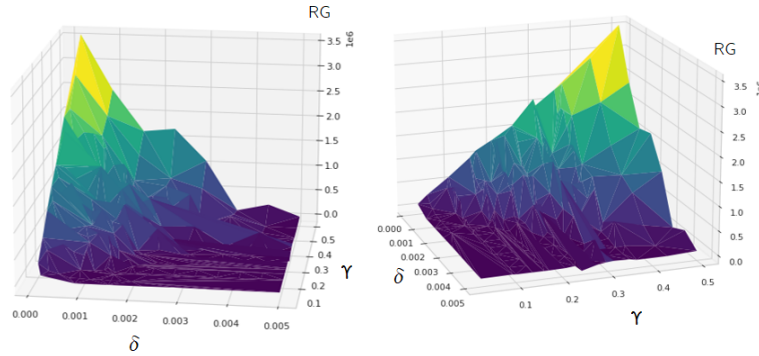


Figure 8.7: 3d plot (δ , γ , RG)

This surface is not surprising and indicates that the higher the acceptance rate and the lower the incentive, the higher the retention gain. The surface gradient can give powerful insights regarding the most efficient commercial efforts to make: is it better for the insurer to propose lower incentives and manage to conserve the same acceptance probability or to put commercial effort into improving the acceptance probability for the same proposed incentive? This surface directly addresses this question.

Remark 8.7

Of course, the interdependency of those parameters should make some part of this surface unrealistic from a management decision-making point of view. The insurer should consider such dependencies when designing a lapse management strategy.

Among the six parameters are also three conjectural parameters that depend on the external state of the world: the insurer's profitability p (that depends on competition, macroeconomic considerations, or regulation), the discount rate d , and the time horizon T (that can be driven by the insurer's vision but also by regulation: the ORSA time horizon with the strategic and the long-term business planning time horizon should be both considered). Among them, we chose to fix p and let d and T vary. Moreover, p and T are obviously interdependent and considered through the management's prospective view of the conjecture's evolution. A given interest rate scenario should represent a curve on the following surface.

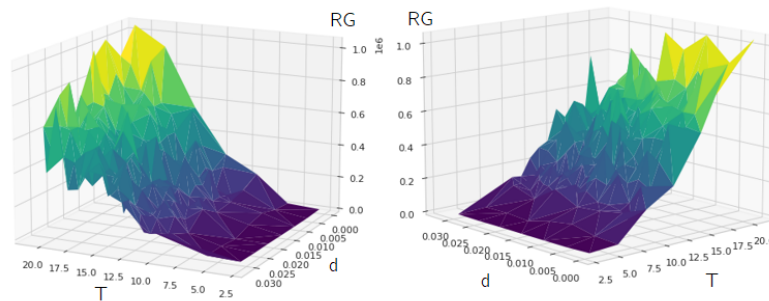


Figure 8.8: 3d plot (d , T , RG)

This surface is less smooth than the one displayed in Figure 8.7 and seems to indicate a more unstable relationship between RG and the conjectural parameters. An explanation of that behavior can be that those surface points are generated by running our framework on a random subsam-

ple of our dataset, for computation time considerations. Generating the same surface with more policyholders is likely to give a smoother behavior.

Remark 8.8

Of course, the interdependency of T and d should make some part of this surface unrealistic from an actuarial point of view. Actuarial rate projections would give precise plausible scenarios on this surface. Such considerations should be taken into account by the insurer when designing a lapse management strategy.

Remark 8.9

The insurer can also use our framework to measure the retention gain to be expected at different time horizons obtained by existing retention campaigns. In that case, the insurer would have to neutralize the effect of the existing LMS in order to estimate the control portfolio's future value. We leave this remark as future work for applied risk management research.

8.7 Conclusion and perspectives

The work carried out in this paper shows that including CLV in lapse management strategy can largely benefit an insurer's decision-making ability regarding lapse management strategy. We showed that survival tree-based models can outperform parametric approaches in such actuarial contexts. Then, our comparison of tree-based models on different lapse management strategies indicated that our CLV-based framework leads to increased predicted gains for any realistic scenario and acts as a loss-limiting targeting approach, regardless of the retention strategy. Moreover, the global results obtained in Section 8.6.1 show that our approach significantly improves on existing ones. Eventually, the discussion section highlighted the fact that our model can give insights to the life insurer regarding commercial and strategic decision-making.

The framework and methodologies described in this paper suffer some limitations. For instance, following one single fixed strategy for every policyholder is arguably unrealistic. We could imagine an extension of our models to individualized lapse management strategies that would vary between subjects and could also be adjusted with time. In the application, we also considered constant parameters p and δ : a limiting assumption whose impact could be studied. There is also room for improvement regarding the correlations of LMS parameters: the value of the incentive and the acceptance probability are evidently interdependent parameters for an insurance company, and this interdependency could be considered.

This paper defines a practical management tool for life insurers as those models can measure the RG and improve real strategies used in existing retention campaigns. Finally, our vision of CLV, and by extension, our whole methodology design could be improved by using longitudinal data that would yield time-dynamic results. We leave those two last observations for future work.

A real-life comparison between an actual retention strategy targeting and both the naïve and CLV-improved methodologies could be insightful for the insurer.

8.8 Suggestions for future work

A few properties of the framework that is detailed in this chapter, as well as some additional suggestions for future work emerged after the publication of the article it is based on. In this section, we will introduce some properties of the ${}^FCLV^{(i)}$ function as defined in Chapter 8, then discuss their consequence on the individual expected profit or loss $z^{(i)}$. Eventually, we will conclude this discussion by suggesting further enrichments to this framework.

8.8.1 $z^{(i)}$ efficiency border

CLV properties

Let us consider the individual ${}^FCLV^{(i)}$ function, defined as

$${}^FCLV^{(i)}(p^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}^{(i)}, \mathbf{d}, T) = \sum_{t=0}^T \frac{p^{(i)} F_t^{(i)} r^{(i)}(t)}{(1 + d_t^{(i)})^t}, \quad (8.8)$$

with $\forall(i, t)$, $d_t^{(i)}$ and $F_t^{(i)} \in \mathbb{R}^+$, $p^{(i)} \in \mathbb{R}$, and $r^{(i)}(t) \in [-1, 1]$ ¹. In this version, we consider a constant individual profitability ratio over time. Here are some trivial properties of this function.

Property 8.8.1. *For any given individual, $\forall t$, $p^{(i)} r^{(i)}(t) \geq 0$ (respectively ≤ 0) $\Rightarrow {}^FCLV^{(i)} \geq 0$ (respectively ≤ 0). In other words, ${}^FCLV^{(i)}$ has the same sign as $p^{(i)} r^{(i)}(t)$.*

Proof. It is trivial to see that the summands always have the same sign as $p^{(i)} r^{(i)}(t)$. The sum of positive terms stays positive and the sum of negative terms stays negative. \square

Property 8.8.2. *For any given individual,*

$${}^FCLV^{(i)}(p_1^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}^{(i)}, \mathbf{d}, T) + {}^FCLV^{(i)}(p_2^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}^{(i)}, \mathbf{d}, T) = {}^FCLV^{(i)}(p_1^{(i)} + p_2^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}^{(i)}, \mathbf{d}, T).$$

In other words, a sum of CLVs with different profitability ratios is itself a CLV with the sum of the ratios.

Proof. We have immediately that $\forall t$,

$$\frac{p_1^{(i)} F_t^{(i)} r^{(i)}(t)}{(1 + d_t^{(i)})^t} + \frac{p_2^{(i)} F_t^{(i)} r^{(i)}(t)}{(1 + d_t^{(i)})^t} = \frac{(p_1^{(i)} + p_2^{(i)}) F_t^{(i)} r^{(i)}(t)}{(1 + d_t^{(i)})^t}.$$

\square

Property 8.8.3. *For any given individual,*

$${}^FCLV^{(i)}(p^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_1^{(i)}, \mathbf{d}, T) + {}^FCLV^{(i)}(p^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_2^{(i)}, \mathbf{d}, T) = {}^FCLV^{(i)}(2p^{(i)}, \mathbf{F}^{(i)}, \bar{\mathbf{r}}^{(i)}, \mathbf{d}, T).$$

With $\bar{\mathbf{r}}^{(i)}(t)$, the mean of $r_1^{(i)}(t)$ and $r_2^{(i)}(t)$. In other words, a sum of CLVs with different risk profiles is itself a CLV with double the profitability ratio and a mean risk profile.

¹For practical applications, $r^{(i)}(t)$ is a probability, hence $\in [0, 1]$. Mathematically, nothing prevents us from considering negative values for $r^{(i)}(t)$ and it will show to be useful for proving Equation 8.12. Hence the definition of ${}^FCLV^{(i)}$ with $r^{(i)}(t) \in [-1, 1]$.

Proof. We have immediately that $\forall t$,

$$\frac{p^{(i)} F_t^{(i)} r_1^{(i)}(t)}{(1+d_t^{(i)})^t} + \frac{p^{(i)} F_t^{(i)} r_2^{(i)}(t)}{(1+d_t^{(i)})^t} = \frac{p^{(i)} F_t^{(i)} (r_1^{(i)}(t) + r_2^{(i)}(t))}{(1+d_t^{(i)})^t} = \frac{2p^{(i)} F_t^{(i)} \bar{r}^{(i)}(t)}{(1+d_t^{(i)})^t}$$

□

Property 8.8.4. For any given individual, let us assume two risk profiles $\mathbf{r}_1^{(i)}$ and $\mathbf{r}_2^{(i)}$. $\forall t$, $r_1^{(i)} \geq r_2^{(i)}$. Let us assume two profitability ratios $p_1^{(i)}$ and $p_2^{(i)}$:

$$\begin{aligned} p_1^{(i)} \geq p_2^{(i)} &\Rightarrow {}^F CLV^{(i)}(p_1^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_1^{(i)}, \mathbf{d}, T) + {}^F CLV^{(i)}(p_2^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_2^{(i)}, \mathbf{d}, T) \geq {}^F CLV^{(i)}(p_1^{(i)} + p_2^{(i)}, \mathbf{F}^{(i)}, \bar{\mathbf{r}}^{(i)}, \mathbf{d}, T) \\ p_1^{(i)} \leq p_2^{(i)} &\Rightarrow {}^F CLV^{(i)}(p_1^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_1^{(i)}, \mathbf{d}, T) + {}^F CLV^{(i)}(p_2^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_2^{(i)}, \mathbf{d}, T) \leq {}^F CLV^{(i)}(p_1^{(i)} + p_2^{(i)}, \mathbf{F}^{(i)}, \bar{\mathbf{r}}^{(i)}, \mathbf{d}, T) \end{aligned}$$

In other words, a sum of CLV's with different profitability ratios and risk profiles can always be compared to a CLV with the sum of the profitability ratios and a mean risk profile.

Proof. We assume $\forall t$, $r_1^{(i)}(t) \geq r_2^{(i)}(t)$.

Thus $r_1^{(i)} = \bar{r}^{(i)} + \frac{|r_1^{(i)} - r_2^{(i)}|}{2} = \bar{r}^{(i)} + \frac{\Delta r^{(i)}}{2}$. Symmetrically, $r_2^{(i)} = \bar{r}^{(i)} - \frac{\Delta r^{(i)}}{2}$.

If $p_1^{(i)} \geq p_2^{(i)}$, we also have $p_1^{(i)} = \bar{p}^{(i)} + \frac{\Delta p^{(i)}}{2}$ and $p_2^{(i)} = \bar{p}^{(i)} - \frac{\Delta p^{(i)}}{2}$.

By replacing every term, and using Properties 8.8.2 and 8.8.3, we deduce that

$$\forall t, \frac{p_1^{(i)} F_t^{(i)} r_1^{(i)}(t)}{(1+d_t^{(i)})^t} + \frac{p_2^{(i)} F_t^{(i)} r_2^{(i)}(t)}{(1+d_t^{(i)})^t} = \frac{2\bar{p}^{(i)} F_t^{(i)} \bar{r}^{(i)}(t)}{(1+d_t^{(i)})^t} + \frac{\Delta p^{(i)} F_t^{(i)} \Delta r^{(i)}(t)}{2(1+d_t^{(i)})^t}.$$

Summing for every t and using the positivity Properties 8.8.1 concludes the proof.

If $p_1^{(i)} \leq p_2^{(i)}$, signs are switched and we deduce that

$$\forall t, \frac{p_1^{(i)} F_t^{(i)} r_1^{(i)}(t)}{(1+d_t^{(i)})^t} + \frac{p_2^{(i)} F_t^{(i)} r_2^{(i)}(t)}{(1+d_t^{(i)})^t} = \frac{2\bar{p}^{(i)} F_t^{(i)} \bar{r}^{(i)}(t)}{(1+d_t^{(i)})^t} - \frac{\Delta p^{(i)} F_t^{(i)} \Delta r^{(i)}(t)}{2(1+d_t^{(i)})^t}.$$

Summing for every t and using the positivity Properties 8.8.1 concludes the proof. □

Borders conditions on $z^{(i)}$

We recall that the framework developed in Chapter 8 eventually describes the following expected profit or loss resulting from tragetting PH i as

$$z^{(i)} = \begin{cases} -{}^F CLV^{(i)}(\delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T) - c & \text{if } y^{(i)} = 0 \\ \gamma \cdot \left[{}^F CLV^{(i)}(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T) \right. \\ \left. - {}^F CLV^{(i)}(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, d^{(i)}, T) \right] - c. & \text{if } y^{(i)} = 1 \end{cases} \quad (8.9)$$

The first case only implies that whenever the policyholder has not been observed to lapse, tragetting her would inevitably result in a loss for the insurer.

The second case can be further discussed. When the PH has lapsed, we have the following inequalities, using Property 8.8.4:

$$\gamma {}^F CLV^{(i)}(-\delta, F^{(i)}, \bar{\mathbf{r}}^{(i)}, d^{(i)}, T) - c \leq z^{(i)} \leq \gamma {}^F CLV^{(i)}(p - \frac{\delta}{2}, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)} - \mathbf{r}_{\text{lapsers}}^{(i)}, T) - c. \quad (8.10)$$

The first inequality is obtained by considering that

$$\begin{aligned} & {}^F CLV^{(i)}\left(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T\right) - {}^F CLV^{(i)}\left(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, d^{(i)}, T\right) \\ &= {}^F CLV^{(i)}\left(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T\right) + {}^F CLV^{(i)}\left(-p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, d^{(i)}, T\right). \end{aligned} \quad (8.11)$$

The second one is obtained by considering that

$$\begin{aligned} & {}^F CLV^{(i)}\left(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T\right) - {}^F CLV^{(i)}\left(p^{(i)}, F^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, d^{(i)}, T\right) \\ &= {}^F CLV^{(i)}\left(p^{(i)} - \delta, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, d^{(i)}, T\right) + {}^F CLV^{(i)}\left(p^{(i)}, F^{(i)}, -\mathbf{r}_{\text{lapsers}}^{(i)}, d^{(i)}, T\right). \end{aligned} \quad (8.12)$$

The first inequality shows that the lower bound of $z^{(i)}$ is always negative. This result is fully expected and confirms that there are no LMS parameters that would ensure a profit for the insurer. Nevertheless, it does exhibit a lower boundary to the expected loss of an individual. With $d^{(i)} \geq 0$ and $\bar{\mathbf{r}}^{(i)} \leq 1$, we deduce that

$$z^{(i)} \geq -\gamma(T+1)\delta F^{(i)} - c. \quad (8.13)$$

By denoting $\bar{F}^{(i)}$, the mean of $F^{(i)}$ over the targetted PH, we conclude that targeting N_t policyholders will at worst produce a loss of

$$N_t[\gamma(T+1)\delta\bar{F}^{(i)} + c], \quad (8.14)$$

which is the undiscounted price of a retention campaign with the offering of an incentive δ , in the worst case where all targetted policyholders stay in the portfolio T years. This lower bound simply states that the insurer will at most lose the initial investment of the retention campaign and not gain anything from it. In other words, with the assumptions of this framework, it is not possible to design a retention campaign that generates a loss superior to its investment, from targeting lapsers. This property is expected by design and is not really informative about our framework.

However, the second inequality is insightful. It gives us a condition ensuring a loss for the insurer. Indeed, it follows from Equation 8.10 that

$$\gamma {}^F CLV^{(i)}\left(p^{(i)} - \frac{\delta}{2}, F^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)} - \mathbf{r}_{\text{lapsers}}^{(i)}, T\right) - c \leq 0 \Rightarrow z^{(i)} \leq 0. \quad (8.15)$$

By majoring every summand, we deduce that

$$\left(\forall t, r_{\text{acceptant}}^{(i)}(t) - r_{\text{lapsers}}^{(i)}(t) \leq \frac{c(1+d^{(i)})^t}{\gamma(p^{(i)} - \frac{\delta}{2})F^{(i)}(T+1)}\right) \Rightarrow z^{(i)} \leq 0 \quad (8.16)$$

Moreover, because $\mathbf{r}_{\text{acceptant}}^{(i)}$ and $\mathbf{r}_{\text{lapsers}}^{(i)}$ were computed with a cause-specific approach, we have $\mathbf{r}_{\text{acceptant}}^{(i)} - \mathbf{r}_{\text{lapsers}}^{(i)} = CIF(\text{EVENT} = \text{LAPSE})$, the Cumulative Incidence Function for the lapse event (See the Appendix A.1.1 for further details). In other words for subject i , we can deduce that if the CIF for lapse is too small at all times, she will not be worth targeting. The framework does not target individuals for which lapse is unlikely to happen. And the threshold for which it is too small is increasing with c , $d^{(i)}$ and δ and decreasing with γ , p , $F^{(i)}$, and T .

Specifically, if the incentive and/or the cost of contact are too high, it might not be worth targeting PH i . Furthermore, the same conclusion arises if the probability of accepting the incentive and/or the horizon considered are too low. Those conclusions are perfectly in line with the empirical observations in Chapter 8 and further indicate that the analysis of the lapse risk alone can be informative enough for an insurer to circumvent designing retention campaigns leading to unavoidable losses.

8.8.2 Other improvements

Eventually, we also became aware of new ways of improving or extending this framework.

Firstly, this work explicitly states the limitation of the assumption of independence for the pair of parameters (δ, γ) . A first idea on this matter would be to consider that γ is an increasing function of δ such as a logistic function. To the best of our knowledge, this has never been tried and some empirical analyses of real retention campaigns could help find realistic parameters for such a function. A study of price sensitivity such as the works of Guelman and Guillen 2014, then Lemmens and Gupta 2020; Verschuren 2022 within a causal inference framework could be adapted to that task. As a matter of fact, the definition of RG , as a difference of profit obtained with and without lapse management is closely related to the Conditional Average Treatment Effect (CATE) which would be given by

$$CATE = \mathbb{E}[CPV | \mathbf{X}] - \mathbb{E}[LMPV | \mathbf{X}],$$

where CPV and $LMPV$ are the so-called *response under control* and *response under treatment*.

Another idea for a more realistic modeling of the probability of acceptance is to replace the constant γ with the realization of a random variable following a well-chosen distribution. This was suggested in other works from Verbraken, Verbeke, and Baesens 2012, or Stripling et al. 2018 within the expected maximum profit measure for customer churn (EMPC) framework. Following this idea, the structure of dependence between γ and δ can be modeled by including δ in the parameters of the distribution of γ . To our knowledge, this has yet to be explored.

Secondly, we noticed that this framework could potentially be used to measure the retention gains obtained from a real campaign, a posteriori. In practice, this is not possible yet, within the proposed framework, as the estimation of the RG requires the observation of a control portfolio (See Chapter 8, Section 3, Equation 2), where no lapse management occurred. This is unobservable a posteriori. Once again, the causal inference framework used by Verschuren 2022 is a possible way of filling that gap and using our framework as a retention campaign's efficiency measuring tool.

Eventually, another limitation of the proposed framework is that it considers that all policyholders are either lapsers or non-lapsers. In reality, it seems obvious that there are more than two extreme risk profiles. There exist nuances of lapse risk between policyholders, but also evolutions of one's PH profiles during her policy's lifetime. In other words, for any PH, there should be a continuous set of potential risk profiles, that can evolve through time. This idea, and others, are tackled with the design of a longitudinal lapse management strategy in Part IV of this thesis.

Bibliography

- Valla, M., X. Milhaud, and A.A. Olympio (Sept. 2023). "Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies". In: *European Actuarial Journal*. DOI: [10.1007/s13385-023-00358-0](https://doi.org/10.1007/s13385-023-00358-0). URL: <https://hal.science/hal-03903047> (HAL), <https://export.arxiv.org/pdf/2307.06651> (arxiv), https://link.springer.com/article/10.1007/s13385-023-00358-0?code=84d3a0d0-b866-48d5-bc60-5ed6832d144a&error=cookies_not_supported (journal).
- Kotler, P. (1996). *Principles of marketing*. eng. European ed. London ; Prentice Hall. ISBN: 0131659030.
- Berger, P.D. and N.I. Nasr (1998). "Customer lifetime value: Marketing models and applications". In: *Journal of Interactive Marketing* 12, pp. 17–30. URL: <https://api.semanticscholar.org/CorpusID:168101567>.
- Dwyer, F.R. (1989). "Customer lifetime valuation to support marketing decision making". In: *Journal of Direct Marketing* 3.4, pp. 8–15. DOI: <https://doi.org/10.1002/dir.4000030404>.
- Wang, P. and T. Splegel (1994). "Database marketing and its measurements of success". In: *Journal of Interactive Marketing* 8, pp. 73–81. URL: <https://api.semanticscholar.org/CorpusID:167929079>.
- Keane, T. and P. Wang (1995). "Applications for the Lifetime Value Model in Modern Newspaper Publishing". In: *Journal of Direct Marketing* 9, pp. 59–66. DOI: [10.1002/dir.4000090209](https://doi.org/10.1002/dir.4000090209).
- Fader, P.S., B.G.S. Hardie, and K.L. Lee (2005). "RFM and CLV: Using Iso-Value Curves for Customer Base Analysis". In: *Journal of Marketing Research* 42.4, pp. 415–430. DOI: [10.1509/jmkr.2005.42.4.415](https://doi.org/10.1509/jmkr.2005.42.4.415).
- Donkers, B., P. Verhoef, and M. Jong (Feb. 2007). "Modeling CLV: A test of competing models in the insurance industry". In: *Quantitative Marketing and Economics* 5, pp. 163–190. DOI: [10.1007/s11129-006-9016-y](https://doi.org/10.1007/s11129-006-9016-y).
- Loisel, S., P. Piette, and C.H.J. Tsai (2021). "Applying economic measures to lapse risk management with Machine Learning approaches". In: *ASTIN Bulletin: The Journal of the IAA* 51.3, pp. 839–871. DOI: [10.1017/asb.2021.10](https://doi.org/10.1017/asb.2021.10).
- Seyerle, M. (2001). *Customer lifetime value in the insurance industry*. Proceedings of the 26th Annual SAS Users Group International Conference.
- Desirena, G. et al. (2019). "Maximizing Customer Lifetime Value using Stacked Neural Networks: An Insurance Industry Application". In: *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pp. 541–544. DOI: [10.1109/ICMLA.2019.00101](https://doi.org/10.1109/ICMLA.2019.00101).
- KPMG (2020). *First Impressions: IFRS 17 insurance contracts (2020 edition)*. URL: <https://assets.kpmg/content/dam/kpmg/ie/pdf/2020/09/ie-ifrs-17-first-impressions.pdf>.
- Buchardt, K. (Mar. 2014). "Dependent interest and transition rates in life insurance". In: *Insurance: Mathematics and Economics* 55. DOI: [10.1016/j.insmatheco.2014.01.004](https://doi.org/10.1016/j.insmatheco.2014.01.004).

- Buchardt, K., T. Moller, and K.B. Schmidt (2015). "Cash flows and policyholder behaviour in the semi-Markov life insurance setup". In: *Scandinavian Actuarial Journal* 2015.8, pp. 660–688. DOI: [10.1080/03461238.2013.879919](https://doi.org/10.1080/03461238.2013.879919).
- Kim, C. (2005). "Modeling Surrender and Lapse Rates With Economic Variables". In: *North American Actuarial Journal* 9.4, pp. 56–70. DOI: [10.1080/10920277.2005.10596225](https://doi.org/10.1080/10920277.2005.10596225).
- Gatzert, N. and H. Schmeiser (2008). "Assessing the Risk Potential of Premium Payment Options in Participating Life Insurance Contracts". In: *The Journal of Risk and Insurance* 75.3, pp. 691–712. ISSN: 00224367, 15396975. URL: <http://www.jstor.org/stable/25145301> (visited on 07/29/2022).
- Eling, M. and M. Kochanski (2013). "Research on lapse in life insurance: what has been done and what needs to be done?" In: *Journal of Risk Finance* 14.4, pp. 392–413. URL: <https://EconPapers.repec.org/RePEc:eme:jrfpps:v:14:y:2013:i:4:p:392-413>.
- Eling, M. and D. Kiesenbauer (2014). "What Policy Features Determine Life Insurance Lapse? An Analysis of the German Market". In: *The Journal of Risk and Insurance* 81.2, pp. 241–269. ISSN: 00224367, 15396975. URL: <http://www.jstor.org/stable/24546804> (visited on 07/29/2022).
- Gupta, S. and D.R. Lehmann (2006). "Customer Lifetime Value and Firm Valuation". In: *Journal of Relationship Marketing* 5.2-3, pp. 87–110. DOI: [10.1300/J366v05n02_06](https://doi.org/10.1300/J366v05n02_06).
- Gupta, S. (2009). "Customer-Based Valuation". In: *Journal of Interactive Marketing* 23.2. Anniversary Issue, pp. 169–178. ISSN: 1094-9968. DOI: <https://doi.org/10.1016/j.intmar.2009.02.006>. URL: <https://www.sciencedirect.com/science/article/pii/S109499680900036X>.
- Ascarza, E. et al. (2018). In *Pursuit of Enhanced Customer Retention Management: Review, Key Issues, and Future Directions*. en. Special Issue on 2016 Choice Symposium. Customer Needs and Solutions 5,
- Dar, A.A. and C. Dodds (1989). "Interest Rates, the Emergency Fund Hypothesis and Saving through Endowment Policies: Some Empirical Evidence for the U.K." In: *Journal of Risk and Insurance* 56, p. 415.
- Poufnas, T. and G. Michaelide (Jan. 2018). "Determinants of Life Insurance Policy Surrenders". In: *Modern Economy* 09, pp. 1400–1422. DOI: [10.4236/me.2018.98089](https://doi.org/10.4236/me.2018.98089).
- Yu, L., J. Cheng, and T. Lin (2019). "Life insurance lapse behaviour: evidence from China". In: *The Geneva Papers on Risk and Insurance - Issues and Practice* 44.4, pp. 653–678. DOI: [10.1057/s41288-018-0104-5](https://doi.org/10.1057/s41288-018-0104-5). URL: https://ideas.repec.org/a/pal/gpprii/v44y2019i4d10.1057_s41288-018-0104-5.html.
- Kuo, W., C. Tsai, and W.K. Chen (2003). "An Empirical Study on the Lapse Rate: The Cointegration Approach". In: *The Journal of Risk and Insurance* 70.3, pp. 489–508. ISSN: 00224367, 15396975. URL: <http://www.jstor.org/stable/3519905> (visited on 07/29/2022).
- Kagraoka, Y. (Jan. 2005). "Modeling Insurance Surrenders by the Negative Binomial Model". In: *JAFEE International Conference*.
- Cox, S.H. and Y. Lin (2006). *Annuity Lapse Modeling: Tobit or not Tobit ?* en. Society of Actuaries.
- Kiesenbauer, D. (2012). "Main Determinants of Lapse in the German Life Insurance Industry". In: *North American Actuarial Journal* 16.1, pp. 52–73. DOI: [10.1080/10920277.2012.10590632](https://doi.org/10.1080/10920277.2012.10590632).
- Russell, D.T. et al. (2013). "An Empirical Analysis of Life Insurance Policy Surrender Activity". In: *Journal of Insurance Issues* 36.1, pp. 35–57. ISSN: 15316076, 23324244. URL: <http://www.jstor.org/stable/41946336> (visited on 07/29/2022).
- Sirak, A.S. (2015). "Income and unemployment effects on life insurance lapse". In: *Retrieved September 18, p. 2020*.

- Vasudev, M., R. Bajaj, and A.A. Escolano (2016). "On the Drivers of Lapse Rates in Life Insurance". en. Sarjana thesis, Barcelona, Spain: University of Barcelona.
- Nolte, S. and J.C. Schneider (2017). "Don't lapse into temptation: a behavioral explanation for policy surrender". In: *Journal of Banking & Finance* 79.C, pp. 12–27. URL: <https://EconPapers.repec.org/RePEc:eee:jbfina:v:79:y:2017:i:c:p:12-27>.
- Shamsuddin, S., I. Noriszura, and N. Roslan (May 2022). "What We Know about Research on Life Insurance Lapse: A Bibliometric Analysis". In: *Risks* 10, p. 97. DOI: [10.3390/risks10050097](https://doi.org/10.3390/risks10050097).
- Renshaw, A.E. and S. Haberman (1986). "Statistical analysis of life assurance lapses". In: *Journal of the Institute of Actuaries* 113, pp. 459–497.
- Milhaud, X., S. Loisel, and V. Maume-Deschamps (Dec. 2011). "Surrender triggers in Life Insurance: what main features affect the surrender behavior in a classical economic context?" In: *Bulletin Français d'Actuariat* 11.22, pp. 5–48. URL: <https://hal.archives-ouvertes.fr/hal-01985261>.
- Hwang, Y., L.F.S. Chan, and J. Tsai (2022). "On Voluntary Terminations of Life Insurance: Differentiating Surrender Propensity From Lapse Propensity Across Product Types". In: *North American Actuarial Journal* 26.2, pp. 252–282. DOI: [10.1080/10920277.2021.1973507](https://doi.org/10.1080/10920277.2021.1973507).
- Ćurak, M., D. Podrug, and K. Poposki (Sept. 2015). "Policyholder and Insurance Policy Features as Determinants of Life Insurance Lapse - Evidence from Croatia". In: *Economics and Business Review* 1 (15), pp. 58–77. DOI: [10.18559/ebrev.2015.3.5](https://doi.org/10.18559/ebrev.2015.3.5).
- Gemmo, I. and M. Gotz (2016). *Life insurance and demographic change: An empirical analysis of surrender decisions based on panel data*. ICIR Working Paper Series 24/16. Goethe University Frankfurt, International Center for Insurance Regulation (ICIR). URL: <https://ideas.repec.org/p/zbw/icirwp/2416.html>.
- Hu, S. et al. (2021). "A spatial machine learning model for analysing customers' lapse behaviour in life insurance". In: *Annals of Actuarial Science* 15.2, pp. 367–393. DOI: [10.1017/S1748499520000329](https://doi.org/10.1017/S1748499520000329).
- Azzone, M. et al. (2022). "A machine learning model for lapse prediction in life insurance contracts". In: *Expert Systems with Applications* 191, p. 116261. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2021.116261>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417421015700>.
- Gupta, S., D. Hanssens, et al. (Nov. 2006). "Modeling Customer Lifetime Value". In: *Journal of Service Research - J SERV RES* 9, pp. 139–155. DOI: [10.1177/1094670506293810](https://doi.org/10.1177/1094670506293810).
- Lemmens, A. and S. Gupta (2020). "Managing Churn to Maximize Profits". In: *Marketing Science* 39.5, pp. 956–973. DOI: [10.1287/mksc.2020.1229](https://doi.org/10.1287/mksc.2020.1229).
- von Mutius, B. and A. Huchzermeier (2021). "Customized Targeting Strategies for Category Coupons to Maximize CLV and Minimize Cost". In: *Journal of Retailing* 97.4. SI: Metrics and Analytics, pp. 764–779. ISSN: 0022-4359. DOI: <https://doi.org/10.1016/j.jretai.2021.01.004>. URL: <https://www.sciencedirect.com/science/article/pii/S002243592100004X>.
- Grinsztajn, L., E. Oyallon, and G. Varoquaux (2022). *Why do tree-based models still outperform deep learning on tabular data?* DOI: [10.48550/ARXIV.2207.08815](https://doi.org/10.48550/ARXIV.2207.08815).
- Routh, P., A. Roy, and J. Meyer (2021). "Estimating customer churn under competing risks". In: *Journal of the Operational Research Society* 72.5, pp. 1138–1155. DOI: [10.1080/01605682.2020.1776166](https://doi.org/10.1080/01605682.2020.1776166).
- Burrows, R. and J. Lang (1997). "Risk discount rates for actuarial appraisal values of life insurance companies". In: *Proceedings of the 7th International AFIR Colloquium*, pp. 283–307.
- Oh, S. et al. (2018). "A Study on the Estimation of the Discount Rate for the Insurance Liability under IFRS 17". In: *Journal of Insurance and Finance* 29.3, pp. 45–75. ISSN: 2384-3209. DOI: [10.23842/jif.2018.29.3.002](https://doi.org/10.23842/jif.2018.29.3.002).

- Blum, V. and P.E. Therond (2019). “Discount rates in IFRS: How practitioners depart the IFRS maze”. PhD thesis. Autorité des Normes Comptables.
- Laurent, J.P., R. Norberg, and F. Planchet, eds. (May 2016). *Modelling in life insurance - A management perspective*. en. 1st ed. European Actuarial Academy (EAA) Series. Cham, Switzerland: Springer International Publishing.
- Milhaud, X. and C. Dutang (Mar. 2018). “Lapse tables for lapse risk management in insurance: a competing risk approach”. In: *European Actuarial Journal* 8.1, pp. 97–126. DOI: [10.1007/s13385-018-0165-7](https://doi.org/10.1007/s13385-018-0165-7). URL: <https://hal.archives-ouvertes.fr/hal-01985256>.
- Cox, D.R. (1972). “Regression Models and Life-Tables”. en. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 34.2. JSTOR, pp. 187–220. URL: <http://www.jstor.org/stable/2985181>..
- Davidson-Pilon, C. (2019). “lifelines: survival analysis in Python”. In: *Journal of Open Source Software* 4.40, p. 1317. DOI: [10.21105/joss.01317](https://doi.org/10.21105/joss.01317).
- Akaike, H. (1973). “Information theory and an extension of the maximum likelihood principle”. en. In: *dans Second International Symposium on Information Theory*, pp. 267–281.
- Harrell, F.E. et al. (1982). “Evaluating the yield of medical tests”. In: *Jama* 247.18, pp. 2543–2546.
- Hanley, J.A. and B.J. McNeil (1982). “The meaning and use of the area under a receiver operating characteristic (ROC) curve”. en. In: *Radiology* Apr;143(1):29-36. PMID: 7063747. DOI: [10.1148/radiology.143.1.7063747](https://doi.org/10.1148/radiology.143.1.7063747)..
- Bou-Hamad, I., D. Larocque, and H. Ben-Ameur (2011). “A review of survival trees”. In: *Statistics Surveys* 5.none, pp. 44–71. DOI: [10.1214/09-SS047](https://doi.org/10.1214/09-SS047).
- Mantel, N. (1966). “Evaluation of survival data and two new rank order statistics arising in its consideration”. en. In: *Cancer Chemotherapy Reports. Part 1* 50, pp. 163–170.
- LeBlanc, M. and J. Crowley (1993). “Survival Trees by Goodness of Split”. In: *Journal of the American Statistical Association* 88.422, p. 457. ISSN: 0162-1459. DOI: [10.2307/2290325](https://doi.org/10.2307/2290325).
- Ishwaran, H., U.B. Kogalur, et al. (2008). “Random survival forests”. In: *The Annals of Applied Statistics* 2.3, pp. 841–860. DOI: [10.1214/08-AOAS169](https://doi.org/10.1214/08-AOAS169).
- Ishwaran, H., T.A. Gerds, et al. (2014). “Random survival forests for competing risks”. io. In: *Biostatistics* Oct;15(4):757-73. Epub 2014 Apr 11. PMID: 24728979 ; PMCID: PMC4173102. DOI: [10.1093/biostatistics/kxu010](https://doi.org/10.1093/biostatistics/kxu010)..
- Polsterl, S. (2020). “scikit-survival: A Library for Time-to-Event Analysis Built on Top of scikit-learn”. In: *Journal of Machine Learning Research* 21.212, pp. 1–6. URL: <http://jmlr.org/papers/v21/20-729.html>.
- Ishwaran, H. and U.B. Kogalur (2007). “Random survival forests for R”. In: *R News* 7.2, pp. 25–31. URL: <https://CRAN.R-project.org/doc/Rnews/>.
- Chinchor, N. (1992). “MUC-4 Evaluation Metrics”. In: *Proceedings of the 4th Conference on Message Understanding. MUC4 '92*. McLean, Virginia: Association for Computational Linguistics, pp. 22–29. ISBN: 1558602739. DOI: [10.3115/1072064.1072067](https://doi.org/10.3115/1072064.1072067).
- Brier, G.W. (1950). “Verification of forecasts expressed in terms of probability”. In: *Monthly Weather Review* 78.1, pp. 1–3. DOI: [https://doi.org/10.1175/1520-0493\(1950\)078<0001:VOFEIT>2.0.CO;2](https://doi.org/10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2). URL: https://journals.ametsoc.org/view/journals/mwre/78/1/1520-0493_1950_078_0001_vofeit_2_0_co_2.xml.
- He, H. and E.A. Garcia (2009). “Learning from Imbalanced Data”. In: *IEEE Transactions on Knowledge and Data Engineering* 21.9, pp. 1263–1284. DOI: [10.1109/TKDE.2008.239](https://doi.org/10.1109/TKDE.2008.239).
- Duchemin, R. and R. Matheus (Dec. 2021). “Forecasting customer churn: Comparing the performance of statistical methods on more than just accuracy”. en. In: *Journal of Supply Chain Management Science* 2.3-4, pp. 115–137.

- Breiman, L. et al. (1984). *Classification and Regression Trees*. Taylor & Francis. ISBN: 9780412048418. URL: <https://books.google.fr/books?id=JwQx-WOmSyQC>.
- Breiman, L. (2001). "Random Forests". English. In: *Machine Learning* 45.1, pp. 5–32. ISSN: 0885-6125. DOI: [10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324).
- Freund, Y. and R.E. Schapire (1996). "Experiments with a New Boosting Algorithm". In: *International Conference on Machine Learning*, pp. 148–156. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.51.6252>.
- Chen, T. and C. Guestrin (2016). "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: ACM, pp. 785–794. ISBN: 978-1-4503-4232-2. DOI: [10.1145/2939672.2939785](https://doi.org/10.1145/2939672.2939785).
- Pedregosa, F. et al. (2011). "Scikit-learn: Machine learning in Python". In: *Journal of machine learning research* 12.Oct, pp. 2825–2830.
- Guelman, L. and M. Guillen (Feb. 2014). "A causal inference approach to measure price elasticity in Automobile Insurance". In: *Expert Systems with Applications: An International Journal* 41, pp. 387–396. DOI: [10.1016/j.eswa.2013.07.059](https://doi.org/10.1016/j.eswa.2013.07.059).
- Verschuren, R.M. (2022). "Customer price sensitivities in competitive insurance markets". In: *Expert Systems with Applications* 202, p. 117133. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2022.117133>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417422005309>.
- Verbraken, T., W. Verbeke, and B. Baesens (Jan. 2012). "A Novel Profit Maximizing Metric for Measuring Classification Performance of Customer Churn Prediction Models". In: *Knowledge and Data Engineering, IEEE Transactions on* 25. DOI: [10.1109/TKDE.2012.50](https://doi.org/10.1109/TKDE.2012.50).
- Stripling, E. et al. (2018). "Profit maximizing logistic model for customer churn prediction using genetic algorithms". In: *Swarm and Evolutionary Computation* 40, pp. 116–130. ISSN: 2210-6502. DOI: <https://doi.org/10.1016/j.swevo.2017.10.010>. URL: <https://www.sciencedirect.com/science/article/pii/S2210650216301754>.

Part IV

Longitudinal setting

Chapter 9

Contributions of Part IV

Chapter 10

Introduction to longitudinal studies

Chapter 11

Longitudinal models

- 11.1 (Semi)-parametric models 108
 - 11.1.1 Mixed Models (MMs) and extensions 108
 - 11.1.2 Cox model and time-varying covariates 108
- 11.2 Overview of TBMs in a longitudinal setting 109
 - 11.2.1 Dynamic regression on a longitudinal outcome with tree-based models 109
 - 11.2.2 Longitudinal survival analysis with tree-based models 112
 - 11.2.3 Metrics 114

Chapter 12

A longitudinal ML framework for lapse management in life insurance

- 12.1 Introduction 115
- 12.2 Longitudinal framework 120
 - 12.2.1 Preliminaries on time-varying covariates and longitudinal notations 120
 - 12.2.2 LMS longitudinal framework 123
- 12.3 Application 126
 - 12.3.1 Data 126
 - 12.3.2 Application: survival step 127
 - 12.3.3 Application: regression step 131
- 12.4 Conclusion, limitations and future work 135

Bibliography

9. Contributions of Part IV

This part is based on the article “ A longitudinal Machine Learning framework for lapse management in life insurance ”, submitted in the Annals of Actuarial Science¹. Part III of this thesis aimed to introduce a PH-centered and profit-driven lapse management framework with temporal and survival considerations. Part IV aims to take this framework one step further by adapting it to longitudinally structured data and making it more dynamic. This work contributes to the fields of actuarial science, management science, and business economics in several ways:

Contributions 5: New longitudinal LMS framework

Development of a new theoretical framework

The first contribution of this Part is the development of a new longitudinal Lapse Management Strategy (LLMS) framework. This new approach enhances the existing lapse management strategies by incorporating time-informed insights into the analysis. By considering time-varying features and targets, the framework offers a more precise evaluation of the stakes of targeting individuals, which is crucial for life insurers to improve their profitability and understand the risks associated with their global portfolio. Importantly, this framework highlights the value of utilizing the complete past trajectory of policyholders, an often overlooked approach.

Contributions 6: Use of longitudinal TBM in life insurance

Application of existing theories or methods in a new context

The second key contribution is the application of existing longitudinal tree-based models, specifically left-truncated and right-censored (LTRC) trees and forests, and mixed-effect tree-based regression models, in a life insurance context. Transposing these existing models into a new application context is a substantial contribution to actuarial science. By applying these models to lapse management, the study enhances the precision of retention targeting and profitability analysis. This innovative approach not only opens up new research avenues but also optimizes the use of data available to insurers, thereby potentially sparking industry-wide advancements

¹See Valla 2023.

10. Introduction to longitudinal studies

We define *longitudinal data* as the data that are obtained from repeated observations of individuals over time¹. This data type is instrumental in creating more accurate and reliable predictive models, as it provides a broader and more comprehensive perspective of the variables and their dynamic relationships. In contrast, cross-sectional data, which is data collected from multiple subjects at a single point in time, provides a snapshot view of the variables. While it can provide valuable insights, it lacks the depth and temporal context that longitudinal data offers. Similarly, time-series data, which tracks the evolution of aggregated numerical features over time misses the individualized information available (see E. Frees and Miller 2004). In a longitudinal study, individuals identified by a unique ID are observed at different time points where features are repeatedly measured: there exist several observations per subject, but every observation is associated with one and only one subject.

Remark 10.1

Three types of covariates emerge from a longitudinal study: baseline covariates (that are non-varying by nature), exogenous longitudinal covariates (their future paths are not directly affected by the outcome^a), and endogenous longitudinal covariates (their future paths are affected by the outcome). As an example, when predicting the occurrence of a lapse for a life insurance policy, the future values of the discount rate are not affected by the outcome: the discount rate is an exogenous longitudinal covariate. Conversely, the outstanding amount of the policy, for instance, is affected by the occurrence of a lapse: the outstanding amount of the policy is an endogenous longitudinal covariate. It is critical to separate the two, as the former is easily dealt with in most modelizations, whereas the latter needs specific treatment (Kalbfleisch and Prentice 2002, Section 6.3).

^aThis distinction is also utterly relevant in a survival context, where the outcome of interest is the occurrence of an event.

In fields such as actuarial science, the importance of longitudinal data cannot be overstated. For instance, it allows businesses to understand not just what is happening at a given point in time, but also how their portfolio and marketing metrics evolve and interact over time. This could include trends, seasonality, and the impact of specific events or interventions. Despite its clear benefits, the actuarial literature is scarce on the use of longitudinal data for forecasting. This is surprising, given that incorporating both cross-sectional and time-based information can significantly enhance predictive accuracy. It provides a more holistic view of the underlying

¹see the complete description of Rizopoulos 2012

patterns and trends, which can inform more effective and strategic decision-making. Moreover, longitudinal studies play a pivotal role in tracking changes in individuals or groups over time. These studies are particularly important in fields like social science, and medical research, as they can help to understand the influence of various environmental and social factors on these changes. For example, a longitudinal study could track a group of individuals' health habits over several years to determine the long-term impact of these habits on their health expenses. This would not be possible with cross-sectional data, which would only provide a snapshot of the individuals' health and expenses at a specific point in time. The time-dynamic property is the main distinguishing feature of longitudinal studies as it aims at modeling the evolution of the response variable over time, which is why longitudinal methods are used to answer research questions such as: is there a systematic change over time? Can we compare the trajectories of different subjects for the same response variable? How does a change in a covariate affect a change in the outcome? The longitudinal response can either be a discrete, continuous, or a time-to-event outcome depending on the research question.

Remark 10.2

Even in the presence of a time-to-event outcome, the fact that subjects are being followed up and covariates are being measured at different times makes the analysis longitudinal. The nature of a study (survival and/or longitudinal) is determined by the nature of the outcome of interest and by the structure of the dataset. Thus, studying a time-to-event response with observations repeatedly measured over time falls under both longitudinal and survival analyses.

While both cross-sectional and longitudinal data have their merits, the latter offers a more thorough understanding of variables and their interrelationships over time. As such, the use of longitudinal data can significantly enhance the relevance, accuracy, and utility of predictive models, providing valuable insights for informed decision-making (see Laird 2022). Actuarial science is a field where research questions that require longitudinal methods are relatively common yet rarely explored with nonparametric approaches. Existing works adopting a Machine Learning approach on subjects like insurance pricing (Henckaerts, Côté, et al. 2021), telematics (Pesantez-Narvaez, Guillen, and Alcañiz 2019; Boucher and Turcotte 2020; Henckaerts and Antonio 2022), asset-liability management (Gu, Kelly, and Xiu 2020) and lapse or death prediction (Loisel, Piette, and Tsai 2021) already exist and would benefit from a time-varying longitudinal framework. Feeding those models with dynamic policyholder's information, contact information, or financial flows² as - possibly time-varying - covariates is a potentially vast source of information. Tree-based models are a suitable option for analyzing longitudinal time-dynamic actuarial data, and understanding how to take advantage of them could benefit the field.

As we already noted, it is naturally common in biomedical research to face regression or classification problems involving time-varying covariates, time-to-event outcomes, correlated data, or repeated discrete measurements of individual attributes over time. One can think of follow-up studies of patients with a pathology where we can observe the evolution of the different biomarkers along with the evolution of the pathology. Yet, where longitudinal studies prevail in medical research, it is still rarely tried in actuarial research.

²e.g., the outstanding amount of a life insurance policy, payments, premiums, claims, fees, profit sharing, macro-economic metrics, etc...

Remark 10.3

An important distinction between biomedical and actuarial studies needs to be addressed. When dealing with biomarkers, we can only observe a continuously evolving attribute at discrete time points. Moreover, these discrete observations can be subject to measurement errors and delays in the measurements. In actuarial science, the majority of the time-varying attributes are financial flows that are truly evolving discretely and actuaries obtain those evolutions without measurement errors (or extremely rarely) and with a precise date of their occurrence. It is important to keep this distinction in mind as we progress through this work: in most actuarial studies, a discrete longitudinal framework is not an oversimplification of reality.

This is precisely the aim of this part of the thesis: gathering the different ideas in the existing literature concerning the integration of time-varying covariates and longitudinal response into tree-based models and bringing it to the actuarial literature. The rest of this part introduces prerequisite knowledge about longitudinal analysis, then it chronologically retraces how the non-survival or survival tree-based models deal with the presence of longitudinal data and time-varying covariates. Eventually, it displays an article that proposes a longitudinal framework and application for lapse behavior analysis in life insurance.

11. Longitudinal models

There are many examples of parametric approaches used to analyze longitudinal data. In this chapter, we will not dive into the mathematical details of each approach but rather give insights and references of two major modeling techniques, mixed-models, and time-varying Cox model, for regression and survival analysis respectively. Regarding non-parametric approaches, Section 11.2 retraces the history of tree-based longitudinal analysis, whether in a survival context or not.

11.1 (Semi)-parametric models

11.1.1 Mixed Models (MMs) and extensions

Mixed models, also known as mixed-effects or multilevel models, are statistical tools used to analyze data with hierarchical structures or dependencies among observations. They combine fixed effects (variables with consistent effects across all observations) and random effects (sources of variation between subjects) within a single framework. They're employed in research problems involving clustered data, longitudinal studies, and situations where accounting for both individual-level and group-level variations is necessary for accurate analysis and inference. Before going any further, it is important to stress that Linear Mixed Models (LMMs) are simple but very common in the literature in general but also in the actuarial field (see Antonio, Beirlant, et al. 2006; Antonio and Beirlant 2008). The idea behind LMMs is to extend the classical linear regression model to the cases where there is a dependence structure in the data. The presence of longitudinal data is a clear example where different observations for a given subject are correlated. In that case, a classical linear regression - which makes the hypothesis of independence between observations - would clearly yield unsatisfying results. LMMs assume a baseline behavior common to every subject, but every individual deviates from it in a specific way. On the one hand, the baseline behavior is described by the *fixed effects*. On the other hand, the individual deviations are described by the *random effects*. In the end, this models the effects of the covariates among subjects and among observations within subjects. The random effects' variance-covariance matrix accounts for the correlation structure of the observation. It can either be set explicitly with assumptions (autoregressive structure or compound symmetry), or it can be left unspecified. Eventually, the random effects b_i can be inferred once the model parameters are estimated. LMMs may have explicit analytic estimators, whereas the more complex and flexible Generalized Linear Mixed Models (GLMMs) require numerical likelihood optimization.

11.1.2 Cox model and time-varying covariates

The Cox model can easily be extended to time-varying covariates as they can be directly incorporated into the hazard function (given by equation A.8). To be more precise, in light of remark 10.1, this model can be easily extended to handle exogenous time-dependent covariates

but not endogenous ones as it has been proven to lead to biased results (see Rizopoulos 2012). As stated by L. Fisher and D. Lin 1999, this is one of the numerous limitations to this model and the concrete application of a time-varying Cox model is challenging to handle and can easily lead to erroneous inferences. We can refer to the works of Meyer 1990, L. Fisher and D. Lin 1999, and Bover, Arellano, and Bentolila 2002, for results and concrete applications of such modelizations.

A complete implementation of the Cox model, generalized to handle time-varying covariates, time-varying effects, and more (competing risks, recurring events...), has been brought to the literature by Scheike and Martinussen 2006; Scheike and M. Zhang 2011 in the R package `timereg`.

11.2 Overview of TBMs in a longitudinal setting

The review focuses on techniques involving regression trees that would be trained on a dataset with both time-varying covariates and fixed covariates and with a time-varying outcome. The outcome of interest in longitudinal studies can be a continuous numerical outcome or a time-to-event response. It is critical to stress that those types of responses are different. Estimating the former with a TBM requires a classical regression approach and is direct as a tree will be built by optimizing a loss function directly depending on the outcome. The latter requires a survival analysis method as it is usually carried out in the presence of censoring data and predicting such response is indirect as a tree will be built by optimizing a loss model depending on the survival distributions of the subjects. We will discuss these differences later and will divide our overview into several sections accordingly. Longitudinal models are compared on their performance for statistical inference, their ability to provide predictions for any individual after her most recent observation, and to highlight the relevant covariates as well as the relevant time points. This overview section aims to review the background literature about longitudinal analysis, focusing on tree-based models, and will be organized into three subsections. In a dynamic longitudinal framework, we will first see how tree-based models can handle regression problems, in a second time, we will show how longitudinal TBMs adapt to a survival analysis context, eventually, we will briefly mention the metrics that can be used to evaluate such models. The goal is mainly to capture for each approach, what strategy or loss function is used, what prediction problems are being solved, or what limitation still stands.

11.2.1 Dynamic regression on a longitudinal outcome with tree-based models

Tree-based models were also used with longitudinal data, and we will keep our focus here on single tree models even if it can be observed that some of them can be or have been extended with bagging and boosting techniques. In this section, most models aim to predict a continuous numerical response in the presence of covariates that can be time-varying depending on the model.

The most naïve model would be a static tree-based model (such as the ones described in Section 4.2.1), trained on all observations in the dataset without taking the correlation between observations of the same subject into account. As stated by Segal 1992a, this would simply ignore the capital aspect of dealing with longitudinal data: *The covariation induced by making several observations of some continuous response on the same unit, as occurs with repeated measures designs, cluster designs, and longitudinal studies, poses data analytic problems. Analysis of such designs that ignore the covariance structure is known to produce incorrect variance estimates.*

We found other attempts in the literature, based on the idea that every time-varying covariate could be summarized by a small number of parameters. For instance, one could think of only keeping the mean value of every longitudinal covariate - or similarly the median, the baseline value, or the most recent one - ignoring all the remaining information. This obviously leads to a loss of precious data. A similar idea is to regress every longitudinal covariate against time and possibly other covariates within subjects to include the regression's parameters -intercept and slope - as baseline covariates. If the longitudinal covariates are all strongly linearly associated with time, which is rarely the case in practice, this kind of alternative solution can be relevant. Of course, that idea can be extended to more complex regressions, with the recent work of Kundu and Harezlak 2019 that developed the concept of resuming information contained in the longitudinal covariates by a combination of splits on baseline covariates and implemented it in the R package LongCART. But the loss of information during the process stands. Moreover, the number of measurements per subject in real datasets can be too small to obtain satisfying regression parameters.

Except that, Segal 1992a and De'Ath 2002 proposed independently the first applications that clearly define an extension to the CART method that considers the correlation problem. Those applications were designed to run a regression on data with fixed covariates and a longitudinal outcome. They both suffered limitations as they were intended for cases where all the subjects were measured at the same observation times, with the same interval between them. On the one hand, Segal's model for regression trees consisted of imputing a covariance structure in the split procedure. This led to many theoretical and practical questions about the definition of that covariance structure and the complexity of the computations. On the other hand, De'Ath 2002 procedure simply modified the CART algorithm by allowing it to train on multivariate data, considering a matrix containing all the observations for one subject as a single training example in the tree. Allowing that was done by using the gain of MSE as a splitting criterion and replacing the 1-dimensional mean in the MSE with a multidimensional mean modified with a covariance structure; the prediction given by the tree would then be the multidimensional mean of the observations in the terminal nodes. In both cases, those methods can be seen as fitting a model to the longitudinal outcome at every node as part of the splitting criterion. More recent works by Larsen and Speckman 2004 and Hsiao and Shih 2007 followed and improved the idea of De'Ath by redefining the node impurity measure with the Mahalanobis distance and estimating the covariance matrix from the whole data set.

For a complete historical review, it is worth mentioning that other works extended the idea of Segal 1992a, to binary responses and classification trees (H. Zhang 1998), in a clustering context using deviance as a goodness-of-fit criterion for partitioning (Abdolell et al. 2002) and then to every type of longitudinal response - not only continuous or binary - using generalized estimating equations (GEE) (see Lee 2005; Lee et al. 2005; Lee 2006) in the splitting process. The latter approach also extends an original idea developed by Chaudhuri et al. 1995 and can be seen as fitting a GEE with a standard maximum likelihood at each node of the tree. Two groups are formed depending on the sign of the model residuals at each node, the covariate selected to split the node is the one with the largest absolute t-statistic - maximizing the separation between the child nodes- and the threshold at which the covariate is split is a weighted average of the covariate means among the two groups.

Finally, other approaches such as Ritschard and Oris 2005 and more recently Moradian et al. 2021 applied trees to data with longitudinal covariates, the former with longitudinal covariates and categorical response, the latter with longitudinal covariates and a time-to-event response by

using lagged response values as potential predictors, but still not treating either the outcome or the covariates as inherently dynamic with time.

All of the methods previously cited, except the work of Moradian et al. 2021, cannot be used to predict the future outcome trajectories of a subject. This is a significant limitation that results from the fact that it would require observations from future periods to compute the means for said periods. Lastly, none of these procedures can satisfyingly handle time-varying covariates - this particular topic is discussed in a dedicated section in Segal's work.

Sela and J.S. Simonoff 2012, Fu and J.S. Simonoff 2015 as well as Galimberti and Montanari 2002 describe a procedure to build regression trees through an iterative two-step process. This idea is a direct extension of the two-step fitting procedures used for mixed-effect models first described by Harville 1977, then developed by Laird and Ware 1982, and later, G. Verbeke and G. Molenberghs 2000 with the Expectation-Maximization (EM) algorithm. Hajjem, Bellavance, and Larocque 2011a; Sela and J.S. Simonoff 2012; Capitaine 2020; Capitaine, Genuer, and Thiébaud 2021 then Hajjem, Bellavance, and Larocque 2014a; Fu and J.S. Simonoff 2015 - with their Random-Effects Expectation-Maximization tree (RE-EM tree) procedure - share the same core idea as Galimberti and Montanari 2002 but differ from the latter on some key points. Their method consists of assuming a mixed model for the longitudinal outcome, estimating the fixed effect parameters with a tree-based model, and inferring the random effect parameters. They estimate the random effects of a mixed model in the first step, then construct a regression tree with the fixed-effect covariates on the original outcome excluding the estimated random effect. The idea is to repeat these two steps until convergence of the random effects, similarly to the two-step well-known EM optimization procedure. Details, along with a general pseudo algorithm for such mixed effect tree-based models (METBM) can be found in Chapter 12.

Galimberti and Montanari's method cannot yield predictions for subjects used to train it. This critical limitation was clearly identified and addressed by Sela and Simonoff and then by Fu and Simonoff, and their model allows predictions for new subjects and future observations of subjects used in the training process. More importantly for our overview, they also allow the presence of time-varying covariates. We can note that similarly to most of our references, observations for a given subject can be spread across different leaves and branches of the tree. The predictive power of some METBM was compared to linear mixed models and CART without random effects. The original RE-EM based on CART outperformed both for large datasets and had comparable results with linear mixed models for smaller ones. In contrast, the unbiased extension seems to outperform linear mixed models and CART regardless of the dimensions of the dataset. We will consider RE-EM and its extension state-of-the-art for numerical regression problems, with time-varying covariates and a longitudinal outcome. This method has been implemented in the R package `REEMtree`. In Mixed Effects Regression Trees (MERT, see Hajjem, Bellavance, and Larocque 2011b), the tree-based model is a single regression tree, in Mixed Effects Random Forest (MERF, see Hajjem, Bellavance, and Larocque 2014b), it is a random forest, whereas in RE-EM (Sela and J.S. Simonoff 2012; Fu and J.S. Simonoff 2015) it can be both. The very recent works of Devaux (see Devaux 2022; Devaux, Genuer, and Peres 2022; Devaux, Helmer, et al. 2023; Devaux, Proust-Lima, and Genuer 2023) contribute to this mixed-effect tree-based model framework by adding a stochastic term in the MM. Eventually, works such as Wei et al. 2020 combine mixed effects models with regression splines to better capture non-linear trajectories among the longitudinal covariates.

Independently, Eo and Cho 2014 proposed a model called mixed-effects longitudinal tree (MELT)

and able to handle longitudinal response. The original idea is to fit a mixed-effect model at each tree node. The sum of the squared difference between subject-specific slopes and the common slope for all the subjects is then considered as an impurity measure for each node. The selected split is the one that maximizes the impurity gain, in other words, the difference between the parent node impurity and the sum of the child nodes' impurities. The algorithm, available in the R package `me1t`, is able to handle time-varying covariates. As in Segal 1992b, each time-varying covariate is regressed against time, and then its regression parameters are considered as fixed covariates. If time-varying covariates are categorical, they are transformed into binary covariates, and the coefficients of a logistic regression are used. Even later, we found references to an extension of RE-EM by Simonof 2016, which modified the regression tree built in the iteration step of RE-EM by estimating a regression tree with a linear function of time ($\beta_0 + \beta_1 \times t$) at each node instead of a constant value. This results in a MODel-basEd RaNdom effects tree, or a MODERN tree but no implementations of this algorithm exist, to the best of our knowledge.

11.2.2 Longitudinal survival analysis with tree-based models

Survival analysis methods with tree-based models have been detailed in Section 5. Indirectly studying the time-to-event outcome through its hazard function is how the vast majority of models handle the time-dependent outcome in survival analysis. We can now focus on how survival trees can handle time-varying covariates, the actual practical difficulty here.

Bacchetti and Segal 1995 and Huang, Chen, and Soong 1998 had the common idea to allow every subject to be potentially divided into pseudo-subjects at each tree node. Let $x_k^{(i)}(t)$ be a numerical time-varying covariate. For a regression tree, the splitting rule at a node would then be $x_k^{(i)}(t) \leq s^1$. A subject for which this rule is true $\forall t$ will go to one child node without any ambiguity. On the other hand, the general case where the rule is true for some periods but false for anywhere else is unclear and needs to be addressed. The simple idea was that the periods where the splitting rule is true would go to the left node and the other to the right node, thus dividing one subject into several pseudo-subjects. This approach cleverly addresses the time-handling issue but doesn't answer the correlation problem between several observations of the same subject - we could argue that it makes it worse by maintaining the correlation between observations and adding correlation between pseudo-subjects - and creates left-truncated observations that need special treatment. Generally speaking, this process creates right-censored and left-truncated (LTRC) data. Any subject can potentially be spread in many different leaves of the tree - even if, at any fixed time, any subject will fall into one unique leaf. The time dynamic is here considered horizontally within the tree structure.

This left-truncation issue was addressed by Bacchetti and Segal 1995 by modifying two-sample tests - like the log-rank test - to handle left-truncated data. Huang, Chen, and Soong 1998's method for handling the LTRC pseudo-subjects observations assumes a splitting criterion derived from the log-likelihood of a model, which presupposes for each subject that the survival time distribution is piecewise exponential. Instead of using the classical log-rank statistic to split every node of a survival tree, a modified log-rank statistic could be used. In 2016, Fu and J.S. Simonoff 2016a proposed a model based on those ideas: they allowed subjects to be divided into pseudo-subjects and adjusted the log-rank test in the splitting procedure to accommodate for LTRC data. This last method has been implemented in the R package `LTRCtrees`. It has been later included in an ensemble framework (see W; Yao et al. 2022) and we will consider it a state-of-the-art method for tree-based survival analysis with time-varying covariates.

¹or $x_k^{(i)}(t) \in S$ for a categorical time-varying covariate

Bou-Hamad 2009 presented a discrete-time survival tree-based procedure able to account for potentially time-varying effects of fixed covariates, meaning that the covariates are fixed but can have different effects depending on time. That can be very insightful in terms of interpretability as this allows us to analyze what attributes most influence survival and at which time points. They then generalized this approach to time-varying covariates (Bou-Hamad, Larocque, and Ben-Ameur 2011) again using the pseudo-subject division idea already discussed. As with most of the survival tree algorithms in the literature (see Section 5), the splitting criteria in the previously described methods are not based on minimizing a loss function but on the maximization of a survival dissimilarity measure. Each split hence separates the subjects with different survival profiles.

It is also worth mentioning that earlier experimental work by Breiman 2002 on survival trees grown used a splitting criterion in which nodes can split either on time or covariates, giving insights into the time-varying effects of the covariates on survival and on the time points at which those effects are the most influential. To the best of our knowledge, no further publications, implementations nor developments following this idea were tried after that. Similarly and independently, Xu and Adak 2001; Xu and Adak 2002 also proposed a method where a tree is grown only to find relevant time points in the data. In this work, they were able to detect at what period the effect of covariates on the outcomes is stronger, thus allowing their model to have time-varying effects but not to handle longitudinal covariates. The model then fits a piecewise Cox model on time intervals found by the tree procedure. This last approach highlights the need for methods able to analyze relevant time points in the data.

More recently, Kundu and Harezlak 2019 extended the idea of resuming information contained in the longitudinal covariates by a combination of splits on baseline covariates, and J. Lin, Li, and Luo 2021a uses a set of scores for every longitudinal covariate and then grows a tree on those scores. Those complex approaches can be related to the naïve strategies we mentioned at the beginning of this section but are way more sophisticated and enlightened. We already discussed Kundu and Harezlak 2019's work in the previous section but it is worth mentioning that it was adapted to survival analysis with the SURVCART algorithm in the R package LONGCART.

Lastly, recent works on this topic have to be mentioned. RSF, described in Section 5.1.3, is a TBM designed for survival analysis and was originally unable to handle longitudinal data. It was then extended to competing risks (Ishwaran et al. 2014) and very recently to handle longitudinal covariates (see Wongvibulsin, Wu, and Zeger 2020; J. Lin, Li, and Luo 2021b; J. Lin, Li, and Luo 2021a; Moradian et al. 2021; Pickett, Suresh, and Campbell 2021). This extension of RSF (J. Lin, Li, and Luo 2021b; J. Lin, Li, and Luo 2021a, in particular) is based on summarizing all longitudinal covariates trajectories, for every subject, with univariate scores that can then be used as baseline covariates by a regular RSF. As seen in other methods, those scores aim to characterize the changing pattern of the time-varying covariates, and they can yield individual future survival predictions. This method has been implemented in the R package `funest`. The goal of Moradian et al. 2021's procedure is to estimate the hazard of a subject at some future time points u for $u = t + 1, t + 2, \dots, T$ until a horizon T . Moradian here performs dynamic prediction by calibrating a hazard function on every combination of observation time and prediction time (t, u) by using lagged response values as potential predictors. This is closely related to the approach of Pickett, Suresh, and Campbell 2021, where RSF is adapted to landmark procedure, thus allowing dynamic prediction at given and presupposed horizon times.

All the tree-based models handling time-varying covariates described in this Chapter either ig-

nore the intrinsic time-dynamic dimension of the data or treat it with the pseudo-subject approach. Further notations and mathematical insights of such models, especially `LTRCtrees` and its extensions are further detailed in Chapter 12. Alternatives to the pseudo-subject approach are discussed in Chapter 14.

11.2.3 Metrics

Models built for longitudinal analysis require a slightly different set of evaluation metrics than those used for cross-sectional analysis. The fundamental reason for this is the time-dependent nature of longitudinal data, which introduces autocorrelation and potential non-stationarity into the data. This can violate the assumption of independence typically made in cross-sectional models (see Singer and Willett 2003).

Therefore, metrics that can handle these temporal dependencies are necessary for non-survival longitudinal analysis. One can think of time-series cross-validation (see Hyndman and Athanasopoulos 2018 and Section 4.1.2), or the mean absolute scaled error (MASE) (see Hyndman and Koehler 2006). In MM or random effects models for regression, individual differences or trajectories over time are modeled via random effects. In this case, the cross-sectional metrics (see Section 4.1.2) can be used without further modifications.

In survival longitudinal analysis, however, very recent works were developed to adapt usual survival evaluation metrics to the longitudinal dynamic context. Modifications of the Brier-Score are detailed in W; Yao et al. 2022. We also use, justify, and detail the formulas for the time-dependent Brier-Score (td-BS) in Chapter 12. To be exhaustive in regards to the metrics introduced in Section 5.2, we refer the astute reader to the adaptations of the C-index (see Hartman et al. 2023) and the AUC (see Lambert and Chevret 2016; van Geloven et al. 2021).

12. A longitudinal ML framework for lapse management in life insurance

Abstract. *Developing an informed lapse management strategy (LMS) is critical for life insurers to improve their profitability, and gain insight into the risk of their global portfolio. When designing a retention campaign, prior research in actuarial science (see Loisel, Piette, and Tsai 2021; Valla, Milhaud, and Olympio 2023) has shown that targeting policyholders by maximizing their individual Customer Lifetime Value is more advantageous and informative for the insurer than targeting all those who are likely to lapse. However, most existing lapse analyses are not taking advantage of the fact that features and targets may vary over time. We propose to define a longitudinal LMS framework, that provides time-informed insights and leads to increased precision in targeting. The strengths and flaws of this new methodology are discussed in various settings. This paper contributes to the field of lapse analysis for life insurers and highlights the importance of using the complete past trajectory of policyholders, which is often available in insurers' information systems but has yet to be exploited.*

Key words: *Lapse management strategy, longitudinal, Machine learning, life insurance, Customer lifetime value*

12.1 Introduction

In this article, we present a novel methodology developed to address the retention challenges faced by life insurers in a French insurance portfolio consisting of equity-linked whole-life insurance policies (see Hardy 2003 for an extensive review on such insurance products). Whole-life insurance provides coverage for the entire lifetime of the insured individual, rather than a specified term and when contracting such an insurance plan, policyholders can choose how the outstanding face amount of their policy is invested between “euro funds” and unit-linked funds. Understanding the fundamental differences between these investment vehicles is essential to comprehending the dynamics of the whole-life insurance market. For savings invested in euro funds, the coverage amount is determined by deducting the policy costs from the total premiums paid, the financial risk associated with these funds is borne by the insurance company itself. The underlying assets of euro funds primarily consist of government and corporate bonds, limiting the potential returns, thus the performance of these funds is directly influenced by factors such as the composition of the euro fund, fluctuations in government bond yields, and the insurance company's profit distribution policy. Additionally, early termination of the policy by the policyholder incurs exit penalties, as determined by the insurance company. In contrast, unit-linked insurance plans operate under a different framework. The coverage amount is determined by the number of units of accounts held by the policyholder, and the financial risk is assumed by the policyholders themselves. Unit-linked funds offer a wide range of underlying assets, among all types of financial instruments, enabling potentially unlimited performance based on the market performance of these assets. The investment strategy is tailored to the specific investment

objectives of the policyholder and while certain limitations exist in terms of asset selection, policyholders generally face no exit penalties for their underlying investments.

Lapse is a critical risk for whole-life insurance products (see Bacinello 2005 or MacKay et al. 2017), thus, policyholders represent a critical asset for life insurers. Therefore, the ability to retain profitable ones is a significant determinant of the insurer’s portfolio value (and more generally a firm’s value, see Gupta, Lehmann, and Stuart 2004). If some historical explanations for lapse are liquidity needs (Outreville 1990) and rise of interest rates, it also appears that individual characteristics are also insightful (see Eling and Kochanski 2013 for a complete review). Consequently, policyholder retention is a strategic imperative, and lapse prediction models are a crucial tool for data-driven policyholder lapse management strategy in any company operating in a contractual setting such as a life insurer. We build an extension of the framework of Valla, Milhaud, and Olympio 2023, that defines an LMS as follows:

Definition 2 (Lapse management strategy (LMS)). *A lapse management strategy for a life insurer is modeled by offering an incentive $\boldsymbol{\eta} = (\eta^{(1)}, \dots, \eta^{(N)})$ to policyholders $(1, \dots, N)$. Their policies yield a profitability ratio of $\mathbf{p} = (p^{(1)}, \dots, p^{(N)})$. The incentive is accepted with probability $\boldsymbol{\gamma} = (\gamma^{(1)}, \dots, \gamma^{(N)})$. and contacting the targeted policyholder has a fixed cost c . A targeted subject who accepts the incentive will be considered as an “acceptant” who will never lapse, and her probability of being active at year $t \in [0, T]$ is denoted $r_{\text{acceptant}}(t)$. Conversely, a subject who refuses the incentive and prefers to lapse will be considered as a “lapse”, and her probability of being active at year t is denoted $r_{\text{lapse}}(t)$. The parameters $(\mathbf{p}, \boldsymbol{\eta}, \boldsymbol{\gamma}, c, T)$ uniquely define a lapse management strategy, while $r_{\text{acceptant}}(t)$ and $r_{\text{lapse}}(t)$ need to be estimated from the portfolio.*

Our goal is not only to model the lapse behavior but also to select which policyholder to target with a given retention strategy to generate an optimized profit for the insurer. Such a lapse management strategy requires estimating what can be considered as the future profit generated by a given policyholder: the individual customer lifetime value or CLV (see Donkers, P. Verhoef, and Jong 2007). The individual CLV over horizon T , for the i -th subject aims at capturing the expected profit or loss that will be generated in the next T years and is expressed as follows, in the general time-continuous case:

$$CLV^{(i)} = \int_{\tau=0}^T \frac{p^{(i)}(\tau) \cdot F^{(i)}(\tau) \cdot r^{(i)}(\tau)}{e^{d(\tau) \cdot \tau}} d\tau, \quad (12.1)$$

with the profitability ratio $p^{(i)}(t)$ being represented as a proportion of the face amount, $F^{(i)}(t)$, observed at time t . The conditional individual retention probability, $r^{(i)}(t)$, is the i -th observation’s probability of still being active at time t . In practice, the individual CLV is often discretized and computed as a sum of annual flows, thus with t , the time in years¹,

$$CLV^{(i)}(\mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}^{(i)}, \mathbf{d}, T) = \sum_{\tau=0}^T \frac{p_{\tau}^{(i)} \cdot F_{\tau}^{(i)} \cdot r_{\tau}^{(i)}}{(1 + d_{\tau})^{\tau}}. \quad (12.2)$$

Equation 12.2 is primarily used in the marketing and actuarial literature (see Berger and Nasr 1998 or Loisel, Piette, and Tsai 2021). If we only consider the future T years of CLV, after time

¹Denoted as a subscript of $p^{(i)}$, $F^{(i)}$ and $d^{(i)}$ for simplicity sake.

t , the sum becomes

$${}^FCLV^{(i)}(t, \mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}^{(i)}, \mathbf{d}, T) = \sum_{\tau=t+1}^{T+t} \frac{p_{\tau}^{(i)} \cdot F_{\tau}^{(i)} \cdot r_{\tau}^{(i)}}{(1 + d_{\tau})^{\tau}}. \quad (12.3)$$

Given an LMS, a policyholder can either be likely to accept the offer of an incentive and behave with an “*acceptant*” risk profile or she can be likely to reject the offer and thus behave with a “*lasper*” risk profile. In this context, *acceptants* and *laspers* will not generate the same CLV as their respective retention probabilities differ. The CLV of an *acceptant* or a *lasper* are estimated using respectively $r_{acceptant}^{(i)}$ and $r_{lasper}^{(i)}$ as retention probabilities. All the expected future financial flows are discounted, with d_t representing the annual discount rate at year t . In definitive, ${}^FCLV^{(i)}(t, \dots)$ represents the future T years of profit following observation at time t .

The analysis of a lapse management strategy, as described in Loisel, Piette, and Tsai 2021, then in Valla, Milhaud, and Olympio 2023, is a two-step framework. The first step consists of using the insurer’s data to train survival models and predict yearly retention probabilities for any subject in the portfolio: we will refer to it as the *survival step*. The retention probabilities are used to compute an individual CLV-based estimation of the profit generated from targetting any policyholder. This estimation is eventually used as a response variable to fit a model predicting which kind of subject is likely to generate profit for the insurer: we will refer to it as the *regression step*. As in Ascarza et al. 2018 or Guelman, Montserrat, and Pérez-Marín 2012, the goal of such a CLV-based methodology is not only to model the lapse behavior but rather to select which policyholder is worth targetting with a given retention strategy in order to generate an optimized profit for the insurer. This existing framework relies on the analysis of the time-to-death and time-to-lapse that can be updated regularly with new information from the policies. It can be summarized as follows:

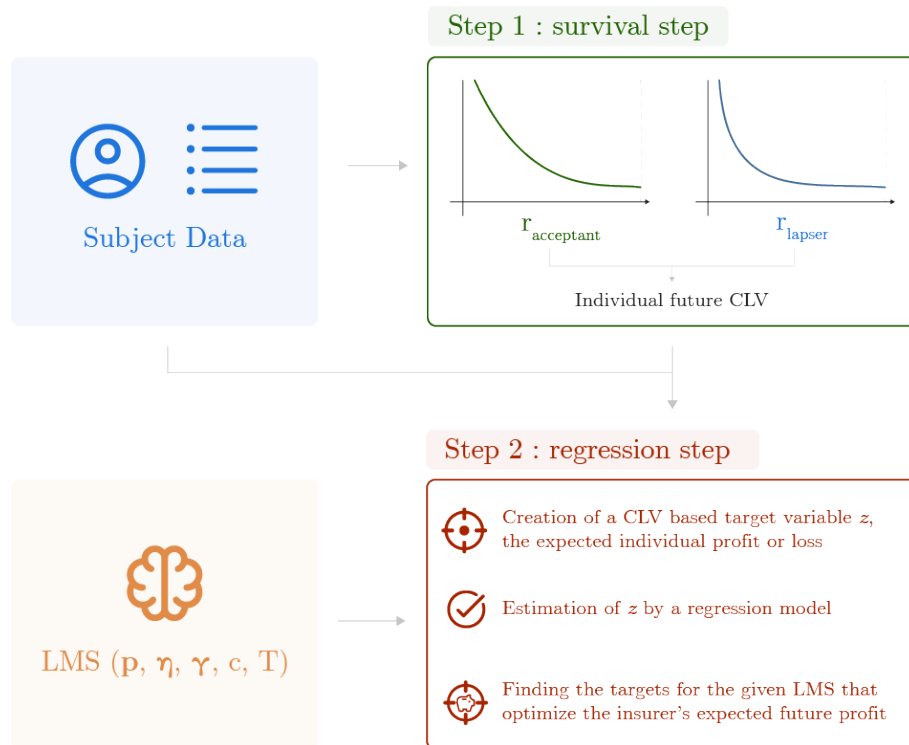


Figure 12.1: General framework for lapse management strategy

At least three limitations of that framework can be addressed. Firstly, it does not consider that an *acceptant* can lapse in the future, which is at best a very optimistic assumption, and at worst a great oversimplification. Secondly, it does not give any information on whether the timing of the retention campaign is optimal or not. Thirdly, it does not allow tightening the criteria on which the targeting of each policyholder is decided, depending on the risk the insurer is willing to take on the uncertainty of the predictions. This work addresses these limitations.

Throughout the lifetime of such insurance policies, a series of significant time-dependent events shape the interactions between policyholders and insurers. Firstly, premium payments play a pivotal role in sustaining the policy: these payments are highly flexible, allowing policyholders to choose their amount and frequency, thus they can be adjusted according to the policyholder’s financial circumstances and preferences. Additionally, policyholders may decide to reduce their coverage by withdrawing a portion of their policy. We refer to these events as partial lapses: they involve a voluntary decrease in the face amount of the policy, enabling policyholders to adjust their coverage to better align with their changing needs. Such flexibility caters to policyholders’ evolving financial situations and offers them greater control over their insurance plans. Over the policy’s lifetime, other financial operations can occur, such as the payment of interest or profit sharing to the policyholder, and the payment of fees to the insurer. Insurance companies’ information systems are usually designed to keep track of those operations at the policy-level, thus actuaries and life insurers often have access to the complete history of their policyholders as the information system is updated in real-time.

In certain instances, a policyholder may choose to lapse their insurance policy entirely. Complete policy lapse typically occurs when the policyholder decides to terminate her policy and receives a surrender value, which represents the accumulated value of the premiums paid, adjusted for fees, expenses, and potential surrender charges. Moreover, the occurrence of a policyholder’s death also terminates the policy and triggers the payment of the policy’s value, often referred to as the death benefit or claim, to the designated beneficiaries.

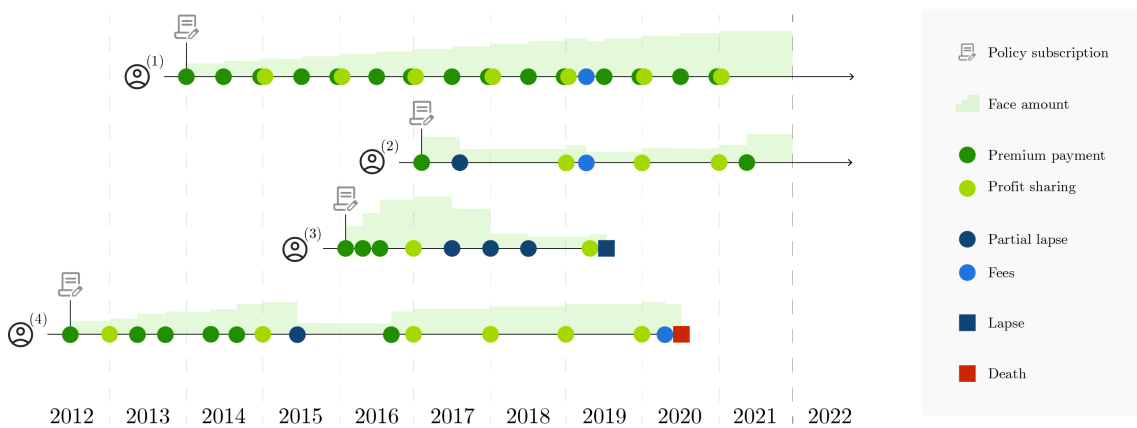


Figure 12.2: Example of policyholders timelines

In the context of our research, a policy can only terminate with a complete lapse or the death of the policyholder, which will be considered as competing risks in the following developments. If none of these events has happened to a policy, it is still active. The cumulated sum of all the financial flows occurring during one’s policy timeline, including premiums, claims, fees, interests, profit-sharing, and lapses, is commonly known as the face amount of the policy. This face amount represents the total value of the policy over its duration and serves as a measure of the

policy's coverage and financial benefits. By comprehensively understanding and analyzing these events and their impact on the face amount of a life insurance policy, insurers can effectively develop lapse management strategies that align with policyholders' preferences and financial goals. Through our research, we aim to shed light on these dynamics and provide insights to optimize the design of such strategies, ultimately enhancing customer retention and overall portfolio performance in the life insurance industry.

In practice, actuaries often have access to the complete trajectories of every policy and it seems that not using them in models is ignoring a significant part of the available information. A data structure where time-varying covariates are measured at different time points is called longitudinal and individual policyholders' timelines can be illustrated as in Figure 12.2. The dynamical aspects of covariates have an impact on the performance of lapse prediction models and Risselada, P.C. Verhoef, and Bijmolt 2010 concludes in favor of the development of dynamic churn models. They showed how the predictive performance of different types of churn prediction models in the insurance market decays quickly over time: this conclusion arguably applies to life insurers and in the case of lapse management strategy, we argue that using the complete longitudinal trajectories of every individual is also justified. Firstly, a change in financial behaviour - recent and frequent withdrawals for instance - can be informative lapse predictors. As an illustration of this point, we can imagine making predictions for two individuals with the exact same characteristics at the time of study but completely different past longitudinal trajectories: one is consistently paying premiums for instance, whereas the other stopped all payments for months and has been withdrawing part of her face amount lately. A prediction model ignoring longitudinal information would produce the exact same lapse prediction for both individuals. Conversely, an appropriate model, trained on longitudinal data is likely to seize the differences between the individuals over time and provide different predictions for the future. Secondly, a longitudinal lapse management framework allows for dynamic predictions with new information. It proves to be insightful in terms of decision-making for the insurer, as it shows how a change in the policy induces a change in the lapse behavior. Eventually, existing lapse management strategy approaches can only provide the insurer with information on whether targeting a given individual now is expected to yield profit, not on whether the timing of targeting is optimal. A longitudinal framework can help answer that last question.

In this paper, we want to account for the time-varying aspect of this problem in both steps of that framework. Firstly, we want to take advantage of the information contained in the historical data from the portfolio and obtain more accurate predictions for $r^{(i)}$ and thus ${}^FCLV^{(i)}$: that is a gain of precision on the survival step. Secondly, we want to evaluate the expected individual retention gains over time to derive the optimal timing to offer the incentive: that is a gain of flexibility and expected profit on the regression step. For that purpose, we introduce tree-based models which are, to the best of our knowledge, yet to be explored in the actuarial literature. Those models, such as left-truncated and right-censored (LTRC) survival trees and LTRC forests by Fu and J.S. Simonoff 2016b and W. Yao et al. 2020, or mixed-effect tree-based regression models (Sela and J.S. Simonoff 2012, Hajjem, Bellavance, and Larocque 2014b, Fu and J.S. Simonoff 2015, Capitaine, Genuer, and Thiébaud 2021) are considered state-of-the-art and have yet to be exploited in the actuarial literature. We propose an application of that framework with data-driven tree-based models but other types of models exist and could fit in this framework (see Appendix C.1)

This extension is not trivial, as time-dependent features and time-dependent response variables are difficult to implement in parametric or tree-based models. Indeed, conventional statistical or machine learning models do not readily accommodate time-varying features. This is the case for

most tree-based models as they assume that records are independently distributed. Of course, this is unrealistic as observations of any given individual are highly correlated. Moreover, time-varying features can generate bias if not dealt with carefully (see L. D. Fisher and D. Y. Lin 1999 for instance). The use of longitudinal data is already a well-studied topic (see G. Molenberghs and G. Verbeke 2006), with rare examples within the actuarial literature (see E. W. Frees et al. 2021 for instance) and, to the best of our knowledge, only a few actuarial uses of time-varying survival trees or mixed-effect tree-based models have been tried or suggested (Dal Pont 2020, Campo and Antonio 2022 or Moradian et al. 2022) and no longitudinal lapse analysis framework based on CLV has been described.

In summary, this work presents a longitudinal lapse analysis framework with time-varying covariates and target variables. This framework accommodates for competing risks and relies on tree-based machine learning models. This work focuses on a lapse management strategy and retention targeting for life insurers and extends the existing lapse management framework proposed in Loisel, Piette, and Tsai 2021 and Valla, Milhaud, and Olympio 2023. It defers from the latter by taking advantage of time-varying features, introducing different tree-based models to the lapse management literature, including the possibility for an *acceptant* to lapse in the future, yielding insights regarding individual targeting times, and adding the possibility to adjust the level of risk which the insurer is willing to take in a retention campaign. The rest of this paper is structured as follows. We describe the specifics of longitudinal analysis and a new longitudinal and time-dynamic lapse management framework which is the main contribution of this work in Section 12.2. This section also includes a brief description of models that can fit in this framework. In Section 12.3, we show a concrete application of our framework on a real-world life insurance portfolio with a discussion of our methodology and results. Eventually, Section 12.4 concludes this paper.

12.2 Longitudinal framework

12.2.1 Preliminaries on time-varying covariates and longitudinal notations

We aim to enrich the existing lapse management frameworks (see Definition 2) with time-varying covariates. To do so, we decide to adapt LMS methods to longitudinal analysis. In order to be perfectly clear on what we mean by *time-varying covariates* or *longitudinal data*, let us introduce some notations. This section borrows notations from the existing literature including Rizopoulos 2012 or W; Yao et al. 2022 for instance. Let us assume a very general setting where we want to build a dataset \mathcal{D} , encompassing the information of N individuals from which features are repeatedly measured over time. These covariates may come in many forms, some of them are time-varying, and others are time-invariant. We denote p_{tv} , p_{ti} the number of covariates in those respective categories, with $p = p_{tv} + p_{ti}$, the total number of covariates. At time t , the covariates matrix is $\mathbf{X}(t) = (x_1, x_2, \dots, x_{p_{ti}}, x_{p_{ti}+1}(t), \dots, x_p(t))$. In order to simplify the notations, we write $\mathbf{X}(t) = (x_1(t), x_2(t), \dots, x_p(t))$ with $x_k(t) = x_k, \forall t$ and $\forall k \in [1, \dots, p_{ti}]$.

These covariates are available for the N individuals, or subjects, which are observed at discrete time points. Subject i has been observed $n^{(i)}$ times, at $t_j^{(i)}, j = 0, 1, \dots, n^{(i)} - 1$. In our life insurance context $t_0^{(i)}$ represents the first measurement of the covariates, i.e the subscription and times $t_j^{(i)}, j = 1, 2, \dots, n^{(i)} - 1$ are the movement dates, i.e times at which a change in the policy has been recorded. If $t_0^{(i)} \neq 0$, this means that the baseline information at subscription is missing and the observation is left-truncated. A given subject i , at time $t_j^{(i)}$ has a vector of covariates

denoted $\mathbf{x}_j^{(i)} = (x_{j,1}^{(i)}, \dots, x_{j,p}^{(i)})$ and generally, has a matrix of covariates denoted

$$\mathbf{X}^{(i)} = \left(\mathbf{x}_0^{(i)}, \dots, \mathbf{x}_{n^{(i)}-1}^{(i)} \right)^\top. \quad (12.4)$$

As stated in Definition 2, the probability of still having an active policy at time t depends on the policyholder's risk profile. *Acceptants* are only at risk for death whereas *lapsers* are at risk for both lapse and death and we consider as the event of interest respectively death and the end of the policy (whatever the cause). Regardless of our outcome of interest, we study the time to an event ending the policy, thus we use the classical survival notations: subject i will eventually experience the event at time $T_*^{(i)}$ and she is no longer observed after censoring time $C^{(i)}$. We let $T^{(i)}$ denote the observed event time for subject i , defined as $t_{n^{(i)}}^{(i)} = T^{(i)} = \min(T_*^{(i)}, C^{(i)})$.

The notations regarding the time dynamics of our data are now clear, we decide to structure this information in a longitudinal dataset. In order to do so, we assume that the time-varying features take constant values between two consecutive observations, that is,

$$\mathbf{x}^{(i)}(t) = \mathbf{x}_j^{(i)}, \quad t \in [t_j^{(i)}, t_{j+1}^{(i)}), \quad j = 0, 1, \dots, n^{(i)} - 1.$$

This assumption is perfectly consistent in an actuarial context where time-varying covariates such as financial flows are immediately updated. Any covariate update leads to a new observation and all variables are in fact constant between two consecutive observations. The only limit of this assumption is that updating the insurer's database usually takes some time and it proves to be unrealistic if a policy change has been reported but not yet processed in the information system.

An insurance policy at any time point is either active or ended. Moreover, it can only end in two ways: the policyholder either lapses her policy or dies. Thus we define three event indicators. $\Delta^{(i)}$ is the event indicator, defined at the subject level, it denotes whether individual (i) has experienced an event (and which one) before censoring time,

$$\Delta^{(i)} = \begin{cases} 0 & \text{if } T_*^{(i)} \leq C^{(i)} \\ 1 & \text{if } T_*^{(i)} > C^{(i)} \text{ and EVENT} = \text{lapse} \\ 2 & \text{if } T_*^{(i)} > C^{(i)} \text{ and EVENT} = \text{death.} \end{cases} \quad (12.5)$$

We also introduce $\delta^{(i)}(t)$, the event indicator defined at the observation level, it denotes whether individual (i) has experienced an event (and which one) by time t :

$$\delta^{(i)}(t) = \Delta^{(i)} \cdot \mathbb{1}\{t \geq T^{(i)}\}. \quad (12.6)$$

At time $t = T_*^{(i)}$, the true event has occurred and we define the ultimate event indicator as

$$\Delta_*^{(i)} = \begin{cases} 1 & \text{if EVENT} = \text{lapse at time } T_*^{(i)} \\ 2 & \text{if EVENT} = \text{death at time } T_*^{(i)}. \end{cases} \quad (12.7)$$

It is constant over the observations for a given subject and represents the final value of $\Delta^{(i)}$ when the subject's policy eventually ends. It can be either equal to 1 or 2. For a subject with an active policy at the censoring time, the value of $\Delta_*^{(i)}$ is unknown.

Eventually, let $\mathcal{X}^{(i)}(t)$ denote the covariate individual information up to time t , and we define

$\pi_*^{(i)}$ as the probability that the policy will eventually end with lapse, given all available information at observation time t . Mathematically speaking, we have

$$\pi_*^{(i)} = P(\Delta_*^{(i)} = 1 | \mathcal{X}^{(i)}(T^{(i)})). \quad (12.8)$$

We can now build \mathcal{D} , a longitudinal dataset encompassing the complete past information of all N subjects. For a given subject i , covariates are stored in rows, one row per observation window $[t_j^{(i)}, t_{j+1}^{(i)})$. Each row contains the unique $(t_j^{(i)}, t_{j+1}^{(i)}, \delta^{(i)}(t_j^{(i)}), \mathbf{x}_j^{(i)})$ element and is completed by the subject unique identifier i and her event indicator $\Delta^{(i)}$: each row is called an *observation*. It is critical to include all those elements in the longitudinal dataset as all columns are inputs of longitudinal models used for the *survival step*.

Any observation only corresponds to one subject and conversely, any subject can be linked to a set of $n^{(i)}$ observations. We build \mathcal{D} as the collection of all observations structured longitudinally :

$$\mathcal{D} = \left\{ \left(i, \left\{ t_j^{(i)}, t_{j+1}^{(i)}, \mathbf{x}_j^{(i)}, \delta^{(i)}(t_j^{(i)}) \right\}_{j=0}^{n^{(i)}-1}, \Delta^{(i)} \right) \right\}_{i=1}^N,$$

or, if displayed in a table:

Table 12.1: A longitudinal dataset, in all generality

ID	Time window Start	Time window End	Covariate 1	...	Covariate p	Observation event indicator	Subject event indicator
1	$t_0^{(1)}$	$t_1^{(1)}$	$x_{0,1}^{(1)}$...	$x_{0,p}^{(1)}$	$\delta^{(1)}(t_0^{(1)})$	Δ^1
1	$t_1^{(1)}$	$t_2^{(1)}$	$x_{1,1}^{(1)}$...	$x_{1,p}^{(1)}$	$\delta^{(1)}(t_1^{(1)})$	Δ^1
1	$t_2^{(1)}$	$t_3^{(1)}$	$x_{2,1}^{(1)}$...	$x_{2,p}^{(1)}$	$\delta^{(1)}(t_2^{(1)})$	Δ^1
1	$t_3^{(1)}$	$C^{(1)}$	$x_{3,1}^{(1)}$...	$x_{3,p}^{(1)}$	$\delta^{(1)}(t_3^{(1)})$	Δ^1
2	$t_0^{(2)}$	$t_1^{(2)}$	$x_{0,1}^{(2)}$...	$x_{0,p}^{(2)}$	$\delta^{(2)}(t_0^{(2)})$	Δ^2
3	$t_0^{(3)}$	$t_1^{(3)}$	$x_{0,1}^{(3)}$...	$x_{0,p}^{(3)}$	$\delta^{(3)}(t_0^{(3)})$	Δ^3
3	$t_1^{(3)}$	$t_2^{(3)}$	$x_{1,1}^{(3)}$...	$x_{1,p}^{(3)}$	$\delta^{(3)}(t_1^{(3)})$	Δ^3
3	$t_2^{(3)}$	$t_3^{(3)}$	$x_{2,1}^{(3)}$...	$x_{2,p}^{(3)}$	$\delta^{(3)}(t_2^{(3)})$	Δ^3
...

Table 12.1 precisely illustrates what we call a longitudinal dataset, and a real-world example of such a dataset can be found in Section 12.3, Table 12.3. Adapting a lapse management strategy framework to a longitudinal setting means we take such a dataset as input and produce enriched predictions of the individual retention probabilities in the *survival step*, but also of individual profit or loss estimated in the *regression step*.

12.2.2 LMS longitudinal framework

We adopt Valla, Milhaud, and Olympio 2023's framework and suggest some modifications and improvements to accommodate for longitudinally structured data. Instead of a top-down approach that consists of estimating the individual contributions to the insurer's profit from a global measure of the portfolio value, we suggest a bottom-up approach and directly evaluate the former and then derive the latter. Thus, we define the control future value of the policy, ${}^F CV^{(i)}(t, \dots)$, which represents the expected T -year individual profit or loss generated by subject i , after time t :

$$\begin{aligned} {}^F CV^{(i)}(t, \mathbf{p}, \boldsymbol{\eta}, \boldsymbol{\gamma}, c, T) = & {}^F CLV^{(i)}\left(t, \mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, \mathbf{d}, T\right) \cdot (1 - \pi_*^{(i)}) \\ & + {}^F CLV^{(i)}\left(t, \mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, \mathbf{d}, T\right) \cdot \pi_*^{(i)}. \end{aligned} \quad (12.9)$$

In other words, it simply represents an individual expected future CLV, if no lapse management is carried out. It highly depends on the probability for the policyholder to be a lapsers.

Let us consider an LMS, let $\odot^{(i)}(t)$ be the individual target vector indicator, designating if subject i is to be targetted at any time t . Our framework aims to find the optimal list of policyholders to target, $\mathcal{T}(t) = \{i \mid \odot^{(i)}(t) = 1\}$ that maximizes the expected profit for the insurer. In order to evaluate the profit or loss generated by an LMS, we must compare the expected profit obtained if no LMS was applied, with the expected profit generated by the lapse-managed portfolio. The former is given by Equation 12.9 and to obtain the latter, we define the lapse managed observation future value as

$$\begin{aligned} {}^F LMV^{(i)}(t, \mathbf{p}, \boldsymbol{\eta}, \boldsymbol{\gamma}, c, T) = & \left[{}^F CLV^{(i)}\left(t, \mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, \mathbf{d}, T\right) \cdot (1 - \pi_*^{(i)}) + {}^F CLV^{(i)}\left(t, \mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, \mathbf{d}, T\right) \cdot \pi_*^{(i)} \right] \cdot (1 - \odot^{(i)}(t)) \\ & + \left[{}^F CLV^{(i)}\left(t, \mathbf{p}^{(i)} - \boldsymbol{\eta}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, \mathbf{d}, T\right) \cdot (1 - \pi_*^{(i)}) + \boldsymbol{\gamma}^{(i)} \cdot {}^F CLV^{(i)}\left(t, \mathbf{p}^{(i)} - \boldsymbol{\eta}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, \mathbf{d}, T\right) \cdot \pi_*^{(i)} \right. \\ & \left. + (1 - \boldsymbol{\gamma}^{(i)}) \cdot {}^F CLV^{(i)}\left(t, \mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, \mathbf{d}, T\right) \cdot \pi_*^{(i)} - c \right] \cdot \odot^{(i)}(t). \end{aligned} \quad (12.10)$$

In simple terms, it is equal to the control future value of the policy (given by Equation 12.9) when subject i is not targetted, otherwise, it depends on whether she intended to lapse in the first place and if so, if she accepts the incentive $\boldsymbol{\eta}$. If a policyholder that would not have lapsed (with probability $(1 - \pi_*^{(i)})$) is targetted, she will rationally accept the incentive and generate the future CLV of an acceptant with profitability $p - \boldsymbol{\eta}$. Conversely, for a policyholder that would have ultimately lapsed, she either accepts the incentive (with probability $\boldsymbol{\gamma}^{(i)}$) and generates the future CLV of an acceptant with profitability $p - \boldsymbol{\eta}$, or she refuses (with probability $(1 - \boldsymbol{\gamma}^{(i)})$) and generates profitability p with the risk profile of a lapsers.

It follows that the individual expected retention gain obtained by applying an LMS is the difference between the expected individual CLVs with and without lapse management:

$$RG^{(i)}(t, \mathbf{p}, \boldsymbol{\eta}, \gamma, c, T) = {}^F\text{LMV}^{(i)}(t, \mathbf{p}^{(i)}, \boldsymbol{\eta}^{(i)}, \gamma^{(i)}, c, T) - {}^F\text{CV}^{(i)}(t, \mathbf{p}^{(i)}, \boldsymbol{\eta}^{(i)}, \gamma^{(i)}, c, T). \quad (12.11)$$

that can be simplified as

$$RG^{(i)}(t, \mathbf{p}, \boldsymbol{\eta}, \gamma, c, T) = \odot^{(i)}(t) \cdot \left[\pi_*^{(i)} \gamma^{(i)} \left[{}^F\text{CLV}^{(i)}(t, \mathbf{p}^{(i)} - \boldsymbol{\eta}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, \mathbf{d}, T) \right. \right. \\ \left. \left. - {}^F\text{CLV}^{(i)}(t, \mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, \mathbf{d}, T) \right] \right. \\ \left. - (1 - \pi_*^{(i)}) \cdot {}^F\text{CLV}^{(i)}(t, \boldsymbol{\eta}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, \mathbf{d}, T) \right] - c \cdot \odot^{(i)}(t). \quad (12.12)$$

An evaluation metric is finally derived to obtain the retention gain, at any observation time, if the policyholder i is targeted. We define $z^{(i)}(t)$ as

$$z^{(i)}(t) = RG^{(i)}(t, \mathbf{p}, \boldsymbol{\eta}, \gamma, c, T | \odot^{(i)}(t) = 1) \\ = \left[\pi_*^{(i)} \gamma^{(i)} \left[{}^F\text{CLV}^{(i)}(t, \mathbf{p}^{(i)} - \boldsymbol{\eta}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, \mathbf{d}, T) \right. \right. \\ \left. \left. - {}^F\text{CLV}^{(i)}(t, \mathbf{p}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{lapsers}}^{(i)}, \mathbf{d}, T) \right] \right. \\ \left. - (1 - \pi_*^{(i)}) \cdot {}^F\text{CLV}^{(i)}(t, \boldsymbol{\eta}^{(i)}, \mathbf{F}^{(i)}, \mathbf{r}_{\text{acceptant}}^{(i)}, \mathbf{d}, T) \right] - c. \quad (12.13)$$

In terms of intuition, it shows that if a policyholder that would have lapsed (with probability $\pi_*^{(i)}$) is targeted and accepts the incentive (with probability $\gamma^{(i)}$), she generates the future CLV of an acceptant with profitability $p - \eta$ instead of her initial future CLV with profitability p and the risk profile of a lapsers. The gain generated by targeting this policyholder is then the difference between the two. On the other hand, if the policyholder is wrongfully targeted and would not have lapsed (with probability $(1 - \pi_*^{(i)})$), she rationally accepts the incentive which is then lost for the insurer. In any case, the contact cost of c is spent.

From a practical point of view, we can see that the value of $z^{(i)}(t)$ depends on parameters that are observed in the portfolio ($\mathbf{F}^{(i)}$), or assumed by the insurer ($\mathbf{p}^{(i)}, \boldsymbol{\eta}^{(i)}, \mathbf{d}, T$), and that only $\mathbf{r}_{\text{acceptant}}^{(i)}$ and $\mathbf{r}_{\text{lapsers}}^{(i)}$ need to be estimated. This estimation is the *survival step* mentioned in Section 12.1. We will show in Section 12.3.2 how to concretely estimate these retention probabilities using time-varying covariates.

Assuming that $z^{(i)}$ has been estimated for every observation in the *survival step*, we can move forward to the *regression step* and use $z^{(i)}$ as a target variable in a regression model handling time-varying covariates to predict whether targeting any policyholder will generate profit, given her previous observations if any. We will show in Section 12.3.3 how to concretely obtain $\hat{z}^{(i)}$ with mixed-effect tree-based models.

With that in mind, we can update Definition 2 and define our LLMS as follows:

Definition 3 (Longitudinal lapse management strategy (LLMS)). *A T -years lapse management strategy is modeled by offering an incentive $\eta^{(i)}$ to subject i if she is targeted. The incentive offered is expressed as a percentage of her face amount at the observation time and is accepted with probability $\gamma^{(i)}$. Contacting the targeted policyholder has a fixed cost of c . A targeted subject who accepts the incentive will be considered an “acceptant” who will be **less likely to lapse**, and her probability of being active at year $t \in [0, T]$, given the information available until then, is denoted $r_{\text{acceptant}}^{(i)}(t \mid \mathcal{X}^{(i)}(t))$. Conversely, a subject who refuses the incentive and prefers to lapse will be considered a “larser”, and her probability of being active at year t , given the information available until then, is denoted $r_{\text{larser}}^{(i)}(t \mid \mathcal{X}^{(i)}(t))$.*

Those probabilities are used to derive a dynamical profit-driven measure $z^{(i)}(t)$ based on CLV (see Equation 12.13). A regression model, allowing for longitudinal data is then used with $z^{(i)}(t)$ as a target variable, which allows us to estimate $\hat{z}^{(i)}(t)$ for any new observations (new observations of known subjects or observations of new subjects). Denoting the standard error of such a model σ_z and any confidence level α , we define the optimal longitudinal LMS at time t as

$$\odot_*^{(i)}(t) = \mathbb{1} \left\{ \hat{z}^{(i)}(t) > \alpha \cdot \sigma_z \right\}. \quad (12.14)$$

This is an indicator variable representing whether it is worth targeting policyholder i at time t , thus, the corresponding list of targetted policyholders is defined as

$$\mathcal{T}(t) = \left\{ i \mid \odot_*^{(i)}(t) = 1 \right\}. \quad (12.15)$$

For any targeted policyholder and any confidence level α desired by the insurer, there is a unique future time $t_*^{(i)} \geq T^{(i)}$ when offering the incentive is optimal, which yields a maximal profit of $\hat{z}_*^{(i)}$. If all policyholders in $\mathcal{T}(t)$ are targetted at time t , the LLMS generates a profit of

$$RG(t, \mathbf{p}, \boldsymbol{\eta}, \boldsymbol{\gamma}, c, T, \alpha) = \sum_{i \in \mathcal{T}(t)} \hat{z}^{(i)}(t). \quad (12.16)$$

If all policyholders are targetted at the optimal time $t_*^{(i)} \geq t$, the LLMS induces a gain for the life insurer of

$$RG^*(t, \mathbf{p}, \boldsymbol{\eta}, \boldsymbol{\gamma}, c, T, \alpha) = \sum_{i \in \mathcal{T}(t)} \frac{\hat{z}_*^{(i)}}{\left(1 + d_{t_*^{(i)}}^{(i)}\right)^{\Delta t}}, \text{ with } \Delta t = t_*^{(i)} - t. \quad (12.17)$$

The addition of a confidence level α contrasts with previous approaches (see Loisel, Piette, and Tsai 2021; Valla, Milhaud, and Olympio 2023). Setting $\alpha = 0$ means that the prediction $\hat{z}^{(i)}(t)$ is trusted with 100% confidence by the insurer, whereas letting α take higher values ensures that $\hat{z}^{(i)}(t)$ is positive with a given confidence interval. Another novelty here is the time dynamic of those results. Not only can we predict whether it is worth targeting a given policyholder, but we can also predict whether there will be some point in the future when targeting her will

be more profitable. Predicting the trajectory of $z^{(i)}(t)$ at future time points requires projecting the time-varying covariates at those future time points. It can be done by either modeling such covariates individually or setting assumptions. It is trivial for covariates such as age or year but more complex for stochastic covariates such as the face amount. This framework does not aim to answer this question, and we assume in our application that stochastic covariates remain constant and equal to their last observed value. Regardless of the assumptions, the framework allows adding a time dimension to the LMS optimization and marketing decision-making. It is also worth noting that our developed framework is consistent in the time-invariant case. By design, it is also fully applicable with uncensored observations, or left-truncated ones. That shows our two-step framework's broad effectiveness and applicability regardless of right-censorship, left-truncation, risk factor, time-varying covariates, or time-varying effects. In that sense, it is a generalized framework for lapse management strategy in life insurance.

12.3 Application

12.3.1 Data

Our framework is inspired by a real-world life insurance dataset used in Valla, Milhaud, and Olympio 2023 (Chapter 8). It initially contains the most recent information from 248 737 unique policies contracted between 1997 and 2018 and 235 076 unique policyholders. A single row originally represented a unique pair policy/policyholder, identified by a unique ID and denoted as a *subject*. Due to great computation times, we restrain our application on a 10,000-subjects subset of this original dataset, but the astute reader will find more information about the complete one in the original article. The 10,000 rows dataset containing the last available information for the 10,000 selected subjects will be denoted \mathcal{D}^{last} , here is a subset for illustrative purposes:

Table 12.2: \mathcal{D}^{last} random subset

ID	EVENT	PRODUCT	SEX	SENIORITY	F_i	CLAIM	CNTRCTS	AGE	YEAR
25737	1	1	1	17	0,73	0	2	76	2015
117322	1	1	2	10	4,32	0	1	63	2012
1322	0	1	2	20	9,82	0	1	75	2019
37433	2	1	2	14	0,99	-50,49	1	88	2011
23902	0	1	1	20	32,66	-13,12	2	71	2019
219281	0	2	2	8	7,08	0	2	71	2019
160112	0	1	2	15	0,04	0	1	51	2019
53108	2	1	2	12	13,11	0	1	92	2010
166078	1	2	2	5	9,02	0	1	64	2013
139644	0	1	1	16	5,65	-107,59	1	66	2019

Here, we were able to retrieve the longitudinal history of every subject present in \mathcal{D}^{last} : this means that for every policy and policyholder, we observe every payment, lapse, fee, profit sharing or discount rate from the policy subscription to the most updated information to date along with baseline covariates such as gender or age at subscription. For operational reasons, the longitudinal data are measured and reported yearly and organized as follows²:

Moreover, all the covariates describing financial flows are observed as cumulated over the years. As an example, let us assume that a subject subscribed in the year 2000: her payment variable for the year 2000 observation contains the sum of all payments that occurred in that year, her

²But it is worth mentioning that covariates in actuarial datasets are usually updated continuously. In that case, we could build a continuous longitudinal dataset with one observation per policy change, and not one per year. The framework detailed here still applies in the continuous case.

Table 12.3: \mathcal{D}^{long} random subset

ID	EVENT	START	END	PRODUCT	SEX	SENIORITY	F_i	CLAIM	CNTRCTS	AGE	YEAR
46784	0	0	1	3	2	0	8,38	0	1	66	2013
46784	0	1	2	3	2	1	8,40	0	1	67	2014
46784	0	2	3	3	2	2	8,57	0	1	68	2015
46784	0	3	4	3	2	3	11,90	0	1	69	2016
46784	0	4	5	3	2	4	12,10	0	1	70	2017
46784	0	5	6	3	2	5	12,28	0	1	71	2018
46784	1	6	7	3	2	7	15,06	-15,06	1	72	2019
7825	0	0	1	2	2	0	3,02	0	1	81	2016
7825	0	1	2	2	2	1	3,05	0	1	82	2017
7825	0	2	3	2	2	2	3,10	0	1	83	2018
7825	0	3	5	2	2	5	3,15	0	1	84	2019
264309	0	0	1	3	2	0	2,61	0	1	66	2016
264309	0	1	2	3	2	1	2,64	0	1	67	2017
264309	0	2	3	3	2	2	2,67	0	1	68	2018
264309	0	3	5	3	2	5	3,48	0	1	69	2019

payment variable for the year 2001 contains the sum of all payments that occurred up to the year 2001 included (hence 2000 and 2001), and so on for the years after. This longitudinal dataset will be denoted \mathcal{D}^{long} . It contains 126,865 observations, in other words, almost 13 for each subject.

For privacy reasons, all the data, statistics, product names, and perimeters presented in this paper have been either anonymized or modified. All analyses, discussions, and conclusions remain unchanged.

12.3.2 Application: survival step

Survival analysis with time-varying covariates

The survival step, described in Section 12.2 requires survival tree-based models that can handle longitudinal time-varying covariates. Most survival tree-based models are analogous to regular tree-based models: survival trees work similarly to regular decision trees, creating partitions of the covariate space. What differentiates them is the splitting criterion that splits by maximizing the difference between two considered child nodes. Typically, at each node and for each split considered, a log-rank test is used to test the null hypothesis that there is no difference between the child nodes in the probability of an event at any time. The split that minimizes the p-value is then selected. By extension, a random survival forest is a random forest of survival trees.

As regression and classification trees, most survival trees are unable to deal with time-varying and longitudinal covariates. Indeed, let $x_1(t)$ be a numerical time-varying covariate. For a single tree, the splitting rule should be able to split subjects into two child nodes at each node. It would then be a rule of the form " $x_1 \leq s$ ". A subject for which this rule is true $\forall t$ will go in one child node without any ambiguity. On the other hand, the general case where the rule is true for some periods but false for anywhere else is unclear and needs to be addressed. Note that the same reasoning can be applied to categorical time-varying covariates as well. A simple idea is that the subject's observations in periods where the splitting rule is true would go to the left node, and the other would go to the right node, thus dividing one subject into several pseudo-subjects. With a longitudinal dataset, that method just implies considering all rows as independent which creates correlated right-censored and left-truncated (LTRC) observations that need special treatment. In such models, any individual can be spread in many different tree leaves - even if, at any fixed time, any individual will have a single observation that will fall into one unique leaf. Fu and J.S. Simonoff 2016a proposed a model based on those ideas: they allowed subjects to be divided into pseudo-subjects and adjusted the log-rank test in the splitting procedure to accommodate for left

truncation and ensure that the independence implicit assumption does not lead to biased results³.

LTRC trees and forests yield an estimate of the survival function:

$$\widehat{S}(t | \mathcal{X}^{(i)}(t)) = P(T^{(i)} > t | \mathcal{X}^{(i)}(t)),$$

that can directly be used to evaluate the conditional incidence functions for competing risks (see Appendix A.1.1). Bagging models of such trees then emerged (W. Yao et al. 2020), with the usual prediction advantages and interpretability drawbacks of such bagging techniques⁴. In order to evaluate the survival models' performance, we chose to use the time-dependent Brier score (td-BS), integrated Brier score (td-IBS), Brier skill score (td-BSS) and integrated Brier skill score (td-IBSS) for longitudinal data (as in W. Yao et al. 2020). More details about these metrics can be found in Appendix 5.2.1.

Comparison settings

We propose here a comparison framework to measure the benefits of including the historical data in \mathcal{D}^{long} , compared to using \mathcal{D}^{last} . The matrices $r_{lapseser}$ and $r_{acceptant}$ are estimated with the algorithms LTRCRRF and LTRCCIF from the R package LTRCforests⁵. In order to assess the advantages of that longitudinal model, we compare its results with those obtained with gradient boosting survival Model (GBSM) as it proved to be a high-performing non-longitudinal model on that dataset (See Valla, Milhaud, and Olympio 2023). With $T^{(i)}$, the ‘‘any event’’ time for subject i (that is the censoring time for active policies and the termination time, whatever the cause, for all others), $r_{lapseser}$ and $r_{acceptant}$ are estimated from the respective survival functions

$$\widehat{S}_{lapseser}(t | \mathcal{X}^{(i)}(t)) = P(T^{(i)} > t | \mathcal{X}^{(i)}(t)),$$

$$\widehat{S}_{acceptant}(t | \mathcal{X}^{(i)}(t)) = P(T^{(i)} > t, \text{EVENT} = \text{death} | \mathcal{X}^{(i)}(t)),$$

with observations that ended with lapse considered as censored in the estimation of $\widehat{S}_{acceptant}$.

We want to compare the performance of all models trained with and without longitudinal data but also compare them on different tasks. Typically, predictions on \mathcal{D}^{last} and \mathcal{D}^{long} do not answer the same questions. The former aims at predicting the last observation of the target variable, and the latter aims at predicting its value at any given point in time. Depending on whether the model has been trained on longitudinal data or only on the most recent observation and with different prediction goals, this naturally designs four settings that answer four prediction problems:

- (a) Models are trained on $\mathcal{D}_{train}^{last}$ and evaluated on predictions from $\mathcal{D}_{test}^{last}$
- (b) Models are trained on $\mathcal{D}_{train}^{long}$ and evaluated on predictions from $\mathcal{D}_{test}^{last}$
- (c) Models are trained on $\mathcal{D}_{train}^{last}$ and evaluated on predictions from $\mathcal{D}_{test}^{long}$
- (d) Models are trained on $\mathcal{D}_{train}^{long}$ and evaluated on predictions from $\mathcal{D}_{test}^{long}$

³See Fu and J.S. Simonoff 2016a for details on that point.

⁴Both methods have been implemented in the R packages LTRCtrees and LTRCforests, and are considered state-of-the-art methods for tree-based survival analysis with time-varying covariates.

⁵In the following sections, we consider LTRCRRF and LTRCCIF: LTRC forests respectively based on regular CART and conditional inference survival tree algorithms. More insights about those models can be found in the references detailed in Section 12.3.2

Setting (a) is the classical setting, where any subject has only one measurement, and the prediction task is also to predict a variable at one given time point. Conversely, setting (d) represents the longitudinal setting, where models are trained with longitudinal time-varying covariates and where the prediction task aims at retrieving the value of a target variable at any given time point during a subject’s lifetime. Setting (c) is un insightful as a model trained on aggregated data cannot retrieve longitudinal information and is expected to perform poorly by design. Intermediate setting (b) is also insightful as it can be used to highlight the added value of the information contained in longitudinal data when training a model. The comparison is made on a time-varying survival evaluation metric: the time-dependent Brier Skill Score (td-BSS) for longitudinal data (see Appendix 5.2.1).

Results

First of all, in order to assess the superiority of longitudinal models in a longitudinal context, we need to compare all our considered models in the classical aggregated setting: with training and testing phases on subsets of \mathcal{D}^{last} . We can see that in this non-longitudinal setting, GBSM and LTRC models (LTRCRRF and LTRCCIF) are close in terms of BSS. Figure 12.3 displays the td-BSS on the y-axis, for which a value of 0 means that the score for the predictions is merely as good as that of a naïve prediction⁶ and a value of 1 is the best score possible. BSSs are computed for every time point, meaning that we can observe and compare the performance of models on estimating retention probabilities for low-seniority policies or high-seniority ones independently.

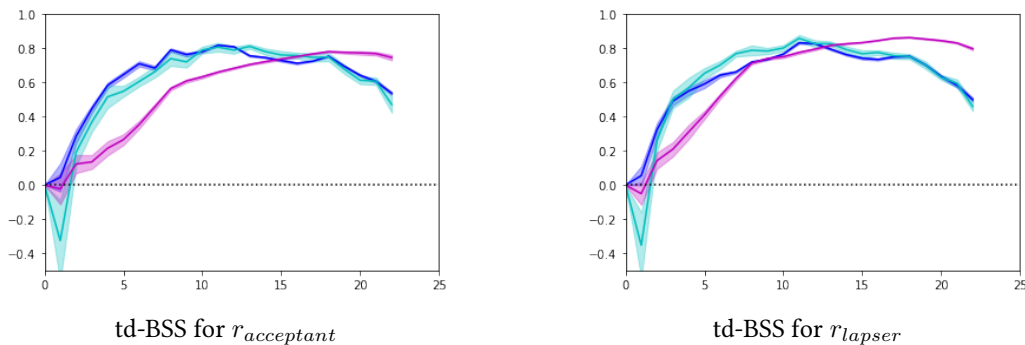


Figure 12.3: td-BSS of models trained on $\mathcal{D}_{train}^{last}$ and tested on $\mathcal{D}_{test}^{last}$

The IBSS, the mean of BSSs over all time points (see Appendix 5.2.1), indicates that LTRCRRF performs slightly better than LTRCCIF, hence we will drop LTRCCIF for the rest of this application. In real-world scenarios, the inherent complexity of the true survival distribution might include time-varying covariates and time-varying effects. The cross-validated Brier scores and Brier Score Skills graphs can potentially lead decision makers to choose different survival estimations at different time points and not a unique choice of method for all time points.

By contrast, the difference between those models is evident and significant whenever they are trained on longitudinal data. The graphs below show the difference in terms of BSS over time in prediction settings (b) and (d):

The conclusion regarding prediction richness contained in longitudinal data and accuracy benefits from using dedicated longitudinal methods is clear. Longitudinal models perform significantly better, and GBSM brings minor improvement over naïve models.

In the end, we select LTRCRRF for estimating the retention probabilities in the *survival step* as it

⁶in our application, the empirical estimate of the survival function has been chosen as the naïve prediction.

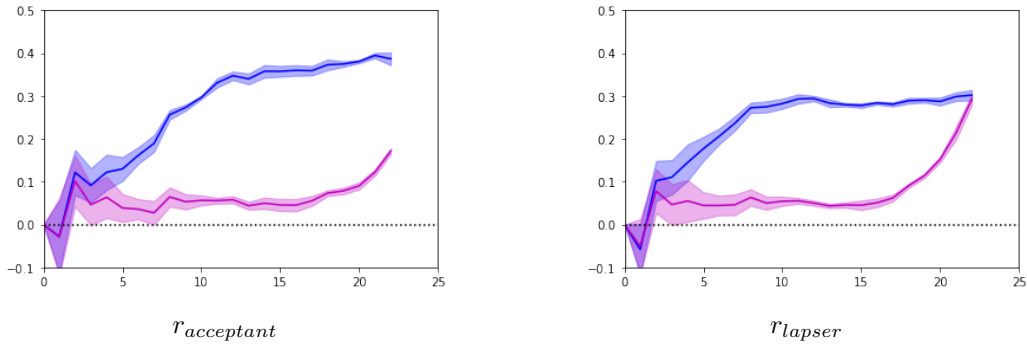


Figure 12.4: td-BSS of models trained on $\mathcal{D}_{train}^{long}$ and tested on $\mathcal{D}_{test}^{last}$ - Setting (b)

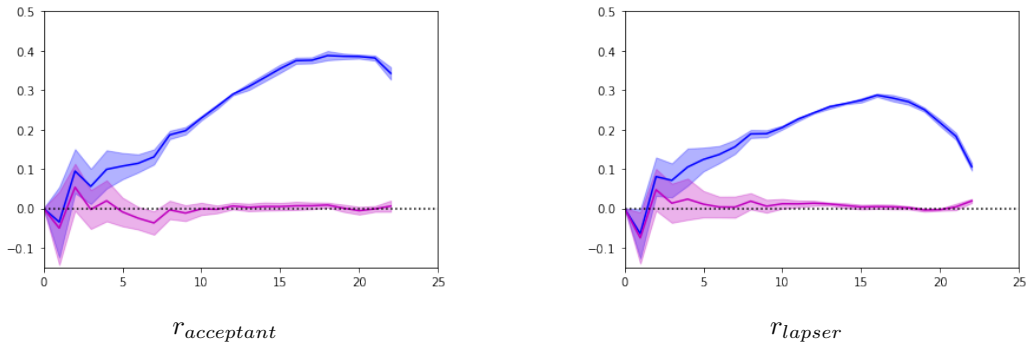


Figure 12.5: td-BSS of models trained on $\mathcal{D}_{train}^{long}$ and tested on $\mathcal{D}_{test}^{long}$ - Setting (d)

shows to be the best model when trained on longitudinal data.

It is to be noted that the results of that modeling approach in terms of global retention gain (Equation 12.16) are not necessarily better than the results obtained without the use of longitudinal data in the estimation of r_{lapses} and $r_{acceptant}$. In other words, a better performance of the model used in the *survival step* does not lead to an increase in the insurer's expected profit, for a given LMS but to a more realistic estimation of it as they model the CLV more accurately.

With that, we determine r_{lapses} and $r_{acceptant}$, the conditional retention probabilities for every observation to derive the trajectory of the observed individual CLV, RG, and eventually $z^{(i)}(t)$ (see Equation 12.13). The latter can then be used as a longitudinal target variable in a regression model: this constitutes the *regression step*, introduced in Section 12.1 and detailed within this application in the next Section.

Another advantage of using longitudinal data for survival analysis is that it helps study how a given subject's retention probabilities are updated with time. We take the example of a randomly selected subject and plot her retention probability at every observation time:

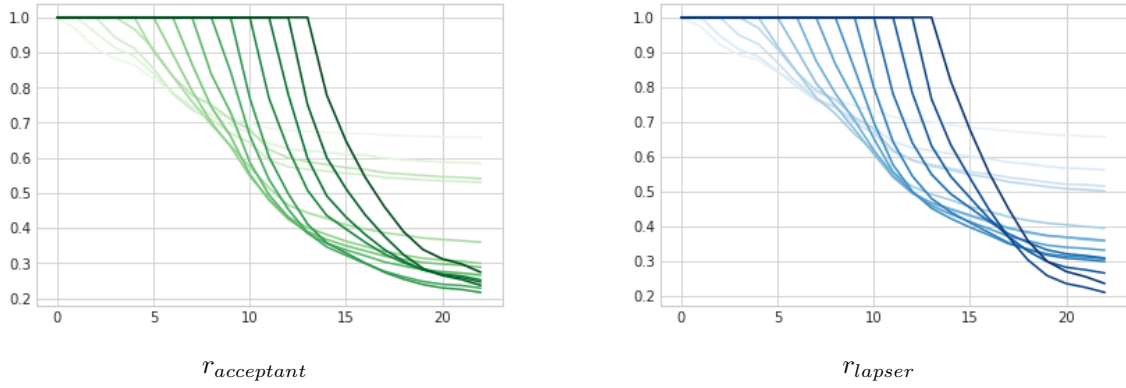


Figure 12.6: Longitudinally updated retention trajectories for a random subject

The further in time the observation is, the more pellucid the survival curve is. The individual retention curves are updated as new measurements are available.

12.3.3 Application: regression step

Regression analysis with time-varying covariates

The *regression step* of the framework introduced in Section 12.2.2 requires using a regression model allowing for longitudinal data to produce an estimate of $z^{(i)}(t)$. We chose to use mixed-effect tree-based models (METBM). First of all, a mixed-effect model is designed to work on clustered data in general, including longitudinal data (see Geert Verbeke, Geert Molenberghs, and Geert Verbeke 1997). Sela and J.S. Simonoff 2012, Capitaine, Genuer, and Thiébaud 2021, Fu and J.S. Simonoff 2015 and Hajjem, Bellavance, and Larocque 2014b describe a procedure to fit a mixed effect model using tree-based models through an iterative two-step process⁷. Mixed effect tree-based algorithms are designed to take clustered data as input. By considering subjects as clusters, they can grasp the dependence structure within the different observations of a single subject and can be used for longitudinal analysis (see Geert Verbeke, Geert Molenberghs, and Geert Verbeke 1997). The underlying idea behind mixed effect tree-based algorithms is to assume a mixed model for the longitudinal outcome and estimate the random effect parameters with a tree-based model. Such approaches estimate the random effects of a mixed model in the first step, then construct a regression tree with the fixed-effect covariates on the original outcome, excluding the estimated random effect. The idea is to repeat these two steps: the model parameters and the random effects are estimated iteratively until convergence, similar to the two-step well-known EM optimization procedure. Suppose that we have p_f covariates with a fixed effect and p_s covariates with a random effect. Initially, a parametric linear mixed-effect model is given by

$$\mathbf{z}^{(i)} = F^{(i)\top} \boldsymbol{\beta} + S^{(i)\top} \mathbf{b}^{(i)} + \boldsymbol{\epsilon}^{(i)}. \quad (12.18)$$

where $\mathbf{z}^{(i)}$ is the $n^{(i)} \times 1$ longitudinal vector outcome of subject i , $\boldsymbol{\beta}$ is the $p_f \times 1$ vector of the fixed effect coefficients and $F^{(i)}$ is the $n^{(i)} \times p_f$ design matrix of the covariates with a fixed effect. The quantity $\mathbf{b}^{(i)}$ is the $p_s \times 1$ vector of random effects and $S^{(i)}$ is the $n^{(i)} \times p_s$ design matrix of the covariates with a subject-specific effect. By construction, $F^{(i)}$ and $S^{(i)}$ are subdivisions of the covariate space. The error term $\boldsymbol{\epsilon}^{(i)}$ is the $n^{(i)} \times 1$ vector of residuals, assumed to come from a normal distribution with mean 0 and variance σ^2 , and we assume $\mathbf{b}^{(i)} \sim \mathcal{N}(0, D)$, $\boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(0, \sigma^2 \cdot \mathbb{1}_{n^{(i)}})$. Eventually, D is the $p_s \times p_s$ variance-covariance matrix for the random

⁷The algorithms corresponding to their respective work are available in the R packages REEMtree and LongituRF, the R function “REEMctree” and the Python library MERF.

effects.

In order to model a longitudinal outcome with non-linear fixed effects, a tree-based model is included in Equation 12.18, as follows:

$$\mathbf{z}^{(i)} = f(F^{(i)}) + S^{(i)\top} \mathbf{b}^{(i)} + \epsilon^{(i)}. \quad (12.19)$$

Here the linear structure of the fixed effect part of the model is generalized: the fixed effects are described by a function of the fixed-effect covariates f , which is the part that a tree-based model will estimate. In MERT (Hajjem, Bellavance, and Larocque 2011b), the tree-based model is a single regression tree, in MERF (Hajjem, Bellavance, and Larocque 2014b), it is a random forest, whereas in RE-EM (Sela and J.S. Simonoff 2012; Fu and J.S. Simonoff 2015) it can be both. A general algorithm for such mixed effect tree-based models can be described as follows:

Algorithm 7 Mixed effect tree-based model pseudo-code

- 1: **Input:** \mathcal{D} , a longitudinal dataset with an outcome $\mathbf{z}^{(i)}$, $\forall i \in [1 \dots N]$
 - 2: **Output:** $\hat{\mathbf{z}}^{(i)}$, \hat{f} , $\hat{\mathbf{b}}^{(i)}$, $\hat{\epsilon}^{(i)}$, $\hat{\sigma}^{(i)2}$, $\hat{D}^{(i)}$, $\forall i \in [1 \dots N]$
 - 3:
 - 4: Initialize: $\hat{b} \leftarrow 0$, $\hat{\sigma}^2 \leftarrow 1$, $\hat{D} \leftarrow \mathbb{1}_{p_s}$
 - 5: **while** GLL < some convergence threshold **do**
 - 6: 1. $\mathbf{z}^{(i)} \leftarrow \mathbf{z}^{(i)} - S^{(i)\top} \mathbf{b}_i$
 - 7: 2. Fit a tree-based model on $\mathbf{z}^{(i)}$ and obtain \hat{f}
 - 8: 3. Infer the updated random effects parameters $\hat{\mathbf{b}}^{(i)}$
 - 9: 4. Compute $\hat{\epsilon}^{(i)} = \mathbf{z}^{(i)} - \hat{f}(F^{(i)}) - S^{(i)\top} \hat{\mathbf{b}}^{(i)}$
 - 10: 5. Update $\hat{\sigma}^{(i)2}$ and $\hat{D}^{(i)}$
 - 11: 6. Update GLL, the generalized log-likelihood criterion used to control for convergence
 - 12: **end while**
-

For further details about all these elements - and notably, the update formulas for $\hat{\sigma}^{(i)2}$, $\hat{D}^{(i)}$ and GLL - we refer the astute reader to the work of Hajjem, Bellavance, and Larocque 2014b (see Section 2 for details on how the between-subject standard error can be estimated from a METBM). Once fit, the mixed-effect tree-based model can be used to predict the vector $\hat{\mathbf{z}}^{(i)}$, the longitudinal predicted trajectory of an LMS-induced profit for any subject. For subjects with past observations included in the training dataset, the prediction includes the random effect correction:

$$\hat{\mathbf{z}}^{(i)} = \hat{f}(F^{(i)}) + S^{(i)\top} \hat{\mathbf{b}}^{(i)}.$$

For a new subject, with a first observation in the testing set, the mixed-effect prediction only includes the fixed effect:

$$\hat{\mathbf{z}}^{(i)} = \hat{f}(F^{(i)}).$$

Moreover, making predictions with such models at given times imposes that we know the value of the longitudinal covariates at those times. This implies that to compute future values of $\mathbf{z}^{(i)}(t)$, future unknown values of the longitudinal covariates are needed. In other words, no predictions for any subject are made beyond that subject's last observation time value unless we assume future values of the longitudinal covariates.

Results

This section contains the results of the *regression step* of our framework. In order to model whether a policyholder is worth targeting or not, we fit a mixed-effect tree-based regression

model to our longitudinal dataset with $z^{(i)}$, the vector of $n^{(i)}$ observations as a longitudinal target variable for every subject i . As $z^{(i)}$ can take any real value, the mean squared error (MSE) in the tree-based part of the mixed-effect model is to be preferred. For a given LLMS, the survival step allows us to compute $z^{(i)}$, the longitudinal variable representing the expected trajectory of the profits or losses generated by subject i . Then, by estimating $z^{(i)}$ on various LLMS with a mixed effect tree-based model, we can hope to find an optimal retention strategy in the sense that it will maximize the expected gain for the life insurer. For this application, we assume parameters p, η, γ , and d to be constant over all policyholders and over time and we fit a mixed effect random forest (MERF). We suggest testing five LLMS:

- one that is obviously and extremely bad and would lead to a loss for the insurer, if applied to a large number of subjects (LLMS n°1)
- one that is unrealistically good, with a small incentive largely accepted and would lead to a sure profit for the insurer (LLMS n°2)
- three realistic strategies, with various degrees of aggressivity (LLMS n°3, 4 and 5)

We train our targeting mixed-effect random forest model on all observations and their respective retention probabilities up to 2020 and test it on all subjects with an observation in 2021. We can note that in 2021, there are predictions on subjects with past observations prior to 2021 but also predictions on new subjects not included in the training set. Overall, the testing set contains “only” 4,472 unique policyholders, hence the order of magnitude of the retention gains presented below. We also chose a very conservative risk parameter, that greatly reduces the number of subjects targeted.

Here are the five strategies, and the corresponding expected profit or loss⁸ they induce:

Table 12.4: Various LMS results with our framework

LMS n°	p	η	γ	c	d	T	RG	# targets	campain investement
1	1%	1%	90%	200	2.00%	10	0	0	0
2	5%	0.01%	80%	5	2.00%	20	134,347.54	141	705
3	3%	0.009%	40%	15	1.50%	20	3,112.03	98	1470
4	2.5%	0.005%	15%	10	1.50%	20	2,940.51	94	940
5	3%	0.001%	5%	5	1.50%	20	2,962.68	122	610

Evidently, the main feature proposed by this framework is that it allows the decision maker to choose the best LLMS among realistic ones. In our application, we immediately see that in terms of profit for the insurer, strategy n°3 is optimal, compared to LLMS n°4 and 5. On the other hand, other factors, such as the number of policyholders to target or the cost of the campaign, are also displayed. they can prove to be critical elements of decisions in a real-world context, as some life insurers could have a limited commercial workforce or investment budget. For instance, an insurer that can only contact up to 95 policyholders this year would choose LLMS n°4, and another that would be limited by a 1,000€ budget for retention would choose LLMS n°5. Moreover, the bad LMS n°1 demonstrates that this framework allows us to detect whenever a strategy should not be carried out. In that case, the conclusion of the targeting step is not

⁸As defined in Definition 3

target any policyholder, thus limiting the insurer’s loss to 0, which is arguably a desirable feature. Finally, the unrealistically good LLMS n°2 shows that this framework cannot detect a “too good to be true” strategy with an unrealistic pair of parameters (η, γ) . This emphasizes the fact that taking this interdependency into account directly in the framework should prevent such unrealistic scenarios and avoid the life insurer the task of selecting in advance a consistent set of LLMS parameters. Another novelty in this framework is the longitudinal structure of the results. Indeed, we can easily retrieve the expected individual loss or profit at any future time. For example, here is a plot of the expected profits generated by targeting randomly selected policyholders:

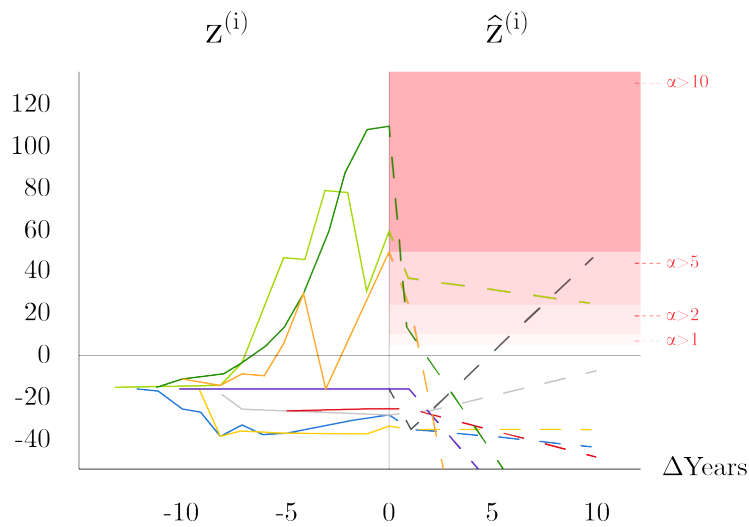


Figure 12.7: Projections of targetted profits over time

Most policyholders have a $\hat{z}^{(i)}$ with a decreasing future trajectory. It makes sense as time is positively correlated with one’s policy probability to end: the more the insurer waits to offer an incentive to a subject, the more the insurer becomes. Usually, if a policyholder does not generate profit by being targeted now, it is even less relevant to target her later in time. For specific profiles, the lapse risk grows faster than the death risk. It can then become more profitable to offer an incentive as the lapse risk increases if the death risk is insignificant.

In any case, we show graphically that depending on the level of risk α that the insurer consent to take, the time at which it is optimal to apply an LLMS to a given policyholder changes. The longitudinal trajectory being estimated with a linear model, the framework as it stands should not be used to evaluate the time when offering an incentive is optimal. It rather yields information about individual tendencies and answers strategical questions: is it profitable to target a given policyholder now? If not now, is it likely to become profitable in the future? And if it is, should the insurer decide quickly or can it wait? The individual intercepts and slopes of the future estimations of $\hat{z}^{(i)}$ answer those questions.

This example of a time-dynamic application shows that including longitudinal data in a lapse management strategy can benefit a life insurer in terms of prediction accuracy and decision-making.

12.4 Conclusion, limitations and future work

In conclusion, this paper presents a novel longitudinal lapse management framework that is tailored specifically for life insurers. The framework enhances the targeting stage of retention campaigns by selectively applying it to policyholders who are likely to generate long-term profits for the life insurer. Our key contribution is the adaptation of existing methodologies to a longitudinal setting through the use of tree-based models. The results of our application demonstrate the advantages of approaching lapse management in a longitudinal context. The use of longitudinally structured data significantly improves the precision of the models in predicting lapse behavior, estimating customer lifetime value, and evaluating individual retention gains. The implementation of mixed-effect random forests enables the production of time-varying predictions that are highly relevant for decision-making. The framework is designed to prevent the application of loss-inducing strategies and allows the life insurer to select the most profitable LMS, under constraints.

However, our work has several limitations that must be acknowledged:

Firstly regarding the framework: the longitudinal lapse management strategy is defined with fixed incentive, probability of acceptance and cost of contact, regardless of the time in the future. Moreover, the γ parameter is constant for a given policyholder, but it could be seen as the realization of a random variable following a chosen distribution. Those points may restrict the framework's practical effectiveness. Moreover, we did not account for the interdependence between different LLMS parameters, which could lead to the implementation of unrealistic strategies. Additionally, the introduction of the confidence level α could be discussed further as it could be linked with actuarial risk measures such as the Value-at-Risk. Eventually, the article describes a discrete-time longitudinal methodology, but in general, the insurer has access to the precise dates of any policy's financial flows. Thus, a continuous-time framework could also be implemented.

Secondly regarding the application: a lot of assumptions have been formulated in the application we propose such as constant parameters where the framework allows them to vary across time and policyholders, or the use of MERF where more complex and completely non-linear models could be tried.

Finally regarding longitudinal tree-based models: the use of LTRC and MERF requires the management of time-varying covariates with the pseudo-subject approach, which has practical limitations and prevents the longitudinal data from being predicted alongside the target variable. Future works could address the latter remark using joint models (see Appendix C.1 for references).

The limitations of the general framework should be discussed and tackled in forthcoming research. Other use-case and applications, with sensitivity analysis over various sets of parameters, models and datasets could constitute an engaging following work. Pseudo-subjects limitations are inherent in the current design of longitudinal tree-based models. Future work will involve developing innovative algorithms to address these issues. Overall, this article opens the field of lapse behavior analysis to longitudinal models, and our framework has the potential to improve retention campaigns and increase long-term profitability for a life insurer.

In Part IV, we aimed to introduce a longitudinal lapse management framework to the actuarial literature and display a concrete application of several longitudinal tree-based models to lapse behavior analysis. The idea was to improve on the application of Part III and go one step further into temporal dynamics considerations.

In this chapter, we have critically examined the literature about tree-based models for longitudinal analysis, particularly those with time-varying covariates. We have outlined why and how actuarial sciences can leverage these methods. Our literature review is bifurcated based on whether the response variable is time-to-event or not. From this comprehensive review, we presented a selection of models which we consider state-of-the-art. We posited that actuarial challenges can be addressed through such tree-based models. For survival analysis, we consider LTRC-like models to be state-of-the-art approaches in the presence of longitudinal survival data. For regression purposes, the best models in the literature proved to be mixed-effect tree-based models (METBM). Both algorithms were detailed in this chapter and included within the lapse management framework introduced in Part III. The results of the application illustrate the benefits of approaching lapse management in a longitudinal context, as it significantly enhances the precision of the models when predicting lapse behavior, estimating customer lifetime value, and evaluating individual retention gains. METBMs have been implemented to produce time-varying predictions that are of high relevance for decision-making. The framework aims to prevent loss-inducing strategies and enables the insurer to select the most profitable LMS under constraints. This framework as well as the application does still have certain limitations as it is subject to a lot of practical assumptions. Moreover, longitudinal tree-based models in general are not always very interpretable and clear visualizations of the temporal insights they yield would be of high interest for decision-makers in an insurance company. As we move forward, we must be mindful of the importance of interpretability and visualization in actuarial studies.

Bibliography

- Valla, M. (July 2023). “A longitudinal framework for lapse management in life insurance”. working paper or preprint. URL: <https://hal.science/hal-04178278>.
- Rizopoulos, D. (2012). *Joint Models for Longitudinal and Time-to-Event Data, with Applications in R*. en. Boca Raton: Chapman & Hall/CRC.
- Frees, E.W. and T.W. Miller (2004). “Sales forecasting using longitudinal data models”. In: *International Journal of Forecasting* 20.1, pp. 99–114. ISSN: 0169-2070. DOI: [https://doi.org/10.1016/S0169-2070\(03\)00005-0](https://doi.org/10.1016/S0169-2070(03)00005-0). URL: <https://www.sciencedirect.com/science/article/pii/S0169207003000050>.
- Kalbfleisch, J.D. and R.L. Prentice (2002). *The Statistical Analysis of Failure Time Data*. en. 2nd. New York: John Wiley and Sons. DOI: [10.1002/9781118032985](https://doi.org/10.1002/9781118032985).
- Laird, N.M. (2022). “Statistical analysis of longitudinal studies”. In: *International Statistical Review* 90.S1, S2–S16. DOI: <https://doi.org/10.1111/insr.12523>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/insr.12523>.
- Henckaerts, R., M.P. Côté, et al. (2021). “Boosting Insights in Insurance Tariff Plans with Tree-Based Machine Learning Methods”. In: *North American Actuarial Journal* 25.2, pp. 255–285. DOI: [10.1080/10920277.2020.1745656](https://doi.org/10.1080/10920277.2020.1745656).
- Pesantez-Narvaez, J., M. Guillen, and M. Alcañiz (2019). “Predicting Motor Insurance Claims Using Telematics Data—XGBoost versus Logistic Regression”. In: *Risks* 7.2. ISSN: 2227-9091. DOI: [10.3390/risks7020070](https://doi.org/10.3390/risks7020070). URL: <https://www.mdpi.com/2227-9091/7/2/70>.
- Boucher, J.P. and R. Turcotte (2020). “A Longitudinal Analysis of the Impact of Distance Driven on the Probability of Car Accidents”. In: *Risks* 8.3. ISSN: 2227-9091. DOI: [10.3390/risks8030091](https://doi.org/10.3390/risks8030091). URL: <https://www.mdpi.com/2227-9091/8/3/91>.
- Henckaerts, R. and K. Antonio (2022). “The added value of dynamically updating motor insurance prices with telematics collected driving behavior data”. In: *Insurance: Mathematics and Economics* 105, pp. 79–95. ISSN: 0167-6687. DOI: <https://doi.org/10.1016/j.insmatheco.2022.03.011>. URL: <https://www.sciencedirect.com/science/article/pii/S0167668722000385>.
- Gu, S., B. Kelly, and D. Xiu (Feb. 2020). “Empirical Asset Pricing via Machine Learning”. In: *The Review of Financial Studies* 33.5, pp. 2223–2273. ISSN: 0893-9454. DOI: [10.1093/rfs/hha009](https://doi.org/10.1093/rfs/hha009). eprint: <https://academic.oup.com/rfs/article-pdf/33/5/2223/33209812/hha009.pdf>.
- Loisel, S., P. Piette, and C.H.J. Tsai (2021). “Applying economic measures to lapse risk management with Machine Learning approaches”. In: *ASTIN Bulletin: The Journal of the IAA* 51.3, pp. 839–871. DOI: [10.1017/asb.2021.10](https://doi.org/10.1017/asb.2021.10).
- Antonio, K., J. Beirlant, et al. (2006). “Lognormal Mixed Models for Reported Claims Reserves”. In: *North American Actuarial Journal* 10.1, pp. 30–48. DOI: [10.1080/10920277.2006.10596238](https://doi.org/10.1080/10920277.2006.10596238).

- Antonio, K. and J. Beirlant (2008). “Issues in Claims Reserving and Credibility: A Semiparametric Approach With Mixed Models”. In: *Journal of Risk and Insurance* 75.3, pp. 643–676. DOI: <https://doi.org/10.1111/j.1539-6975.2008.00278.x>.
- Fisher, L.D. and D.Y. Lin (1999). “Time-dependent covariates in the Cox proportional hazards regression model”. In: *Annual Review of Public Health* 20.1. PMID: 10352854, pp. 145–157. DOI: [10.1146/annurev.publhealth.20.1.145](https://doi.org/10.1146/annurev.publhealth.20.1.145).
- Meyer, B.D. (1990). “Unemployment Insurance and Unemployment Spells”. fr. In: *PDF). Econometrica* 58.4, pp. 757–782. DOI: [10.2307/2938349](https://doi.org/10.2307/2938349).
- Bover, O., M. Arellano, and S. Bentolila (2002). “Unemployment Duration, Benefit Duration, and the Business Cycle”. en. In: *PDF). The Economic Journal* 112.479, pp. 223–265. DOI: [10.1111/1468-0297.00034](https://doi.org/10.1111/1468-0297.00034).
- Scheike, T.H. and T. Martinussen (2006). *Dynamic Regression models for survival data*. Springer, NY.
- Scheike, T.H. and M.J. Zhang (2011). “Analyzing Competing Risk Data Using the R timereg Package”. In: *Journal of Statistical Software* 38.2, pp. 1–15. URL: <https://www.jstatsoft.org/v38/i02/>.
- Segal, M.R. (1992a). “Tree-structured methods for longitudinal data”. In: *Journal of the American Statistical Association* 87.418, pp. 407–418.
- Kundu, M.G. and J. Harezlak (2019). “Regression trees for longitudinal data with baseline covariates”. In: *Biostatistics & Epidemiology* 3.1, pp. 1–22. DOI: [10.1080/24709360.2018.1557797](https://doi.org/10.1080/24709360.2018.1557797).
- De’Ath, G. (2002). “Multivariate regression trees: a new technique for modeling species-environment relationships”. en. In: *Ecology* 83, pp. 1105–1117.
- Larsen, D.R. and P.L. Speckman (2004). “Multivariate regression trees for analysis of abundance data”. en. In: *Biometrics* 60, pp. 543–549.
- Hsiao, W.C. and Y.S. Shih (2007). “Splitting variable selection for multivariate regression trees”. en. In: *Statistics and Probability Letters* 77, pp. 265–271.
- Zhang, H. (1998). “Classification trees for multiple binary responses”. en. In: *Journal of the American Statistical Association* 93, pp. 180–193.
- Abdolell, M. et al. (2002). “Binary partitioning for continuous longitudinal data: categorizing a prognostic variable”. it. In: *Statistics in Medicine* 21, pp. 3395–3409.
- Lee, S.K. (2005). “On generalized multivariate decision tree by using GEE”. en. In: *Computational Statistics & Data Analysis* 49, pp. 1105–1119.
- Lee, S.K. et al. (2005). “Using generalized estimating equations to learn decision trees with multivariate responses”. en. In: *Data Mining and Knowledge Discovery* 11, pp. 273–293.
- Lee, S.K. (2006). “On classification and regression trees for multiple responses and its application”. en. In: *Journal of Classification* 23, pp. 123–141.
- Chaudhuri, P. et al. (1995). “Generalized regression trees”. it. In: *Stat. Sinica* 5, pp. 641–666.
- Ritschard, G. and M. Oris (2005). “Life course data in demography and social sciences: statistical and data mining approaches”. en. In: *Towards an interdisciplinary perspective on the life course, advances in life course research*. Ed. by R. Levy et al. Amsterdam: Elsevier, pp. 289–320.
- Moradian, H. et al. (2021). “Dynamic estimation with random forests for discrete-time survival data”. In: *Canadian Journal of Statistics*.
- Sela, R. and J.S. Simonoff (Feb. 2012). “RE-EM trees: A data mining approach for longitudinal and clustered data”. In: *Machine Learning* 86, pp. 169–207. DOI: [10.1007/s10994-011-5258-3](https://doi.org/10.1007/s10994-011-5258-3).
- Fu, W. and J.S. Simonoff (2015). “Unbiased regression trees for longitudinal and clustered data”. In: *Computational Statistics & Data Analysis* 88, pp. 53–74. ISSN: 0167-9473. DOI: <https://doi.org/10.1016/j.csda.2015.05.001>.

- [//doi.org/10.1016/j.csda.2015.02.004](https://doi.org/10.1016/j.csda.2015.02.004). URL: <https://www.sciencedirect.com/science/article/pii/S0167947315000432>.
- Galimberti, G. and A. Montanari (2002). “Regression trees for longitudinal data with time-dependent covariates”. en. In: ed. by K. Jajuga, A. Sokolowski, and H.-H. Bock, pp. 391–398.
- Harville, D.A. (1977). “Maximum likelihood approaches to variance component estimation and to related problems”. en. In: *Journal of the American Statistical Association* 72, pp. 320–340.
- Laird, N.M. and J.H. Ware (1982). “Random-Effects Models for Longitudinal Data”. In: *Biometrics* 38.4, pp. 963–974. ISSN: 0006341X, 15410420. URL: <http://www.jstor.org/stable/2529876>.
- Verbeke, G. and G. Molenberghs (2000). “Linear mixed models for longitudinal data”. da. In: Hajjem, A., F. Bellavance, and D. Larocque (2011a). “Mixed effects regression trees for clustered data”. In: *Statistics & probability letters* 81.4, pp. 451–459.
- Capitaine, L. (Dec. 2020). “Forêts aléatoires pour données longitudinales de grande dimension.” Theses. Université de Bordeaux. URL: <https://theses.hal.science/tel-03151753>.
- Capitaine, L., R. Genuer, and R. Thiébaud (2021). “Random forests for high-dimensional longitudinal data”. In: *Statistical Methods in Medical Research* 30.1. PMID: 32772626, pp. 166–184. DOI: [10.1177/0962280220946080](https://doi.org/10.1177/0962280220946080).
- Hajjem, A., F. Bellavance, and D. Larocque (2014a). “Mixed-effects random forest for clustered data”. In: *Journal of Statistical Computation and Simulation* 84.6, pp. 1313–1328.
- Fu, W. and J.S. Simonoff (2015). “Unbiased regression trees for longitudinal and clustered data”. In: *Computational Statistics & Data Analysis* 88, pp. 53–74.
- Hajjem, A., F. Bellavance, and D. Larocque (2011b). “Mixed effects regression trees for clustered data”. In: *Statistics & Probability Letters* 81.4, pp. 451–459. URL: <https://ideas.repec.org/a/eee/stapro/v81y2011i4p451-459.html>.
- (2014b). “Mixed-effects random forest for clustered data”. In: *Journal of Statistical Computation and Simulation* 84.6, pp. 1313–1328. DOI: [10.1080/00949655.2012.741599](https://doi.org/10.1080/00949655.2012.741599).
- Devaux, A. (Nov. 2022). “Modélisation et prédiction dynamique individuelle d’événements de santé à partir de données longitudinales multivariées”. Theses. Université de Bordeaux. URL: <https://theses.hal.science/tel-03909257>.
- Devaux, A., R. Genuer, and K. Peres (2022). “Individual dynamic prediction of clinical endpoint from large dimensional longitudinal biomarker history: a landmark approach”. In: *BMC Med Res Methodol* 22 188.
- Devaux, A., C. Helmer, et al. (2023). *Random survival forests with multivariate longitudinal endogenous covariates*. arXiv: [2208.05801](https://arxiv.org/abs/2208.05801) [stat.ML].
- Devaux, A., C. Proust-Lima, and R. Genuer (2023). *Random Forests for time-fixed and time-dependent predictors: The DynForest R package*. arXiv: [2302.02670](https://arxiv.org/abs/2302.02670) [stat.ML].
- Wei, Y. et al. (2020). “Precision medicine: Subgroup identification in longitudinal trajectories”. In: *Statistical methods in medical research* 29.9, pp. 2603–2616.
- Eo, S.H. and H.J. Cho (2014). “Tree-Structured Mixed-Effects Regression Modeling for Longitudinal Data”. In: *Journal of Computational and Graphical Statistics* 23.3, pp. 740–760. DOI: [10.1080/10618600.2013.794732](https://doi.org/10.1080/10618600.2013.794732).
- Segal, M.R. (1992b). “Tree-structured models for longitudinal data”. en. In: *Journal of the American Statistical Association* 87, pp. 407–418.
- Simonof, J.S. (2016). *Regression Trees for Longitudinal and Clustered Data: Methods, Applications, and Extensions*. en. URL: <https://modeling.uconn.edu/wp-content/uploads/sites/1188/2016/05/Regression-Trees-for-Longitudinal-and-Clustered-Data-Methods-Applications-and-Extensions.pdf>.

- Bacchetti, P. and M.R. Segal (1995). “Survival trees with time-dependent covariates: application to estimating changes in the incubation period of aids”. en. In: *Lifetime Data Analysis* 1, pp. 35–47.
- Huang, X.Y., S.C. Chen, and S.J. Soong (1998). “Piecewise exponential survival trees with time-dependent covariates.” In: *Biometrics* 54 4, pp. 1420–33.
- Fu, W. and J.S. Simonoff (June 2016a). “Survival trees for left-truncated and right-censored data, with application to time-varying covariate data”. In: *Biostatistics (Oxford, England)* 18. DOI: [10.1093/biostatistics/kxw047](https://doi.org/10.1093/biostatistics/kxw047).
- Yao, W; et al. (2022). “Ensemble methods for survival function estimation with time-varying covariates”. In: *Statistical Methods in Medical Research* 31.11. PMID: 35895510, pp. 2217–2236. DOI: [10.1177/09622802221111549](https://doi.org/10.1177/09622802221111549). URL: <https://export.arxiv.org/pdf/2006.00567>.
- Bou-Hamad, I. (2009). “Discrete–Time Survival Trees”. en. In: *Canadian Journal of Statistics* 37. MR2509459, pp. 17–32.
- Bou-Hamad, I., D. Larocque, and H. Ben-Ameur (2011). *Discrete–Time Survival Trees and Forests with Time-Varying Covariates: Application to Bankruptcy Data*. en. To appear in *Statistical Modeling*.
- Breiman, L. (2002). “Software for the masses”. en. In: *Wald Lectures, Meeting of the Institute of Mathematical Statistics*. Available at Banff, Canada. URL: <http://www.stat.berkeley.edu/users/breiman..>
- Xu, R. and S. Adak (2001). “Survival Analysis with Time–Varying Relative Risks: A Tree–Based Approach”. et. In: *Methods of information in medicine* 40, pp. 141–147.
- (2002). “Survival Analysis with Time–Varying Regression Effects Using a Tree–Based Approach”. en. In: *Biometrics* 58, pp. 305–315.
- Lin, J., K. Li, and S. Luo (2021a). “Functional survival forests for multivariate longitudinal outcomes: Dynamic prediction of Alzheimer’s disease progression”. en. In: *Stat Methods Med Res* Jan;30(1):99–111. PMID: 32726189; PMCID: PMC7855476. DOI: [10.1177/0962280220941532..](https://doi.org/10.1177/0962280220941532..)
- Ishwaran, H. et al. (2014). “Random survival forests for competing risks”. io. In: *Biostatistics* Oct;15(4):757–73. Epub 2014 Apr 11. PMID: 24728979; PMCID: PMC4173102. DOI: [10.1093/biostatistics/kxu010..](https://doi.org/10.1093/biostatistics/kxu010..)
- Wongvibulsin, S., K.C. Wu, and S.L. Zeger (2020). “Clinical risk prediction with random forests for survival, longitudinal, and multivariate (RF-SLAM) data analysis”. en. In: *BMC Med Res Methodol* 20, 1. DOI: [10.1186/s12874-019-0863-0](https://doi.org/10.1186/s12874-019-0863-0).
- Lin, J., K. Li, and S. Luo (2021b). “Functional survival forests for multivariate longitudinal outcomes: Dynamic prediction of Alzheimer’s disease progression”. en. In: *Stat Methods Med Res* Jan;30(1):99–111. PMID: 32726189; PMCID: PMC7855476. DOI: [10.1177/0962280220941532..](https://doi.org/10.1177/0962280220941532..)
- Pickett, K.L., K. Suresh, and K.R. Campbell (2021). “Random survival forests for dynamic predictions of a time-to-event outcome using a longitudinal biomarker”. en. In: *BMC Med Res Methodol* 21, p. 216. DOI: [10.1186/s12874-021-01375-x](https://doi.org/10.1186/s12874-021-01375-x).
- Singer, J.D. and J.B. Willett (2003). *Applied longitudinal data analysis: Modeling change and event occurrence*. Oxford University Press.
- Hyndman, R.J. and G. Athanasopoulos (2018). *Forecasting: principles and practice*. OTexts.
- Hyndman, R.J. and A.B. Koehler (2006). “Another look at measures of forecast accuracy”. In: *International journal of forecasting* 22.4, pp. 679–688.
- Hartman, N. et al. (2023). “Pitfalls of the concordance index for survival outcomes”. In: *Statistics in Medicine* 42.13, pp. 2179–2190. DOI: <https://doi.org/10.1002/sim.9717>.
- Lambert, J. and S. Chevret (2016). “Summary measure of discrimination in survival models based on cumulative/dynamic time-dependent ROC curves”. In: *Statistical Methods in Medical Research* 25.5. PMID: 24395866, pp. 2088–2102. DOI: [10.1177/0962280213515571](https://doi.org/10.1177/0962280213515571).

- van Geloven, N. et al. (2021). “Estimation of incident dynamic AUC in practice”. In: *Computational Statistics & Data Analysis* 154, p. 107095. ISSN: 0167-9473. DOI: <https://doi.org/10.1016/j.csda.2020.107095>. URL: <https://www.sciencedirect.com/science/article/pii/S0167947320301869>.
- Valla, M., X. Milhaud, and A.A. Olympio (Sept. 2023). “Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies”. In: *European Actuarial Journal*. DOI: [10.1007/s13385-023-00358-0](https://doi.org/10.1007/s13385-023-00358-0). URL: <https://hal.science/hal-03903047> (HAL), <https://export.arxiv.org/pdf/2307.06651> (arxiv), <https://link.springer.com/article/10.1007/s13385-023-00358-0?code=84d3a0d0-b866-48d5-bc60-5ed6832d144a> (EAJ).
- Hardy, M. (2003). *Investment guarantees: modeling and risk management for equity-linked life insurance*. Vol. 168. John Wiley & Sons.
- Bacinello, A.R. (2005). “Endogenous model of surrender conditions in equity-linked life insurance”. In: *Insurance: Mathematics and Economics* 37.2. Papers presented at the 8th IME Conference, Rome, 14-16 June 2004, pp. 270–296. ISSN: 0167-6687. DOI: <https://doi.org/10.1016/j.insmathco.2005.02.002>. URL: <https://www.sciencedirect.com/science/article/pii/S0167668705000156>.
- MacKay, A. et al. (2017). “Risk Management of Policyholder Behavior in Equity-Linked Life Insurance”. In: *Journal of Risk and Insurance* 84.2, pp. 661–690. DOI: <https://doi.org/10.1111/jori.12094>.
- Gupta, S., D.R. Lehmann, and J.A. Stuart (2004). “Valuing customers”. In: *Journal of marketing research* 41.1, pp. 7–18.
- Outreville, J.F. (1990). “Whole-life insurance lapse rates and the emergency fund hypothesis”. In: *Insurance: Mathematics and Economics* 9.4, pp. 249–255. URL: <https://EconPapers.repec.org/RePEc:eee:insuma:v:9:y:1990:i:4:p:249-255>.
- Eling, M. and M. Kochanski (2013). “Research on lapse in life insurance: what has been done and what needs to be done?” In: *Journal of Risk Finance* 14.4, pp. 392–413. URL: <https://EconPapers.repec.org/RePEc:eme:jrfpps:v:14:y:2013:i:4:p:392-413>.
- Donkers, B., P. Verhoef, and M. Jong (Feb. 2007). “Modeling CLV: A test of competing models in the insurance industry”. In: *Quantitative Marketing and Economics* 5, pp. 163–190. DOI: [10.1007/s11129-006-9016-y](https://doi.org/10.1007/s11129-006-9016-y).
- Berger, P. and N. Nasr (Dec. 1998). “Customer Lifetime Value: Marketing Models and Applications”. In: *Journal of Interactive Marketing* 12, pp. 17–30. DOI: [10.1002/\(SICI\)1520-6653\(199824\)12:1<17::AID-DIR3>3.0.CO;2-K](https://doi.org/10.1002/(SICI)1520-6653(199824)12:1<17::AID-DIR3>3.0.CO;2-K).
- Ascarza, E. et al. (2018). In *Pursuit of Enhanced Customer Retention Management: Review, Key Issues, and Future Directions*. en. Special Issue on 2016 Choice Symposium. Customer Needs and Solutions 5,
- Guelman, L., G. Montserrat, and A.M. Pérez-Marín (2012). “Random Forests for Uplift Modeling: An Insurance Customer Retention Case”. In: *Modeling and Simulation in Engineering, Economics and Management*. Ed. by Kurt J. Engemann, Anna M. Gil-Lafuente, and José M. Merigó. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 123–133. ISBN: 978-3-642-30433-0.
- Risselada, H., P.C. Verhoef, and T.H.A. Bijmolt (2010). “Staying Power of Churn Prediction Models”. In: *Journal of Interactive Marketing* 24.3, pp. 198–208. DOI: [10.1016/j.intmar.2010.04.002](https://doi.org/10.1016/j.intmar.2010.04.002).
- Fu, W. and J.S. Simonoff (Dec. 2016b). “Survival trees for left-truncated and right-censored data, with application to time-varying covariate data”. In: *Biostatistics* 18.2, pp. 352–369. ISSN: 1465-4644. DOI: [10.1093/biostatistics/kxw047](https://doi.org/10.1093/biostatistics/kxw047). eprint: <https://academic.oup.com/biostatistics/advance-article-abstract/doi/10.1093/biostatistics/kxw047/352369>

- oup.com/biostatistics/article-pdf/18/2/352/11057459/kxw047.pdf.
- Yao, W. et al. (2020). *Ensemble methods for survival function estimation with time-varying covariates*. DOI: [10.48550/ARXIV.2006.00567](https://doi.org/10.48550/ARXIV.2006.00567). URL: <https://arxiv.org/abs/2006.00567>.
- Fisher, Lloyd D. and D. Y. Lin (1999). “Time-dependent covariates in the cox proportional-hazards regression model”. In: *Annual Review of Public Health* 20.1. PMID: 10352854, pp. 145–157. DOI: [10.1146/annurev.publhealth.20.1.145](https://doi.org/10.1146/annurev.publhealth.20.1.145).
- Molenberghs, G. and G. Verbeke (2006). *Models for Discrete Longitudinal Data*. Springer Series in Statistics. Springer New York. ISBN: 9780387289809. URL: <https://books.google.fr/books?id=LjfyKpw36S8C>.
- Frees, Edward W. et al. (2021). “Dependence modeling of multivariate longitudinal hybrid insurance data with dropout”. In: *Expert Systems with Applications* 185, p. 115552. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2021.115552>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417421009581>.
- Dal Pont, M. (2020). “Construction d’une table de mortalité d’expérience en assurance emprunteur”. PhD thesis. ISFA, Université Lyon 1. URL: <https://www.institutdesactuaire.com/docs/mem/1e992efa93786a498553e9a184326e4a.pdf>.
- Campo, B.D.C. and K. Antonio (2022). *Insurance pricing with hierarchically structured data: An illustration with a workers’ compensation insurance portfolio*. DOI: [10.48550/ARXIV.2206.15244](https://doi.org/10.48550/ARXIV.2206.15244). URL: <https://arxiv.org/abs/2206.15244>.
- Moradian, H. et al. (2022). “Dynamic estimation with random forests for discrete-time survival data”. In: *Canadian Journal of Statistics* 50.2, pp. 533–548. DOI: <https://doi.org/10.1002/cjs.11639>.
- Verbeke, Geert, Geert Molenberghs, and Geert Verbeke (1997). *Linear mixed models for longitudinal data*. Springer.

Part V

Towards time-dependent trees

Chapter 13 Contributions of Part V

Chapter 14 Time-penalized trees: a new tree-based data mining algorithm for time-varying covariates

- 14.1 Introduction 145
- 14.2 Preliminaries 146
 - 14.2.1 Classification and regression trees 146
 - 14.2.2 Longitudinal notations 150
 - 14.2.3 Existing longitudinal tree-based algorithms 151
- 14.3 Time penalized trees 153
 - 14.3.1 TpT splitting criterion 155
 - 14.3.2 TpT pruning process 160
 - 14.3.3 Time-to-event 160
- 14.4 Applications 160
 - 14.4.1 Properties of TpT for the maximal tree 161
 - 14.4.2 Use-case with minsplit 164
 - 14.4.3 Pathways vizualizations 166
- 14.5 Conclusion, limitations and future work 168
 - 14.5.1 Conclusion 168
 - 14.5.2 Limitations and future work 169

Bibliography

13. Contributions of Part V

This part is based on the article “Time-penalized trees (TpT): a new tree-based data mining algorithm for time-varying covariates”, submitted in the Annals of Mathematics and Artificial Intelligence¹. This work contributes to the fields of machine learning, data mining, and actuarial science by introducing a new tree-based algorithm handling time-varying covariates and using it in a life insurance clustering application. In the scope of this thesis, Part V points out the potential drawbacks of the state-of-the-art longitudinal TBM and proposes a new approach with an innovative design. The main contributions of Part III and IV reside in the applicative methodology they detail whereas Part V mainly contributes to the data mining field by designing a novel algorithm that can detect relevant time points. This last idea is capital: some behaviors will have a different influence on the outcome, depending on when they have been measured. TpT can be used to tell when a change in the outcome is more likely to happen, or when a covariate is highly influencing the outcome. A list of various contributions found in this Part are listed below:

Contributions 7: Gaps within the longitudinal TBM literature

Identification of gaps in the existing literature and proposing solutions

Firstly, Part V identifies gaps in the existing literature about longitudinal tree-based models and proposes solutions to address these gaps. This critical evaluation enhances the understanding of the current limitations in the field, which can lead to undesired properties of the models in dynamic environments. We argue that existing methods like the pseudo-subject approach - discussed in more detail in Chapter 14 - are not entirely satisfying in terms of design and interpretability.

Contributions 8: Introduction of TpT

Development of a new theoretical model

Secondly, the article introduces a new theoretical model named “Time-penalized Tree” (TpT). This new decision tree algorithm accounts for time-varying covariates in the decision-making process, distinguishing it from existing longitudinal tree-based algorithms by utilizing a different structure and a time-penalized splitting criterion. It enables recursive partitioning of both the covariates space and time. We detail this algorithm and demonstrate its application for life insurance through real data mining and visualization examples. This innovative approach allows for a more interpretable analysis in settings where the covariates are subject to change over time, expanding its potential applications in various fields.

¹See Valla 2023a.

14. Time-penalized trees: a new tree-based data mining algorithm for time-varying covariates

Abstract. *This article proposes a new decision tree algorithm that accounts for time-varying covariates in the decision-making process. Traditional decision tree algorithms assume that the covariates are static and do not change over time, which can lead to inaccurate predictions in dynamic environments. Other existing methods suggest workaround solutions such as the pseudo-subject approach, discussed in the article. The proposed algorithm utilizes a different structure and a time-penalized splitting criterion that allow a recursive partitioning of both the covariates space and time. Relevant historical trends are then inherently involved in the construction of a tree, and are visible and interpretable once it is fit. This approach allows for innovative and highly interpretable analysis in settings where the covariates are subject to change over time. The effectiveness of the algorithm is demonstrated through real-world data analysis, highlighting its potential applications in various fields, including healthcare, finance, insurance, environmental monitoring, and data mining in general.*

Key words: *decision tree, time-varying covariate, data mining, longitudinal study, algorithm*

14.1 Introduction

Decision trees are a popular machine learning tool for data mining as well as classification and regression predictions. Growing such a tree is a data-driven process based on a set of input covariates and a target variable. The most famous decision tree algorithm is arguably Classification and Regression Trees (CART), introduced by Breiman et al. Breiman et al. 1984. CART constructs a binary tree by recursively partitioning the feature space into smaller and smaller subsets, based on a splitting criterion that maximizes the separation between the target variable's values in each subset. However, traditional decision tree algorithms like CART assume that the input features or covariates are static and do not change over time. In many real-world settings, this assumption is unrealistic, and the time-dynamic nature of the covariates is highly informative and should be included in the tree construction. In such settings, not accounting for dynamic features results in information loss, hence a loss of accuracy and richness of analysis.

Other data-driven approaches can already efficiently seize the time dimension of features in prediction and data-mining settings. One can think of neural networks (see Mena et al. 2023 or Wong et al. 2022 for instance). Yet conventional parametric statistical models or machine learning approaches such as logistic regression or most tree-based models cannot handle time-varying covariates straightforwardly. They assume that individual observations are independently distributed. Because of the longitudinal structure of a time-varying dataset (see Section 14.2.2 for

more details), this independence hypothesis cannot be met: different observations of a single subject are naturally strongly correlated. To address this limitation, some existing tree-based methods suggest workarounds such as the pseudo-subject approach in survival trees (Fu and J.S. Simonoff 2016), which create artificial left-truncated and right-censored subjects by pooling observations over time, or the inclusion of a mixed effect model structure around a tree-based core (Hajjem, Bellavance, and Larocque 2011a; Sela and J.S. Simonoff 2012; Hajjem, Bellavance, and Larocque 2014). Such computationally intensive methods proved to yield competitive results in many prediction frameworks, yet we argue in the following sections that they are not entirely satisfying in terms of interpretability.

In this article, we propose “Time-penalized Tree” (TpT), a new decision tree algorithm that accounts for time-varying covariates in the decision-making process. Our algorithm utilizes a different structure and a time-penalized splitting criterion that allows for recursive partitioning of both time and the features space. We detail the algorithm and show simulated real data-mining and visualization applications. However, it is crucial to underscore the inherent limitations in the current scope of this study. Recognizing the need for further refinement, this work primarily concentrates on introducing and demonstrating the applicability of TpT. Nevertheless, two crucial aspects remain unaddressed: firstly, the imperative need for a comprehensive exploration into the statistical properties and theoretical foundations of this new tool; and secondly, the essential comparative analysis of TpT results against existing longitudinal techniques, trained on well-studied datasets and evaluated with consistent indicators. This introduction sets the stage for future investigations, acknowledging the identified gaps and emphasizing their significance in shaping the future trajectory of our research.

The rest of this paper is structured as follows. We recall the basics about classification and regression trees as well as time-varying covariates analysis in Section 14.2, we also briefly present existing approaches and frame their interpretability flaws. Then we detail the specificities of TpT in Section 14.3 and explain its benefits, which is the main contribution of this work. In Section 14.4, we show a concrete application of our framework on a real-world life-insurance dataset, with visuals and illustration work, demonstrating the interpretability properties of TpT. Eventually, Section 14.5 concludes this paper and details future works.

14.2 Preliminaries

14.2.1 Classification and regression trees

In this section, we briefly describe the mechanisms of a simple yet powerful data-mining and prediction model: decision trees, and more specifically, classification and regression trees or CART Breiman et al. 1984. Here, we assume that all covariates are time-independent. Let $\mathcal{D} = (\mathbf{x}^{(i)}, y^{(i)})_{i=1}^N$ be a dataset of N individuals with $\mathbf{x}^{(i)} = (x_1^{(i)}, \dots, x_p^{(i)})$, the vector of p covariates and $y^{(i)}$ the target variable for the i -th subject. The covariates and target spaces are respectively denoted \mathcal{X} and \mathcal{Y} . Decision trees create a recursive partitioning of \mathcal{X} based on binary decision rules. This partitioning can be visualized directly in the case where there are two covariates x_1 and x_2 (see Figure 14.1). In that case, individual observations are represented as dots that are eventually clustered into n_L distinct, non-overlapping regions of \mathcal{X} denoted (L_1, \dots, L_{n_L}) .

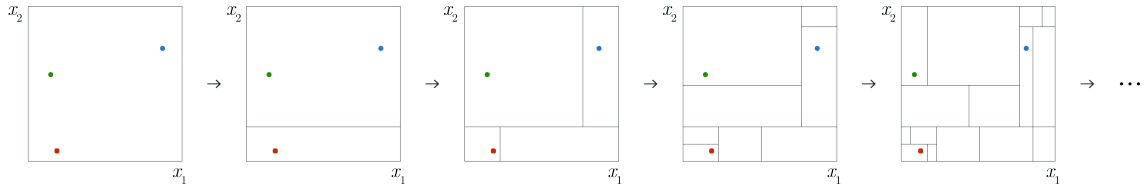


Figure 14.1: Decision tree recursive partitioning

More generally it can be visualized as a tree (see Figure 14.2), with yes/no questions within each node and terminal nodes - or leaves - corresponding to the n_L regions of the covariates space. Because the regions defined by leaves are non-overlapping, every individual i belongs to a single leaf, and a unique prediction is made for all individuals falling in a specific leaf. More generally, let g be a node, at g , we define $\mathcal{D}(g) \subseteq \mathcal{D}$ such as $\mathcal{D}(g) = \{(\mathbf{x}^{(i)}, y^{(i)}) \subseteq \mathcal{D} \mid \mathbf{x}^{(i)} \in g\}$, the set of observations in the node g . The quantity $\mathcal{N}(g) = |\mathcal{D}(g)|$ is then the number of individuals in the node.

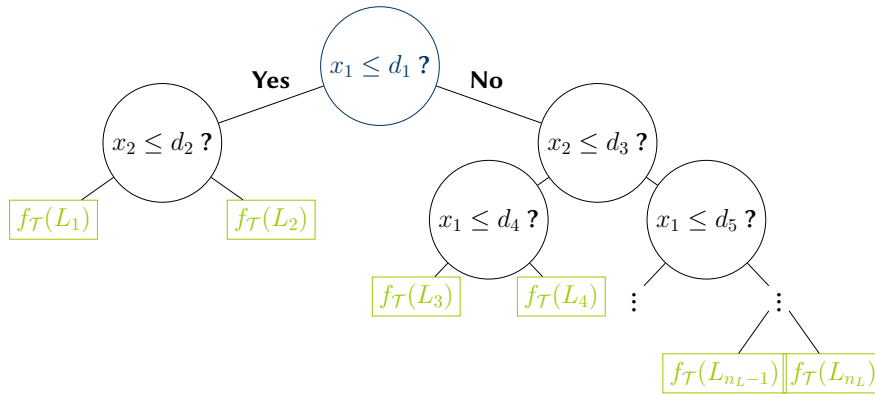


Figure 14.2: Decision tree example

In a classification context, the label given by the tree \mathcal{T} for subject i , falling in leaf L is given by

$$f_{\mathcal{T}}(\mathbf{x}^{(i)}) = \text{mode} \left(\{y^{(i)}, \forall i \mid \mathbf{x}^{(i)} \in L\} \right) = f_{\mathcal{T}}(L).$$

In a regression context, the label given by the tree \mathcal{T} for subject i in leaf L is given by

$$f_{\mathcal{T}}(\mathbf{x}^{(i)}) = \text{mean} \left(\{y^{(i)}, \forall i \mid \mathbf{x}^{(i)} \in L\} \right) = f_{\mathcal{T}}(L).$$

In both cases, a decision tree yields a single constant label¹ for an entire region: its mode or mean. The accuracy of a tree is then based on its ability to minimize the error it commits when assigning labels. Among all possible trees - thus, all possible partitions of \mathcal{X} - the optimal one should maximize a predetermined objective measure (such as the label assignment accuracy, for instance). Such a tree theoretically exists but cannot generally be found in a computationally reasonable time. Therefore algorithms like CART use a top-down greedy approach: they start from an initial node - the root - containing all observations in \mathcal{D} . Then they find the covariate x_j and the threshold d^2 such that they optimize a splitting criterion. The root is then split into those two child nodes for which the same splitting process is repeated until a stopping criterion is triggered. Once grown, this tree is called *maximal tree*. From an algorithmic perspective, growing a maximal CART can be summarized as such:

¹or prediction, in such contexts

²The set of classes for categorical covariates

Algorithm 8 Growing a maximal CART

```
1: Input: Training dataset  $\mathcal{D}$ 
2: Output: Maximal CART  $\mathcal{T}_{max}$ 
3: Initialize the root node  $g$  with the entire dataset  $\mathcal{D}$ 
4: Grow( $g$ )
5:
6: Function Grow( $g$ ):
7: if Stopping criteria met (e.g., maximum depth, minimum samples) then
8:   Let  $g$  be a leaf with the prediction  $f_{\mathcal{T}}(g)$ .
9: else
10:   For all possible covariates and thresholds find the pair  $(x_k, d)$  that obtain the best splitting
      criterion.
11:
12:   Split the node  $g$  along covariate  $x_k$  at threshold  $d$  into two child nodes  $g_r$  and  $g_l$ .
13:   Grow( $g_r$ )
14:   Grow( $g_l$ )
15: end if
```

Such a tree overfits the data, and predictions made on observations that were not used to grow the tree are usually inaccurate. That is why a last step is required: the maximal tree is pruned to a subtree that has better generalization abilities. The pruning step is described in Section 14.2.1. A decision tree is therefore defined by its splitting criterion, stopping rule(s), and its pruning process.

Splitting Criterion

Originally, CART produces, at every node, a split that minimizes the heterogeneity regarding the target variable within each child node. Equivalently, the optimal split is to maximize the loss of heterogeneity between the considered node and its child nodes: the so-called *goodness-on-split*. Therefore, measures of heterogeneity are needed when the target variable is categorical (for classification tasks) and when it is numerical (for regression tasks).

Classification: In a P -classes classification problem, let us define $p_l, l \in \{1, \dots, P\}$ as the proportion of observations of class l in \mathcal{D} . We extend this idea by defining $p_l(g)$ as the proportion of observation of class l in $\mathcal{D}(g)$. An impurity function ϕ , is a function measuring the heterogeneity, defined for $p_l, l \in \{1, \dots, P\}$, with $p_l \geq 0$ and $\sum_l p_l = 1$ such that:

- $\phi(p_1, \dots, p_P) \geq 0$,
- The minimum of ϕ is reached whenever any of the $p_l = 1$, then $\phi(p_1, \dots, p_P) = 0$,
- The maximum of ϕ is reached for $\phi(\frac{1}{P}, \dots, \frac{1}{P})$,
- ϕ is symmetric with regard to its arguments.

For CART, usual classification impurities are Gini ($\phi(p_1, \dots, p_K) = -\sum_i p_i \log(p_i)$), the entropy ($\phi(p_1, \dots, p_K) = \frac{1}{2} \sum_i p_i (1 - p_i)$) or the twoing measure. For our purposes, no further specificities are needed and in full generality, the impurity - or heterogeneity - of node g is measured by $I(g) = \phi(p_1(g), \dots, p_K(g))$. At each node of a CART, the optimal split is chosen as the split that reduces the impurity the most. That is to say, the split that maximizes the following gain

function by splitting the parent node g_p into the two child nodes g_l and g_r is

$$G(g_p; g_l, g_r) = I(g_p) - \left(\frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} I(g_l) + \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)} I(g_r) \right). \quad (14.1)$$

Of course, various other criteria and ideas for splitting exist. This paper does not aim to review all of them but we refer the astute reader to such comparisons of splitting methods (see Mingers 1989, Buntine and Niblett 1992, Breiman 1996, Shih 1999 or Drummond and Holte 2000 for instance). The efficacy of each splitting criterion has been discussed but no definitive consensus over which one is the finest exists. All measures prove desirable properties in particular scenarios while demonstrating drawbacks in others.

Regression: In a regression context, the best split can be chosen with the target variable empirical variance or mean squared error, a natural choice of heterogeneity measure. We define $MSE(g)$ the mean squared error at node g , as

$$MSE(g) = \sum_{\{i|\mathbf{x}^{(i)} \in g\}} \left(\bar{y}_g - y^{(i)} \right)^2, \quad (14.2)$$

with $\bar{y}_g = \frac{1}{\mathcal{N}(g)} \sum_{\{i|\mathbf{x}^{(i)} \in g\}} y^{(i)}$.

Then, the gain function to maximize when splitting the parent node g_p into the two child nodes g_l and g_r is obviously

$$G(g_p; g_l, g_r) = MSE(g_p) - \left(\frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} MSE(g_l) + \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)} MSE(g_r) \right). \quad (14.3)$$

Even if technically, MSE is not an impurity function, we clearly see that Equation 14.3 is the perfect regression equivalent of Equation 14.1.

Thus in the following sections, we use the general notations of equation 14.1 with $I(g) \equiv MSE(g)$ when the target variable is numerical.

Stopping rules

Stopping rules can be specified. In that case, the growing phase continues until one of them is met. First of all, a node will not split any further if all observations it contains have the same target variable value. Other commonly used stopping rules are: a minimum improvement in the splitting criterion, a maximum depth of the tree (parameter: `maxdepth`), a minimum number of observations in a node (parameter: `minsplit`), or a minimum number of observations in the hypothetical child nodes that would result from a new split.

Tree pruning

The stopping rules affect the size of the maximal tree. No or weak stopping rules will generate a high-variance/low-bias over-fitted tree whereas constraining ones will lead to smaller, more interpretable low-variance/high-bias under-fitted trees. The idea of cost-complexity pruning developed by Breiman emerged from the need to find a compromise between the two extremes.

The main idea behind cost-complexity pruning is to consider sub-trees of the maximal tree and evaluate them with a cost function that increases as the error rate rises and decreases as the

number of leaves drops. When a tree is pruned at a node, the weighted summed error of the leaves increases while the number of leaves reduces, thus a pruned sub-tree is selected only if the error gain is counter-balanced by the complexity loss. The cost of a tree \mathcal{T} is given by:

$$C_\alpha(\mathcal{T}) = R(\mathcal{T}) + \alpha\psi(n_L), \quad \alpha \geq 0, \quad (14.4)$$

where $R(\mathcal{T})$ is the sum of all errors or impurities of the leaves of \mathcal{T} , weighted by the number of individuals they represent. The function ψ is an increasing function of n_L , it is originally set to $\psi(n_L) = n_L$ in Breiman's work Breiman et al. 1984, but as demonstrated relevant properties when set to $\psi(n_L) = \sqrt{n_L}$ in classification settings (see Appendix D.1 for more details and references). The penalty α is the complexity parameter: the higher it is, the smaller the pruned tree. With a reasonable choice of ψ , the interest of α is that for a fixed complexity parameter value, there exists a unique smallest subtree \mathcal{T} of the maximal tree \mathcal{T}_{max} that minimizes $C_\alpha(\mathcal{T})$. Thus by decreasing α , we can construct a sequence of pruned optimal subtrees $[\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{max}]$ of different sizes. This tree sequence is such that \mathcal{T}_1 is the root node, \mathcal{T}_2 a sub-tree of \mathcal{T} with more leaves and accuracy than \mathcal{T}_1 and so on until \mathcal{T}_{max} , the unpruned maximal tree. With Breiman's notation, we have

$$\mathcal{T}_{max} \supseteq \dots \supseteq \mathcal{T}_2 \supseteq \mathcal{T}_1.$$

The optimal complexity parameter value, hence the best tree in the sequence is usually selected using cross-validation.

14.2.2 Longitudinal notations

This paper aims to enrich the growing process of decision trees in the presence of time-varying covariates. To do so, let us introduce some notations borrowed from the existing longitudinal literature including works of Rizopoulos or Yao et al Rizopoulos 2012; W; Yao et al. 2022. Let us assume a very general setting where we want to build a dataset \mathcal{D}_{long} , encompassing the time-varying features of N subjects, which are repeatedly measured over time. In all generality, let us assume that among the p covariates, p_{TV} of them are time-varying and p_{TI} others are time-invariant. At time t , the set of covariates is given by $\mathbf{x}(t) = (x_1, x_2, \dots, x_{p_{TI}}, x_{p_{TI}+1}(t), \dots, x_p(t))$. In order to simplify the notations, we consider all constant features as a special case of time-varying covariates, with $\mathbf{x}(t) = (x_1(t), x_2(t), \dots, x_p(t))$ with $x_k(t) = x_k, \forall t$ and $\forall k \in [1, \dots, p_{TI}]$. Let $n^{(i)}$ be the number of distinct times $t_j^{(i)}, j = 0, 1, \dots, n^{(i)} - 1$ at which subject i has been observed. At time $t_j^{(i)}$, subject i has a vector of covariates denoted $\mathbf{x}_j^{(i)} = (x_{j,1}^{(i)}, \dots, x_{j,p}^{(i)})$.

Classical longitudinal setting: For a given subject i , covariates are stored in rows, one row per observation window $[t_j^{(i)}, t_{j+1}^{(i)})$. Each row contains the unique $(t_j^{(i)}, t_{j+1}^{(i)}, \mathbf{x}_j^{(i)}, y_j^{(i)})$ elements, with $y_j^{(i)}$ the target variable observed at time $t_j^{(i)}$. They are completed by the subject unique identifier i . Each row represents what we will now call an *observation*. We build \mathcal{D}_{long} as the collection of all observations structured longitudinally :

$$\mathcal{D}_{long} = \left\{ \left(i, \left\{ t_j^{(i)}, t_{j+1}^{(i)}, \mathbf{x}_j^{(i)}, y_j^{(i)} \right\}_{j=0}^{n^{(i)}-1} \right) \right\}_{i=1}^N$$

Or, if displayed in a table:

Table 14.1: A longitudinally structured dataset

ID	Time window Start	Time window End	Covariate 1	...	Covariate p	Target variable
1	$t_0^{(1)}$	$t_1^{(1)}$	$x_{0,1}^{(1)}$...	$x_{0,p}^{(1)}$	$y_0^{(1)}$
1	$t_1^{(1)}$	$t_2^{(1)}$	$x_{1,1}^{(1)}$...	$x_{1,p}^{(1)}$	$y_1^{(1)}$
1	$t_2^{(1)}$	$t_3^{(1)}$	$x_{2,1}^{(1)}$...	$x_{2,p}^{(1)}$	$y_2^{(1)}$
1	$t_3^{(1)}$	$t_4^{(1)}$	$x_{3,1}^{(1)}$...	$x_{3,p}^{(1)}$	$y_3^{(1)}$
2	$t_0^{(2)}$	$t_1^{(2)}$	$x_{0,1}^{(2)}$...	$x_{0,p}^{(2)}$	$y_0^{(2)}$
3	$t_0^{(3)}$	$t_1^{(3)}$	$x_{0,1}^{(3)}$...	$x_{0,p}^{(3)}$	$y_0^{(3)}$
3	$t_1^{(3)}$	$t_2^{(3)}$	$x_{1,1}^{(3)}$...	$x_{1,p}^{(3)}$	$y_1^{(3)}$
3	$t_2^{(3)}$	$t_3^{(3)}$	$x_{2,1}^{(3)}$...	$x_{2,p}^{(3)}$	$y_2^{(3)}$
...

14.2.3 Existing longitudinal tree-based algorithms

The problem when splitting time-varying covariates: Whether they are designed for survival analysis or not, longitudinal tree-based models exist and propose various methods to include time-varying covariates that cannot naturally fit in the tree-growing algorithm described in Algorithm 8. As an illustrative example, let $x_1(t)$ be a numerical time-varying covariate. At each node, a splitting rule of the form “ $x_1(t) \leq d$ ”³ should be able to split subjects into two child nodes. A subject for which this rule is true at all observed times will go in one child node without any ambiguity. On the other hand, the general case where the rule is true for some periods but false for anywhere else is unclear and needs to be addressed.

The “eventually not longitudinal” methods: The most naïve model would be a regular CART, trained on all observations in the longitudinal dataset without taking the correlation between observations of the same subject into account. As stated by Segal Segal 1992, this would simply ignore the capital aspect of dealing with longitudinal data: “*The covariation induced by making several observations of some continuous response on the same unit, as occurs with repeated measures designs, cluster designs, and longitudinal studies, poses data analytic problems. Analysis of such designs that ignore the covariance structure are known to produce incorrect variance estimate.*”. Other naïve attempts consist of summarising the longitudinal trajectories of time-varying covariates with a small number of parameters. For instance, one could think of only keeping the mean value of every trajectory, the median, its final slope, the baseline value, or the most recent one, ignoring all the remaining information. This leads to a loss of precious information. A similar idea is to regress every longitudinal covariate against time and possibly other features, within-subjects to include the parameters of the regression - intercept and slope - as baseline covariates. It can be argued that if the longitudinal covariates are all strongly linearly associated with time, which is rarely the case in practice, this kind of alternative solution can be relevant. Eo and Cho Eo and Cho 2014 proposed a model called mixed-effects longitudinal tree (MELT) able to handle a longitudinal response by fitting a mixed-effect model at each node of the tree. Subjects are then split based on the heterogeneity of their slopes. Kundu Kundu and Harezlak 2019 extended this idea of resuming information contained in the longitudinal covariates by a combination of splits on baseline covariates and implemented it in the R package LONGCART.

³Note that the same reasoning can be applied to categorical time-varying covariates as well.

Other approaches (such as Ritschard and Oris 2005 and more recently Moradian et al. 2021) designed longitudinal trees that use lagged response values as potential predictors, but still do not treat either the outcome or the covariates as inherently dynamic with time. Overall, in these methods, information is lost during the process, and the number of measurements per subject in real datasets can be too small to obtain consistent time-invariant surrogates to the time-varying covariates.

The “CART-extended” methods: Segal and De’Ath Segal 1992; De’Ath 2002 independently proposed the first applications that clearly define an extension to the CART method and directly account for correlation in the response variable. They both suffered limitations as they were designed for a longitudinal response but time-fixed covariates where all the subjects were measured at the same observation times, with the same interval between them. On the one hand, Segal’s regression tree consisted of imputing a covariance structure in the split procedure. This led to many theoretical questions about defining that covariance structure as well as practical ones regarding the complexity of the computations. On the other hand, De’Ath procedure simply modified the CART algorithm by allowing it to consider an entire matrix containing all the observations for one subject as a single observation. Allowing that was done by using the gain of MSE as a splitting criterion, and replacing the 1-dimensional mean in the MSE with a multi-dimensional mean modified with a covariance structure; the prediction given by the tree would then be the multi-dimensional mean of the observation in the terminal nodes. In both cases, those methods can be seen as fitting a model to the longitudinal outcome at every node as part of the splitting criterion. More recent works by Larsen and Speckman Larsen and Speckman 2004 as well as Hsiao et al. Hsiao and Shih 2007 followed and improved the idea of De’Ath, by redefining the node impurity measure with the Mahalanobis distance and estimating the covariance matrix from the whole data set. It is worth mentioning that other articles extended the idea of Segal, to binary responses and classification trees (see Zhang 1998), in a clustering context using deviance as a goodness-of-fit criterion for partitioning (see Abdollell et al. 2002) and then to every type of longitudinal response - not only continuous or binary - using Generalized estimating equations (see the works of Lee Lee 2005; Lee et al. 2005; Lee 2006). Such models show advantages in terms of predictive ability and interpretability but do not handle time-varying covariates.

The “state-of-the-art” methods: In the work of Hajjem et al. Hajjem, Bellavance, and Larocque 2011b, Sela et al. Sela and J.S. Simonoff 2012 and their respective extensions (see Hajjem, Bellavance, and Larocque 2014; Capitaine, Genuer, and Thiébaud 2021⁴ and Fu and J.S. Simonoff 2015), a general mixed-effect model is assumed for the longitudinal outcome. The tree-based part only predicts fixed effects whereas individual estimated parameters account for all the time-varying effects. Such approaches can estimate longitudinal outcomes but the inclusion of time-varying covariates is handled via the pseudo-subject workaround detailed in the next paragraph. It relies on the assumption that all the dependency between several observations of the same subject is captured by the random effect of the mixed model. In a survival setting, Du et al. Fu and J.S. Simonoff 2016 and its extensions (see W. Yao et al. 2020) proposed a model based on those ideas: they allowed subjects to be divided into pseudo-subjects and used an adjusted log-rank test in the splitting procedure to accommodate for left truncation and ensure that the independence implicit assumption does not lead to biased results. We refer the astute reader to the works mentioned

⁴Louis Capitaine also worked on a promising generalization of decision trees and forests that must be acknowledged. We refer to Appendix D.2 for further details.

in this paragraph as we consider they are the most advantageous approaches today. The algorithms corresponding to their respective work are the R packages `REEMtree`, `LongituRF`, `LTRCtrees` and `LTRCforests`, the R function `REEMctree` and the Python library `MERF`.

Pseudo-subjects LTRC and mixed-effect tree-based models (at least their tree-based part) consider the unmodified \mathcal{D}_{long} as an input and run through a CART-like growing process, finding optimal binary decision rules at each node of the tree. Whenever a split produces an ambiguity as described in Section 14.2.3, the periods $[t_j^{(i)}, t_{j+1}^{(i)})$ where their splitting rule “ $x_1(t) \leq d$ ” is true would go to the left node, and the other would go to the right node, thus dividing one subject into several pseudo-subjects. It cleverly addresses the time-handling issue when the bias that comes with correlated right-censored and left-truncated (LTRC) observations is neutralized otherwise. In such models, any individual can be spread in many different tree leaves - even if, at any fixed time, any individual will have a single observation that will fall into a unique one. Treating one subject’s observations, not as an indivisible block of information but rather as multiple pseudo subject’s data leads to a loss of interpretability. In our opinion, none of these procedures can inherently handle time-varying covariates, while maintaining CART’s interpretability. A unique trajectory per subject would ensure a clear visualization of the data: the algorithm should be designed to separate **individuals** whose features are significantly diverging regarding the target variable rather than **pseudo-subjects**.

14.3 Time penalized trees

We present here the building blocks of a new way to think about decision trees in the presence of time-varying covariates: time-penalized trees or TpT. Let \mathcal{D}_{long} be a longitudinal dataset, and $T = [0; \max_{i,j}(t_j^{(i)})]$ be the continuous observation interval of time. We define $\mathcal{D}(t)$ as the dataset containing $\overset{i,j}{i,j}$, for every subject i , her unique observation with the maximal observation time $t_j^{(i)}$ such that $t_j^{(i)} \leq t$ and $t_{n^{(i)}-1}^{(i)} \geq t$, where $\mathbf{x}^{(i)}(t) \in \mathcal{X}(t)$ is the vector of covariates and $y^{(i)}(t) \in \mathcal{Y}(t)$ the target variable at time t . Eventually, $\mathcal{N}(t) = |\mathcal{D}(t)|$ is the total number of subjects under study at time t . Let g be a node, which is also identified with a subregion of $\mathcal{X} \otimes T$ it represents. We thus define $\mathcal{D}(g)$, the set of observations in the node g and $\mathcal{N}(g) = |\mathcal{D}(g)|$ the number of subjects it contains.

The idea behind TpT is to build a tree that benefits from all the longitudinal information available and where the concept of time is central: at each node, we chose to split along covariates and time. As stated in Section 14.2.1, a tree-growing algorithm is defined by its splitting criterion, stopping rule(s), and pruning process. This applies to TpT and the algorithm we propose can be described as in Algorithm 9. In the end, a final Time-penalized Tree would look like the tree depicted in Figure 14.3.

Algorithm 9 Growing a maximal TpT

- 1: **Input:** Training longitudinal dataset \mathcal{D}_{long}
 - 2: **Output:** Maximal TpT $_{max}$
 - 3: Initialize the root node g_p with the entire dataset at time $t = 0$, $\mathcal{D}_{long}(0)$
 - 4: $\text{Grow}(g_p, 0)$
 - 5:
 - 6: **Function** $\text{Grow}(g_p, t_p)$:
 - 7: **if** Stopping criteria met (e.g., maximum depth, minimum samples) **then**
 - 8: Let g_p be a “terminal leaf”.
 - 9: **else**
 - 10: For all possible covariates x_k , thresholds d and time points $t_c \geq t_p$ find the triplet (x_k, d, t_c) such that a partitioning of $\mathcal{D}_{long}(t_c)$ along x_k , at threshold d obtains the best splitting criterion.
 - 11:
 - 12: Split the node g_p : all subjects with $t_{n^{(i)}-1}^{(i)} < t_c$ go to a “duration leaf” g_t . All other subjects - with $t_{n^{(i)}-1}^{(i)} \geq t_c$ - are split along covariate x_k at threshold d into two child nodes g_r and g_l .
 - 13: $\text{Grow}(g_r, t_c)$
 - 14: $\text{Grow}(g_l, t_c)$
 - 15: **end if**
-

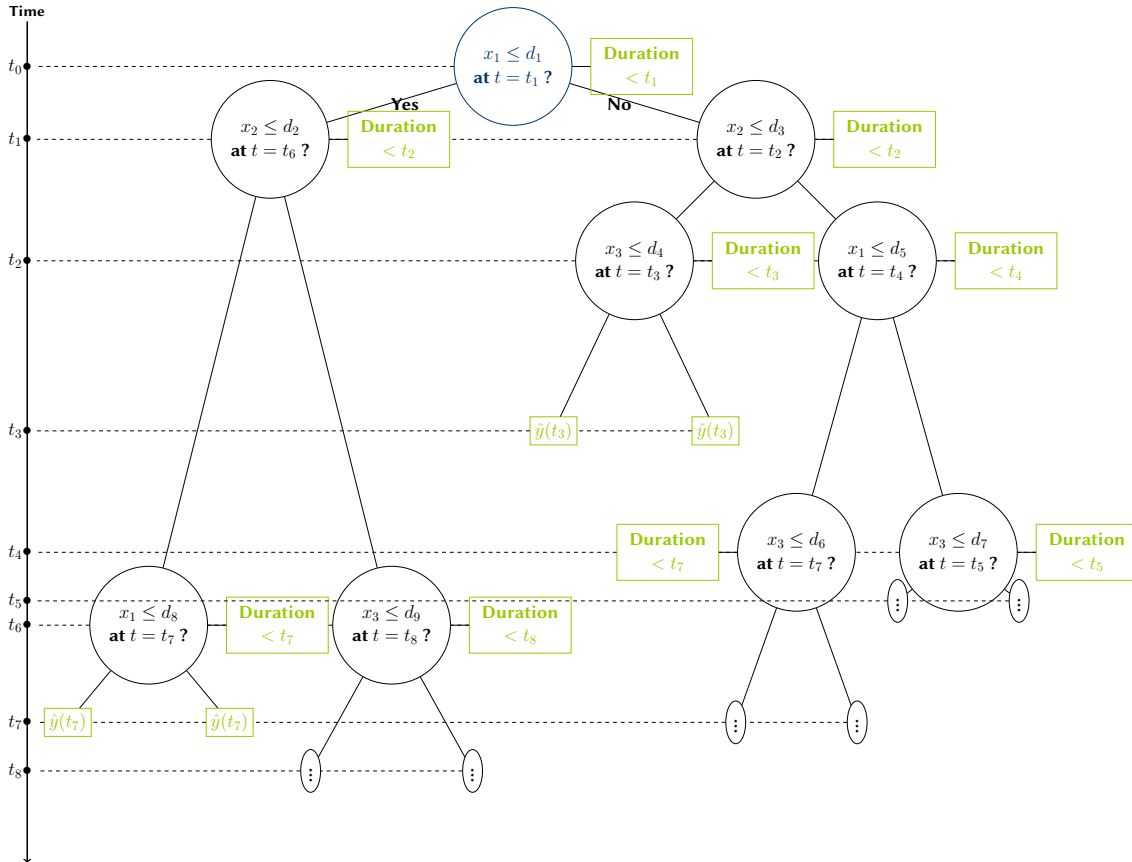


Figure 14.3: Illustration of a TpT

Let us detail how to understand the TpT depicted in Figure 14.3. The root node appears in blue, and leaf nodes appear in green. The root node contains all subjects of $\mathcal{D}(t_0)$ and is then split into three nodes:

- a left child node containing all subjects from the root node for whom the covariate x_1 is inferior or equal to the threshold d_1 , at time t_1 ,
- a right child node containing all subjects from the root node for whom the covariate x_1 is greater than d_1 , at time t_1 ,
- a third node (depicted horizontally from the root node in Figure 14.3) containing all subjects from the root node without any observation at time t_1 or later. Without any information about the value of x_1 at time t_1 , such subjects cannot be spread into one of the child nodes. As this third node cannot be split any further, constitutes a *duration leaves*.

The right and left child nodes, thus, each contain non-overlapping subsets of $\mathcal{D}(t_1)$ and are themselves split further, along optimal covariates, thresholds, and at a times that are $\geq t_1$. This iterative splitting process continues until a stopping criterion is met and the nodes cannot be partitioned any further. The final nodes obtained at the very end of every branch constitute *terminal leaves*.

Remark 14.1

A remark about the time notations of Figure 14.3 needs to be made, in order to avoid any confusion. The time points that figure along the vertical axis on the left of Figure 14.3 can be understood as the times of arrival to the node: the last time that was used to split the subjects in the previous node. Conversely, the time point mentioned inside any given node is part of the decision rule: the optimal time at which the node is split. For instance, all subjects from $\mathcal{D}(t_0)$ arrive to the root node (hence the “ t_0 ” on the left axis) and the root node is then split based on the value of covariate x_1 , at time $t_1 \geq t_0$ (hence the “ t_1 ” inside the root node).

Defining TpT stopping rules is exactly similar to CART (see Section 14.2.1). Its splitting criterion to be optimized at each node, as well as its pruning process, meanwhile, are modified and discussed in the sections below.

14.3.1 TpT splitting criterion

The split function for TpT is rather straightforward. We want to select the split on a covariate, at a threshold and a time that will maximize a time-penalized split criterion. The division of a node into two child nodes and a duration leaf has been detailed for the root node of Figure 14.3, and in all generality, a single split of a parent node g_p into the three nodes g_l (the left child node), g_r (the right child node), and g_t (the duration leaf), is illustrated in Figure 14.4.

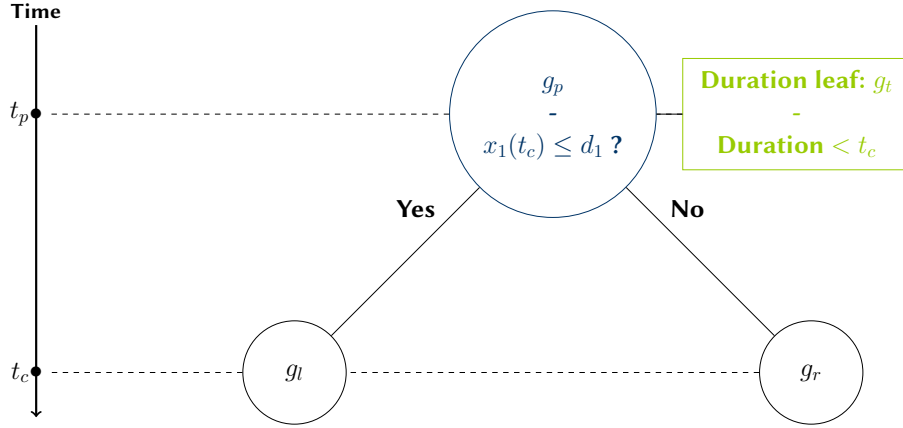


Figure 14.4: Single split of a TpT

To obtain such a split, we have to define a time-penalized split criterion, as

$$G_\gamma(g_p; g_l, g_r, g_t) = \left[I(g_p) - \left(\frac{\mathcal{N}(g_l)}{\mathcal{N}(g_p)} I(g_l) + \frac{\mathcal{N}(g_r)}{\mathcal{N}(g_p)} I(g_r) + \frac{\mathcal{N}(g_t)}{\mathcal{N}(g_p)} I(g_t) \right) \right] \cdot e^{-\gamma \cdot (t_c - t_p)}, \quad (14.5)$$

with $\gamma \in \mathbb{R}^+$, $I(g)$ an impurity or MSE function as described in Section 14.2.1, t_p and t_c the respective times of the parent node and child nodes and γ the penalty parameter. We can immediately see that this is simply the classical CART splitting criterion with an additional exponential penalty term, depending on how distant in time the considered split is. The exponential penalty that we propose induces that the more time distance there is between a parent node and its potential child node, the more penalized the split. Without that penalty term, a TpT would have early splits at advanced times, and much information contained in early observations would be lost. It ensures that early observations are explored and exploited and that distant splits are selected early in the tree if and only if they are greatly informative. In other words, splits are chosen where covariates AND time points are informative about the target variable; we first try to find close splits if they can detect heterogeneity but distant splits will be considered if they are very informative. We can find examples of this type of exponential consideration of time in time series analysis with exponential smoothing (see Brown 1956; Holt 2004), where exponential functions are used to assign exponentially decreasing weights over time. As far as our knowledge extends, instances of tree-based modified splitting criteria where exponential weights were introduced are very rare. A first reference can be found in Section 5.5 of Bremner's PhD thesis Bremner 2004 that used localized splitting criteria that are based on local averaging in regression trees or local proportions in classification trees, weighted by exponential weights. The weights have no link to time or a measure of distance from the previous node. Goldstein Goldstein 2014 also suggested using exponential weights in tree-based algorithms to promote splits on covariates that were already used in previous splits over others.

The partitioning procedure of TpT can also be visualized similarly to Figure 14.1, the only difference is that the iterative splits occur on different versions of the longitudinal dataset. Instead of partitioning the feature space alone, we need to illustrate how TpT partitions the feature space at different times.

Consider two time-varying covariates $x_1(t)$ and $x_2(t)$ et let us assume that $t_0 = 0$: at depth 0 and $t = 0$, the tree is only a root and $\mathcal{D}(0)$ is not partitioned (as it can be seen on the left side of

Figure 14.5). We can see that on the first iteration of the algorithm, a first split, at $t = 0$ creates a division of $\mathcal{D}(t_1)$ ⁵ such as illustrated on the right side of Figure 14.5.

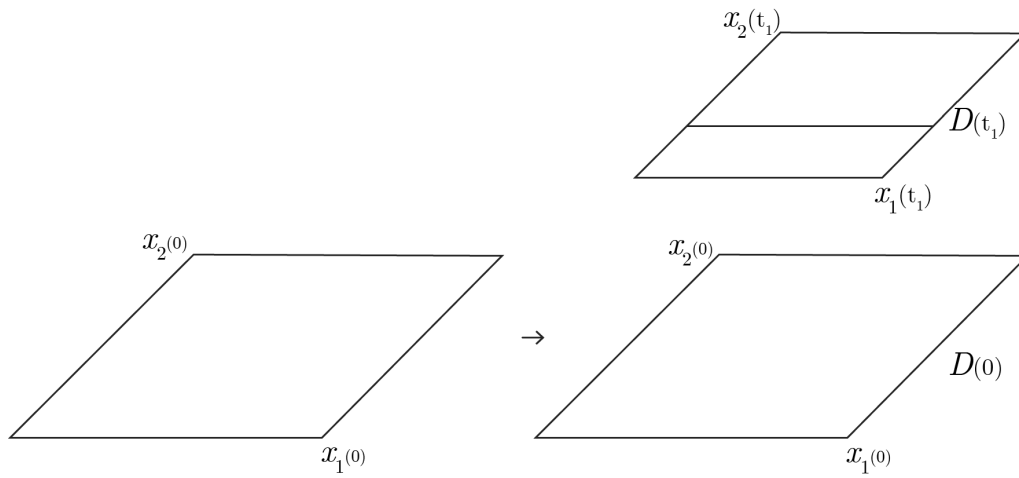


Figure 14.5: TpT 1-depth recursive partitioning

If we go on with the iterative partitioning, at depth 2 and $t = t_1$, all subjects that have been observed up to $t = t_2$ within each partition, are once again split into two subgroups. This creates a division of $\mathcal{D}(t_2)$ such as depicted in Figure 14.6.

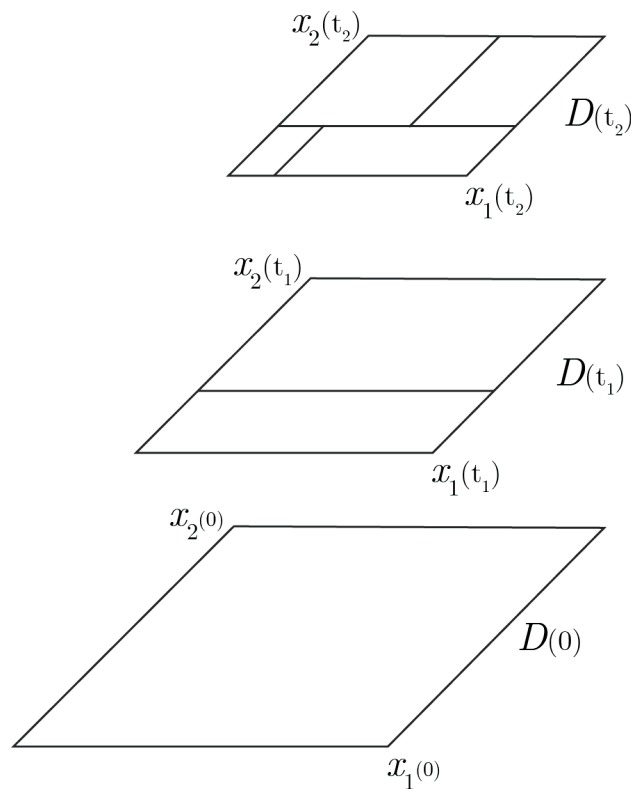


Figure 14.6: TpT 2-depth recursive partitioning

⁵Please refer to Remark 14.1 for more insights on why the split of subjects observed at a given time occurs at an ulterior time

Eventually, a few more steps of the iterative partitioning procedure can be visualized as in Figure 14.7. It is the representation of a classical binary split procedure, with the inclusion of a time dimension. The routes of all subjects can be displayed in that representation: the red, blue and green paths in Figure 14.7 are examples of such individual trajectories.

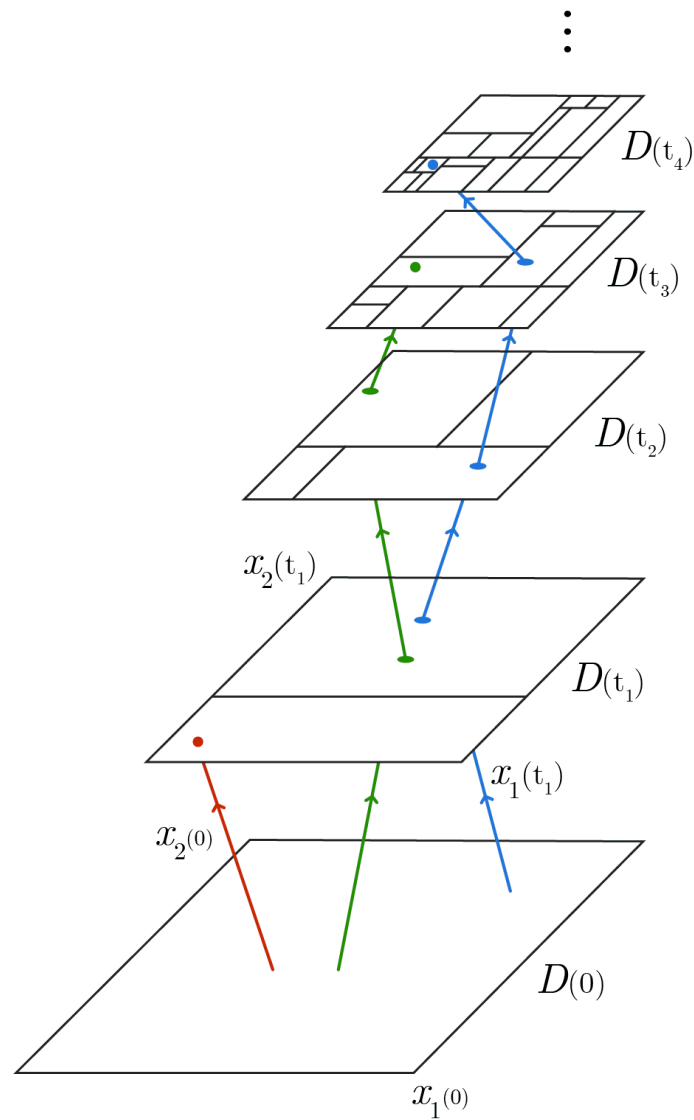


Figure 14.7: TpT recursive partitioning and individual trajectories

Remark 14.2

Several things need to be noted regarding the partitioning illustration depicted in Figure 14.7.

First, duration leaves are not represented here: the red trajectory for instance, does not split after time $t = t_1$ because the subject it represents has not been observed at time t_2 or later. Its course at time t_2 being unknown, it stops in the region of $\mathcal{D}(t_1)$, which is then a duration leaf for similar subjects.

Secondly, this illustration shows that divisions of early steps transpose into continuous partitions in further steps: this is not true in general. The two groups formed by the partition of $\mathcal{D}(t_1)$ may not be represented by a unique region of $\mathcal{D}(t_2)$, split at a constant threshold over one covariate.

To illustrate that point with a concrete example, let us assume that one of the covariates in \mathcal{D} is the subject's salary. The set of all individuals with a salary $\leq 1,000$ at time t_1 is composed of individuals without any observation at time t_2 , of subjects with a salary that increased by t_2 and is $> 1,000$ and of others still earning $\leq 1,000$. A set of subjects within a unique connected region of the feature space at time t_1 generally lies within disconnected subregions of the feature space at time t_2 .

Eventually, it is also to be noted that not every disjoint region splits at every time step of the partitioning. There are times when several splits occur, others where only one region is partitioned, and others where none. All those points are not depicted in Figure 14.7 for simplicity's sake.

We can already foresee that higher values of γ ensure that the next optimal split is more likely to be close in time to the previous node (a distant split is to be chosen only if it is very interesting). The produced TpT will be close to a CART with all longitudinal covariates values blocked at $t = t_0$. And it can be easily proven that

$$TpT(\mathcal{D}_{long}) \xrightarrow{\gamma \rightarrow +\infty} CART(\mathcal{D}_{long}(t_0)). \quad (14.6)$$

It allows a TpT to explore the covariates space but prevents it from exploring the time dimension. On the contrary, lower values of γ are more likely to produce distant splits and the constructed TpT will show similarities with a CART where all longitudinal covariates values rapidly approach their final value. It allows a TpT to split along the time dimension quickly but prevents it from exploring the covariates space at any given time.

Remark 14.3

Because the impurities of the parent and child nodes can be computed at different time points, it can happen that $G_\gamma < 0$. Such cases imply that a specific stopping rule must be enforced for TpT: G_γ must be positive for a node to split. Otherwise, it would allow ineffective splits.

14.3.2 TpT pruning process

For a TpT, the penalty parameter γ affects the tree's dimensions (depth and number of leaves, see Section 14.4 for an analysis on the matter). An optimal γ that minimizes the impurity of the tree (the weighted sum of all leaves impurities) can be chosen but it is not a pruning process comparable to cost-complexity pruning. For a given γ , a maximal TpT can be grown and may overfit the data. To control for bias and overfitting, various pruning strategies can be considered. First, Breiman's cost-complexity pruning is still well-defined under the TpT framework, for a given γ , and can be applied as long as all term nodes - denoted as g_t in previous illustrations and algorithm - are considered as leaves. We suggest a slightly different adaptation of this pruning strategy to select both α and γ simultaneously. It consists of selecting the pair (α, γ) that minimizes $C_\alpha(\mathcal{T})$, the cost of the tree. Simpler pruning strategies such as Reduced Error Pruning (see Quinlan 1987) can also be used. Their advantages and flaws are notably discussed in Esposito's article Esposito et al. 1997 as well as their tendency to over/under-prune.

14.3.3 Time-to-event

Survival analysis can also be carried out directly under the TpT framework. Indeed, for data mining purposes, subjects are distributed in the final tree considering their last observed time $T^{(i)}$. Censorship and event occurrences are visible in the term nodes g_t . Adapting TpT for prediction tasks would require additional work to account for censorship (see Vock et al. 2016), but this specific topic is not in the scope of our paper.

14.4 Applications

Such a longitudinal data mining algorithm can prove useful in various fields (medicine, sports analytics, taxonomy, biology), here we applied it to a life insurance customer segmentation analysis. For that purpose, we use a real-world dataset of 983 policyholders (PHs), a subset of the dataset used in Chapters 8 and 12, and we investigate the link between the PH's characteristics through time and the final outcome of their policies, a categorical response variable. Throughout the lifetime of such insurance policies, a series of events can occur. Firstly, one policyholder's coverage can be increased with premium payments that are highly flexible, both in terms of amount and frequency, and are adjusted according to the policyholder's financial circumstances and preferences. Additionally, policyholders may decide to reduce their coverage by withdrawing a portion of their policy. We refer to these events as partial lapses, enabling PHs to adjust their coverage to better align with their changing needs. Other financial operations can occur, such as the payment of interest or profit sharing from the insurer to the PH, and the payment of fees from the PH to the insurer. Insurance companies' information systems are usually designed to keep track of those operations at the policy level, thus actuaries and life insurers often have access to the complete history of their policyholders as the information system is updated in real-time. Eventually, one's insurance plan ends whenever the PH dies or decides to terminate it by lapsing. In the end, the timeline of such insurance policies can be illustrated in Figure 14.8⁶, below.

⁶Illustration taken from Valla 2023b

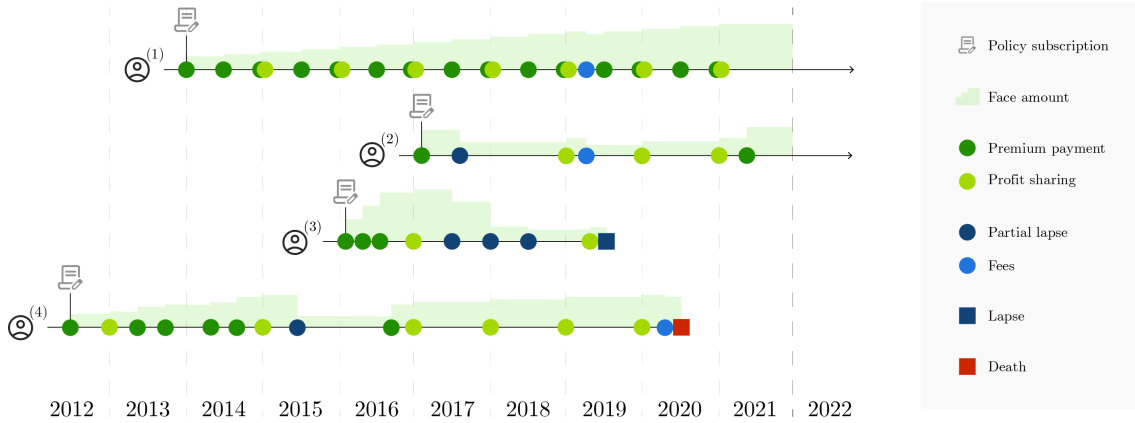


Figure 14.8: Example of policyholders timelines

At any given time a policy is either active, has been lapsed by the policyholder, or has ended because of her death. Among all insurance plans subscribed between 1998 and 2019, in our dataset, 57.4% are active, 22.8% ended with the death of the policyholders, and 19.8% lapsed. We only consider uncensored observations here, we thus have 46% of churned policies and 54% that ended with death. For this application, our data mining goal is to gain insights into the PH’s pathways that lead to these different outcomes. We want to find time-dependent clusters of individuals with similar timelines and outcomes at a given time. This is thus a time-dependent classification problem, where the target variable is the final outcome of the policies, the tree grows with the survival time and splits on potentially time-varying covariates such as age, rate, Customer Lifetime Value (CLV), face amount (FA) or gender. More detailed descriptions of the dataset used can be found in Valla et al. and in Valla’s works Valla, Milhaud, and Olympio 2023; Valla 2023b. In all visualizations of the following sections, all leaves or regions that contain a majority of policies that ended with the PH’s death are labeled “D”, and all those that contain a majority of policies that ended with lapse (or churn) are labeled “C”. In terms of colors, the proportion of each class is represented by a nuance between (for a 100% proportion of “D”) and (for a 100% proportion of “C”). For example, a leaf or region with 50% of churners is represented by the color . Since we only consider PHs that were observed until the termination of their policy, there are no censored observations to consider.

14.4.1 Properties of TpT for the maximal tree

First of all, Table D.1 in Appendix D.3 displays the results obtained by TpT with various choices for the time penalty parameter γ . It shows the dimensions of TpTs (depth and number of leaves), their global impurities and costs, the highest time point when a split occurred, and the average time at which any subject is split. Graphs of those results can be found in Figure D.1. Here we considered unpruned trees using the time-penalized version of the Gini impurity measure as a splitting rule (Equation 14.5) and without any stopping criterion. For this application, we computed the cost of the tree with a choice of $\alpha = \sqrt{\frac{3 \log 2}{2N}}$, suggested by Scott Scott and Nowak 2006. The pruning process then only consist of selecting γ as the solution of $\arg \min_{\gamma} C_{\alpha}(\mathcal{T})$.

We can observe that the depth and number of leaves grow with γ . This was to be expected, as a TpT that does not penalize time-distant splits will quickly find high impurity-gain splits at distant times thus preventing the exploration of less distant time periods. Conversely, the same phenomenon explains that the average time when splits occur is a decreasing function of γ . As the penalty parameters get high values, any future split is heavily penalized and can not com-

pete with splits at time t_0 , regardless of their potential unpenalized gain. Eventually and very interestingly, we observe that the unpenalized TpT, as well as the heavily penalized one, are not optimal in terms of global impurity. There exists an optimal choice of γ that generates a TpT minimizing the sum of its leaves impurities. This tree has a penalty parameter of 0.2725, a depth of 17 and a number of leaves of 190 - 173 terminal leaves and 17 duration leaves - and is displayed in Appendix D.3 as long with more results and graphs obtained with diverse settings, with various impurity measures.

Such trees, without stopping criterion and post-pruning are useful to discuss the properties of TpTs but do not yield immediate insights on our dataset. Nevertheless, there is one statistic that proves to be insightful: the distribution of times when splits occur. Obviously, with an exponentially penalized splitting criterion, the more distant from its parent time t_p a split time t_c is, the more penalized it is and the less likely it is to be selected. The a priori probability for time to be selected as a split time is $\propto e^{-\gamma \cdot (t_p - t_c)}$. Thus, by weighting the frequency of times when splits occur with an exponential factor, we balance this bias and retrieve the importance of the time periods. In the optimal unpruned and unstopped tree, the splitting time points are distributed as such:

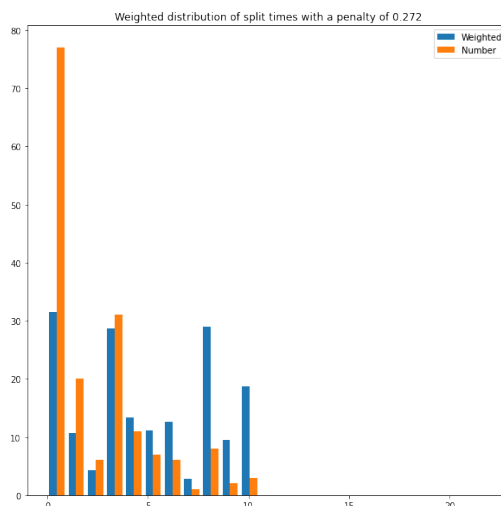


Figure 14.9: Split times distribution for the optimal unstopped and unpruned TpT

In the weighted histogram, we can clearly see that some periods seem to be critical split points that differentiate active policies from lapsed or policies that are likely to end with the policyholder's death. Interestingly, we see that $t = 0$ and $t = 8$ are particularly important in terms of differentiation between policies' outcomes. For $t = 0$, the insight is clear: most of the information that separates the churners from policies that end with the PH's death can be retrieved from the baseline covariates: for instance, it can be seen in the early splits of Figure 14.10 that the age at subscription seems to be very informative - older PHs are more exposed to the mortality risk - and thus is selected at baseline. Regarding the important splits at $t = 8$, we see in Figure 14.10 that they correspond to splits on age, CLV, or FA. CLV is highly dependent on both age and FA, thus we could argue that age and FA are the most informative covariates at $t = 8$. By law, French life insurance plans ensure that when a given policy is at least eight years old, the policyholder can lapse without any surrender penalty. This is a clear incentive not to churn before one's policy reaches 8 years of seniority. It seems consistent to observe that this threshold is pointed out in our analysis. The third year of seniority comes right after $t = 0$ and $t = 8$ in terms of temporal importance, which does not have any obvious business justification. However, every split at $t = 3$ (see Figure 14.10) is either a split on CLV or the FA of the policy, thus we can

argue that the final outcome of a policy seems dependent on its FA 3 years after subscription. Similarly, the unpenalized suboptimal TpT with $\gamma = 0$, depicted in Figure 14.12 only splits at times $t = 0$, $t = 3$ or $t \geq 8$, with respectively 1, 1 and 24 splits.

This application has also been tried on a larger longitudinal dataset, containing 119,431 observations of 9,873 PHs. Characteristics of TpTs grown with various γ are described in Figure 14.10. It gives the split times distribution in Figure 14.11. Due to the heavy computation time, all other analyses are carried out on the smaller dataset.

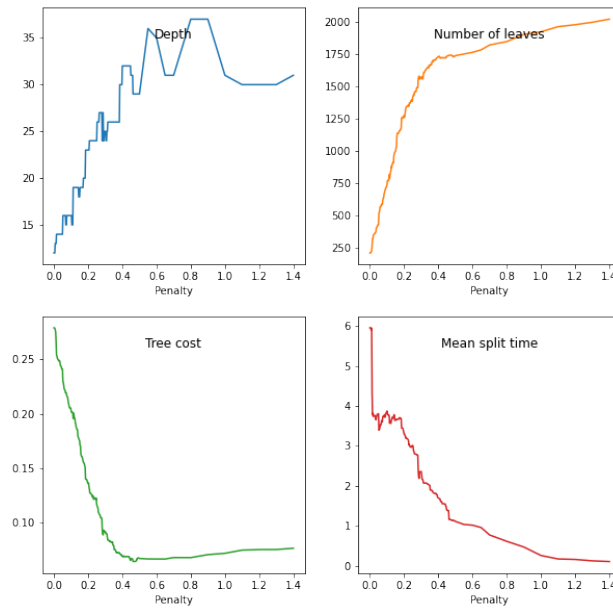


Figure 14.10: Characteristics of unstopped and unpruned TpT, trained on 9,873 PHs with various time penalties

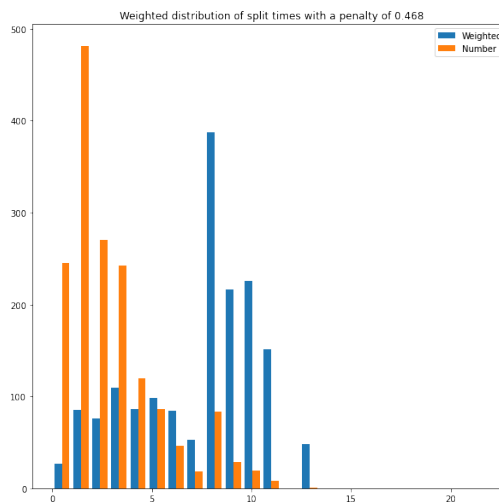


Figure 14.11: Split times distribution for the optimal unstopped and unpruned TpT, trained on 9,873 PHs

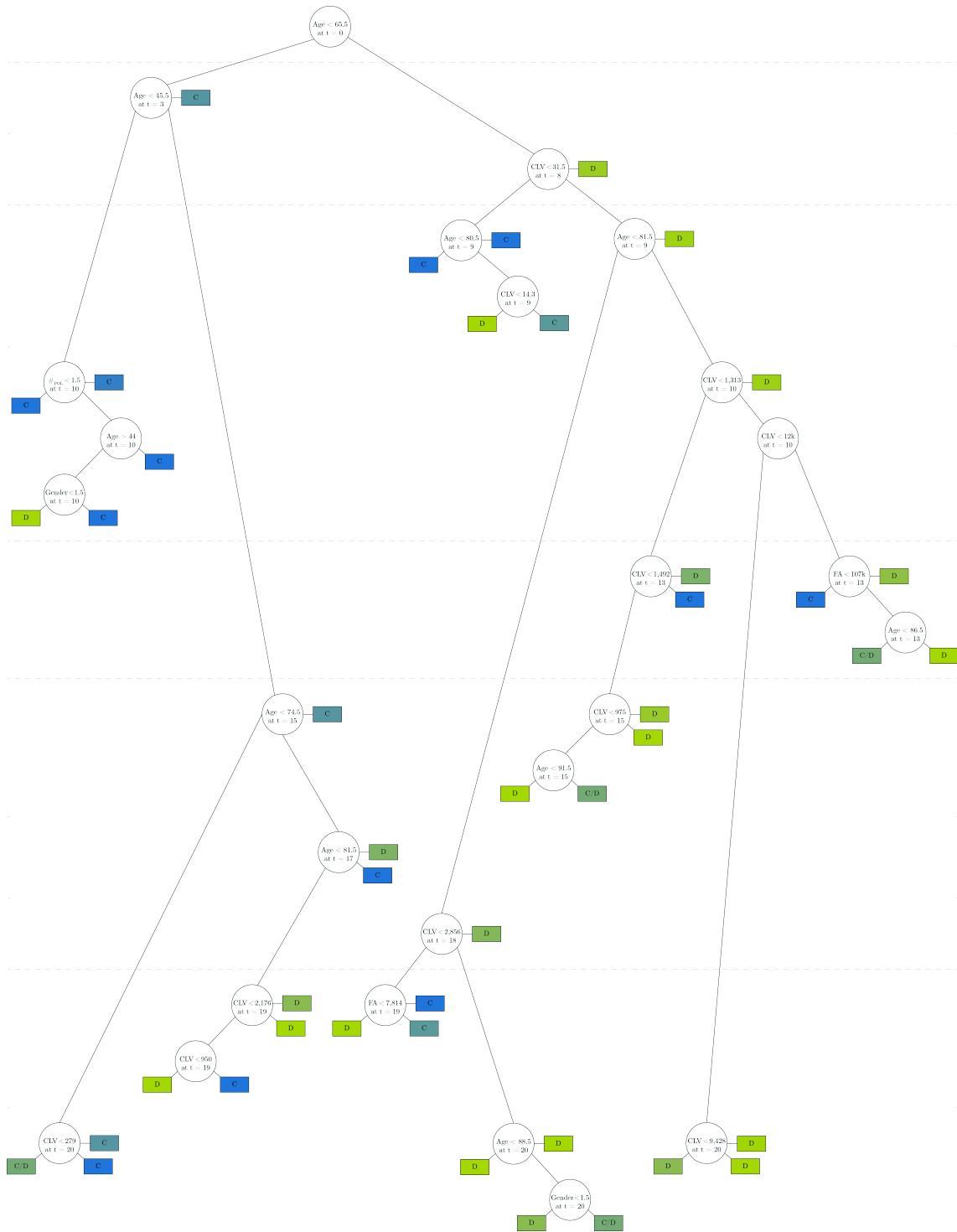


Figure 14.12: Overfitted unpenalized, unstopped and unpruned TpT

14.4.2 Use-case with minsplit

A clear strength of decision trees is their interpretability. Obviously, trees with hundreds of leaves each containing a handful of subjects can not be interpreted. Here we decided to investigate the results obtained by TpTs with various γ , using the time-penalized version of the Gini impurity measure as a splitting rule and including a stopping criterion: any leaf must contain at least 50 individuals otherwise it does not split. This choice of stopping rule is not really close to the

default value for the `minsplit` parameter in most CART implementations, but it will generate shorter, less overfitted TpTs, better suited for direct interpretability and data analysis. Here are the results for TpT on our longitudinal dataset, with `minsplit= 50`. Graphs of those results can be found in Figure 14.13.

Time penalty γ	Runtime	Depth	# of terminal leaves	# of duration leaves	Total # of leaves	Tree cost	Max of split times	Mean of split times
0.0000	587.03	4	9	7	16	0.319	15.0	4.309
0.0025	736.31	6	11	6	17	0.306	15.0	2.134
0.0075	730.35	6	11	5	16	0.306	15.0	2.102
0.0200	726.84	6	11	4	15	0.306	15.0	1.97
0.0275	789.2	6	13	6	19	0.300	8.0	2.203
0.0325	784.67	6	13	5	18	0.300	8.0	2.131
0.0350	817.64	6	14	4	18	0.296	9.0	1.762
0.0650	840.38	7	15	3	18	0.297	9.0	1.129
0.0925	841.11	7	15	2	17	0.299	8.0	0.88
0.1100	850.24	7	15	1	16	0.300	8.0	0.564
0.1250	873.15	7	16	1	17	0.298	5.0	0.423
0.1375	873.64	6	15	2	17	0.297	5.0	0.303
0.1900	899.41	6	15	1	16	0.297	3.0	0.157
0.2050	886.07	6	15	1	16	0.298	3.0	0.125
0.7000	752.84	6	15	0	15	0.299	1.0	0.032
0.8000	743.89	6	15	0	15	0.300	0.0	0.0

Table 14.2: Characteristics of TpT (`minsplit: 50`) depending on the time penalty

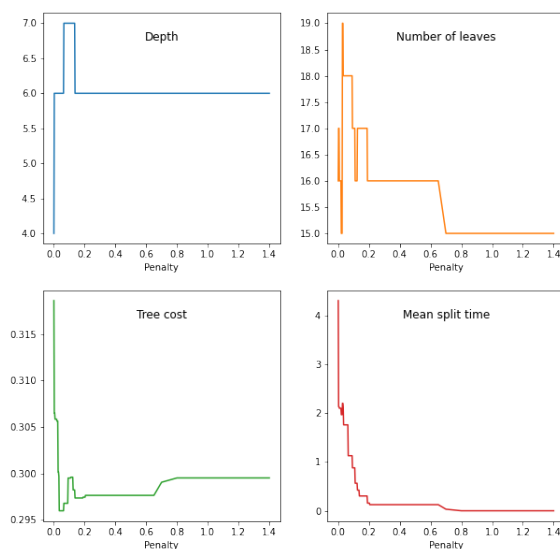


Figure 14.13: Characteristics of TpT (`minsplit: 50`) depending on the time penalty

Among all the different TpTs in table 14.2, we can discuss which one minimizes the tree cost. First of all, we see here that the trees with $\gamma = 0$ and $\gamma \rightarrow \infty$ are not the best in terms of global cost. This is a critical result: $\gamma = 0$ is the case where the last observed observation points are quickly considered whereas early periods are not really considered, and high γ represents the case where a tree is grown only on the baseline values of all time-varying covariates. Thus, TpT shows that considering time in the splitting process improves the global purity of the tree, it better differentiates between individuals with different outcomes and trajectories. In terms of interpretability, Figure 14.14 shows that the optimal TpT is a compromise between small TpTs with time-distant splits and a large baseline tree without any temporal information. Whole-page versions of those trees can be found in Appendix D.3.3.

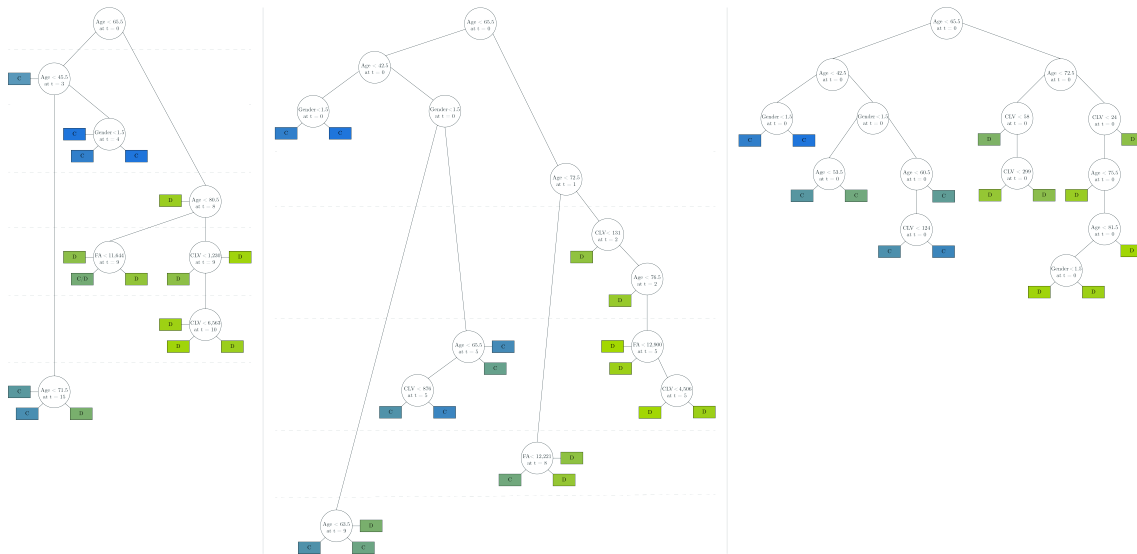


Figure 14.14: TpTs with $\gamma = 0$, $\gamma = 0.035$ and $\gamma \rightarrow \infty$, respectively

An important temporal dependence that can be learnt from the tree is the fact that there exists an incentive not to lapse before eight years of seniority. It is clearly depicted in the optimal TpT - $\gamma = 0.035$ - as the duration leaves generated by splits occurring at times ≥ 8 contain a majority of policyholders that did not lapse. It means that regardless of their age, subjects with a seniority ≤ 8 years do not lapse. The TpT with no time penalty - $\gamma = 0$ - can capture the same temporal dependence for splits that occur immediately after 8 years for older PH but fails to do so for younger ones. This is explained by the fact that for the latter, the unpenalized TpT quickly finds an excellent split at time $t = 15$, which prevents splits around 8 years from being found. This is a compelling argument in favor of a time penalty. Furthermore, the TpT with a very high time penalty produces a tree that only splits at time $t = 0$, thus no temporal insights can be found with it. If we were to conclude from such a tree, we could say that Age is the most important covariate, and allow for a good partitioning of \mathcal{D} but we cannot have any temporal analysis. This is an argument in favor of TpT and the suggested γ selection process.

14.4.3 Pathways vizualizations

In terms of data mining and clustering, let us focus on the optimal TpT obtained in the previous section and depicted in Figure 14.15. In the same way a decision tree is a representation of all observations in a cross-sectional dataset, a TpT is a complete representation of a longitudinal one and we can highlight the pathway of any given policyholder in the tree. Unlike any other longitudinal tree-based model, any individual has a unique continuous trajectory in the tree. The pathways of five policyholders selected at random from our dataset are represented in the following TpT.

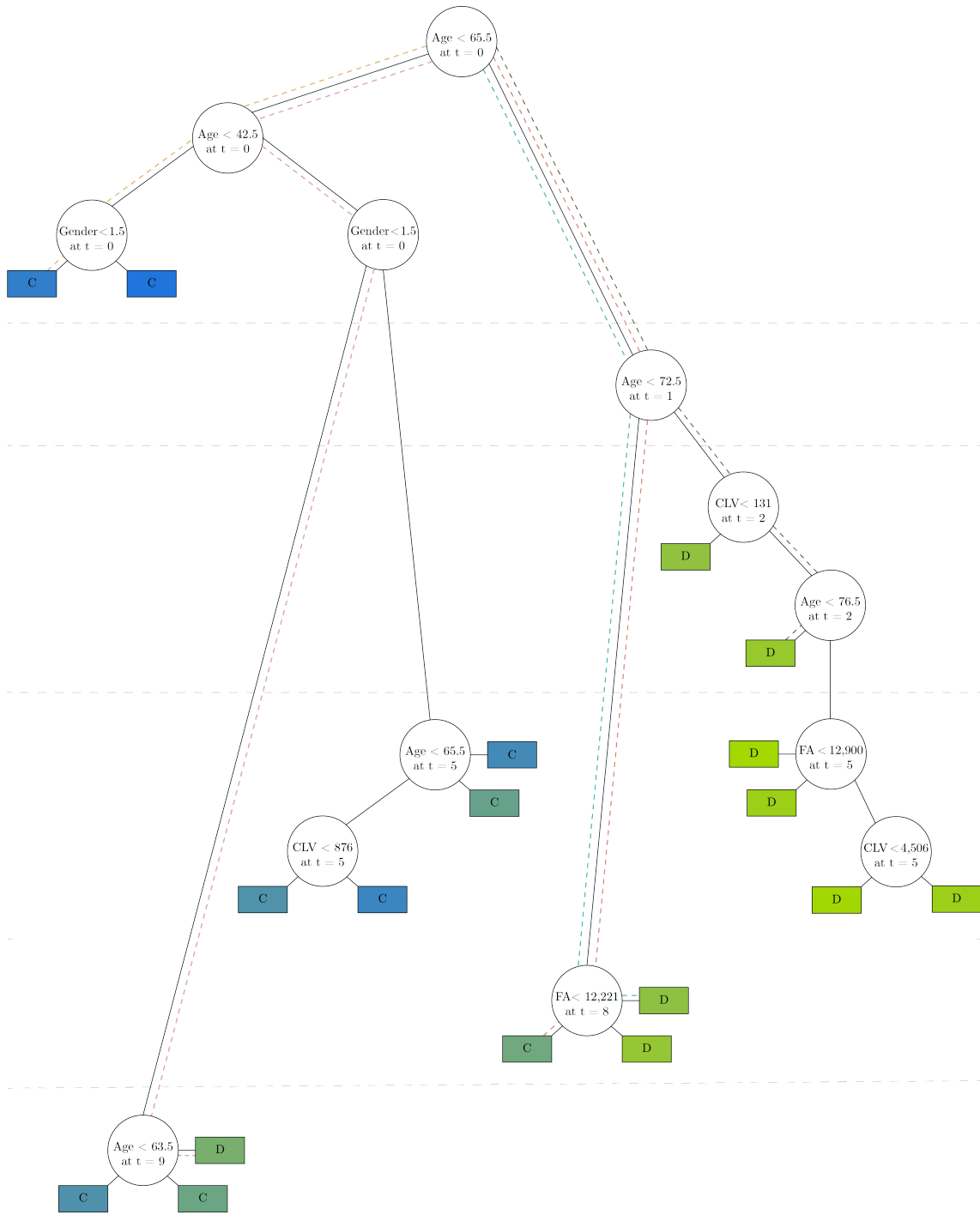


Figure 14.15: Individual longitudinal trajectories in the optimal TpT (minsplit = 50)

Thus, the longitudinal dataset and all individual timelines can be easily represented as a partitioning. All PH are represented on the y-axis and the region of the covariate space where they belong changes as a function of time, on the x-axis. In this example, the 18 leaves of Figure 14.15 correspond to the 18 final regions of Figure 14.16, as $t \rightarrow \infty$.

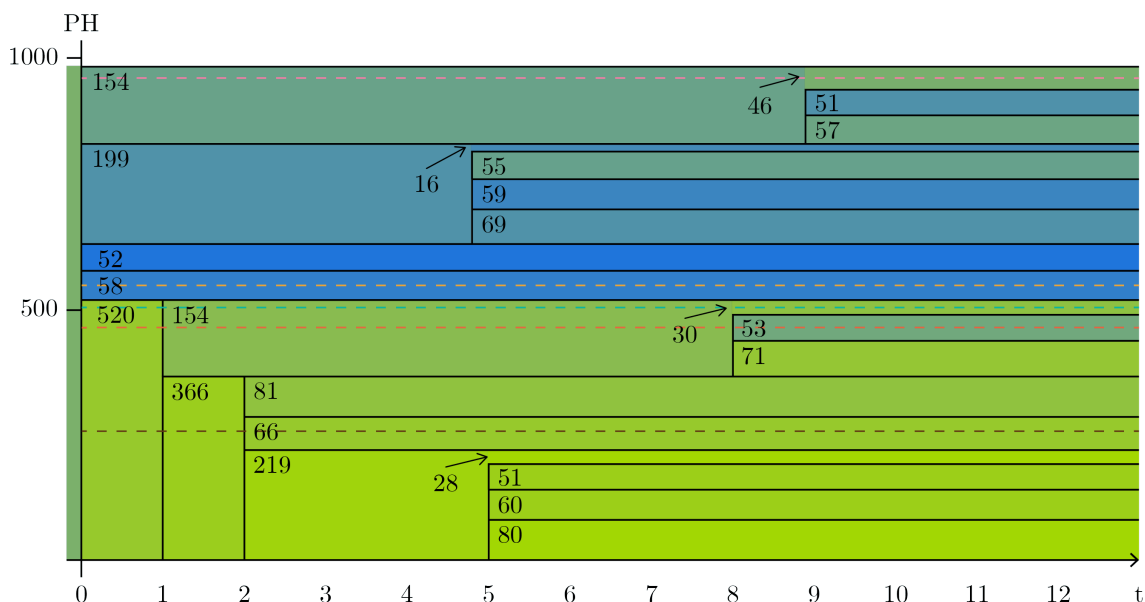


Figure 14.16: Global timeline and individual longitudinal trajectories

The numbers in each region of Figure 14.16, as well as their heights, are the number of PH they contain, and the five individual trajectories represented as pathways in the tree correspond to the five horizontal lines within the global timeline. Let us take a few examples to understand this Figure. In the TpT displayed in Figure 14.15, there are 520 policyholders that are older than 62.5 years old, at subscription, they all go from the root to the first right node of the tree. At that point, we see the trajectories of three policyholders (depicted in light blue, red and brown dashed lines) taking that path to the right. Their trajectories spread after the next split at $t = 1$. Similarly, those 520 PHs can be found in the lower region of Figure 14.16, with $t \in [0, 1]$ on the y-axis. The light blue, red, and brown dashed paths can be found in that region. After that, when $t > 1$, these PHs' trajectories are never in the same region again, the same way they can never be found in the same node of the corresponding TpT. This type of visualization allows to better interpret the periods where changes in the outcome can occur.

14.5 Conclusion, limitations and future work

14.5.1 Conclusion

This paper exhibits TpT, a new tree-based data-mining algorithm that accounts for time-varying covariates through time-penalized splitting criteria. Our methods handle time-varying covariates as well as longitudinal target variables inherently. Contrary to existing approaches, it does not need workaround strategies such as the pseudo-subject method and provides a tree that separates “complete individuals”, as each subject covariates trajectory corresponds to a single unique trajectory in the final tree. Pruning strategies were proposed and tested with real datasets and illustrative examples. The algorithm proves to have appealing data-mining and visualization potential in various fields that could be explored more deeply in the future.

14.5.2 Limitations and future work

Right away, it is crucial to acknowledge the general limitations of this work, before going into more technical details. First of all, the need for a thorough investigation into the statistical properties and theoretical underpinnings of the developed tools is evident to ensure their reliability and robustness. Such theoretical work is critical and constitutes forthcoming research. Secondly, conducting comprehensive comparisons of TpT with existing longitudinal techniques, employing well-studied datasets and consistent indicators is pivotal for a more rigorous evaluation of the proposed methods. These identified gaps in the current work underscore the necessity for subsequent research on TpT.

Besides those points, and with the algorithm as it is defined for data-mining purposes, many improvement paths can be considered:

Firstly, the introduction of a penalized splitting criterion, and thus a penalty parameter could be discussed more thoroughly. The current multiplicative exponential form of penalization has been dually justified but one could explore the effects of different approaches. Other distributions of the future time cut-off penalties such as Gamma (with parameters $\alpha < 1, \beta \geq 1$ or $\alpha = 1, \beta > 2$), Pareto, Weibull (with parameter $k < 1$) or Log-logistic ($\beta \leq 1$) could be justified on concrete examples. Moreover, in the algorithm as it stands, every point in time can be considered for a potential cut-off; some time-horizon limit where distant splits would simply be ignored would have an impact on the shape of the final tree. Eventually, the possibility of a penalty parameter that changes along the growth of the tree is yet to be explored. In all those scenarios, the penalty parameter affects the width and length of the final tree and can even be interpreted as a pre-pruning parameter. The properties of that pre-pruning as well as the choice of an optimal γ are yet to be discussed. On a final note, we do not know if any technical properties (see Breiman 1996; Buntine and Niblett 1992) of the penalized splitting criterion still hold. That knowledge will certainly not affect the concrete applications of TpT but is more of a theoretical interest.

Secondly, we showed in illustrative applications that time-outliers can be easily miscategorized as the TpT can send them early in one direction of the tree from which they will not escape. In addition to that, those observations are likely to end up being isolated in a leaf if the stopping rules allow it. On the one hand, it forces observation into an early path that may not be consistent with later observations. On the other hand, this behavior is linked to an abrupt change of the covariates and target variable trajectories in time, which is a discriminating feature that can justify that such subjects end up in a specific leaf. We see two ways to handle this specific property:

- A first idea would be to modify the TpT algorithm to make it less greedy. Instead of choosing the best split at each node, we could consider finding the best sequence of several consecutive splits. This *multi-step ahead* strategy would ensure that abrupt changes in covariates in the future are anticipated in early splits. In cases where the penalty parameter is low, this approach also ensures that the TpT does not grow too fast with time.
- Another innovative solution is to introduce the possibility for an outlier in a node to teleport into another one, at a similar depth/split time in the tree. For instance, if it so happens that a subject trajectory suddenly becomes significantly different from the other ones in the same node, it can be clever to acknowledge that it is no longer consistent to keep it in the said node. This solution has drawbacks: it requires testing for outliers in every node, at every step. If one is found, it can only be teleported if another node within which the subject would not be an outlier is found. Moreover, it is a straightforward solution for

data mining but other adaptations are necessary if TpT is to be used for predictive tasks. Despite this, it would still ensure individual trajectories for every subject in the tree and it would consolidate the global within-node homogeneity.

Then, our last point raises another capital question: the applications shown in Section 14.4 only exhibit the potential of TpT for clustering tasks; can it be adapted to prediction ones? In our context, a prediction is an estimation of a subject's target variable $y^{(i)}$ at a time t , given its covariate history up to t . An obvious research path in that direction is to mimic the example of CART. For a subject in node g , predicting the mean of the target variable at time t of all subjects emerging from node g is to be tried. If the target variable is a time-to-event and the observations are censored, it could be weighted by the inverse probabilities of censoring weights (IPCW). It perfectly translates in terms of interpretability: the prediction of an outcome at time t for individual i is the mean of the outcomes at time t of all subjects taking the same trajectory in the tree. There are several obvious drawbacks to this approach: there needs to exist observations of other subjects at time t . And even if some exist, the variance of the prediction is directly linked to the number of such subjects. Exploring the properties and predictive performance of this approach is left as future work and other methods such as fitting a longitudinal model⁷ at every node, not for splitting but for prediction purposes are also studied.

Eventually, if prediction is made possible in the future, exploring the performance of ensemble methods for TpT looks like a reasonable next step. Such approaches are in contradiction with the research of interpretability that motivated TpT, but competitive predictive performance could justify them.

Within the specific context of this thesis, we can add some broader thoughts. The elaboration of the original article on which this part is based naturally arose from the gaps detected in the longitudinal tree-based models' literature for data mining and the potential use of TpT for actuarial applications. Nevertheless, similar research problems also emerged from different origins, and answer different research problems. For instance, sequence analysis in social science (see Liao et al. 2022) or time-sequence/clinical pathway/treatment sequence analysis in the medical field whether its purpose is data visualization (see Prodel et al. 2020) or data-mining (see Augusto et al. 2016; De Oliveira et al. 2020; Chouaid et al. 2022) also deal with individual trajectories of subjects with time-varying features. Such topics are closely related to our work and demonstrate a broader interest in analyzing time-varying behaviors in any decision-making field. The recent work of W; Yao et al. 2022 (specifically appendices A and B) suggests a survival method that yields a decision tree that splits along covariates and time points. Such a tree and its similarity with a TpT can be observed in Figure 14.17. The derivation of the survival function estimate is described in Appendix B. Appendix A displays a medical application that analyses the COVID survival probabilities of various groups and how such a tree allows to update their survival functions dynamically. This could be adapted to TpT for survival prediction purposes and future actuarial research could benefit from such works.

Then, even if references for the exponential penalization of time-distant splits have been suggested in Section 14.3.1, links to other tree-like structures with an exponential design have been made through discussion with other researchers on that topic. Specifically, links between TpT and Yule Processes (see Athreya and Ney 2004, chapter 3, section 11) or Mondrian trees (see Lakshminarayanan 2016, chapter 5) could be explored in the future.

⁷A mixed effect model in regression or a joint model in survival setting for instance

Eventually, the consistency results depicted in Section 4.2.3 for TBMs trained on cross-sectional datasets could be extended for TpT. As a matter of fact, the splitting criterion at each node is evaluated at a given time point, thus on independent observations and a single deep TpT can be studied as a concatenation of numerous time-invariant shallow CART. Thus, given a sufficiently large training set, we obtain sufficiently deep concatenated CARTs, all consistent for a fixed time. Conditions for the consistency of TpT, in the sense that the trajectory of any individual along the tree tends to a minimal classification error at each time step, could be derived. The mathematical work that consists of extending the consistency results of cross-sectional TBMs to TpT constitutes future research.

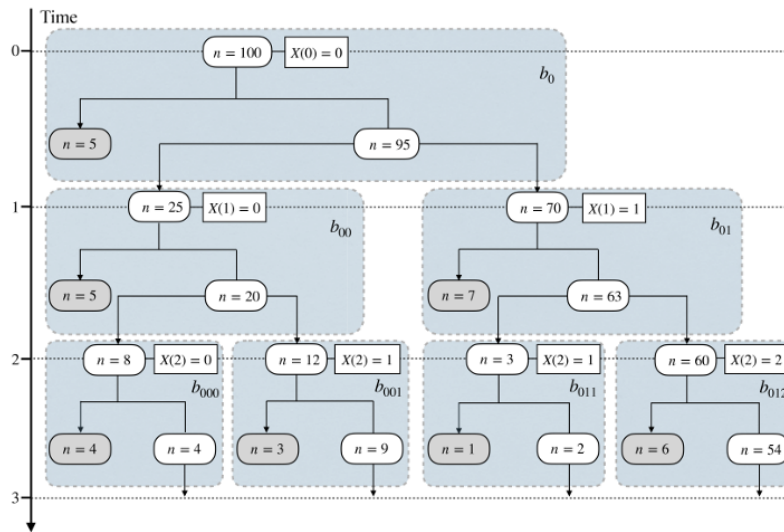


Figure 14.17: Yao et al. survival decision tree, taken from W; Yao et al. 2022

Bibliography

- Valla, M. (Aug. 2023a). “Time-penalized trees (TpT): a new tree-based data mining algorithm for time-varying covariates”. working paper or preprint. URL : <https://hal.science/hal-04178282> (HAL), <https://www.researchsquare.com/article/rs-3400744/v1> (research square).
- Breiman, L. et al. (1984). *Classification and Regression Trees*. Taylor & Francis. ISBN: 9780412048418. URL: <https://books.google.fr/books?id=JwQx-WOmSyQC>.
- Mena, G. et al. (2023). “Exploiting time-varying RFM measures for customer churn prediction with deep neural networks”. In: *Annals of Operations Research*. URL: <https://hal.science/hal-04027550>.
- Wong, S.Y.K. et al. (2022). “Time-varying neural network for stock return prediction”. In: *Intelligent Systems in Accounting, Finance and Management* 29.1, pp. 3–18. DOI: <https://doi.org/10.1002/isaf.1507>.
- Fu, W. and J.S. Simonoff (Dec. 2016). “Survival trees for left-truncated and right-censored data, with application to time-varying covariate data”. In: *Biostatistics* 18.2, pp. 352–369. ISSN: 1465-4644. DOI: [10.1093/biostatistics/kxw047](https://doi.org/10.1093/biostatistics/kxw047). eprint: <https://academic.oup.com/biostatistics/article-pdf/18/2/352/11057459/kxw047.pdf>.
- Hajjem, A., F. Bellavance, and D. Larocque (2011a). “Mixed effects regression trees for clustered data”. In: *Statistics & Probability Letters* 81.4, pp. 451–459. URL: <https://ideas.repec.org/a/eee/stapro/v81y2011i4p451-459.html>.
- Sela, R. and J.S. Simonoff (Feb. 2012). “RE-EM trees: A data mining approach for longitudinal and clustered data”. In: *Machine Learning* 86, pp. 169–207. DOI: [10.1007/s10994-011-5258-3](https://doi.org/10.1007/s10994-011-5258-3).
- Hajjem, A., F. Bellavance, and D. Larocque (2014). “Mixed-effects random forest for clustered data”. In: *Journal of Statistical Computation and Simulation* 84.6, pp. 1313–1328. DOI: [10.1080/00949655.2012.741599](https://doi.org/10.1080/00949655.2012.741599).
- Mingers, J. (1989). “An empirical comparison of selection measures for decision-tree induction”. In: *Machine learning* 3.4, pp. 319–342.
- Buntine, W. and T. Niblett (1992). “A further comparison of splitting rules for decision-tree induction”. In: *Machine Learning* 8.1, pp. 75–85.
- Breiman, L. (1996). “Technical note: Some properties of splitting criteria”. In: *Machine Learning* 24.1, pp. 41–47.
- Shih, Y.S. (1999). “Families of splitting criteria for classification trees”. In: *Statistics and Computing* 9.4, pp. 309–315.
- Drummond, C. and R.C. Holte (2000). “Exploiting the cost (in) sensitivity of decision tree splitting criteria”. In: *ICML*, pp. 239–246.
- Rizopoulos, D. (2012). *Joint Models for Longitudinal and Time-to-Event Data, with Applications in R*. Boca Raton: Chapman & Hall/CRC.

- Yao, W; et al. (2022). “Ensemble methods for survival function estimation with time-varying covariates”. In: *Statistical Methods in Medical Research* 31.11. PMID: 35895510, pp. 2217–2236. DOI: [10.1177/09622802221111549](https://doi.org/10.1177/09622802221111549). URL: <https://export.arxiv.org/pdf/2006.00567>.
- Segal, M.R. (1992). “Tree-structured models for longitudinal data”. en. In: *Journal of the American Statistical Association* 87, pp. 407–418.
- Eo, S.H. and H.J. Cho (2014). “Tree-Structured Mixed-Effects Regression Modeling for Longitudinal Data”. In: *Journal of Computational and Graphical Statistics* 23.3, pp. 740–760. DOI: [10.1080/10618600.2013.794732](https://doi.org/10.1080/10618600.2013.794732).
- Kundu, M.G. and J. Harezlak (2019). “Regression trees for longitudinal data with baseline covariates”. In: *Biostatistics & Epidemiology* 3.1, pp. 1–22. DOI: [10.1080/24709360.2018.1557797](https://doi.org/10.1080/24709360.2018.1557797).
- Ritschard, G. and M. Oris (2005). “Life course data in demography and social sciences: statistical and data mining approaches”. en. In: *Towards an interdisciplinary perspective on the life course, advances in life course research*. Ed. by R. Levy et al. Amsterdam: Elsevier, pp. 289–320.
- Moradian, H. et al. (2021). “Dynamic estimation with random forests for discrete-time survival data”. In: *Canadian Journal of Statistics*.
- De’Ath, G. (2002). “Multivariate regression trees: a new technique for modeling species-environment relationships”. en. In: *Ecology* 83, pp. 1105–1117.
- Larsen, D.R. and P.L. Speckman (2004). “Multivariate regression trees for analysis of abundance data”. en. In: *Biometrics* 60, pp. 543–549.
- Hsiao, W.C. and Y.S. Shih (2007). “Splitting variable selection for multivariate regression trees”. en. In: *Statistics and Probability Letters* 77, pp. 265–271.
- Zhang, H. (1998). “Classification trees for multiple binary responses”. en. In: *Journal of the American Statistical Association* 93, pp. 180–193.
- Abdolell, M. et al. (2002). “Binary partitioning for continuous longitudinal data: categorizing a prognostic variable”. it. In: *Statistics in Medicine* 21, pp. 3395–3409.
- Lee, S.K. (2005). “On generalized multivariate decision tree by using GEE”. en. In: *Computational Statistics & Data Analysis* 49, pp. 1105–1119.
- Lee, S.K. et al. (2005). “Using generalized estimating equations to learn decision trees with multivariate responses”. en. In: *Data Mining and Knowledge Discovery* 11, pp. 273–293.
- Lee, S.K. (2006). “On classification and regression trees for multiple responses and its application”. en. In: *Journal of Classification* 23, pp. 123–141.
- Hajjem, A., F. Bellavance, and D. Larocque (2011b). “Mixed effects regression trees for clustered data”. In: *Statistics & Probability Letters* 81.4, pp. 451–459. ISSN: 0167-7152. DOI: <https://doi.org/10.1016/j.spl.2010.12.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0167715210003433>.
- Capitaine, L., R. Genuer, and R. Thiébaud (2021). “Random forests for high-dimensional longitudinal data”. In: *Statistical Methods in Medical Research* 30.1. PMID: 32772626, pp. 166–184. DOI: [10.1177/0962280220946080](https://doi.org/10.1177/0962280220946080).
- Fu, W. and J.S. Simonoff (2015). “Unbiased regression trees for longitudinal and clustered data”. In: *Computational Statistics & Data Analysis* 88, pp. 53–74. ISSN: 0167-9473. DOI: <https://doi.org/10.1016/j.csda.2015.02.004>. URL: <https://www.sciencedirect.com/science/article/pii/S0167947315000432>.
- Yao, W. et al. (2020). *Ensemble methods for survival function estimation with time-varying covariates*. DOI: [10.48550/ARXIV.2006.00567](https://doi.org/10.48550/ARXIV.2006.00567). URL: <https://arxiv.org/abs/2006.00567>.
- Brown, R.G. (1956). *Exponential Smoothing for Predicting Demand*. Little.

- Holt, C.C. (2004). “Forecasting seasonals and trends by exponentially weighted moving averages”. In: *International Journal of Forecasting* 20.1, pp. 5–10. URL: <https://EconPapers.repec.org/RePEc:eee:intfor:v:20:y:2004:i:1:p:5-10>.
- Bremner, A.P. (2004). “Localised splitting criteria for classification and regression trees”. eng. PhD thesis. Murdoch University.
- Goldstein, A.L. (2014). “Topics in Tree-Based Methods”. Doctoral dissertation. University of Pennsylvania. URL: <https://repository.upenn.edu/dissertations/AAI3622048/>.
- Quinlan, J.R. (1987). “Simplifying decision trees”. In: *Int. J. Man Mach. Stud.* 27, pp. 221–234.
- Esposito, F. et al. (1997). “A comparative analysis of methods for pruning decision trees”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.5, pp. 476–491. DOI: [10.1109/34.589207](https://doi.org/10.1109/34.589207).
- Vock, D.M. et al. (June 2016). “Adapting machine learning techniques to censored time-to-event health record data: A general-purpose approach using inverse probability of censoring weighting”. In: *J. Biomed. Inform.* 61, pp. 119–131.
- Valla, M. (July 2023b). “A longitudinal framework for lapse management in life insurance”. working paper or preprint. URL: <https://hal.science/hal-04178278>.
- Valla, M., X. Milhau, and A.A. Olympio (Sept. 2023). “Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies”. In: *European Actuarial Journal*. DOI: [10.1007/s13385-023-00358-0](https://doi.org/10.1007/s13385-023-00358-0). URL: <https://hal.science/hal-03903047> (HAL), <https://export.arxiv.org/pdf/2307.06651> (arxiv), https://link.springer.com/article/10.1007/s13385-023-00358-0?code=84d3a0d0-b866-48d5-bc60-5ed6832d144a&error=cookies_not_supported (journal).
- Scott, C. and R.D. Nowak (2006). “Minimax-optimal classification with dyadic decision trees”. In: *IEEE Transactions on Information Theory* 52.4, pp. 1335–1353. DOI: [10.1109/TIT.2006.871056](https://doi.org/10.1109/TIT.2006.871056).
- Liao, T.F. et al. (2022). “Sequence analysis: Its past, present, and future”. In: *Social Science Research* 107, p. 102772. ISSN: 0049-089X. DOI: <https://doi.org/10.1016/j.ssresearch.2022.102772>. URL: <https://www.sciencedirect.com/science/article/pii/S0049089X22000783>.
- Prodel, M. et al. (Dec. 2020). “PCN273 Meta-TAK, a Scalable Double-Clustering Method for Treatment Sequences Visualization: Case Study in Breast Cancer Using Claim DATA”. In: *Value in Health* 23, S471. ISSN: 1098-3015. DOI: [10.1016/j.jval.2020.08.410](https://doi.org/10.1016/j.jval.2020.08.410).
- Augusto, V. et al. (Dec. 2016). “Evaluation of discovered clinical pathways using process mining and joint agent-based discrete-event simulation”. In: pp. 2135–2146. DOI: [10.1109/WSC.2016.7822256](https://doi.org/10.1109/WSC.2016.7822256).
- De Oliveira, H. et al. (2020). “Optimal process mining of timed event logs”. In: *Information Sciences* 528, pp. 58–78. ISSN: 0020-0255. DOI: <https://doi.org/10.1016/j.ins.2020.04.020>. URL: <https://www.sciencedirect.com/science/article/pii/S0020025520303200>.
- Chouaid, C. et al. (2022). “Machine Learning-Based Analysis of Treatment Sequences Typology in Advanced Non-Small-Cell Lung Cancer Long-Term Survivors Treated With Nivolumab”. In: *JCO Clinical Cancer Informatics* 6. PMID: 35113656, e2100108. DOI: [10.1200/CCI.21.00108](https://doi.org/10.1200/CCI.21.00108).
- Athreya, K.B. and PE Ney (2004). *Branching processes*. Dover Publications.
- Lakshminarayanan, B. (2016). “Decision Trees and Forests: A Probabilistic Perspective”. PhD thesis. University College London.

Part VI

Epilogue

Chapter 15
Final words

Bibliography

15. Final words

The purpose of our conclusion is not to detail once more the contributions of this thesis. Indeed, these have already been discussed in the abstracts, the introductions and conclusions of each part. We propose in this general conclusion a very brief reminder of the different themes addressed in each part of the thesis, and the way in which they are articulated to tell a coherent story. We then propose some general research perspectives that align with the work presented and have not been specifically explained in the conclusions of each part.

This thesis focuses on new frameworks, methodologies, and algorithms to analyze lapse behavior in life insurance. It aims to include temporal dynamics of the data in policyholder-centered modeling approaches. Parts I and II are introductory parts of the thesis arguing why and how the insurance sector could benefit from customer-centered methods using ML tools that allow a time-dependent analysis. Recent advancements in ML have significantly enhanced the efficiency of predictive models, leading to novel uses across various sectors. Nonetheless, the insurance industry struggles to work extensively with these innovative tools. On the one hand, it is due to its fundamental role in safeguarding society from economic losses: any decision-making process in insurance potentially affects significantly policyholders' lives. On the other hand, it is also a consequence of the stringent regulations, implemented to ensure the equitable treatment of all individuals. These aspects underscore the necessity of explainability and interpretability in insurance-related decisions. Moreover, these points stress the need for customer-centric strategies, firstly to meet the needs and expectations of PHs, then to better identify individual behaviors and their consequences on the insurer's profitability. Part III introduces a new framework and gives tools for managing lapses in a life insurance portfolio. The temporal dimension is seized here with the use of survival models and the analysis of time-to-event outcomes. Part IV further improves on this lapse management strategy framework by allowing the insurer to use the complete historical data of the policyholders to increase the precision of its predictions, thus allowing for a deeper inclusion of time in the framework. Eventually, Part V suggests an innovative way to adapt decision tree algorithms to longitudinal data, providing new ways to visualize and represent the data.

The research perspectives on the general study of temporal dynamics and the use of longitudinal data in insurance are significant. The use of ML approaches and longitudinal data is rising, yet still at its commencement. Such techniques can be more difficult to implement, necessitate more computation power, and must convince insurers that they lead to better decisions, in terms of prediction accuracy, interpretability and visualization. This is what we want to show in this thesis: there could be potential to leverage these emerging modeling techniques while still complying with the distinct needs of the insurance industry. We hope that our research as well as future works that will follow will participate in bridging the gap between academic discoveries and practical applications in the insurance sector.

The research perspectives on the modeling approaches for longitudinal analysis are numerous. First of all, a major parametric tool for longitudinal analysis has been completely ignored in this thesis: joint models, a powerful technique allowing for the joint analysis of a survival outcome and time-varying covariates. It can handle several time-varying features and competing risks (van Niekerk, Bakka, and Rue 2021). The book of Rizopoulos 2012 is a great introduction to this topic and it has already been considered for actuarial purposes and churn analysis (see Ascarza and Hardie 2013). A machine learning adaptation of joint modeling could be considered, by growing a mob tree (see Section 4.2.1) that fits a joint model at each child node, one can obtain a decision tree that separates individuals based on their differences both in terms of survival profile and longitudinal features and yield predictions for the survival probabilities and the future values of the longitudinal covariates. This approach will constitute future work.

Secondly, we decided to tackle our research problems with tree-based ML techniques, as argued in Section 1.3. Despite the opposition to the use of DL for tabular analysis, one might think of using NN in the framework we proposed. We do not see it as a priority as it would further hinder the interpretability of our methodologies and require greater computation capabilities without the guarantee of improved results for insurance applications. That being said, several studies have shown the potential of NN for insurance applications (see Wuthrich 2019), churn prediction in other sectors (see Tsai and Lu 2009; Zoric 2016), when adding non-tabular data such as text analysis (see De Caigny et al. 2020), or in combination with tree-based models (see Li, Xia, and Zhang 2023 or Holvoet, Antonio, and Henckaerts 2023). We have reasons to believe that neural networks indeed serve as excellent alternatives for processing unstructured text or image data, but such analyses are perspectives beyond the scope of this thesis. Henckaerts 2021 and Holvoet, Antonio, and Henckaerts 2023 suggested another outlook of NNs: they could be used as an *adjustment model* on top of a baseline, more interpretable, model. It is a promising approach for pricing, but could also be used for lapse analysis as it can increase predictive performance without conceding too much explainability from the baseline model.

In drawing this expedition through the labyrinth of statistics and probabilities to a close, I can only be truly confident about one thing: the potential for enhancing and refining this research remains immense, either by extending it or exhibiting its flaws, inaccuracies, and uncertainties. While aware of the modest and seemingly insignificant nature of this contribution to the vast ocean of scientific knowledge, I would like to end this work as it had begun: by invoking the profound wisdom and conclusive words of a great figure of science.

“ The worthwhile problems are the ones you can really solve or help solve, the ones you can really contribute something to. [...] No problem is too small or too trivial if we can really do something about it. ”

Richard P. Feynman

Bibliography

- van Niekerk, J., H. Bakka, and H. Rue (2021). “Competing risks joint models using R-INLA”. In: *Statistical Modelling* 21.1-2, pp. 56–71. DOI: [10.1177/1471082X20913654](https://doi.org/10.1177/1471082X20913654).
- Rizopoulos, D. (2012). *Joint Models for Longitudinal and Time-to-Event Data, with Applications in R*. en. Boca Raton: Chapman & Hall/CRC.
- Ascarza, E. and B.G.S. Hardie (2013). “A Joint Model of Usage and Churn in Contractual Settings”. In: *Marketing Science* 32.4, pp. 570–590. DOI: [10.1287/mksc.2013.0786](https://doi.org/10.1287/mksc.2013.0786).
- Wuthrich, M.V. (2019). “From Generalized Linear Models to Neural Networks, and Back”. In: This paper has been integrated into SSRN Manuscript 3822407. DOI: [10.2139/ssrn.3491790](https://doi.org/10.2139/ssrn.3491790). URL: <https://ssrn.com/abstract=3491790>.
- Tsai, C.F. and Y.H. Lu (2009). “Customer churn prediction by hybrid neural networks”. In: *Expert Systems with Applications* 36.10, pp. 12547–12553. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2009.05.032>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417409004758>.
- Zoric, A.B. (2016). “Predicting Customer Churn in Banking Industry using Neural Networks”. In: *Interdisciplinary Description of Complex Systems* 14.2, pp. 116–124. DOI: [10.7906/indecs.14.2.1](https://doi.org/10.7906/indecs.14.2.1).
- De Caigny, A. et al. (2020). “Incorporating textual information in customer churn prediction models based on a convolutional neural network”. In: *International Journal of Forecasting* 36.4, pp. 1563–1578. ISSN: 0169-2070. DOI: <https://doi.org/10.1016/j.ijforecast.2019.03.029>. URL: <https://www.sciencedirect.com/science/article/pii/S0169207019301499>.
- Li, S., G. Xia, and X. Zhang (2023). “Customer Churn Combination Prediction Model Based on Convolutional Neural Network and Gradient Boosting Decision Tree”. In: *Proceedings of the 2022 5th International Conference on Algorithms, Computing and Artificial Intelligence*. ACAI '22. Sanya, China: Association for Computing Machinery. ISBN: 9781450398336. DOI: [10.1145/3579654.3579666](https://doi.org/10.1145/3579654.3579666).
- Holvoet, F., K. Antonio, and R. Henckaerts (Oct. 2023). *Neural networks for insurance pricing with frequency and severity data: a benchmark study from data preprocessing to technical tariff*. Tech. rep. arXiv.org. URL: <https://ideas.repec.org/p/arx/papers/2310.12671.html>.
- Henckaerts, R. (2021). “Insurance pricing in the era of machine learning and telematics technology.” PhD thesis. KU Leuven.

Part VII
Appendices

A. General Appendix

A.0.1 Permutation test

A permutation test, also referred to as a re-randomization or shuffle test, stands as an exact statistical method that operates on the basis of disproving a hypothesis. It's employed when examining two or more samples with, a priori, distributions F and G to test if they originate from the same distribution. The null hypothesis, denoted as

$$H_0 : F = G,$$

assumes no distinction between the sample distributions. Under this premise, the test statistic's distribution is determined by calculating all potential values resulting from rearrangements of the observed data, making permutation tests a form of resampling technique.

Permutation tests are interesting as they are non-parametric and only require the exchangeability assumption: given any permutation function $\pi(\cdot)$, it is assumed that

$$(x^1, x^2, \dots, x^N) \stackrel{d}{=} (x^{\pi(1)}, x^{\pi(2)}, \dots, x^{\pi(N)}).$$

The notion of exchangeability plays a crucial role, ensuring that the labels or treatments can be interchanged without affecting the test outcomes. Consequently, permutation tests yield precise significance levels under this condition, enabling the derivation of confidence intervals.

Essentially, these tests generate surrogate data by permuting the original observations, reflecting the initial treatment allocation in experimental designs. Originating from the works of Fisher 1935 and Pitman 1937; Pitman 1938, permutation tests deviate from traditional statistical tests as they do not rely on theoretical probability distributions. They offer an approach that generates the distribution of a chosen test statistic under the assumption that no distinction exists between groups based on the measured variable, providing an alternative to parametric tests by deriving p-values from sample-specific permutation distributions rather than theoretical assumptions.

A.0.2 Pearson's chi-square test

Pearson's chi-square test is a statistical method used to either determine the fit of a distribution or if there is a significant dependence between categorical variables. Most notations of this section are directly borrowed from the Pearson's chi-squared test Wikipédia page 2023.

When used to test for the fit of a given distribution, the value of the test statistic is given by

$$\chi^2 = \sum_{i=1}^n \frac{(O_i - E_i)^2}{E_i} = N \sum_{i=1}^n \frac{(O_i/N - p_i)^2}{p_i},$$

or

$$\chi^2 = \sum_{i=1}^n \frac{O_i^2}{E_i} - N,$$

where:

- χ^2 = Pearson's cumulative test statistic, which asymptotically approaches a χ^2 distribution,
- O_i = the number of observations of type i ,
- $E_i = Np_i$ = the expected (theoretical) count of type i , asserted by the null hypothesis that the fraction of type i in the population is p_i ,
- n = the number of cells in the table.

The chi-squared statistic can then be used to calculate a p-value by comparing the value of the statistic to a chi-squared distribution. The number of degrees of freedom is equal to the number of cells n , minus the reduction in degrees of freedom, p .

When used as a test for statistical independence (also known as a test of homogeneity), it assesses whether the observed frequencies of categorical data differ significantly from the expected frequencies under a null hypothesis of no association between the variables. The test involves comparing observed frequencies in a contingency table (which cross-tabulates the categorical variables) with the frequencies that would be expected if the variables were independent. The value of the chi-square statistic involves calculating the sum of the squared differences between observed and expected frequencies, divided by the expected frequencies. The contingency table contains r rows and c columns, and the expected frequency for a cell of the table, given the hypothesis of independence, is

$$E_{i,j} = Np_{i \cdot} p_{\cdot j},$$

where

$$p_{i \cdot} = \frac{O_{i \cdot}}{N} = \sum_{j=1}^c \frac{O_{i,j}}{N},$$

is the proportion of type i observation, ignoring the column attribute (fraction of row totals), and

$$p_{\cdot j} = \frac{O_{\cdot j}}{N} = \sum_{i=1}^r \frac{O_{i,j}}{N}$$

is the proportion of type j observation ignoring the row attribute. The value of the test statistic is then given by

$$\begin{aligned} \chi^2 &= \sum_{i=1}^r \sum_{j=1}^c \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}} \\ &= N \sum_{i,j} p_{i \cdot} p_{\cdot j} \left(\frac{(O_{i,j}/N) - p_{i \cdot} p_{\cdot j}}{p_{i \cdot} p_{\cdot j}} \right)^2. \end{aligned}$$

The null hypothesis for the chi-square test asserts that there is no relationship between the categorical variables being studied, and the alternative hypothesis corresponds to the variables having an association or relationship where the structure of this relationship is not specified.

A.0.3 Bonferroni correction

The Bonferroni correction is a statistical method for correcting the significance threshold in multiple comparisons. The American mathematician and statistician Olive Jean Dunn worked on confidence intervals in biostatistics and originally developed the Bonferroni correction as a

solution to the problem of multiple comparisons. The method is somewhat wrongfully named after the Italian mathematician Carlo Emilio Bonferroni, although he was not its original author.

The correction procedure works as follows. Let H_1, \dots, H_m be a family of m hypotheses and p_1, \dots, p_m their corresponding p-values. Let m be the total number of null hypotheses, and let m_0 be the number of true null hypotheses (a priori unknown). The family-wise error rate (FWER) is the probability of rejecting at least one true H_i , that is, of making at least one type I error. The Bonferroni correction rejects the null hypothesis for each $p_i \leq \frac{\alpha}{m}$, thereby controlling the FWER at $\leq \alpha$. Proof of this control follows from Boole's inequality, as follows:

$$\text{FWER} = P \left\{ \bigcup_{i=1}^{m_0} \left(p_i \leq \frac{\alpha}{m} \right) \right\} \leq \sum_{i=1}^{m_0} \left\{ P \left(p_i \leq \frac{\alpha}{m} \right) \right\} = m_0 \frac{\alpha}{m} \leq \alpha.$$

This control does not require any assumptions about dependence among the p-values or about how many of the null hypotheses are true. It is the simplest correction method, although it is conservative as it carries a substantial risk of type II error. Indeed, this method does not take into account some information, such as the distribution of p-values for the different comparisons and has been later extended for those purposes.

A.0.4 Generalized M-Fluctuation Tests

As the mechanisms behind generalized M-Fluctuation tests are too complex to be summarized in an Appendix subsection, and as their full explanation does not lie within the scope of this thesis we will not detail them here. However, it is largely complex enough to deserve some more references in this appendix.

The astute reader can then find all the details about MOB's inference framework in Zeileis, Hothorn, and Hornik 2008 but other works such as Merkle and Zeileis 2013 use a more standard - and accessible - notation as they focus on the maximum likelihood special case. As stated in Zeileis, Hothorn, and Hornik 2008: "many other test statistics known from the statistics and econometrics literature are contained as special cases in the rich class of generalized M-fluctuation tests. Specifically, the residuals-based tests of Kuan and Hornik 1995, the ML score-based framework of Koning and Hjort 2002 as well as other tests based on F statistics (Andrews 1993; Andrews and Ploberger 1994) are contained. A unifying view is given in Zeileis 2005."

A.0.5 Lifetime function estimate by Kaplan-Meier

All survival notations being introduced, our focus now turns to the estimation process for $S(t)$. The first method to estimate $S(t)$ is to model $\lambda(t)$ by specifying its distribution, a Weibull distribution for instance (see Wang and Hu 2016). Such a model is called "parametric". On the other hand, nonparametric estimations exist and consist of deriving an empirical estimate of $S(t)$, from the observations. The most common nonparametric method that is used throughout this thesis is the Kaplan-Meier (KM) estimator.

When the data is uncensored, a natural empirical estimator of $S(t)$ is $\hat{S}(t)$, the proportion of individuals with survival times greater than t , or mathematically:

$$\hat{S}(t) = P(T > t) = \frac{1}{N} \sum_{i=1}^N I(T^{(i)} > t) \tag{A.1}$$

Another approach would be to access $S(t)$ through an estimate of $\lambda(t)$. Indeed, the instantaneous hazard rate can be estimated empirically by the proportion of individuals that experience the event exactly at time t among the population at risk at that same time:

$$\lambda(t) = \frac{\sum_{i=1}^N I(T^{(i)} = t)}{\sum_{i=1}^N I(T^{(i)} \geq t)} \quad (\text{A.2})$$

Thus, an estimate of $S(t)$ is given by:

$$\hat{S}(t) = \prod_{k=1}^{t-1} \left[1 - \lambda(k) \right] = \prod_{k=1}^{t-1} \left[1 - \frac{\sum_{i=1}^N I(T^{(i)} = k)}{\sum_{i=1}^N I(T^{(i)} \geq k)} \right] \quad (\text{A.3})$$

However, in the presence of censorship, such estimates must be adjusted. A generalisation of Equation A.3 for censored data has been obtained by Kaplan and Meier 1958 and is given by:

$$\hat{S}(t) = P(T > t) = \prod_{i:T^{(i)} \leq t} \left[1 - \frac{d^{(i)}}{n^{(i)}} \right] \quad (\text{A.4})$$

where $d^{(i)}$ is the number of events at time $T^{(i)}$, and $n^{(i)}$ is the number of individuals at risk at time $T^{(i)}$.

Furthermore, the use of the Greenwood variance estimate (see Greenwood 1926) and the assumption of asymptotic normality for $\hat{S}(t)$ allows to derive a confidence interval for $\hat{S}(t)$, given by:

$$\left(\hat{S}(t) \pm z_{1-\alpha/2} \cdot \hat{\sigma} / \sqrt{(n)} \right), \quad (\text{A.5})$$

Or, to ensure that the interval is bounded between 0 and 1, it can be transformed as:

$$\left(\hat{S}(t) \pm e^{z_{1-\alpha/2} \cdot \frac{\hat{\sigma}}{\hat{S}(t) \ln \hat{S}(t)}} \right). \quad (\text{A.6})$$

The KM estimator is widely used in survival analysis and proves to be useful for simple survival probability estimations. It is nevertheless limited by the fact that it cannot take covariates in consideration while parametric survival models or the Cox proportional hazards model (see Section A.0.7) can.

A.0.6 Log-rank test (Mantel–Cox)

The survival function $S(t)$ can be defined among several groups. Survival probabilities can be derived for men and women for instance, or among different age classes. A natural question that arises is to find a statistical test to estimate if two survival functions are significantly different. In the case of two groups, we denote $S_1(t)$ and $S_2(t)$ are the survival functions for groups 1 and 2 respectively. If there is no censored observation, any rank test (the Wilcoxon rank sum for instance) can be used. In the presence of censoring, however, the differences in risk profile regarding survival can be tested between these two groups using the log-rank test (or Mantel-Cox test). Harrington and Fleming 1982. We find relevant to explicit the mechanisms of this statistical test, as it is internally used in most survival tree-based models, a critical subject of this thesis that will be discussed in Section 5.1.

The hypothesis of the log-rank test are:

- $H_0 : S_1(t) = S_2(t), \forall t$
- $H_1 : S_1(t) \neq S_2(t)$ for at least one t

The log-rank test compares the number of events observed in each group with the number of events expected under the H_0 hypothesis. The test statistic is an χ^2 distribution with one degree of freedom.

To construct a test statistic, let $t_{(1)}, t_{(2)}, \dots, t_{(k)}$ be the distinct ordered event times in groups 1 and 2 combined, let $O_{1,i}$ and $O_{2,i}$ be the numbers of observations just before time $t_{(i)}$ in samples 1 and 2, with $O_i = O_{1,i} + O_{2,i}$. Finally, let $\delta_{1,i}$ be a variable that takes the value 1 if the event at time $t_{(i)}$ occurs in sample 1, and 0 otherwise. Then, the log-rank (LR) test is given as:

$$Z_{\log-rank} = \frac{\sum_{i=1}^k \delta_{1,i} - \sum_{i=1}^k \frac{O_{1,i}}{O_i}}{\sqrt{\sum_{i=1}^k \frac{O_{1,i} O_{2,i}}{O_i^2}}} \quad (\text{A.7})$$

By the central limit theorem, the distribution of $Z_{\log-rank}$ converges towards a standard normal distribution as k approaches infinity and therefore can be approximated by the standard normal distribution for a sufficiently large k .

The null hypothesis can thus be rejected if $2P(\mathcal{N}(0, 1) \geq Z_{\log-rank}) \leq 0.05$.

A.0.7 Cox-Model

One of the most common survival models is the Cox proportional hazard (CPH) model (Cox 1972). It postulates that the hazard function can be modeled as the product of a time-dependent and a covariate-dependent functions.

The hazard function at time t for subject i with covariate vector \mathbf{X}_i , under Cox proportionnal hazard model can be expressed as:

$$\underbrace{\lambda(t|X_i^1, X_i^2, \dots)}_{\text{hazard function}} = \underbrace{\lambda_0(t)}_{\text{baseline hazard}} \underbrace{e^{\left(\mathbf{x}^{(i)} \cdot \beta^{(i)}\right)}}_{\text{partial hazard}} \quad (\text{A.8})$$

It is crucial to note that in this model, the hazard function is the product of the baseline hazard, which only varies with time, and the partial hazard, which only varies depending on the covariates.

The parameters of this model are the β , and they can easily be estimated with a maximum likelihood approach. Their estimation can be carried out without having to model $\lambda_0(t)$ - which is why CPH is considered semi-parametric.

A Cox model can be fit in R, using the packages `survival` and `timereg` (see Scheike and Zhang 2011), and in Python, with the library `lifelines` (see Davidson-Pilon 2019).

A.1 Competing risk framework

In practice, survival analysis is not limited to a single event, since subjects are likely to be at risk from several events at the same time, in contrast to multi-state models (see Andersen and Keiding 2002) where the transition between the different events is possible. When studying a cyclical event of interest such as death, for example, the different causes are in competition (or

concurrence), and then when the subject dies from one cause such as cancer, he cannot die from another. There are several regression models to estimate the global hazard and the hazard of one risk in settings where competing risks are present: modeling the cause-specific hazard and the subdistribution hazard function. They account for competing risks differently, obtaining different hazard functions and thus distinct advantages, drawbacks, and interpretations. Here, we will introduce those approaches' theoretical and practical implications and justify which one we will use in our modeling approaches.

A.1.1 Cause-specific approach

In cause-specific regression, each cause-specific hazard is estimated separately, in our case, the cause-specific hazards of lapse and death, by considering all subjects that experienced the competing event as censored. Here, t is the traditional time variable of a survival model, with $t = 0$ being the beginning of a policy. It is not to be confused with the use of t in Sections 8.3 and 8.4. We remind that $J_T = 0$ corresponds to an active subject that did not experience lapse $J_T = 1$ or death $J_T = 2$. The cause-specific hazard rates regarding the j -th risk ($j \in [1, \dots, J]$) are defined as

$$\lambda_{T,j}(t) = \lim_{dt \rightarrow 0} \frac{P(t \leq T < t + dt, J_T = j \mid T \geq t)}{dt}.$$

We can recover the global hazard rate as $\lambda_{T,1}(t) + \dots + \lambda_{T,J}(t) = \lambda_T(t)$, and derive the global survival distribution of T as

$$\begin{aligned} P(T > t) &= 1 - F_T(t) = S_T(t) \\ &= \exp\left(-\int_0^t (\lambda_{T,1}(s) + \dots + \lambda_{T,J}(s)) ds\right). \end{aligned}$$

This approach aims at analysing the cause-specific “distribution” function: $F_{T,j}(t) = P(T \leq t, J_T = j)$. In practice, it is called the Cumulative Incidence Function (*CIF*) for cause j and not a distribution function since $F_{T,j}(t) \rightarrow P(J_T = j) \neq 1$ as $t \rightarrow +\infty$. By analogy with the classical survival framework, the *CIF* can be characterised as $F_{T,j}(t) = \int_0^t f_{T,j}(s) ds$ ¹, where $f_{T,j}$ is the improper² density function for cause j . It follows that

$$f_{T,j}(s) = \lim_{dt \rightarrow 0} \frac{P(t \leq T < t + dt, J_T = j)}{dt} = \lambda_{T,j}(t) S_T(t).$$

The equation above is self-explanatory: the probability of experiencing cause j at time t is simply the product of surviving the previous time periods by the cause-specific hazard at time t . We finally obtain the *CIF* for cause j as

$$F_{T,j}(t) = \int_0^t \lambda_{T,j}(s) \exp\left(-\int_0^s \lambda_T(u) du\right) ds.$$

There are several advantages to that approach. First of all, cause-specific hazard models can be easily fit with any classical implementation of CPH by simply considering as censored any subject that experienced the competing event. Then the *CIF* is clearly interpretable and summable $P(T \leq t) = F_{T,1}(s) + \dots + F_{T,J}(s)$ ³. On the other hand, the *CIF* estimation of one given cause depends on all other causes: it implies that the study of a specific cause requires estimating

¹We suppose that T has a continuous distribution

²Because derived from the *CIF*, an improper cumulative distribution function

³unlike to the function $1 - \exp\left(-\int_0^t \lambda_{T,j}(u) du\right)$.

the global hazard rate and interpreting the effects of covariates on this cause is difficult. Indeed, part of the effects on a specific cause comes from the competing causes, but in our setting, we are only interested in the prediction of the survival probabilities, not their interpretation as such.

A.1.2 Subdistribution approach

We have introduced it at the beginning of this section; another approach is often considered to analyze competing risks and derive a cause-specific *CIF*. This other approach called the subdistribution hazard function of Fine and Gray regression, works by considering a new competing risk process τ . Without loss of generality, let's consider death as our cause of interest,

$$\tau = T \times \mathbb{1}_{J_T=2} + \infty \times \mathbb{1}_{J_T \neq 2}.$$

It has the same as T regarding the risk of death, $P(\tau \leq t) = F_{T,2}(t)$ and a mass point at infinity $1 - F_{T,2}(\infty)$, probability to observe other causes ($J_T \neq 2$) or not to observe any failure. In other words, if the previous approach considered every subject that experienced competing events as censored, this approach considers a new and artificial at-risk population. This last consideration is made clear when deriving the hazard rate of τ ,

$$\lambda_\tau(t) = \lim_{dt \rightarrow 0} \frac{P(t \leq T < t + dt, J_T = 2 \mid \{T \geq t\} \cup \{T \leq t, J_T \neq 2\})}{dt}.$$

Finally, we obtain the *CIF* for the risk of death as

$$F_{T,2}(t) = 1 - \exp\left(-\int_0^t \lambda_\tau(s) ds\right).$$

This subdistribution approach resolves the most important drawback to cause-specific regression, as the coefficients resulting from it do have a direct relationship with the cumulative incidence: estimating the *CIF* for a specific cause does not depend on the other causes, which makes the interpretation of *CIF* easier. The subdistribution hazard models can be fit in R by using the `crr` function in the `cmprsk` package or using the `timereg` package. Still, to our knowledge, there is no implementation of a Fine and Gray model in Lifelines or, more generally, Python. We can also note that these two approaches are linked, Putter, Schumacher, and Houwelingen 2020 and the link between $\lambda_\tau(t)$ and $\lambda_{T,j}(t)$ is given by

$$\lambda_\tau(t) = r_j(t) \lambda_{T,j}(t), \text{ with } r_j(t) = \frac{P(J_T = 0)}{\sum_{p \neq j} P(J_T = p)}.$$

In other words, if the probability of any competing risk is low, the two approaches give very close results.

B. Appendix of the first article (chap. 8)

B.1 Survival analysis results

The quantity $r_{lapse}^{(i)}(t)$ represents the probability that the policy of subject i is still active at time t , given that it was active at its last observed time. Predicting the overall conditional survival with the competing risks, in that case, can be achieved by creating a combined outcome. The policy ends with death or lapse, whichever comes first, and to compute r_{lapse} , we recode the competing events as a combined event. In terms of statistical guarantees, this approach is compatible with any survival analysis method.

In the following sections of this appendix, $r_{acceptant}^{(i)}(t)$ indicates the probability of survival for subject i at time t given that it will not lapse. In other words, it is the survival probability regarding only the risk of death. As detailed in Section 8.4.1, this corresponds to the cause-specific survival probability for death. It is to be noted that the density from which we derive our survival probabilities is improper as it derives itself from the *CIF*, which is not a proper distribution function.¹ Therefore, any conclusion about those probabilities should be drawn with care. Similarly to r_{lapse} , covariates selection and tuning are performed by minimizing AIC.

All graphs representing survival curves below are plotted with the same axis. The x-axes are the time in years, the y-axes represent the survival probability.

B.1.1 Cox-model

We first decide to estimate survival with a Cox Proportional hazard model with a spline baseline hazard from the Python library Lifelines. Covariate selection and tuning are performed by minimizing AIC. Here is what $r_{acceptant}$, the vector of cause-specific probabilities, looks like, and we can compare it to r_{lapse} on some subjects.

¹as it does not tend to 1 as t goes to $+\infty$

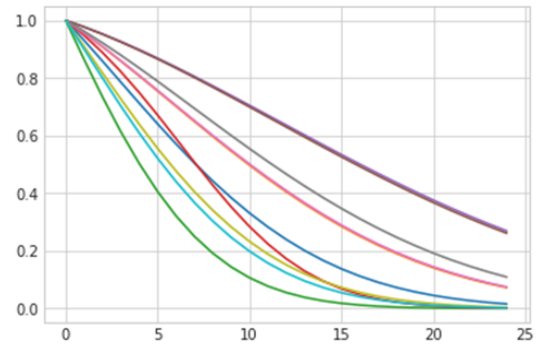
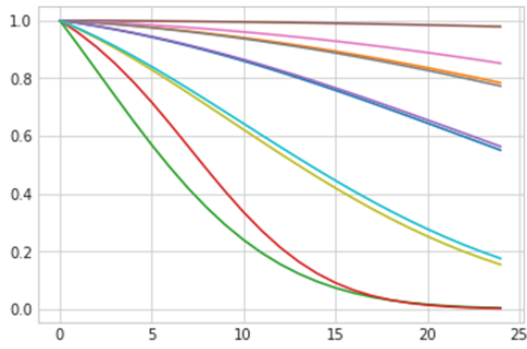


Figure B.1: 10 policyholders' survival curve for $r_{acceptant}$ with Cox model
 Figure B.2: 10 policyholders' survival curve for r_{lapser}
 The effect of various covariates on the survival outcome can be found below.

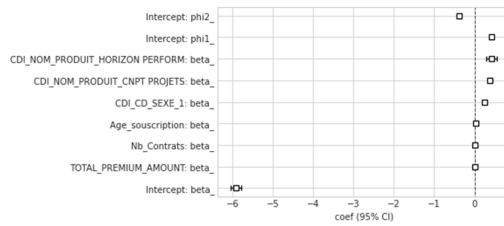


Figure B.3: Coefficient plot for r_{lapser}

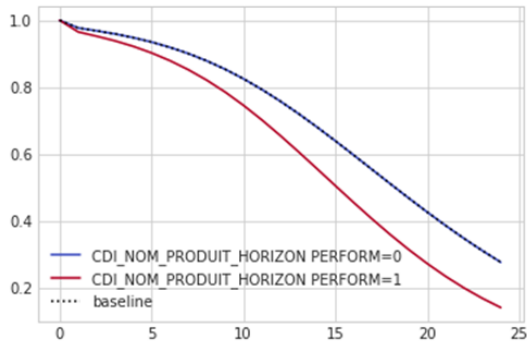


Figure B.4: r_{lapser} trajectories for different products

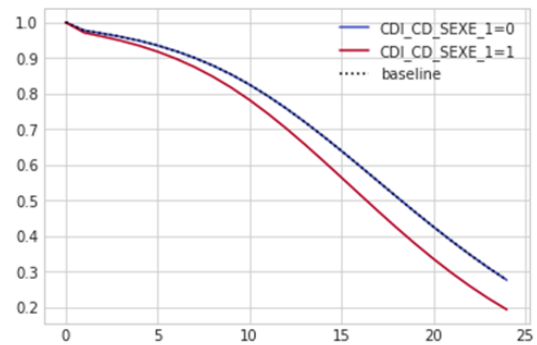


Figure B.5: r_{lapser} trajectories by gender

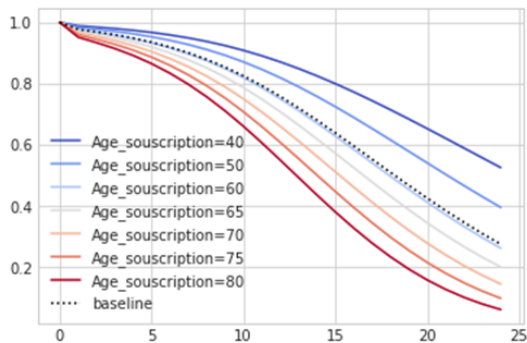


Figure B.6: r_{lapser} trajectories for different ages

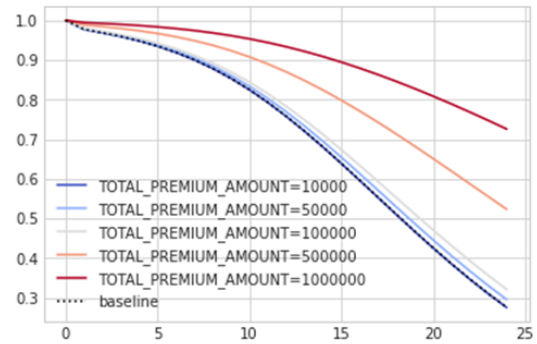


Figure B.7: r_{lapser} trajectories for different face amounts

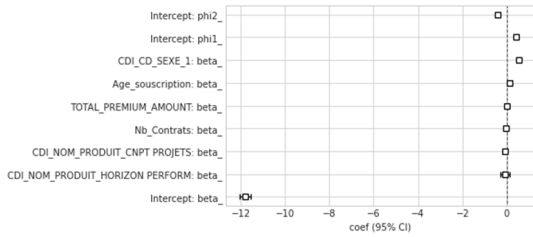


Figure B.8: Coefficient plot for $r_{acceptant}$

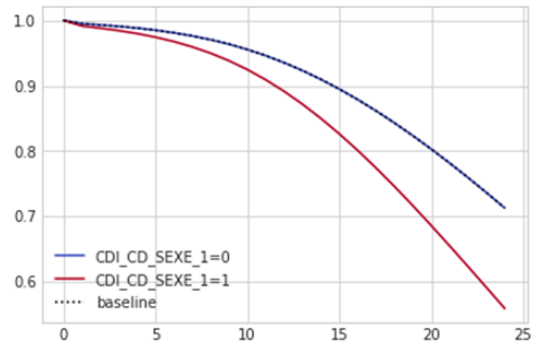


Figure B.9: $r_{acceptant}$ trajectories by gender

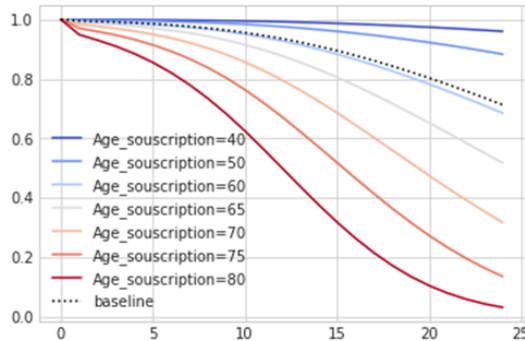


Figure B.10: $r_{acceptant}$ trajectories for different ages

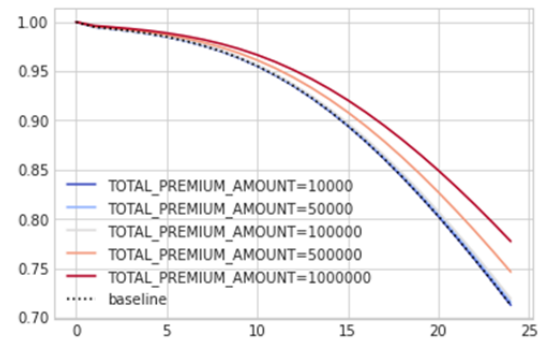


Figure B.11: $r_{acceptant}$ trajectories for different face amounts

B.1.2 RSF

We obtain better results than Cox in terms of concordance index at the cost of very high computation time for one training with one set of parameters - 5days without parallelisation, 4 hours with - compared to a few seconds for cox model.

Some of the results we obtain are displayed below.

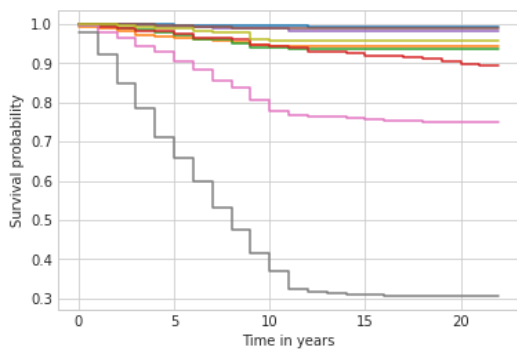


Figure B.12: 10 policyholders' survival curve for $r_{acceptant}$ with RSF

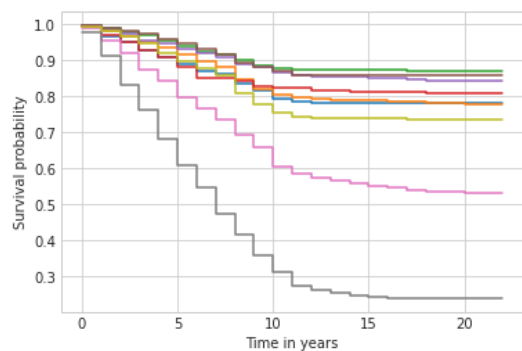


Figure B.13: 10 policyholders' survival curve for r_{lapser} with RSF

Weight		Feature
0.3148	± 0.0064	Age_souscription
0.0100	± 0.0008	CDI_CD_SEXE_1
0.0091	± 0.0014	PRODUIT_2
0.0077	± 0.0006	TOTAL_PREMIUM_AMOUNT
0.0013	± 0.0004	Nb_Contrats
0.0010	± 0.0003	PRODUIT_3

Table B.1: Covariates importance for $r_{acceptant}$ with RSF

Weight		Feature
0.1838	± 0.0045	Age_souscription
0.0415	± 0.0018	TOTAL_PREMIUM_AMOUNT
0.0083	± 0.0011	CDI_CD_SEXE_1
0.0026	± 0.0013	PRODUIT_2
0.0022	± 0.0006	PRODUIT_3
0.0020	± 0.0006	Nb_Contrats

Table B.2: Covariates importance for $r_{lapsier}$ with RSF

B.1.3 XGSB

We obtain better results than Cox and slightly better results than RSF in terms of concordance index at the cost of even higher computation time for one training with one set of parameters - 10h with great parallelisation - compared to a few seconds for Cox model. Some of the results we obtain are displayed below.

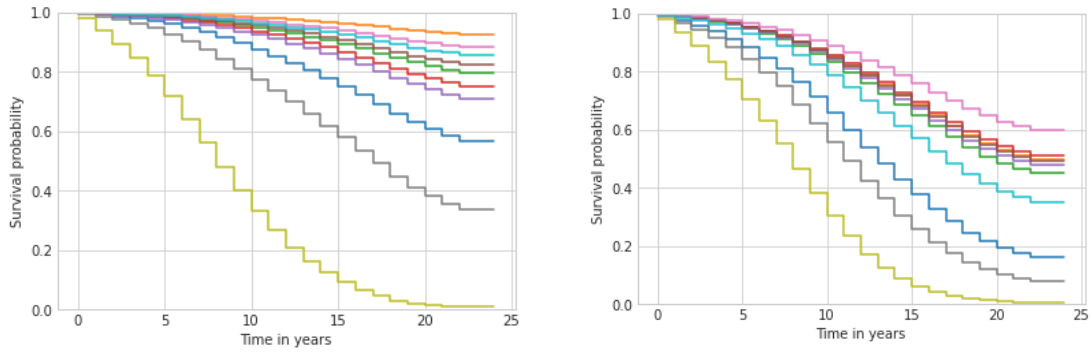


Figure B.14: 10 policyholders' survival curve for $r_{acceptant}$ with GBSM

Figure B.15: 10 policyholders' survival curve for $r_{lapsier}$ with GBSM

Weight		Feature
0.3274	± 0.0071	Age_souscription
0.0104	± 0.0006	TOTAL_PREMIUM_AMOUNT
0.0100	± 0.0008	CDI_CD_SEXE_1
0.0025	± 0.0005	PRODUIT_2
0.0005	± 0.0001	Nb_Contrats
0.0000	± 0.0001	PRODUIT_3

Table B.3: Covariates importance for $r_{acceptant}$ with GBSM

Weight		Feature
0.1872	± 0.0039	Age_souscription
0.0438	± 0.0020	TOTAL_PREMIUM_AMOUNT
0.0134	± 0.0014	PRODUIT_2
0.0076	± 0.0009	CDI_CD_SEXE_1
0.0051	± 0.0006	PRODUIT_3
0.0011	± 0.0004	Nb_Contrats

Table B.4: Covariates importance for $r_{lapsers}$ with GBSM

B.1.4 Final survival model

The final concordance index scores are displayed below:

	Concordance Index	
	$r_{lapsers}$	$r_{acceptant}$
Cox model	69,5%	80,7%
RSF	71,6%	83,7%
GBSM	73,0%	84,1%

Table B.5: Survival models comparison

B.2 Other results

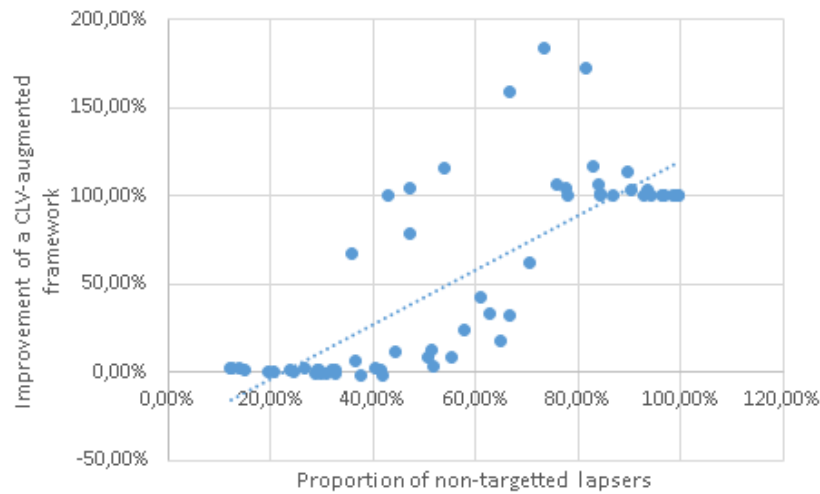


Figure B.16: Correlation between the proportion of non-targetted lapsers and the improvement² of a CLV-augmented LMS

B.3 Considering various statistical metrics

The table below contains the results of the LMSs listed in Table 8.3, evaluated on accuracy, recall, F1-score and AUC. For every metric, it displays the results of a classification over $y^{(i)}$ tuned and cross-validated with each of the metrics - respectively $y^{(i)}$, $y^{(i)}$, $y^{(i)}$ and $y^{(i)}$ - or over $\tilde{y}^{(i)}$ which is always tuned and cross-validated with RG .

N°	Model	Accuracy		Recall		F1-score		AUC		Retention gain					RG/target				
		$accuracy_{\tilde{y}^{(i)}}$	$\tilde{y}^{(i)}$	$recall_{\tilde{y}^{(i)}}$	$\tilde{y}^{(i)}$	$F1-score_{\tilde{y}^{(i)}}$	$\tilde{y}^{(i)}$	$AUC_{\tilde{y}^{(i)}}$	$\tilde{y}^{(i)}$	$accuracy_{\tilde{y}^{(i)}}$	$recall_{\tilde{y}^{(i)}}$	$F1-score_{\tilde{y}^{(i)}}$	$AUC_{\tilde{y}^{(i)}}$	$\tilde{y}^{(i)}$	$accuracy_{\tilde{y}^{(i)}}$	$recall_{\tilde{y}^{(i)}}$	$F1-score_{\tilde{y}^{(i)}}$	$AUC_{\tilde{y}^{(i)}}$	$\tilde{y}^{(i)}$
A-1	CART	92.3%	85.3%	75.7%	18.0%	76.4%	28.9%	85.6%	58.4%	114,661	128,251	135,085	128,251	219,655	4.48	5.13	5.46	5.13	38.20
	RF	92.9%	85.4%	74.7%	14.7%	77.7%	24.9%	85.6%	57.0%	232,314	203,821	203,821	203,821	287,884	9.82	8.66	8.66	8.66	56.65
	XGB	93.4%	85.8%	79.6%	18.5%	80.0%	30.1%	87.8%	58.8%	243,365	261,553	266,198	266,198	324,952	9.61	10.25	10.55	10.55	54.64
A-5	CART	92.3%	83.6%	76.8%	2.5%	76.7%	4.8%	86.0%	51.1%	-514,477	-497,277	-494,414	-497,277	-112,372	-20.08	-19.28	-19.21	-19.28	-86.48
	RF	92.9%	83.4%	74.9%	0.6%	77.9%	1.3%	85.7%	50.3%	-323,544	-323,543	-325,110	-323,543	-3,937	-13.65	-13.64	-13.72	-13.64	-28.28
	XGB	93.4%	83.3%	79.8%	0.4%	80.0%	0.7%	87.9%	50.2%	-383,004	-379,736	-379,736	-379,736	0	-15.14	-14.88	-14.88	-14.88	0
A-25	CART	92.3%	89.2%	76.2%	50.2%	76.7%	60.1%	85.8%	73.6%	4,160,423	4,322,030	4,267,310	4,322,030	3,882,623	162.44	170.53	251.34	251.34	241.06
	RF	92.9%	89.5%	75.3%	47.5%	78.0%	60.3%	85.8%	72.7%	4,018,432	3,687,705	3,687,705	3,687,705	3,666,219	169.65	154.49	154.49	154.49	249.54
	XGB	93.4%	90.0%	80.1%	51.2%	80.1%	63.0%	88.0%	74.4%	4,455,108	4,633,404	4,578,684	4,633,404	4,410,629	176.09	179.36	180.16	179.36	267.87

Table B.6: Results of representative LMS with various statistical metrics

It is to be noted that regardless of the evaluation metric used for tuning and validation purposes, the objective function used with XGB to generate those results is always the log-loss function. Using the area under the ROC curve or the area under the Precision-Recall curve as an objective function in this boosting algorithm would surely yield better results when trained on $y^{(i)}$ and even better on the more unbalanced $\tilde{y}^{(i)}$. As stated in Section 8.4.2, this analysis is not within the scope of our article.

B.4 Complete LMS numerical results

²Taking the results of XGBoost and excluding LMS n°B-27 that has a very high improvement ratio.

LMS	p	δ	γ	c	d	T	LMS	p	δ	γ	c	d	T
A-1	2,50%	0,04%	25%	10	1,50%	5	B-1	2,50%	0,08%	20%	10	1,50%	5
A-2	2,50%	0,04%	25%	10	1,50%	20	B-2	2,50%	0,08%	20%	10	1,50%	20
A-3	2,50%	0,04%	25%	100	1,50%	5	B-3	2,50%	0,08%	20%	100	1,50%	5
A-4	2,50%	0,04%	25%	100	1,50%	20	B-4	2,50%	0,08%	20%	100	1,50%	20
A-5	2,50%	0,04%	5%	10	1,50%	5	B-5	2,50%	0,08%	10%	10	1,50%	5
A-6	2,50%	0,04%	5%	10	1,50%	20	B-6	2,50%	0,08%	10%	10	1,50%	20
A-7	2,50%	0,04%	5%	100	1,50%	5	B-7	2,50%	0,08%	10%	100	1,50%	5
A-8	2,50%	0,04%	5%	100	1,50%	20	B-8	2,50%	0,08%	10%	100	1,50%	20
A-9	2,50%	0,10%	25%	10	1,50%	5	B-9	2,50%	0,20%	20%	10	1,50%	5
A-10	2,50%	0,10%	25%	10	1,50%	20	B-10	2,50%	0,20%	20%	10	1,50%	20
A-11	2,50%	0,10%	25%	100	1,50%	5	B-11	2,50%	0,20%	20%	100	1,50%	5
A-12	2,50%	0,10%	25%	100	1,50%	20	B-12	2,50%	0,20%	20%	100	1,50%	20
A-13	2,50%	0,10%	5%	10	1,50%	5	B-13	2,50%	0,20%	10%	10	1,50%	5
A-14	2,50%	0,10%	5%	10	1,50%	20	B-14	2,50%	0,20%	10%	10	1,50%	20
A-15	2,50%	0,10%	5%	100	1,50%	5	B-15	2,50%	0,20%	10%	100	1,50%	5
A-16	2,50%	0,10%	5%	100	1,50%	20	B-16	2,50%	0,20%	10%	100	1,50%	20
A-17	5,00%	0,04%	25%	10	1,50%	5	B-17	5,00%	0,08%	20%	10	1,50%	5
A-18	5,00%	0,04%	25%	10	1,50%	20	B-18	5,00%	0,08%	20%	10	1,50%	20
A-19	5,00%	0,04%	25%	100	1,50%	5	B-19	5,00%	0,08%	20%	100	1,50%	5
A-20	5,00%	0,04%	25%	100	1,50%	20	B-20	5,00%	0,08%	20%	100	1,50%	20
A-21	5,00%	0,04%	5%	10	1,50%	5	B-21	5,00%	0,08%	10%	10	1,50%	5
A-22	5,00%	0,04%	5%	10	1,50%	20	B-22	5,00%	0,08%	10%	10	1,50%	20
A-23	5,00%	0,04%	5%	100	1,50%	5	B-23	5,00%	0,08%	10%	100	1,50%	5
A-24	5,00%	0,04%	5%	100	1,50%	20	B-24	5,00%	0,08%	10%	100	1,50%	20
A-25	5,00%	0,10%	25%	10	1,50%	5	B-25	5,00%	0,20%	20%	10	1,50%	5
A-26	5,00%	0,10%	25%	10	1,50%	20	B-26	5,00%	0,20%	20%	10	1,50%	20
A-27	5,00%	0,10%	25%	100	1,50%	5	B-27	5,00%	0,20%	20%	100	1,50%	5
A-28	5,00%	0,10%	25%	100	1,50%	20	B-28	5,00%	0,20%	20%	100	1,50%	20
A-29	5,00%	0,10%	5%	10	1,50%	5	B-29	5,00%	0,20%	10%	10	1,50%	5
A-30	5,00%	0,10%	5%	10	1,50%	20	B-30	5,00%	0,20%	10%	10	1,50%	20
A-31	5,00%	0,10%	5%	100	1,50%	5	B-31	5,00%	0,20%	10%	100	1,50%	5
A-32	5,00%	0,10%	5%	100	1,50%	20	B-32	5,00%	0,20%	10%	100	1,50%	20

Table B.7: More LMS

N°	time (s)	Model	% target diff	Accuracy		Retention gain		RG/target		Improvement ²
				$y^{(i)}$	$\hat{y}^{(i)}$	$y^{(i)}$	$\hat{y}^{(i)}$	$y^{(i)}$	$\hat{y}^{(i)}$	
A-1	4949	CART	62,58%	92,3%	85,3%	114 661	219 655	4,48	38,20	91,57%
		RF		92,9%	85,4%	232 314	287 884	9,82	56,65	23,92%
		XGB		93,4%	85,8%	243 365	324 952	9,61	54,64	33,52%
A-2	6111	CART	26,66%	92,3%	89,8%	7 092 097	6 142 119	277,00	353,83	-13,39%
		RF		92,9%	90,2%	6 596 374	5 696 455	278,47	351,02	-13,64%
		XGB		93,4%	90,9%	7 308 721	7 432 688	288,92	404,84	1,70%
A-3	4603	CART	93,50%	92,3%	83,3%	- 2 187 622	- 8 224	- 85,52	- 31,09	99,62%
		RF		92,9%	83,4%	- 1 900 265	45 483	- 80,18	194,35	102,39%
		XGB		93,4%	83,5%	- 2 032 650	77 481	- 80,39	174,44	103,81%
A-4	5555	CART	55,37%	92,3%	86,5%	4 789 814	5 117 844	187,00	577,74	6,85%
		RF		92,9%	86,4%	4 463 796	4 255 175	188,47	566,05	-4,67%
		XGB		93,4%	86,8%	5 032 706	5 433 366	198,92	610,26	7,96%
A-5	4753	CART	86,72%	92,3%	83,6%	- 514 477	- 112 372	- 20,08	- 86,48	78,16%
		RF		92,9%	83,4%	- 323 544	- 3 937	- 13,65	- 28,28	98,78%
		XGB		93,4%	83,3%	- 383 004	0	- 15,14	0	100,00%
A-6	5803	CART	44,27%	92,3%	87,9%	335 810	517 224	13,17	39,91	54,02%
		RF		92,9%	87,9%	655 350	661 021	27,68	61,13	0,87%
		XGB		93,4%	88,6%	654 219	729 493	25,86	58,22	11,51%
A-7	4241	CART	99,09%	92,3%	83,3%	- 2 816 759	- 10 205	- 110,08	- 384,04	99,64%
		RF		92,9%	83,3%	- 2 456 122	1 013	- 103,65	66,30	100,04%
		XGB		93,4%	83,3%	- 2 659 020	243	- 105,14	15,92	100,01%
A-8	5164	CART	82,78%	92,3%	84,0%	- 1 966 473	- 46 323	- 76,83	- 22,31	97,64%
		RF		92,9%	84,0%	- 1 477 229	253 885	- 62,32	149,67	117,19%
		XGB		93,4%	84,1%	- 1 621 796	273 243	- 64,14	117,83	116,85%
A-9	4781	CART	77,60%	92,3%	83,7%	- 825 372	- 161 100	- 32,19	- 127,87	80,48%
		RF		92,9%	83,4%	- 384 736	8 596	- 16,22	32,12	102,23%
		XGB		93,4%	83,6%	- 498 263	22 337	- 19,70	35,47	104,48%
A-10	6075	CART	29,10%	92,3%	89,7%	4 614 513	4 483 831	180,36	266,33	-2,83%
		RF		92,9%	89,9%	4 973 929	4 328 724	210,01	280,90	-12,97%
		XGB		93,4%	90,7%	5 354 770	5 368 917	211,69	301,57	0,26%
A-11	4506	CART	96,56%	92,3%	83,2%	- 3 127 655	- 118 886	- 122,19	- 2 230,39	96,20%
		RF		92,9%	83,3%	- 2 517 315	1 340	- 106,22	87,71	100,05%
		XGB		93,4%	83,3%	- 2 774 278	736	- 109,70	52,00	100,03%
A-12	5534	CART	57,93%	92,3%	86,2%	2 312 231	3 310 314	90,36	412,71	43,17%
		RF		92,9%	86,1%	2 841 351	3 129 652	120,01	465,74	10,15%
		XGB		93,4%	86,6%	3 078 755	3 825 920	121,69	475,53	24,27%
A-13	4640	CART	92,91%	92,3%	83,3%	- 1 201 626	- 163 056	- 46,87	- 1 838,44	86,43%
		RF		92,9%	83,3%	- 717 620	- 5 339	- 30,28	- 354,24	99,26%
		XGB		93,4%	83,3%	- 875 378	508	- 34,60	16,26	100,06%
A-14	5739	CART	47,12%	92,3%	87,3%	- 1 476 651	- 831 019	- 57,49	- 77,99	43,72%
		RF		92,9%	86,0%	- 380 683	126 532	- 16,03	21,14	133,24%
		XGB		93,4%	85,5%	- 644 389	29 382	- 25,47	7,10	104,56%
A-15	4216	CART	99,61%	92,3%	83,3%	- 3 503 908	- 97 263	- 136,87	- 2 354,34	97,22%
		RF		92,9%	83,3%	- 2 850 198	0	- 120,28	0	100,00%
		XGB		93,4%	83,3%	- 3 151 393	0	- 124,60	0	100,00%
A-16	5096	CART	84,46%	92,3%	83,8%	- 3 778 933	- 734 773	- 147,49	- 418,58	80,56%
		RF		92,9%	83,5%	- 2 513 261	8 914	- 106,03	20,13	100,35%
		XGB		93,4%	83,6%	- 2 920 405	34 492	- 115,47	45,75	101,18%

N°	time (s)	Model	% target diff	Accuracy		Retention gain		RG/target		Improvement ²
				$y^{(i)}$	$\hat{y}^{(i)}$	$y^{(i)}$	$\hat{y}^{(i)}$	$y^{(i)}$	$\hat{y}^{(i)}$	
A-17	5390	CART	28,74%	92,3%	89,5%	5 100 456	4 899 479	199,11	279,88	-3,94%
		RF		92,9%	89,8%	4 635 482	4 226 648	195,69	276,06	-8,82%
		XGB		93,4%	90,2%	5 196 736	5 138 253	205,40	299,27	-1,13%
A-18	6452	CART	12,12%	92,3%	91,3%	52 090 240	47 706 070	2 034,15	2 170,64	-8,42%
		RF		92,9%	91,9%	46 171 160	42 049 900	1 949,05	2 082,36	-8,93%
		XGB		93,4%	92,5%	51 629 950	52 606 740	2 040,95	2 339,70	1,89%
A-19	4913	CART	64,89%	92,3%	85,2%	2 798 173	3 182 143	109,11	481,60	13,72%
		RF		92,9%	85,2%	2 502 903	2 743 070	105,69	554,76	9,60%
		XGB		93,4%	85,6%	2 920 720	3 438 303	115,40	576,64	17,72%
A-20	6160	CART	29,03%	92,3%	89,6%	49 787 960	45 366 730	1 944,15	2 616,32	-8,88%
		RF		92,9%	90,0%	44 038 580	39 947 830	1 859,05	2 547,89	-9,29%
		XGB		93,4%	90,6%	49 353 940	49 789 670	1 950,95	2 796,17	0,88%
A-21	5079	CART	51,69%	92,3%	86,8%	482 682	544 887	18,85	53,99	12,89%
		RF		92,9%	86,8%	557 090	554 195	23,52	65,17	-0,52%
		XGB		93,4%	87,1%	607 670	624 556	24,01	64,79	2,78%
A-22	6199	CART	23,94%	92,3%	90,2%	9 335 438	8 527 444	364,60	454,78	-8,66%
		RF		92,9%	90,6%	8 570 307	7 931 029	361,80	460,42	-7,46%
		XGB		93,4%	91,2%	9 518 466	9 581 934	376,27	501,56	0,67%
A-23	4601	CART	89,51%	92,3%	83,6%	- 1 819 600	135 305	- 71,15	121,80	107,44%
		RF		92,9%	83,5%	- 1 575 489	159 620	- 66,48	215,65	110,13%
		XGB		93,4%	83,7%	- 1 668 346	228 226	- 65,99	208,69	113,68%
A-24	5650	CART	50,83%	92,3%	87,0%	7 033 156	7 124 100	274,60	680,08	1,29%
		RF		92,9%	87,0%	6 437 729	6 364 477	271,80	711,89	-1,14%
		XGB		93,4%	87,4%	7 242 450	7 840 770	286,27	771,71	8,26%
A-25	5379	CART	30,97%	92,3%	89,2%	4 160 423	3 882 623	162,44	241,06	-6,68%
		RF		92,9%	89,5%	4 018 432	3 666 219	169,65	249,54	-8,76%
		XGB		93,4%	90,0%	4 455 108	4 410 629	176,09	267,87	-1,00%
A-26	6410	CART	12,52%	92,3%	91,3%	49 612 660	45 948 690	1 937,51	2 083,30	-7,39%
		RF		92,9%	91,9%	44 548 720	40 814 960	1 880,59	2 029,68	-8,38%
		XGB		93,4%	92,5%	49 676 000	50 549 740	1 963,72	2 260,20	1,76%
A-27	4887	CART	66,67%	92,3%	85,1%	1 858 140	2 575 538	72,44	442,86	38,61%
		RF		92,9%	85,0%	1 885 853	2 387 018	79,65	531,25	26,57%
		XGB		93,4%	85,4%	2 179 093	2 879 880	86,09	544,35	32,16%
A-28	6047	CART	29,42%	92,3%	89,4%	47 310 370	43 168 880	1 847,51	2 519,41	-8,75%
		RF		92,9%	89,9%	42 416 140	38 573 620	1 790,59	2 504,61	-9,06%
		XGB		93,4%	90,5%	47 399 990	47 812 830	1 873,72	2 721,63	0,87%
A-29	5070	CART	53,79%	92,3%	86,5%	- 204 467	- 5 098	- 7,95	- 1,66	97,51%
		RF		92,9%	86,1%	163 014	273 435	6,90	40,30	67,74%
		XGB		93,4%	86,8%	115 297	248 982	4,55	28,64	115,95%
A-30	6179	CART	24,36%	92,3%	90,3%	7 522 978	7 058 487	293,94	382,06	-6,17%
		RF		92,9%	90,6%	7 534 275	7 068 293	318,08	411,80	-6,18%
		XGB		93,4%	91,2%	8 219 857	8 265 167	324,94	442,88	0,55%
A-31	4627	CART	90,18%	92,3%	83,6%	- 2 506 749	- 139 983	- 97,95	- 121,44	94,42%
		RF		92,9%	83,5%	- 1 969 564	73 101	- 83,10	111,49	103,71%
		XGB		93,4%	83,6%	- 2 160 719	76 641	- 85,45	93,28	103,55%
A-32	5679	CART	51,25%	92,3%	86,8%	5 220 695	5 811 833	203,94	583,55	11,32%
		RF		92,9%	86,9%	5 401 696	5 269 505	228,08	605,69	-2,45%
		XGB		93,4%	87,4%	5 943 841	6 682 230	234,94	670,03	12,42%

N°	time (s)	Model	% target diff	Accuracy		Retention gain		RG/target		Improvement ²
				$y^{(i)}$	$\hat{y}^{(i)}$	$y^{(i)}$	$\hat{y}^{(i)}$	$y^{(i)}$	$\hat{y}^{(i)}$	
B-1	4778	CART	75,89%	92,3%	84,0%	- 627 165	- 148 913	- 24,46	- 65,19	76,26%
		RF		92,9%	83,7%	- 280 855	11 973	- 11,84	9,57	104,26%
		XGB		93,4%	84,1%	- 366 103	25 099	- 14,47	12,30	106,86%
B-2	6074	CART	29,70%	92,3%	89,7%	3 862 156	3 397 247	150,95	203,11	-12,04%
		RF		92,9%	89,9%	4 127 224	3 550 730	174,26	230,67	-13,97%
		XGB		93,4%	90,6%	4 451 686	4 408 819	175,99	250,17	-0,96%
B-3	4528	CART	96,60%	92,3%	83,2%	- 2 929 448	- 85 465	- 114,46	- 1 482,06	97,08%
		RF		92,9%	83,3%	- 2 413 433	3 724	- 101,84	- 108,33	100,15%
		XGB		93,4%	83,3%	- 2 642 119	9 092	- 104,47	93,79	100,34%
B-4	5476	CART	60,93%	92,3%	85,9%	1 559 874	2 471 262	60,95	329,63	58,43%
		RF		92,9%	85,8%	1 994 645	2 517 111	84,26	422,45	26,19%
		XGB		93,4%	86,3%	2 175 670	3 089 897	85,99	422,77	42,02%
B-5	4708	CART	84,33%	92,3%	83,4%	- 857 439	- 159 856	- 33,45	- 218,16	81,36%
		RF		92,9%	83,3%	- 484 459	40	- 20,44	7,23	100,01%
		XGB		93,4%	83,3%	- 596 203	897	- 23,57	46,96	100,15%
B-6	5906	CART	36,63%	92,3%	88,8%	705 721	922 490	27,69	60,21	30,72%
		RF		92,9%	88,9%	1 352 182	1 269 349	57,11	97,63	-6,13%
		XGB		93,4%	89,6%	1 342 882	1 428 722	53,09	96,76	6,39%
B-7	4400	CART	98,49%	92,3%	83,2%	- 3 159 722	- 39 633	- 123,45	- 1 230,61	98,75%
		RF		92,9%	83,3%	- 2 617 037	1 024	- 110,44	0,56	100,04%
		XGB		93,4%	83,3%	- 2 872 219	295	- 113,57	19,31	100,01%
B-8	5278	CART	73,18%	92,3%	84,6%	- 1 596 562	169 852	- 62,31	41,78	110,64%
		RF		92,9%	84,6%	- 780 396	637 625	- 32,89	194,52	181,71%
		XGB		93,4%	85,0%	- 933 133	780 845	- 36,91	188,79	183,68%
B-9	4601	CART	94,12%	92,3%	83,3%	- 2 380 789	- 113 444	- 92,86	- 840,25	95,24%
		RF		92,9%	83,3%	- 1 403 468	317	- 59,21	7,96	100,02%
		XGB		93,4%	83,3%	- 1 724 731	3 980	- 68,17	149,44	100,23%
B-10	5947	CART	35,98%	92,3%	89,0%	- 760 449	429 196	- 29,35	29,80	156,44%
		RF		92,9%	88,5%	1 175 540	1 354 131	49,71	118,11	15,19%
		XGB		93,4%	89,8%	871 455	1 456 080	34,48	96,25	67,09%
B-11	4229	CART	99,16%	92,3%	83,3%	- 4 683 072	- 48 985	- 182,86	- 1 186,22	98,95%
		RF		92,9%	83,3%	- 3 536 046	0	- 149,21	0	100,00%
		XGB		93,4%	83,3%	- 4 000 747	0	- 158,17	0	100,00%
B-12	5391	CART	66,76%	92,3%	85,0%	- 3 062 732	- 388 289	- 119,35	- 80,44	87,32%
		RF		92,9%	84,7%	- 957 039	710 688	- 40,29	220,55	174,26%
		XGB		93,4%	85,3%	- 1 404 561	834 198	- 55,52	163,88	159,39%
B-13	4493	CART	96,30%	92,3%	83,3%	- 2 358 179	- 159 922	- 91,98	- 2 793,13	93,22%
		RF		92,9%	83,3%	- 1 384 098	0	- 58,40	0	100,00%
		XGB		93,4%	83,3%	- 1 705 577	0	- 67,42	0	100,00%
B-14	5851	CART	42,98%	92,3%	87,8%	- 3 251 762	- 1 761 821	- 126,63	- 143,20	45,82%
		RF		92,9%	86,4%	- 1 013 089	79 273	- 42,69	11,90	107,82%
		XGB		93,4%	83,3%	- 1 582 006	4 396	- 62,52	287,68	100,28%
B-15	4040	CART	99,67%	92,3%	83,3%	- 4 660 462	- 38 969	- 181,98	- 2 075,03	99,16%
		RF		92,9%	83,3%	- 3 516 676	0	- 148,40	0	100,00%
		XGB		93,4%	83,3%	- 3 981 592	161	- 157,42	10,53	100,00%
B-16	5182	CART	77,97%	92,3%	84,2%	- 5 554 044	- 1 491 522	- 216,63	- 549,23	73,15%
		RF		92,9%	83,6%	- 3 145 668	52 475	- 132,69	84,54	101,67%
		XGB		93,4%	83,3%	- 3 858 022	0	- 152,52	0	100,00%

N°	time (s)	Model	% target diff	Accuracy		Retention gain		RG/target		Improvement ²
				$y^{(i)}$	$\hat{y}^{(i)}$	$y^{(i)}$	$\hat{y}^{(i)}$	$y^{(i)}$	$\hat{y}^{(i)}$	
B-17	5324	CART	32,66%	92,3%	88,9%	3 361 471	3 037 200	131,25	191,31	-9,65%
		RF		92,9%	89,3%	3 241 680	2 911 023	136,86	204,43	-10,20%
		XGB		93,4%	89,6%	3 596 593	3 546 671	142,15	222,04	-1,39%
B-18	6411	CART	13,83%	92,3%	91,1%	39 860 670	37 695 680	1 556,66	1 778,71	-5,43%
		RF		92,9%	91,7%	35 787 050	32 345 100	1 510,72	1 654,32	-9,62%
		XGB		93,4%	92,0%	39 908 670	40 886 810	1 577,61	1 848,71	2,45%
B-19	4853	CART	70,34%	92,3%	84,7%	1 059 189	1 813 631	41,25	392,14	71,23%
		RF		92,9%	84,8%	1 109 101	1 808 616	46,86	474,33	63,07%
		XGB		93,4%	85,0%	1 320 578	2 141 271	52,15	482,34	62,15%
B-20	5973	CART	31,76%	92,3%	89,2%	37 558 390	34 068 550	1 466,66	2 125,97	-9,29%
		RF		92,9%	89,4%	33 654 470	30 032 580	1 420,72	2 072,47	-10,76%
		XGB		93,4%	90,1%	37 632 650	38 008 480	1 487,61	2 277,17	1,00%
B-21	5228	CART	41,79%	92,3%	87,7%	1 136 879	1 179 837	44,40	92,50	3,78%
		RF		92,9%	88,1%	1 276 808	1 188 256	53,91	104,81	-6,94%
		XGB		93,4%	88,7%	1 385 145	1 356 864	54,74	104,76	-2,04%
B-22	6296	CART	19,52%	92,3%	90,7%	18 704 980	17 177 190	730,55	852,81	-8,17%
		RF		92,9%	91,1%	17 182 100	15 732 340	725,34	859,29	-8,44%
		XGB		93,4%	91,5%	19 071 370	19 050 020	753,90	939,00	-0,11%
B-23	4746	CART	81,36%	92,3%	84,1%	- 1 165 404	458 223	- 45,60	172,83	139,32%
		RF		92,9%	84,0%	- 855 770	525 335	- 36,09	288,55	161,39%
		XGB		93,4%	84,1%	- 890 871	645 445	- 35,26	310,86	172,45%
B-24	5845	CART	40,47%	92,3%	88,2%	16 402 700	15 013 310	640,55	1 093,43	-8,47%
		RF		92,9%	88,4%	15 049 520	13 423 040	635,34	1 122,81	-10,81%
		XGB		93,4%	88,9%	16 795 360	17 144 260	663,90	1 247,50	2,08%
B-25	5274	CART	37,42%	92,3%	88,6%	1 607 847	1 839 864	62,84	126,33	14,43%
		RF		92,9%	88,7%	2 119 067	1 923 982	89,49	152,71	-9,21%
		XGB		93,4%	89,2%	2 237 965	2 194 469	88,45	155,54	-1,94%
B-26	6425	CART	14,83%	92,3%	91,1%	35 238 060	32 690 970	1 376,37	1 558,26	-7,23%
		RF		92,9%	91,6%	32 835 370	29 986 540	1 386,17	1 543,12	-8,68%
		XGB		93,4%	92,0%	36 328 440	36 803 630	1 436,10	1 688,53	1,31%
B-27	4811	CART	73,92%	92,3%	84,3%	- 694 436	751 404	- 27,16	226,99	208,20%
		RF		92,9%	84,4%	- 13 512	1 018 369	- 0,51	356,48	7636,98%
		XGB		93,4%	84,7%	- 38 050	1 253 252	- 1,55	345,94	3393,68%
B-28	5995	CART	32,61%	92,3%	89,1%	32 935 780	29 342 930	1 286,37	1 847,71	-10,91%
		RF		92,9%	89,4%	30 702 790	27 725 620	1 296,17	1 933,38	-9,70%
		XGB		93,4%	90,0%	34 052 420	34 390 060	1 346,10	2 094,90	0,99%
B-29	5143	CART	47,03%	92,3%	87,3%	- 363 861	55 985	- 14,12	3,38	115,39%
		RF		92,9%	87,4%	377 170	488 284	15,95	49,62	29,46%
		XGB		93,4%	88,0%	275 772	491 567	10,89	44,89	78,25%
B-30	6243	CART	20,47%	92,3%	90,7%	14 747 500	13 838 380	576,23	690,22	-6,16%
		RF		92,9%	91,1%	14 816 830	13 378 460	625,54	743,34	-9,71%
		XGB		93,4%	91,5%	16 146 490	16 169 440	638,30	814,80	0,14%
B-31	4730	CART	83,83%	92,3%	83,7%	- 2 666 144	- 487 716	- 104,12	- 267,75	81,71%
		RF		92,9%	83,7%	- 1 755 409	139 545	- 74,05	102,66	107,95%
		XGB		93,4%	83,7%	- 2 000 244	134 199	- 79,11	130,13	106,71%
B-32	5865	CART	41,41%	92,3%	88,0%	12 445 210	11 693 070	486,23	884,49	-6,04%
		RF		92,9%	88,3%	12 684 250	11 381 260	535,54	971,28	-10,27%
		XGB		93,4%	88,8%	13 870 470	14 101 470	548,30	1 048,38	1,67%

²In order to account for negative retention gains, the improvement is computed with an absolute value for the denominator. This leads to a rather unintuitive improvement measure whenever one of the models yields negative RG and the other positive RG.

C. Appendix of the second article (chap. 12)

C.1 Note on parametric models

This work focuses on the ability of non-parametric tree-based approaches to perform in both steps of our framework. For comparison's sake, a semi-parametric survival model had been fitted in Valla, Milhaud, and Olympio 2023; it is important to explain why we did not investigate such models here. Time-varying Cox-like models also exist and can even take competing risks into account. They can be compared and yield survival curves for any individual but only up to their last observed time. Predicting survival probabilities at future time points is not possible. For the astute reader, a complete implementation of those techniques can be found in the R package `timeereg` by Scheike and Martinussen 2006; Scheike and Zhang 2011.

Moreover, other prediction biases can appear in the presence of endogenous longitudinal covariates, with Cox-like models Austin, Latouche, and Fine 2019, which is typically our situation. This is why we decided to leave such modeling approaches out of this paper.

It is to be noted that a statistical learning approach addressing research questions involving the association structure between longitudinal data and an event time exists: joint models. This type of modeling technique is primarily used in time-to-event contexts, with censored data and can handle multiple exogenous and endogenous longitudinal covariates with possibly multiple competing risks. Joint models outweigh time-dependent Cox models in terms of prediction; by predicting both the longitudinal trajectories and the survival probabilities simultaneously, it is possible to compute the conditional probability of surviving later than the last observed time for which a longitudinal measurement was available. They have been extensively studied and extended and have proved to yield competitive predictive results for relatively small datasets. A complete overview of such models can be found in Rizopoulos 2012, and their implementation is available in R packages `JM`, `JMBayes` and `JMBayes2`. Joint models are performant but computationally expensive for large datasets and multiple longitudinal covariates or outcomes. We did not implement this approach in this paper for those reasons and instead implement tree-based models handling time-varying covariates that we will compare to tree-based models with time-fixed covariates.

C.2 Model selection methodology

Regardless of their size, \mathcal{D}^{last} and \mathcal{D}^{long} both relate to 10,000 subjects. In order to tune the models detailed in the next Sections, we adopt a 5-fold Monte-Carlo cross-validation methodology. We randomly select 80% of subjects' observations in \mathcal{D}^{last} and \mathcal{D}^{long} as training sets, and the remaining 20% of subjects' observations go in testing sets. Models are trained on the training sets

and tested on both training and testing sets to control for overfitting. We repeat this step 5 times such that we obtain 20 different datasets: ${}^k\mathcal{D}_{train}^{last}$, ${}^k\mathcal{D}_{test}^{last}$, ${}^k\mathcal{D}_{train}^{long}$ and ${}^k\mathcal{D}_{test}^{long}$ for $k \in [1, \dots, 5]$. We can illustrate this as follows:

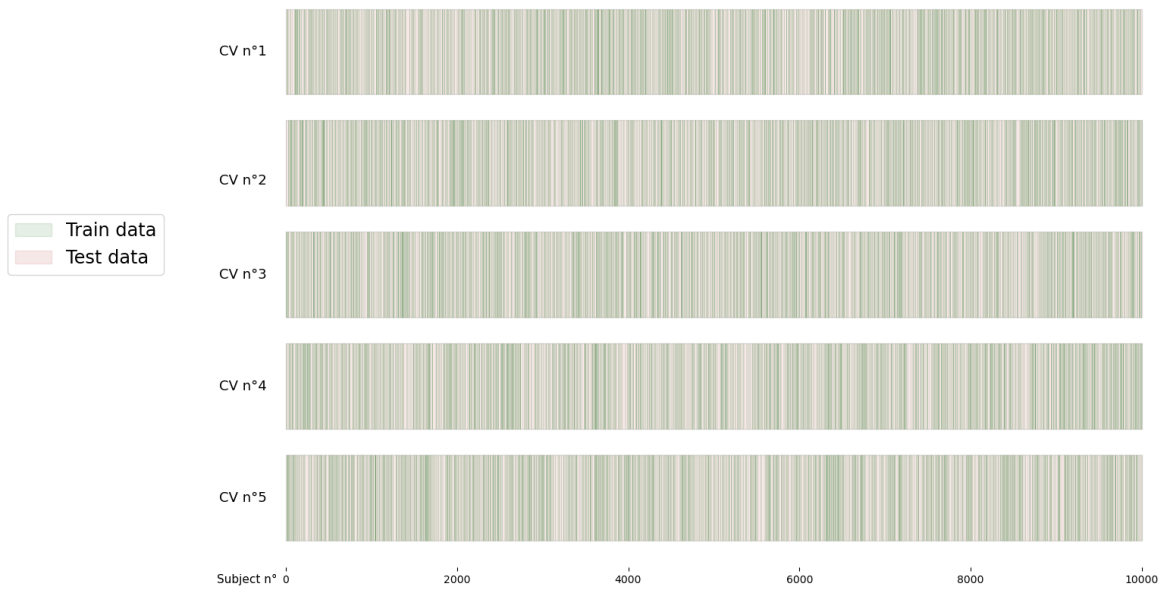


Figure C.1: Monte-Carlo cross-validation

In the following Sections, this will be our methodology for studying the mean and variance of all considered models' performances. All presented conclusions are the results of a 5-fold Monte-Carlo cross-validation.

C.3 Estimation of π_*

Very intuitively, for policyholders linked to a non-active policy, the last observation ended with either lapse or death and $\Delta^{(i)} \neq 0$. For any observation related to a policyholder that eventually lapsed $\pi_*^{(i)} = 1$. For any observation related to a policy that eventually ended with the policyholder's death, we have $\pi_*^{(i)} = 0$. Deriving $\pi_*^{(i)}$ is more complex for policyholders with an active policy where we have

$$\pi_*^{(i)} = P(\Delta_*^{(i)} = 1 | \Delta^{(i)} = 0, \mathcal{X}^{(i)}(T^{(i)})) = \frac{P(\Delta_*^{(i)} = 1, \Delta^{(i)} = 0 | \mathcal{X}^{(i)}(T^{(i)}))}{P(\Delta^{(i)} = 0 | \mathcal{X}^{(i)}(T^{(i)}))}. \quad (\text{C.1})$$

By treating the competing risks within the cause-specific framework, we have that the probability of having an active policy, in other words having survived every cause of events, is the product of the cause-specific probabilities (See Heisey and Patterson 2006). Given the risk profiles that we introduced in Section 8.1, we define $r_{lapses}^{(i)}(t)$ the all-causes survival probability of subject i at time t and $r_{acceptant}^{(i)}(t)$ the death survival probability of subject i at time t . Moreover, in practice, we only have access to a limited history $T_{max} = \max(T^{(i)})$, corresponding to the longest time a policy was ever observed to last. In order to estimate $\pi_*^{(i)}$, we will consider that the ultimate event time $T_*^{(i)}$ is bounded by T . Thus we have

$$\pi_*^{(i)} = \frac{1 - r_{lapses}^{(i)}(T_{max}) / r_{acceptant}^{(i)}(T_{max})}{r_{lapses}^{(i)}(T^{(i)}) / r_{acceptant}^{(i)}(T^{(i)})} = \frac{r_{acceptant}^{(i)}(T^{(i)})}{r_{lapses}^{(i)}(T^{(i)})} \cdot \left(1 - \frac{r_{lapses}^{(i)}(T_{max})}{r_{acceptant}^{(i)}(T_{max})} \right). \quad (\text{C.2})$$

D. Appendix of the third article (chap. 14)

D.1 About cost-complexity pruning

Even though the original cost function of the CART algorithm described by Breiman et al. 1984 is penalized proportionally to its number of leaves n_L , several works on the matter suggest other types of penalty. Barron 1991 shows that applying risk bounds to CART implies a penalty with $\psi(n_L) = \sqrt{n_L}$. In later works, Mansour and McAllester 2000, Nobel 2002, then Scott and Nowak 2002 also showed that risk bounds with a penalty using $\psi(n_L) = \sqrt{n_L}$ can be derived for classification trees whereas penalties proportional to n_L can only be derived in specific cases discussed by Blanchard, Schäfer, and Rozenholc 2004. In summary, square-root penalties appear to have a much stronger theoretical foundation than n_L proportional ones in various contexts, notably for classification tasks.

D.2 Fréchet trees

Another very interesting and general approach is Fréchet trees - and Fréchet forest - by Capitaine et al. 2020. It is a tree-building procedure that allows handling data for which input covariates and the outcome take values in general metric spaces. Concretely, it is designed to handle covariates and outcomes that can be any functions and can be, in particular, functions of time. In this article, they illustrate the prediction ability of Fréchet forests on longitudinal data and the robustness of their method to missing data and time shifts. Several limitations can be pointed out: firstly the mathematical assumption of the existence of the Fréchet mean in the output space must be verified and that mean must be approximated as precisely as possible. Another limitation is the interpretability, as it is always the case with bagging techniques, but here it is also true for individual Fréchet trees: if covariates' importance can be analyzed, relevant threshold and time points can not be easily observed. Eventually, the computational burden of this algorithm is also important. This method has been implemented in the R package `FréchForest`.

D.3 More results

D.3.1 Results without stopping criterion

The maximal unpruned and unstopped TpT, obtained with the time-penalized Gini splitting criterion and an optimal time penalty achieves a depth of 17, has 190 leaves - 173 terminal leaves, 17 duration leaves - and is too large to be fully displayed as a tree here. However, we can still represent it as a list of decisions describing how the dataset is partitioned:

The maximum depth achieved is 17
The number of leaves is 190
173 terminal leaves and 17 duration leaves
The tree impurity is : 0.06270710057403012
The penalized tree impurity is : 0.3556124586066797
The maximum time where a split occurred is 10.0
The average split time is 0.9805922147055561

The tree is:

```

depth = 0 if Age <= 65.5 at t = 0.0, samples: 983
and no duration leaf
  then depth = 1 if Age <= 42.5 at t = 0.0, samples: 463
  and no duration leaf
    then depth = 2 if GENDER <= 1.5 at t = 0.0, samples: 110
    and no duration leaf
      then depth = 3 if Age <= 30.5 at t = 0.0, samples: 58
      and no duration leaf
        then depth = 4{value: CHURNED, samples: 29}
        else depth = 4 if CLV <= 9.16 at t = 0.0, samples: 29
        and no duration leaf
          then depth = 5{value: CHURNED, samples: 9}
          else depth = 5 if FACE_AMOUNT <= 7197.19 at t = 2.0, samples: 20
          and no duration leaf
            then depth = 6 if FACE_AMOUNT <= 3654.22 at t = 2.0, samples: 9
            and no duration leaf
              then depth = 7 if Age <= 39.5 at t = 2.0, samples: 5
              and no duration leaf
                then depth = 8{value: CHURNED, samples: 3}
                else depth = 8{value: CHURNED 0.5, samples: 2}
                else depth = 7{value: DEATH, samples: 4}
              else depth = 6{value: CHURNED, samples: 11}
            else depth = 3{value: CHURNED, samples: 52}
          else depth = 2 if GENDER <= 1.5 at t = 0.0, samples: 353
          and no duration leaf
            then depth = 3 if Age <= 53.5 at t = 0.0, samples: 154
            and no duration leaf
              then depth = 4 if Age <= 52.5 at t = 0.0, samples: 53
              and no duration leaf
                then depth = 5 if CLV <= 13.11 at t = 0.0, samples: 47
                and no duration leaf
                  then depth = 6 if FACE_AMOUNT <= 3196.44 at t = 3.0, samples: 10
                  and duration leaf has 1 samples. Label is: CHURNED 1.0
                  then depth = 7{value: CHURNED, samples: 7}
                  else depth = 7{value: CHURNED 0.5, samples: 2}
                else depth = 6 if CLV <= 88.4 at t = 0.0, samples: 37
                and no duration leaf
                  then depth = 7 if Nb_Contrats <= 1.5 at t = 0.0, samples: 14
                  and no duration leaf
                    then depth = 8 if CLV <= 40.05 at t = 0.0, samples: 12
                    and no duration leaf
                      then depth = 9 if Age <= 45.5 at t = 0.0, samples: 8
                      and no duration leaf
                        then depth = 10{value: DEATH, samples: 3}
                        else depth = 10 if CLV <= 17.03 at t = 0.0, samples: 5
                        and no duration leaf
                          then depth = 11{value: DEATH, samples: 2}
                          else depth = 11{value: CHURNED, samples: 3}
                        else depth = 9{value: DEATH, samples: 4}
                      else depth = 8{value: CHURNED, samples: 2}
                    else depth = 7 if CLV <= 591.46 at t = 0.0, samples: 23
                    and no duration leaf
                      then depth = 8 if CLV <= 352.28 at t = 0.0, samples: 18
                      and no duration leaf
                        then depth = 9 if CLV <= 155.3 at t = 0.0, samples: 16
                        and no duration leaf
                          then depth = 10 if CLV <= 190.4 at t = 1.0, samples: 9
                          and no duration leaf
                            then depth = 11{value: CHURNED 0.5, samples: 2}
                            else depth = 11{value: CHURNED, samples: 7}
                          else depth = 10 if CLV <= 178.0 at t = 0.0, samples: 7
                          and no duration leaf
                            then depth = 11{value: DEATH, samples: 2}
                            else depth = 11 if CLV <= 259.66 at t = 0.0, samples: 5
                            and no duration leaf
                              then depth = 12{value: CHURNED, samples: 3}
                              else depth = 12{value: CHURNED 0.5, samples: 2}
                            else depth = 9{value: DEATH, samples: 2}
                          else depth = 8{value: CHURNED, samples: 5}
                        else depth = 5{value: CHURNED, samples: 6}
                      else depth = 4 if CDI_NOM_PRODUIT <= 1.5 at t = 0.0, samples: 101
                      and no duration leaf
                        then depth = 5 if FACE_AMOUNT <= 10325.88 at t = 4.0, samples: 83
                        and duration leaf has 3 samples. Label is: DEATH 1.0

```

```

then depth = 6 if CLV <= 16.28 at t = 4.0, samples: 41
and no duration leaf
then depth = 7 if Age <= 60.5 at t = 6.0, samples: 8
and duration leaf has 1 samples. Label is: CHURNED 1.0
then depth = 8{value: CHURNED 0.5, samples: 2}
else depth = 8{value: DEATH, samples: 5}
else depth = 7 if CLV <= 89.04 at t = 4.0, samples: 33
and no duration leaf
then depth = 8{value: CHURNED, samples: 8}
else depth = 8 if CLV <= 100.4 at t = 4.0, samples: 25
and no duration leaf
then depth = 9{value: DEATH, samples: 2}
else depth = 9 if CLV <= 181.96 at t = 4.0, samples: 23
and no duration leaf
then depth = 10{value: CHURNED, samples: 5}
else depth = 10 if CLV <= 310.27 at t = 5.0, samples: 18
and no duration leaf
then depth = 11{value: DEATH, samples: 3}
else depth = 11 if Age <= 68.5 at t = 5.0, samples: 15
and no duration leaf
then depth = 12 if FACE_AMOUNT <= 3972.54 at t = 5.0, samples: 11
and no duration leaf
then depth = 13{value: DEATH, samples: 3}
else depth = 13 if CLV <= 748.3 at t = 6.0, samples: 8
and no duration leaf
then depth = 14{value: CHURNED, samples: 4}
else depth = 14 if CLV <= 917.37 at t = 6.0, samples: 4
and no duration leaf
then depth = 15{value: DEATH, samples: 2}
else depth = 15{value: CHURNED, samples: 2}
else depth = 12{value: CHURNED, samples: 4}
else depth = 6 if FACE_AMOUNT <= 17894.43 at t = 4.0, samples: 39
and no duration leaf
then depth = 7{value: DEATH, samples: 8}
else depth = 7 if Age <= 65.5 at t = 5.0, samples: 31
and no duration leaf
then depth = 8 if CLV <= 1745.92 at t = 5.0, samples: 17
and no duration leaf
then depth = 9 if FACE_AMOUNT <= 21616.0 at t = 5.0, samples: 5
and no duration leaf
then depth = 10{value: DEATH, samples: 3}
else depth = 10{value: CHURNED, samples: 2}
else depth = 9 if CLV <= 3172.41 at t = 5.0, samples: 12
and no duration leaf
then depth = 10{value: CHURNED, samples: 6}
else depth = 10 if FACE_AMOUNT <= 64766.89 at t = 6.0, samples: 6
and no duration leaf
then depth = 11{value: DEATH, samples: 3}
else depth = 11{value: CHURNED, samples: 3}
else depth = 8 if FACE_AMOUNT <= 20931.16 at t = 6.0, samples: 14
and duration leaf has 1 samples. Label is: DEATH 1.0
then depth = 9{value: CHURNED, samples: 2}
else depth = 9 if CLV <= 10948.21 at t = 9.0, samples: 11
and duration leaf has 4 samples. Label is: DEATH 1.0
then depth = 10 if CLV <= 5539.86 at t = 9.0, samples: 4
and no duration leaf
then depth = 11{value: DEATH, samples: 2}
else depth = 11{value: CHURNED, samples: 2}
else depth = 10{value: DEATH, samples: 3}
else depth = 5 if CLV <= 3.37 at t = 0.0, samples: 18
and no duration leaf
then depth = 6{value: CHURNED 0.5, samples: 2}
else depth = 6 if CLV <= 21.3 at t = 0.0, samples: 16
and no duration leaf
then depth = 7{value: CHURNED, samples: 6}
else depth = 7 if CLV <= 363.44 at t = 0.0, samples: 10
and no duration leaf
then depth = 8{value: CHURNED 0.5, samples: 4}
else depth = 8 if CLV <= 2068.21 at t = 0.0, samples: 6
and no duration leaf
then depth = 9{value: CHURNED, samples: 4}
else depth = 9{value: CHURNED 0.5, samples: 2}
else depth = 3 if CDI_NOM_PRODUIT <= 1.5 at t = 0.0, samples: 199
and no duration leaf
then depth = 4 if Age <= 60.5 at t = 0.0, samples: 167
and no duration leaf
then depth = 5 if Age <= 43.5 at t = 0.0, samples: 115
and no duration leaf
then depth = 6 if CLV <= 82.12 at t = 0.0, samples: 4
and no duration leaf
then depth = 7{value: DEATH, samples: 2}
else depth = 7{value: CHURNED 0.5, samples: 2}
else depth = 6 if CLV <= 2145.92 at t = 3.0, samples: 111

```

```

and duration leaf has 1 samples. Label is: DEATH 1.0
  then depth = 7 if FACE_AMOUNT <= 28288.52 at t = 3.0, samples: 94
  and no duration leaf
  then depth = 8 if CLV <= 910.67 at t = 3.0, samples: 92
  and no duration leaf
  then depth = 9 if CLV <= 840.53 at t = 3.0, samples: 80
  and no duration leaf
  then depth = 10 if FACE_AMOUNT <= 12512.22 at t = 6.0, samples: 78
  and no duration leaf
  then depth = 11 if CLV <= 217.81 at t = 8.0, samples: 65
  and duration leaf has 4 samples. Label is: CHURNED 0.5
  then depth = 12 if Age <= 66.5 at t = 8.0, samples: 24
  and no duration leaf
  then depth = 13 if Age <= 54.5 at t = 8.0, samples: 20
  and no duration leaf
  then depth = 14 if Age <= 55.5 at t = 10.0, samples: 7
  and duration leaf has 1 samples. Label is: CHURNED

  then depth = 15{value: CHURNED, samples: 3}
  else depth = 15{value: DEATH, samples: 3}
  else depth = 14 if CLV <= 74.87 at t = 8.0, samples: 13
  and no duration leaf
  then depth = 15{value: CHURNED, samples: 10}
  else depth = 15{value: CHURNED 0.67, samples: 3}
else depth = 13 if Age <= 67.5 at t = 8.0, samples: 4
  and no duration leaf
  then depth = 14{value: DEATH, samples: 2}
  else depth = 14{value: CHURNED 0.5, samples: 2}
else depth = 12 if Age <= 53.0 at t = 8.0, samples: 37
  and no duration leaf
  then depth = 13{value: CHURNED 0.5, samples: 2}
  else depth = 13 if FACE_AMOUNT <= 3848.48 at t = 10.0, samples: 35
  and duration leaf has 5 samples. Label is: CHURNED

  then depth = 14 if FACE_AMOUNT <= 3086.41 at t = 10.0, samples: 8
  and no duration leaf
  then depth = 15{value: CHURNED, samples: 6}
  else depth = 15{value: CHURNED 0.5, samples: 2}
  else depth = 14{value: CHURNED, samples: 22}
else depth = 11 if Age <= 60.5 at t = 7.0, samples: 13
  and duration leaf has 1 samples. Label is: DEATH 1.0
  then depth = 12 if FACE_AMOUNT <= 16037.28 at t = 8.0, samples: 8
  and no duration leaf
  then depth = 13{value: DEATH, samples: 3}
  else depth = 13 if Age <= 57.5 at t = 8.0, samples: 5
  and no duration leaf
  then depth = 14{value: DEATH 0.67, samples: 3}
  else depth = 14{value: CHURNED, samples: 2}
  else depth = 12{value: CHURNED, samples: 4}
  else depth = 10{value: DEATH, samples: 2}
  else depth = 9{value: CHURNED, samples: 12}
  else depth = 8{value: DEATH, samples: 2}
  else depth = 7{value: CHURNED, samples: 16}
else depth = 5 if Age <= 64.5 at t = 0.0, samples: 52
  and no duration leaf
  then depth = 6 if CLV <= 251.48 at t = 0.0, samples: 43
  and no duration leaf
  then depth = 7 if FACE_AMOUNT <= 73.34 at t = 3.0, samples: 29
  and no duration leaf
  then depth = 8{value: DEATH, samples: 3}
  else depth = 8 if CLV <= 119.45 at t = 3.0, samples: 26
  and no duration leaf
  then depth = 9 if CLV <= 54.46 at t = 4.0, samples: 11
  and no duration leaf
  then depth = 10{value: CHURNED 0.67, samples: 3}
  else depth = 10{value: CHURNED, samples: 8}
  else depth = 9 if FACE_AMOUNT <= 3331.54 at t = 3.0, samples: 15
  and no duration leaf
  then depth = 10{value: DEATH, samples: 2}
  else depth = 10 if CLV <= 445.55 at t = 3.0, samples: 13
  and no duration leaf
  then depth = 11{value: CHURNED, samples: 3}
  else depth = 11 if CLV <= 709.34 at t = 3.0, samples: 10
  and no duration leaf
  then depth = 12 if CLV <= 622.04 at t = 3.0, samples: 4
  and no duration leaf
  then depth = 13{value: CHURNED 0.5, samples: 2}
  else depth = 13{value: DEATH, samples: 2}
  else depth = 12 if FACE_AMOUNT <= 16863.76 at t = 3.0, samples: 6
  and no duration leaf
  then depth = 13{value: CHURNED, samples: 3}
  else depth = 13{value: DEATH 0.67, samples: 3}
else depth = 7 if Nb_Contrats <= 1.5 at t = 0.0, samples: 14

```

```

        and no duration leaf
        then depth = 8 if FACE_AMOUNT <= 22229.94 at t = 1.0, samples: 11
        and no duration leaf
            then depth = 9{value: DEATH 0.67, samples: 3}
            else depth = 9{value: DEATH, samples: 8}
            else depth = 8{value: CHURNED, samples: 3}
    else depth = 6 if CLV <= 633.18 at t = 0.0, samples: 9
    and no duration leaf
        then depth = 7{value: CHURNED, samples: 7}
        else depth = 7{value: CHURNED 0.5, samples: 2}
else depth = 4 if CLV <= 2081.39 at t = 0.0, samples: 32
and no duration leaf
    then depth = 5 if Age <= 63.5 at t = 0.0, samples: 30
    and no duration leaf
        then depth = 6 if FACE_AMOUNT <= 37433.94 at t = 3.0, samples: 26
        and duration leaf has 2 samples. Label is: CHURNED 1.0
        then depth = 7{value: CHURNED, samples: 21}
        else depth = 7{value: CHURNED 0.67, samples: 3}
    else depth = 6 if Age <= 64.5 at t = 0.0, samples: 4
    and no duration leaf
        then depth = 7{value: CHURNED 0.5, samples: 2}
        else depth = 7{value: CHURNED, samples: 2}
    else depth = 5{value: CHURNED 0.5, samples: 2}
else depth = 1 if Age <= 72.5 at t = 0.0, samples: 520
and no duration leaf
    then depth = 2 if CLV <= 2.73 at t = 0.0, samples: 180
    and no duration leaf
        then depth = 3 if CLV <= 99.7 at t = 2.0, samples: 13
        and no duration leaf
            then depth = 4 if CLV <= 11.97 at t = 3.0, samples: 11
            and no duration leaf
                then depth = 5{value: CHURNED, samples: 9}
                else depth = 5{value: CHURNED 0.5, samples: 2}
            else depth = 4{value: DEATH, samples: 2}
    else depth = 3 if CLV <= 2982.24 at t = 0.0, samples: 167
    and no duration leaf
        then depth = 4 if Nb_Contrats <= 2.5 at t = 0.0, samples: 165
        and no duration leaf
            then depth = 5 if CDI_NOM_PRODUIT <= 1.5 at t = 0.0, samples: 159
            and no duration leaf
                then depth = 6 if CLV <= 153.51 at t = 0.0, samples: 146
                and no duration leaf
                    then depth = 7 if Nb_Contrats <= 1.5 at t = 1.0, samples: 61
                    and no duration leaf
                        then depth = 8 if Age <= 70.5 at t = 1.0, samples: 58
                        and no duration leaf
                            then depth = 9 if GENDER <= 1.5 at t = 1.0, samples: 30
                            and no duration leaf
                                then depth = 10 if CLV <= 152.49 at t = 1.0, samples: 17
                                and no duration leaf
                                    then depth = 11{value: DEATH, samples: 10}
                                    else depth = 11 if CLV <= 212.33 at t = 1.0, samples: 7
                                    and no duration leaf
                                        then depth = 12{value: CHURNED 0.5, samples: 2}
                                        else depth = 12{value: DEATH, samples: 5}
                                else depth = 10 if FACE_AMOUNT <= 3475.12 at t = 1.0, samples: 13
                                and no duration leaf
                                    then depth = 11 if FACE_AMOUNT <= 1499.92 at t = 1.0, samples: 7
                                    and no duration leaf
                                        then depth = 12{value: DEATH, samples: 2}
                                        else depth = 12 if Age <= 69.5 at t = 1.0, samples: 5
                                        and no duration leaf
                                            then depth = 13{value: CHURNED, samples: 3}
                                            else depth = 13{value: CHURNED 0.5, samples: 2}
                                    else depth = 11 if CLV <= 195.51 at t = 1.0, samples: 6
                                    and no duration leaf
                                        then depth = 12{value: DEATH, samples: 4}
                                        else depth = 12{value: CHURNED 0.5, samples: 2}
                                else depth = 9 if FACE_AMOUNT <= 2307.93 at t = 1.0, samples: 28
                                and no duration leaf
                                    then depth = 10 if CLV <= 35.73 at t = 1.0, samples: 6
                                    and no duration leaf
                                        then depth = 11{value: DEATH, samples: 4}
                                        else depth = 11{value: CHURNED 0.5, samples: 2}
                                    else depth = 10{value: DEATH, samples: 22}
                                else depth = 8{value: CHURNED 0.67, samples: 3}
                            else depth = 7 if CLV <= 161.62 at t = 0.0, samples: 85
                            and no duration leaf
                                then depth = 8{value: CHURNED, samples: 2}
                                else depth = 8 if CLV <= 1185.78 at t = 0.0, samples: 83
                                and no duration leaf
                                    then depth = 9 if CLV <= 1072.6 at t = 0.0, samples: 69
                                    and no duration leaf

```

```

then depth = 10 if Nb_Contrats <= 1.5 at t = 0.0, samples: 67
  and no duration leaf
  then depth = 11 if CLV <= 296.66 at t = 0.0, samples: 61
    and no duration leaf
    then depth = 12 if CLV <= 240.56 at t = 0.0, samples: 22
      and no duration leaf
      then depth = 13 if CLV <= 396.72 at t = 1.0, samples: 14
        and no duration leaf
        then depth = 14 if CLV <= 342.37 at t = 1.0, samples: 7
          and no duration leaf
          then depth = 15{value: CHURNED 0.5, samples: 2}
            else depth = 15{value: DEATH, samples: 5}
          else depth = 14 if CLV <= 428.3 at t = 1.0, samples: 7
            and no duration leaf
            then depth = 15{value: CHURNED, samples: 3}
              else depth = 15 if GENDER <= 1.5 at t = 1.0, samples: 4
                and no duration leaf
                then depth = 16{value: DEATH, samples: 2}
                  else depth = 16{value: CHURNED 0.5, samples: 2}
                else depth = 13{value: DEATH, samples: 8}
              else depth = 12 if CLV <= 522.24 at t = 0.0, samples: 39
                and no duration leaf
                then depth = 13 if CLV <= 388.65 at t = 0.0, samples: 24
                  and no duration leaf
                  then depth = 14 if Age <= 70.5 at t = 0.0, samples: 12
                    and no duration leaf
                    then depth = 15 if CLV <= 308.23 at t = 0.0, samples: 8
                      and no duration leaf
                      then depth = 16{value: CHURNED 0.5, samples: 2}
                        else depth = 16{value: DEATH, samples: 6}
                      else depth = 15 if CLV <= 320.33 at t = 0.0, samples: 4
                        and no duration leaf
                        then depth = 16{value: CHURNED, samples: 2}
                          else depth = 16{value: CHURNED 0.5, samples: 2}
                        else depth = 14 if CLV <= 427.18 at t = 0.0, samples: 12
                          and no duration leaf
                          then depth = 15{value: CHURNED, samples: 4}
                            else depth = 15 if CLV <= 507.19 at t = 0.0, samples: 8
                              and no duration leaf
                              then depth = 16 if GENDER <= 1.5 at t = 0.0, samples: 6
                                and no duration leaf
                                then depth = 17{value: DEATH, samples: 3}
                                  else depth = 17{value: CHURNED 0.67, samples: 3}
                                  else depth = 16{value: CHURNED, samples: 2}
                                else depth = 13 if CLV <= 735.13 at t = 0.0, samples: 15
                                  and no duration leaf
                                  then depth = 14{value: DEATH, samples: 8}
                                    else depth = 14 if CLV <= 767.92 at t = 0.0, samples: 7
                                      and no duration leaf
                                      then depth = 15{value: CHURNED, samples: 2}
                                        else depth = 15 if FACE_AMOUNT <= 47527.88 at t = 1.0, samples: 2
                                          and no duration leaf
                                          then depth = 16{value: DEATH, samples: 3}
                                            else depth = 16{value: CHURNED 0.5, samples: 2}
                                          else depth = 11{value: DEATH, samples: 6}
                                        else depth = 10{value: CHURNED, samples: 2}
                                      else depth = 9 if Age <= 71.0 at t = 0.0, samples: 14
                                        and no duration leaf
                                        then depth = 10{value: DEATH, samples: 11}
                                          else depth = 10{value: DEATH 0.67, samples: 3}
                                        else depth = 6 if FACE_AMOUNT <= 7905.44 at t = 2.0, samples: 13
                                          and no duration leaf
                                          then depth = 7 if FACE_AMOUNT <= 1385.41 at t = 3.0, samples: 8
                                            and duration leaf has 2 samples. Label is: CHURNED 1.0
                                            then depth = 8{value: DEATH, samples: 2}
                                              else depth = 8{value: CHURNED, samples: 4}
                                              else depth = 7{value: DEATH, samples: 5}
                                            else depth = 5 if Nb_Contrats <= 4.5 at t = 1.0, samples: 6
                                              and no duration leaf
                                              then depth = 6{value: CHURNED, samples: 4}
                                                else depth = 6{value: DEATH, samples: 2}
                                              else depth = 4{value: CHURNED, samples: 2}
                                            else depth = 2 if CLV <= 24.19 at t = 0.0, samples: 340
                                              and no duration leaf
                                              then depth = 3 if CLV <= 23.77 at t = 0.0, samples: 70
                                                and no duration leaf
                                                then depth = 4 if Age <= 81.5 at t = 0.0, samples: 68
                                                  and no duration leaf
                                                  then depth = 5 if Age <= 76.5 at t = 0.0, samples: 53
                                                    and no duration leaf
                                                    then depth = 6 if CDI_NOM_PRODUIIT <= 1.5 at t = 0.0, samples: 32
                                                      and no duration leaf
                                                      then depth = 7 if CLV <= 1.72 at t = 0.0, samples: 24

```

```

and no duration leaf
  then depth = 8 if CLV <= 2.86 at t = 1.0, samples: 7
  and no duration leaf
    then depth = 9{value: DEATH, samples: 3}
    else depth = 9{value: CHURNED 0.5, samples: 4}
  else depth = 8{value: DEATH, samples: 17}
else depth = 7 if GENDER <= 1.5 at t = 4.0, samples: 8
  and duration leaf has 2 samples. Label is: CHURNED 0.5
  then depth = 8{value: CHURNED, samples: 2}
  else depth = 8 if CLV <= 56.71 at t = 4.0, samples: 4
  and no duration leaf
    then depth = 9{value: CHURNED 0.5, samples: 2}
    else depth = 9{value: DEATH, samples: 2}
else depth = 6 if CLV <= 1.5 at t = 0.0, samples: 21
  and no duration leaf
  then depth = 7{value: CHURNED, samples: 3}
  else depth = 7 if CLV <= 101.49 at t = 3.0, samples: 18
  and duration leaf has 1 samples. Label is: DEATH 1.0
  then depth = 8 if Age <= 79.5 at t = 3.0, samples: 13
  and no duration leaf
  then depth = 9{value: CHURNED, samples: 2}
  else depth = 9 if GENDER <= 1.5 at t = 3.0, samples: 11
  and no duration leaf
  then depth = 10 if CLV <= 24.93 at t = 3.0, samples: 4
  and no duration leaf
  then depth = 11{value: DEATH, samples: 2}
  else depth = 11{value: CHURNED, samples: 2}
  else depth = 10 if Age <= 80.5 at t = 3.0, samples: 7
  and no duration leaf
  then depth = 11{value: DEATH 0.67, samples: 3}
  else depth = 11{value: DEATH, samples: 4}
  else depth = 8{value: CHURNED, samples: 4}
  else depth = 5{value: DEATH, samples: 15}
else depth = 4{value: CHURNED, samples: 2}
else depth = 3 if CDL_NOM_PRODUIT <= 1.5 at t = 0.0, samples: 270
  and no duration leaf
  then depth = 4 if Age <= 74.5 at t = 0.0, samples: 240
  and no duration leaf
  then depth = 5 if CLV <= 303.99 at t = 0.0, samples: 41
  and no duration leaf
  then depth = 6 if Age <= 73.5 at t = 0.0, samples: 23
  and no duration leaf
  then depth = 7 if CLV <= 391.43 at t = 2.0, samples: 6
  and no duration leaf
  then depth = 8{value: CHURNED 0.5, samples: 2}
  else depth = 8{value: DEATH, samples: 4}
  else depth = 7{value: DEATH, samples: 17}
  else depth = 6 if CLV <= 334.97 at t = 0.0, samples: 18
  and no duration leaf
  then depth = 7{value: CHURNED, samples: 3}
  else depth = 7 if CLV <= 1380.47 at t = 0.0, samples: 15
  and no duration leaf
  then depth = 8{value: DEATH, samples: 13}
  else depth = 8{value: CHURNED 0.5, samples: 2}
else depth = 5 if Age <= 89.5 at t = 0.0, samples: 199
  and no duration leaf
  then depth = 6 if FACE_AMOUNT <= 65229.84 at t = 3.0, samples: 192
  and duration leaf has 2 samples. Label is: DEATH 1.0
  then depth = 7 if FACE_AMOUNT <= 5858.16 at t = 3.0, samples: 160
  and no duration leaf
  then depth = 8 if FACE_AMOUNT <= 5693.4 at t = 4.0, samples: 27
  and duration leaf has 3 samples. Label is: DEATH 1.0
  then depth = 9{value: DEATH, samples: 22}
  else depth = 9{value: CHURNED, samples: 2}
  else depth = 8 if Age <= 78.5 at t = 3.0, samples: 133
  and no duration leaf
  then depth = 9 if FACE_AMOUNT <= 14620.96 at t = 3.0, samples: 13
  and no duration leaf
  then depth = 10 if CLV <= 743.21 at t = 3.0, samples: 5
  and no duration leaf
  then depth = 11{value: DEATH, samples: 3}
  else depth = 11{value: CHURNED 0.5, samples: 2}
  else depth = 10{value: DEATH, samples: 8}
  else depth = 9 if Age <= 81.5 at t = 3.0, samples: 120
  and no duration leaf
  then depth = 10 if Age <= 80.5 at t = 3.0, samples: 49
  and no duration leaf
  then depth = 11{value: DEATH, samples: 33}
  else depth = 11 if CLV <= 1337.19 at t = 3.0, samples: 16
  and no duration leaf
  then depth = 12 if CLV <= 1236.12 at t = 3.0, samples: 7
  and no duration leaf
  then depth = 13{value: DEATH, samples: 5}

```

```

else depth = 13{value: CHURNED 0.5, samples: 2}
else depth = 12{value: DEATH, samples: 9}
else depth = 10{value: DEATH, samples: 71}
else depth = 7 if Age <= 79.5 at t = 3.0, samples: 30
and no duration leaf
then depth = 8 if CLV <= 6469.44 at t = 3.0, samples: 6
and no duration leaf
then depth = 9{value: CHURNED, samples: 2}
else depth = 9{value: DEATH, samples: 4}
else depth = 8 if CLV <= 4697.78 at t = 4.0, samples: 24
and duration leaf has 2 samples. Label is: DEATH 1.0
then depth = 9{value: CHURNED 0.5, samples: 2}
else depth = 9{value: DEATH, samples: 20}
else depth = 6 if GENDER <= 1.5 at t = 0.0, samples: 7
and no duration leaf
then depth = 7{value: CHURNED 0.5, samples: 2}
else depth = 7{value: DEATH, samples: 5}
else depth = 4 if Age <= 80.5 at t = 0.0, samples: 30
and no duration leaf
then depth = 5 if GENDER <= 1.5 at t = 0.0, samples: 11
and no duration leaf
then depth = 6 if Age <= 75.5 at t = 0.0, samples: 4
and no duration leaf
then depth = 7{value: CHURNED 0.5, samples: 2}
else depth = 7{value: DEATH, samples: 2}
else depth = 6 if Age <= 79.5 at t = 0.0, samples: 7
and no duration leaf
then depth = 7{value: CHURNED, samples: 4}
else depth = 7{value: CHURNED 0.67, samples: 3}
else depth = 5{value: DEATH, samples: 19}

```

The unstopped and unpruned TpTs, obtained with the time-penalized gini splitting criterion, and various time penalties yields the following results:

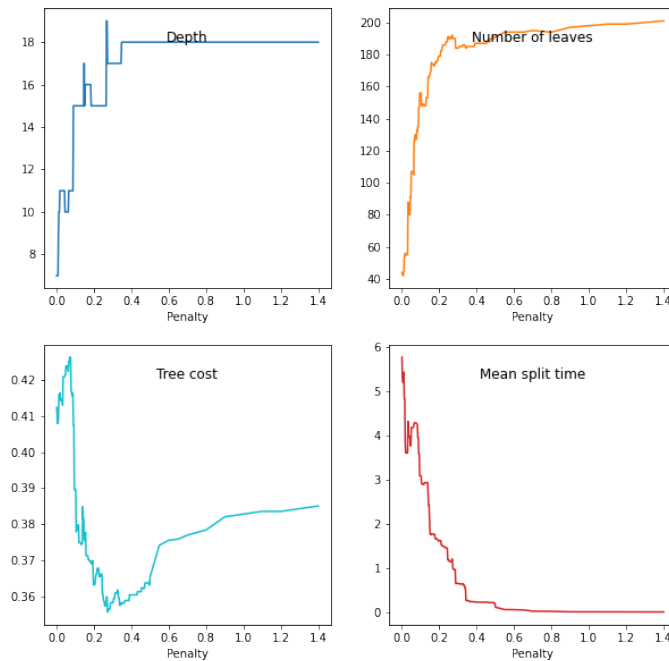


Figure D.1: Characteristics of unstopped and unpruned TpT depending on the time penalty

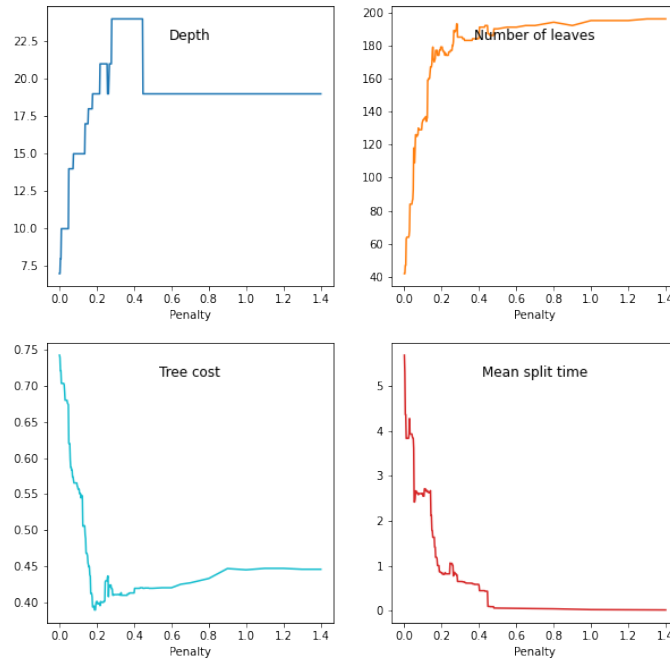
The unstopped and unpruned TpTs, obtained with the time-penalized entropy splitting criterion, and various time penalties yields the following results:

Time penalty γ	Runtime	Depth	# of terminal leaves	# of duration leaves	Total # of leaves	Tree cost	Max of split times	Mean of split times
0.0000	719.07	7	27	17	44	0.412	20.0	5.768
0.0025	733.95	7	27	16	43	0.411	20.0	5.276
0.0050	732.73	7	27	15	42	0.408	20.0	5.192
0.0100	725.95	9	30	15	45	0.410	20.0	5.429
0.0125	795.26	10	35	19	54	0.416	20.0	4.832
0.0175	862.99	11	35	21	56	0.416	20.0	3.81
0.0200	883.2	11	35	20	55	0.414	20.0	3.599
0.0300	868.98	11	35	20	55	0.415	20.0	3.601
0.0325	1007.65	11	55	33	88	0.413	20.0	4.315
0.0350	970.33	11	53	27	80	0.421	20.0	3.977
0.0450	1046.33	10	61	30	91	0.421	20.0	3.763
0.0500	1062.24	10	71	36	107	0.424	20.0	4.175
0.0625	1069.06	10	69	36	105	0.422	20.0	4.209
0.0650	1141.95	11	79	47	126	0.425	18.0	4.277
0.0700	1137.4	11	81	49	130	0.426	18.0	4.282
0.0775	1117.22	11	83	44	127	0.417	18.0	4.272
0.0800	1122.76	11	87	46	133	0.417	18.0	4.26
0.0825	1133.39	11	87	45	132	0.416	18.0	4.238
0.0850	1165.53	11	89	46	135	0.416	18.0	3.947
0.0900	1287.23	15	98	49	147	0.408	18.0	3.579
0.0950	1350.29	15	105	51	156	0.390	18.0	3.085
0.1025	1340.77	15	106	50	156	0.389	18.0	3.083
0.1050	1346.53	15	104	44	148	0.378	18.0	2.906
0.1075	1349.57	15	104	44	148	0.379	18.0	2.898
0.1125	1351.12	15	105	44	149	0.380	18.0	2.888
0.1200	1347.56	15	105	43	148	0.375	18.0	2.929
0.1300	1378.96	15	112	41	153	0.374	16.0	2.929
0.1400	1561.86	15	118	48	166	0.385	16.0	2.645
0.1425	1559.3	15	122	44	166	0.381	16.0	2.416
0.1475	1685.14	17	124	43	167	0.377	16.0	2.129
0.1500	1712.25	15	128	40	168	0.375	15.0	1.778
0.1525	1677.49	16	130	39	169	0.377	15.0	1.76
0.1550	1709.18	16	131	39	170	0.378	15.0	1.749
0.1575	1692.19	16	137	38	175	0.371	14.0	1.773
0.1700	1686.58	16	137	36	173	0.370	14.0	1.762
0.1775	1687.16	16	140	35	175	0.370	13.0	1.663
0.1850	1695.6	15	143	33	176	0.369	13.0	1.647
0.1925	1687.49	15	144	33	177	0.370	13.0	1.64
0.1950	1704.68	15	146	33	179	0.367	13.0	1.608
0.1975	1704.85	15	146	33	179	0.363	13.0	1.618
0.2075	1679.89	15	149	33	182	0.364	12.0	1.617
0.2100	1721.42	15	150	32	182	0.366	12.0	1.521
0.2125	1741.25	15	152	31	183	0.366	12.0	1.519
0.2150	1740.3	15	155	29	184	0.367	12.0	1.502
0.2175	1746.67	15	157	29	186	0.368	12.0	1.492
0.2275	1730.43	15	158	28	186	0.365	12.0	1.465
0.2350	1717.29	15	160	28	188	0.366	12.0	1.457
0.2375	1724.14	15	161	27	188	0.366	12.0	1.455
0.2500	1796.99	15	165	25	190	0.359	14.0	1.184
0.2525	1809.4	15	168	23	191	0.359	13.0	1.158
0.2550	1810.04	15	169	22	191	0.359	13.0	1.143
0.2575	1832.64	15	169	21	190	0.357	13.0	1.142
0.2675	1854.5	19	169	23	192	0.360	11.0	1.198
0.2725	1850.47	17	173	17	190	0.356	10.0	0.981
0.2775	1857.1	17	174	16	190	0.356	10.0	0.965
0.2875	2025.16	17	173	11	184	0.358	10.0	0.645
0.2900	2045.07	17	174	10	184	0.358	10.0	0.648
0.3050	1976.4	17	175	10	185	0.359	10.0	0.639
0.3125	1969.5	17	176	9	185	0.361	10.0	0.635
0.3250	1960.95	17	176	10	186	0.362	10.0	0.64
0.3300	1949.79	17	176	10	186	0.360	10.0	0.606
0.3325	1976.88	17	177	9	186	0.359	10.0	0.604
0.3375	1965.69	17	176	9	185	0.357	11.0	0.534
0.3425	1953.18	17	179	5	184	0.358	6.0	0.267
0.3475	1947.54	18	181	4	185	0.358	5.0	0.265
0.3650	1962.15	18	181	4	185	0.359	5.0	0.236
0.3900	1918.64	18	183	4	187	0.360	5.0	0.226
0.4325	1809.19	18	183	4	187	0.361	5.0	0.223
0.4625	1683.22	18	185	3	188	0.362	5.0	0.216
0.4725	1651.89	18	187	3	190	0.364	5.0	0.213
0.4950	1620.57	18	188	3	191	0.363	5.0	0.199
0.5000	1626.91	18	189	2	191	0.365	5.0	0.111
0.5500	1636.21	18	193	1	194	0.374	5.0	0.057
0.6000	1629.73	18	193	1	194	0.375	5.0	0.055
0.7000	1572.03	18	194	1	195	0.377	5.0	0.021
0.8000	1566.33	18	194	0	194	0.378	5.0	0.016
0.9000	1571.42	18	197	0	197	0.382	5.0	0.007
1.0000	1575.33	18	198	0	198	0.383	5.0	0.006
1.1000	1549.67	18	199	0	199	0.384	5.0	0.005
1.3000	1509.43	18	200	0	200	0.384	5.0	0.004
1.4000	1511.22	18	201	0	201	0.385	5.0	0.003

Table D.1: Characteristics of TpT depending on the time penalty

Time penalty γ	Runtime	Depth	# of terminal leaves	# of duration leaves	Total # of leaves	Tree cost	Max of split times	Mean of split times
0.0000	671.79	7	26	16	42	0.743	20.0	5.689
0.0025	703.08	7	27	15	42	0.740	20.0	5.306
0.0050	796.89	8	30	17	47	0.721	20.0	4.366
0.0100	939.1	10	40	23	63	0.704	20.0	3.835
0.0125	937.04	10	40	24	64	0.704	20.0	3.837
0.0250	940.39	10	41	24	65	0.702	20.0	3.933
0.0275	944.95	10	43	25	68	0.692	20.0	4.277
0.0300	1034.06	10	53	31	84	0.680	20.0	3.933
0.0425	1054.6	10	54	32	86	0.677	20.0	3.875
0.0450	1058.25	10	55	32	87	0.675	20.0	3.845
0.0475	1056.82	10	59	34	93	0.674	20.0	3.84
0.0500	1246.9	14	73	45	118	0.620	19.0	3.613
0.0525	1354.92	14	69	40	109	0.620	20.0	2.419
0.0575	1383.59	14	77	41	118	0.598	20.0	2.484
0.0600	1389.81	14	82	44	126	0.587	19.0	2.666
0.0650	1383.68	14	82	43	125	0.583	19.0	2.639
0.0700	1379.63	14	83	43	126	0.574	19.0	2.605
0.0750	1427.5	15	86	44	130	0.569	19.0	2.581
0.0775	1427.62	15	87	42	129	0.566	19.0	2.606
0.0950	1421.46	15	91	42	133	0.561	19.0	2.591
0.0975	1421.71	15	92	42	134	0.560	19.0	2.606
0.1000	1420.78	15	93	42	135	0.557	19.0	2.549
0.1075	1447.24	15	95	41	136	0.551	19.0	2.712
0.1150	1438.93	15	97	40	137	0.545	19.0	2.688
0.1175	1447.51	15	97	37	134	0.549	19.0	2.657
0.1200	1458.82	15	98	38	136	0.547	19.0	2.674
0.1250	1502.87	15	114	45	159	0.506	18.0	2.643
0.1300	1504.6	15	115	45	160	0.506	18.0	2.628
0.1375	1504.56	17	119	45	164	0.494	15.0	2.647
0.1400	1647.49	17	121	46	167	0.488	15.0	2.672
0.1425	1724.77	17	124	43	167	0.468	15.0	2.119
0.1475	1750.56	17	125	44	169	0.467	17.0	1.787
0.1500	1756.44	17	132	44	176	0.464	17.0	1.78
0.1525	1766.5	17	135	44	179	0.455	15.0	1.63
0.1550	1791.99	18	135	37	172	0.452	15.0	1.633
0.1575	1798.21	18	136	35	171	0.449	15.0	1.638
0.1600	1800.5	18	136	34	170	0.450	15.0	1.636
0.1625	1844.87	18	143	28	171	0.438	15.0	1.415
0.1650	1825.93	18	143	29	172	0.436	15.0	1.416
0.1675	1884.66	18	154	23	177	0.412	11.0	1.187
0.1750	1889.17	18	155	22	177	0.407	11.0	1.158
0.1775	1910.45	19	156	18	174	0.394	11.0	1.003
0.1825	1901.92	19	157	17	174	0.395	11.0	1.003
0.1875	1940.97	19	164	13	177	0.390	11.0	0.865
0.1950	1938.27	19	166	13	179	0.398	11.0	0.846
0.1975	1909.02	19	166	13	179	0.401	11.0	0.843
0.2050	1931.62	19	165	12	177	0.398	11.0	0.809
0.2100	1921.77	19	165	11	176	0.399	11.0	0.807
0.2175	1938.66	21	165	9	174	0.395	12.0	0.844
0.2200	1940.89	21	166	9	175	0.401	12.0	0.828
0.2250	1946.0	21	166	8	174	0.400	12.0	0.818
0.2375	1941.67	21	167	9	176	0.402	12.0	0.819
0.2450	1939.31	21	160	16	176	0.428	14.0	1.059
0.2475	1917.16	21	161	16	177	0.430	14.0	1.05
0.2575	1920.18	19	162	18	180	0.435	14.0	1.013
0.2600	1916.36	19	163	17	180	0.437	14.0	1.007
0.2625	1968.98	19	172	10	182	0.408	11.0	0.771
0.2650	1965.17	20	172	17	189	0.422	13.0	0.849
0.2675	1969.52	21	173	16	189	0.423	13.0	0.85
0.2700	1960.23	21	174	14	188	0.421	13.0	0.85
0.2725	1997.83	21	175	14	189	0.424	13.0	0.817
0.2775	1977.08	21	176	13	189	0.420	13.0	0.816
0.2800	2194.84	24	182	11	193	0.419	13.0	0.791
0.2850	2167.05	24	179	6	185	0.410	7.0	0.649
0.2925	2138.61	24	179	6	185	0.411	7.0	0.647
0.3125	2115.89	24	179	5	184	0.411	7.0	0.641
0.3225	2088.42	24	178	5	183	0.410	7.0	0.626
0.3300	2076.5	24	178	5	183	0.413	7.0	0.612
0.3650	2027.6	24	180	4	184	0.412	7.0	0.596
0.3750	2051.34	24	180	4	184	0.413	7.0	0.584
0.4000	2079.19	24	181	4	185	0.413	7.0	0.56
0.4025	2015.83	24	186	5	191	0.419	7.0	0.444
0.4325	1864.67	24	187	5	192	0.420	7.0	0.431
0.4475	1757.62	19	182	5	187	0.420	7.0	0.095
0.4575	1731.83	19	183	3	186	0.420	4.0	0.092
0.4800	1577.62	19	187	3	190	0.420	4.0	0.057
0.5500	1577.43	19	188	3	191	0.420	4.0	0.053
0.6500	1577.71	19	189	3	192	0.425	4.0	0.048
0.7000	1573.9	19	190	2	192	0.427	4.0	0.045
0.8000	1578.4	19	193	1	194	0.433	4.0	0.04
0.9000	1589.0	19	192	0	192	0.447	3.0	0.032
1.0000	1592.38	19	195	0	195	0.445	3.0	0.022
1.1000	1576.86	19	195	0	195	0.447	2.0	0.02
1.3000	1576.14	19	196	0	196	0.446	2.0	0.015

Table D.2: Characteristics of unstopped and unpruned entropy TpTs depending on the time penalty



D.3.2 Results with `minsplit = 25`

The maximal unpruned and unstopped TpTs, obtained with the time-penalized entropy splitting criterion, `minsplit = 25`, and various time penalties yields the following results:

Time penalty γ	Runtime	Depth	# of terminal leaves	# of duration leaves	Total # of leaves	Tree cost	Max of split times	Mean of split times
0.0000	668.38	5	14	8	22	0.664	15.0	5.722
0.0025	690.38	5	14	7	21	0.664	15.0	5.172
0.0050	782.22	6	17	8	25	0.638	15.0	4.346
0.0100	840.15	7	16	8	24	0.646	15.0	3.552
0.0150	914.61	8	17	8	25	0.637	15.0	3.301
0.0250	912.15	8	18	9	27	0.636	15.0	3.307
0.0275	917.44	8	18	7	25	0.640	15.0	3.033
0.0300	978.72	8	18	8	26	0.637	15.0	2.347
0.0400	1085.14	8	25	10	35	0.618	15.0	1.953
0.0475	1102.22	8	27	12	39	0.613	15.0	1.959
0.0500	1216.25	8	31	14	45	0.586	9.0	1.578
0.0575	1213.44	8	31	15	46	0.584	9.0	1.568
0.0675	1213.17	8	31	13	44	0.585	9.0	1.5
0.0850	1205.48	9	31	15	46	0.581	10.0	1.52
0.1000	1196.86	9	32	14	46	0.571	10.0	1.434
0.1125	1194.82	9	32	13	45	0.569	10.0	1.418
0.1175	1195.85	8	30	11	41	0.581	11.0	1.244
0.1200	1253.33	11	33	9	42	0.562	9.0	0.823
0.1350	1242.11	11	33	9	42	0.561	9.0	0.807
0.1625	1345.51	11	34	9	43	0.562	9.0	0.714
0.1700	1321.99	11	34	8	42	0.562	9.0	0.708
0.1925	1288.5	11	34	9	43	0.563	9.0	0.689
0.1950	1279.26	11	34	8	42	0.567	9.0	0.651
0.2000	1304.58	11	34	5	39	0.565	9.0	0.38
0.2050	1323.32	11	34	4	38	0.563	8.0	0.316
0.2400	1333.81	11	34	3	37	0.567	6.0	0.275
0.2525	1340.74	11	33	3	36	0.568	6.0	0.242
0.3150	1358.63	11	32	1	33	0.570	3.0	0.173
0.3725	1310.12	11	32	0	32	0.574	2.0	0.045
0.3750	1350.37	11	32	0	32	0.575	2.0	0.014

Table D.3: Characteristics of Entropy TpTs (`minsplit=25`) depending on the time penalty

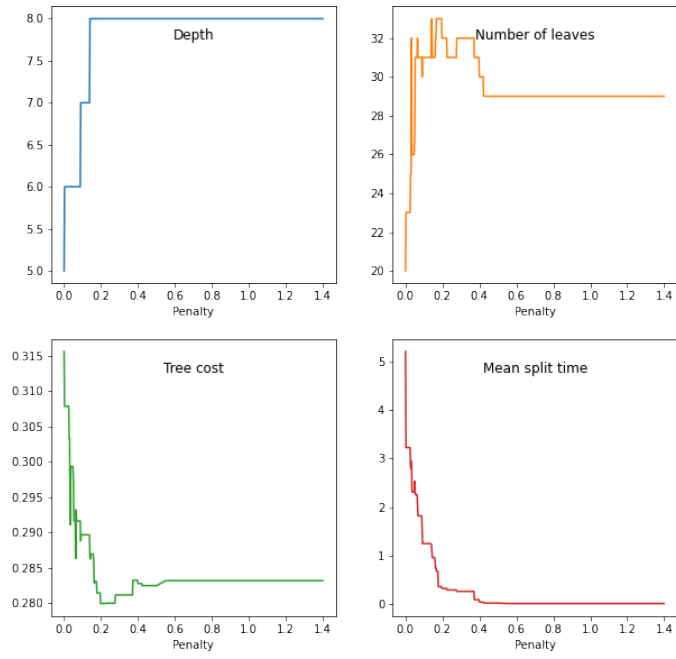


Figure D.2: Characteristics of Entropy TpTs (`minsplit=25`) depending on the time penalty

D.3.3 Results with `minsplit = 50`

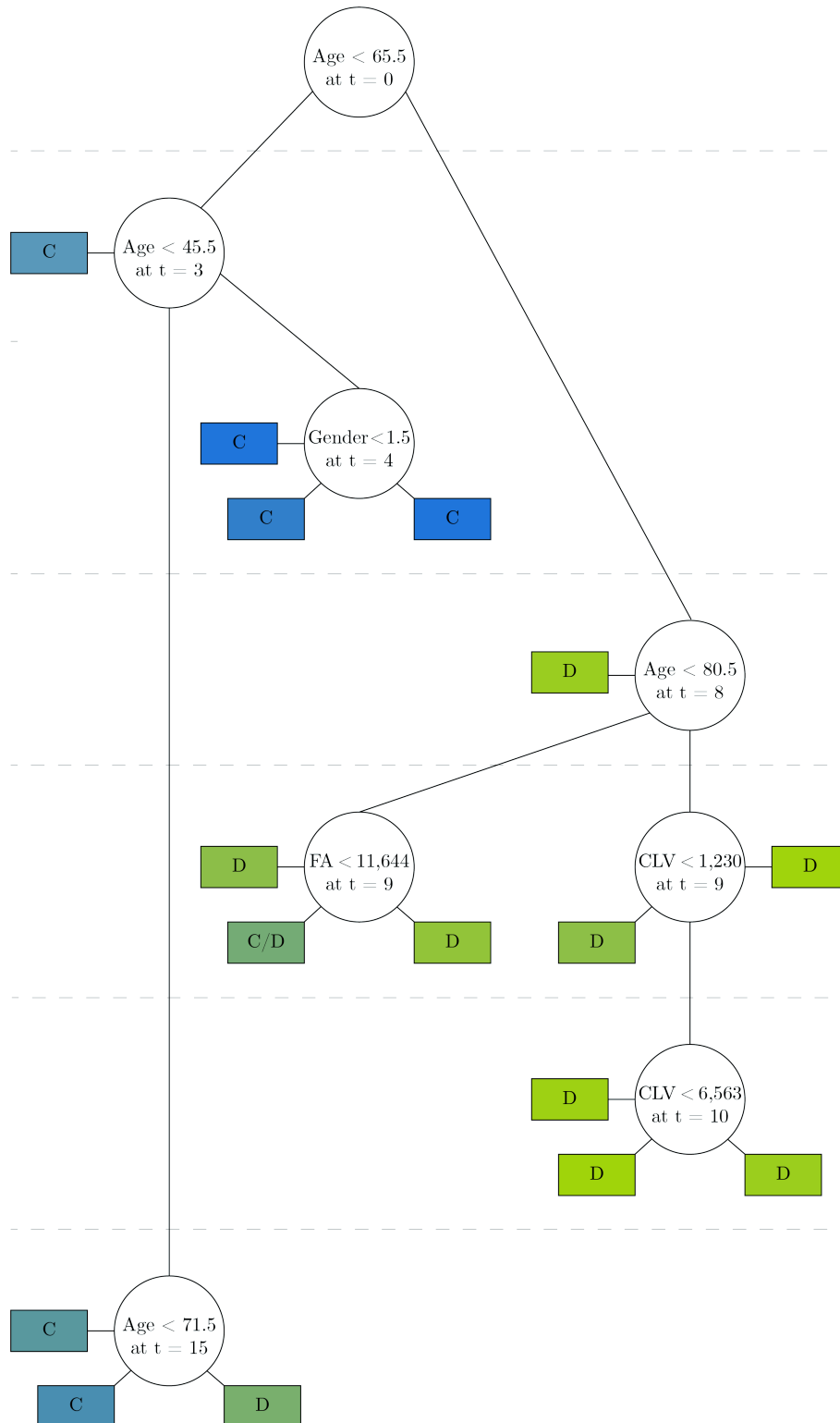


Figure D.3: Gini TpT (`minsplit=50`) with $\gamma = 0$

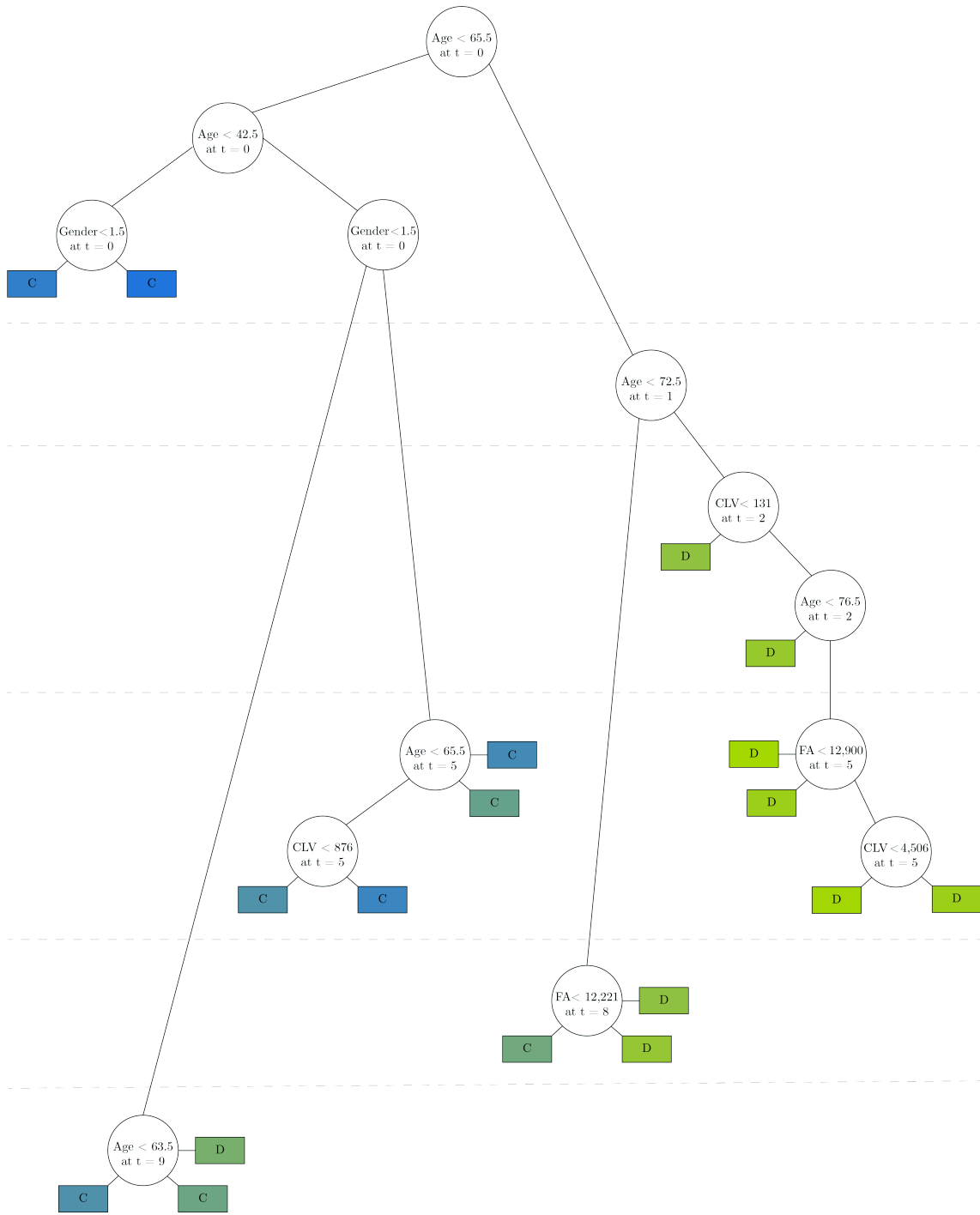


Figure D.4: Gini TpT (minsplit=50) with the optimal time penalty

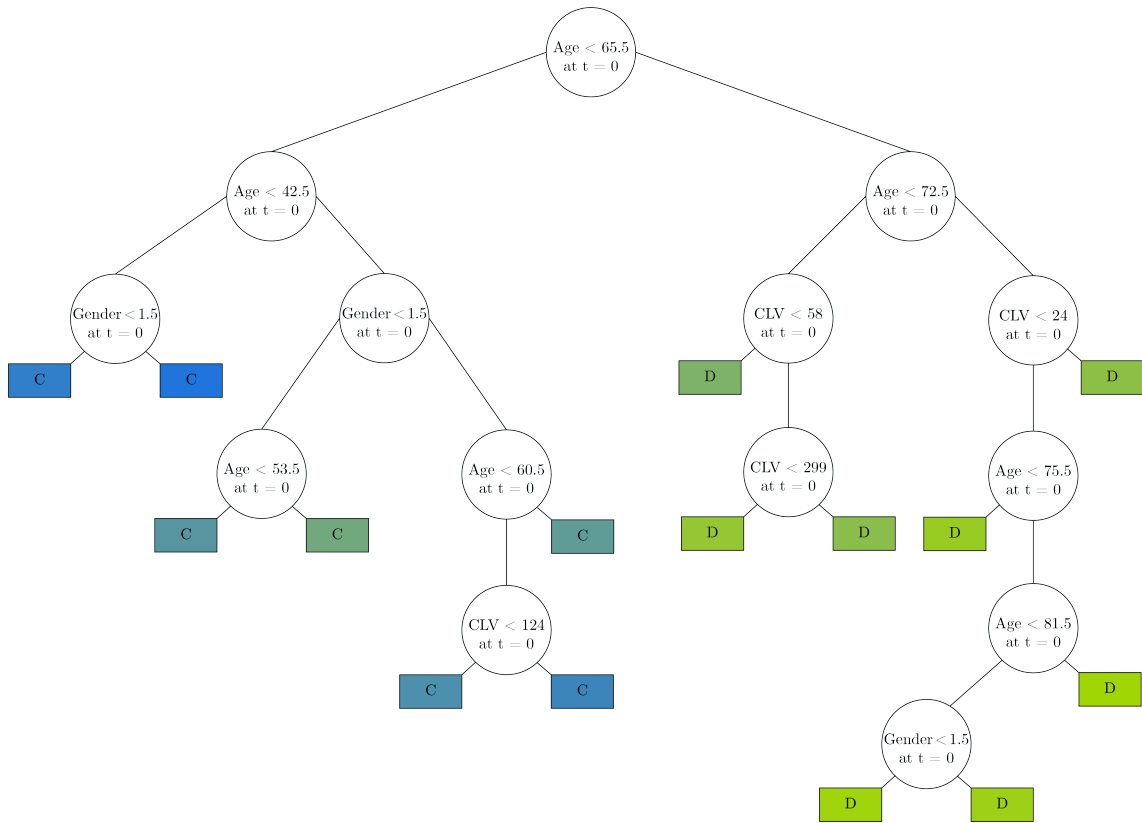


Figure D.5: Gini TpT (minsplit=50) with $\gamma \rightarrow \infty$

LIST OF FIGURES

4.1	Bias-variance tradeoff U-shaped curve	24
4.2	Train-test split procedure	27
4.3	k -fold cross-validation	28
4.4	Leave-p-out cross-validation	28
4.5	Monte-Carlo CV	29
4.6	Time series rolling CV	29
4.7	Illustration of the Areas under the ROC and PR curves	32
4.8	A simple decision tree	35
4.9	A M5 tree	39
4.10	Mechanisms of a random forest	42
4.11	Mechanisms of tree boosting	43
5.1	Illustration of survival analysis and censorship	50
5.2	Illustration of survival analysis and censorship - Starting times aligned	50
5.3	Illustration of a survival tree	53
7.1	Past and future CLV	68
7.2	The potential of CLV in the insurance industry	70
8.1	Seniorities and face amounts distributions	75
8.2	Strategy n°A-1: (Positive result on $y^{(i)}$ and an improved result on $\tilde{y}^{(i)}$.)	87
8.3	Strategy n°A-5: (Very negative result on $y^{(i)}$ and a loss-limiting result on $\tilde{y}^{(i)}$.)	87
8.4	Strategy n°B-27: (Negative result on $y^{(i)}$ and positive one on $\tilde{y}^{(i)}$)	88
8.5	Strategy n°B-6: (High positive result on $y^{(i)}$ slightly improved with $\tilde{y}^{(i)}$.)	88
8.6	Strategy n°A-25: (Results on $y^{(i)}$ better than results on $\tilde{y}^{(i)}$.)	89
8.7	3d plot (δ , γ , RG)	92
8.8	3d plot (d, T, RG)	92
12.1	General framework for lapse management strategy	117
12.2	Example of policyholders timelines	118
12.3	td-BSS of models trained on $\mathcal{D}_{train}^{last}$ and tested on $\mathcal{D}_{test}^{last}$	129
12.4	td-BSS of models trained on $\mathcal{D}_{train}^{long}$ and tested on $\mathcal{D}_{test}^{last}$ - Setting (b)	130
12.5	td-BSS of models trained on $\mathcal{D}_{train}^{long}$ and tested on $\mathcal{D}_{test}^{long}$ - Setting (d)	130
12.6	Longitudinally updated retention trajectories for a random subject	131
12.7	Projections of targeted profits over time	134
14.1	Decision tree recursive partitioning	147
14.2	Decision tree example	147
14.3	Illustration of a TpT	154
14.4	Single split of a TpT	156
14.5	TpT 1-depth recursive partitioning	157

14.6	TpT 2-depth recursive partitioning	157
14.7	TpT recursive partitioning and individual trajectories	158
14.8	Example of policyholders timelines	161
14.9	Split times distribution for the optimal unstopped and unpruned TpT	162
14.10	Characteritics of unstopped and unpruned TpT, trained on 9,873 PHs with various time penalties	163
14.11	Split times distribution for the optimal unstopped and unpruned TpT, trained on 9,873 PHs	163
14.12	Overfitted unpenalized, unstopped and unpruned TpT	164
14.13	Characteritics of TpT (<code>minsplit</code> : 50) depending on the time penalty	165
14.14	TpTs with $\gamma = 0$, $\gamma = 0.035$ and $\gamma \rightarrow \infty$, respectively	166
14.15	Individual longitudinal trajectories in the optimal TpT (<code>minsplit</code> = 50)	167
14.16	Global timeline and individual longitudinal trajectories	168
14.17	Yao et al. survival decision tree, taken from W; Yao et al. 2022	171
B.1	10 policyholders' survival curve for $r_{acceptant}$ with Cox model	188
B.2	10 policyholders' survival curve for r_{lapses}	188
B.3	Coefficient plot for r_{lapses}	188
B.4	r_{lapses} trajectories for different products	188
B.5	r_{lapses} trajectories by gender	188
B.6	r_{lapses} trajectories for different ages	188
B.7	r_{lapses} trajectories for different face amounts	188
B.8	Coefficient plot for $r_{acceptant}$	189
B.9	$r_{acceptant}$ trajectories by gender	189
B.10	$r_{acceptant}$ trajectories for different ages	189
B.11	$r_{acceptant}$ trajectories for different face amounts	189
B.12	10 policyholders' survival curve for $r_{acceptant}$ with RSF	189
B.13	10 policyholders' survival curve for r_{lapses} with RSF	189
B.14	10 policyholders' survival curve for $r_{acceptant}$ with GBSM	190
B.15	10 policyholders' survival curve for r_{lapses} with GBSM	190
B.16	Correlation between the proportion of non-targeted lapsers and the improvement ¹ of a CLV-augmented LMS	191
C.1	Monte-Carlo cross-validation	199
D.1	Characteritics of unstopped and unpruned TpT depending on the time penalty	208
D.2	Characteritics of Entropy TpTs (<code>minsplit</code> =25) depending on the time penalty	212
D.3	Gini TpT (<code>minsplit</code> =50) with $\gamma = 0$	213
D.4	Gini TpT (<code>minsplit</code> =50) with the optimal time penalty	214
D.5	Gini TpT (<code>minsplit</code> =50) with $\gamma \rightarrow \infty$	215

LIST OF TABLES

4.1	Confusion matrix for binary classification	31
8.1	Data set description	74
8.2	Insightful LMS	85
8.3	Means of the results obtained on considered LMS	86
12.1	A longitudinal dataset, in all generality	122
12.2	\mathcal{D}^{last} random subset	126
12.3	\mathcal{D}^{long} random subset	127
12.4	Various LMS results with our framework	133
14.1	A longitudinally structured dataset	151
14.2	Characteritics of TpT (minsplit: 50) depending on the time penalty	165
B.1	Covariates importance for $r_{acceptant}$ with RSF	190
B.2	Covariates importance for r_{lapper} with RSF	190
B.3	Covariates importance for $r_{acceptant}$ with GBSM	190
B.4	Covariates importance for r_{lapper} with GBSM	191
B.5	Survival models comparison	191
B.6	Results of representative LMS with various statistical metrics	192
B.7	More LMS	193
D.1	Characteritics of TpT depending on the time penalty	209
D.2	Characteritics of unstopped and unpruned entropy TpTs depending on the time penalty	210
D.3	Characteritics of Entropy TpTs (minsplit=25) depending on the time penalty	211

E. Doctoral booklet Curriculum Vitae Research Data Management

One of KU Leuven's requirements is to write and share a doctoral booklet. Out of transparency, here it is, shared within the PhD Thesis. It is a document that serves as a crucial progress reporting tool for Ph.D. students throughout their academic journey. This booklet is designed to document and track the progress, activities, and achievements of the doctoral student throughout their PhD program.

The doctoral booklet includes details about seminars and conferences attended by the doctoral student, their international publications, teaching activities, Research Data Management and any other involvement in academic or field-related initiatives.

It is periodically updated, often on an annual basis, to reflect the ongoing progress and achievements of the doctoral student. The booklet serves as a record of their academic and professional development, providing an overview of their contributions, activities, and engagements during their PhD studies. Both the doctoral student and their supervisors review and sign the booklet to confirm the accuracy of the information provided.

Ultimately, the doctoral booklet at KU Leuven functions as a formalized record-keeping tool that supports the supervision, assessment, and evaluation of the doctoral student's progress throughout their PhD journey. It aids in maintaining transparency and accountability while documenting the breadth and depth of their scholarly activities and achievements.

The author's individual doctoral booklet, updated on the 31st of December 2023 can be found on the next page.

PHD BUSINESS ECONOMICS: DOCTORAL BOOKLET

DOCTORAL BOOK: Mathias VALLA

Promotors for the 3 first years: Katrien Antonio, Xavier Milhaud, Denys Pommeret

Promotors for the 4th year: Katrien Antonio, Xavier Milhaud, Christian Yann Robert

2020-2021 – First year of PhD

2021-2022 – Second year of PhD

2022-2023 – Third year of PhD

2023-2024 – Fourth year of PhD (defense in 2024)

Topic of PhD: Temporal dynamics in tree-based models and applications to lapse behaviour in life insurance

Note:

- Every year this doctoral book should be updated and signed by promotor and doctoral student. It needs to be submitted with the KULoket “PhD Progress”

COURSEWORK

YEAR	COURSE (course title – hours – validation state)
2020-2021	Economics and Management of Organizations : An Experimental Emphasis – 21 hours - Passed
2020-2021	Ethics for Researchers in Economics and Management– 12 hours - Passed
2020-2021	Qualitative Research Design – 12 hours - Passed
2020-2021	Various Approaches to Study a Case– 12 hours - Passed
2020-2021	Atelier Statistique sur Traitement statistique des données manquantes (<i>Statistical workshop on missing values statistical analysis</i>) – 18 hours - Passed
2021-2022	Intégrité scientifique dans les métiers de la recherche – 15 hours – Passed
2021-2022	Comprendre la Propriété Intellectuelle– 8 hours – Passed
2021-2022	Epistémologie des sciences de gestion– 28 hours – Passed
2021-2022	Workshop de l'EDSEG 2022– 12 hours – Passed
2021-2022	Theoretical Computer Science Spring School: Machine Learning, at CIRM – 30 hours - Passed

SEMINAR ON RESEARCH METHODOLOGY/DEONTOLOGY AND ETHICS IN RESEARCH

Two courses passed on this topic – *see the COURSEWORK section.*

SEMINAR AND CONFERENCES ATTENDANCE:

- Petit Déjeuner Chaire DIALog, 11/12/2020
- L2 ISFA Lyon and DSA-HEC Lausanne Seminar, 19/01/2021
- Math & IA, Mathematics and Artificial Intelligence, 09/03/2021

- Explicabilité et Interprétabilité des méthodes d'Intelligence Artificielle pour la classification et compréhension des scènes visuelles, 07/04/2021
- 2021 ASTIN Colloquium, 18-21/05/2021
- L2 ISFA Lyon and DSA-HEC Lausanne Seminar, 14/06/2021
- 3rd Insurance Data Science Conference, 16-18/06/2021
- United As One: 24th International Congress on Insurance: Mathematics and Economics (IME) 05-09/07/2021
- ASTIN Webinar: Unsupervised Learning applied to the Customer Lifetime Value (CLV), 21/09/2021
- Wiki-meeting CNP, 22/03/2022
- Indice de risque de dangerosité de Lundberg-Aumann-Serrano pour les processus de risques en temps discret, Etienne Marceau presentation at ISFA, 25/03/2022
- ASTIN Webinar: What is AGLM from Technical Viewpoint, Hirokazu Iwasawa 03/05/2022
- Theoretical Computer Science Spring School: Machine Learning, at CIRM, 23-27/05/2022
- L2 ISFA Lyon and DSA-HEC Lausanne Seminar, 13/06/2022
- Doctoral course for international seminar validation, ISFA, 14/06/2022
- Petit Déjeuner Chaire DIALog, 01/07/2022
- Web-coffee de rentrée Chaire DIALog 09/09/2022
- Convention A: CONNECTING KNOWLEDGE 19-23/09/2022
- Participation to SHAPE-Med@Lyon from December 2022 until today
- « Atelier ORCID » Créer et mettre à jour son identifiant ORCID : les enjeux – la pratique 13/12/2022

- Petit Déjeuner Chaire DIALog, 14/03/2023
- Petit-déjeuner - Projections climatiques et assurance, Chaire DIALog, 07/06/2023

ACTUARIAL SEMINAR AND CONFERENCES ATTENDANCE – ORGANISED BY THE FRENCH INSTITUTE OF ACTUARIES

- Assurance collaborative: Théorie des graphes et Actuariat, 28/01/2021
- Les perspectives de l'informatique quantique pour les actuaires, 15/04/2021
- Les risques psychosociaux : impact sur l'absentéisme et prévention en entreprise, 01/07/2021
- 1er Colloque International de l'Actuariat Francophone, 04-08/10/2021
- Principes généraux de la data visualization, 17/11/2021
- Peut-on mesurer la qualité d'anonymisation d'un jeu de données anonymisé ? 11/01/2022
- Journées IARD 2022, 25/03/2022
- Webinaire des filières – Dauphine, 28/03/2022
- Statistique bayésienne, data sciences et nouveaux risques – 22/09/2022
- Colloque International de l'Actuariat Francophone 03-07/10/2022
- Zoom sur les modèles de mesure de l'équité en machine learning 01/06/2023
- IA et éthique en assurance : une nouvelle solution pour atténuer la discrimination par proxy dans la modélisation du risqué 28/06/2023
- 3e Colloque International de l'Actuariat Francophone 11-15/09/2023

- Pourquoi DORA change-t-il la donne pour la gouvernance du numérique ?, 13/11/2023
- Les risques privacy pour les IA génératives, 16/11/2023
- Modélisation dynamique des résiliations sur un portefeuille d'assurance emprunteur en contrat groupe 28/11/2023
- "Actuaries as Data Scientists: how our Professionalism and Training keeps us ahead" - AAE Professionalism Committee – 29/11/2023
- Contributions des données de l'assurance à l'étude des risques naturels - mention spéciale du jury du Prix des Sciences du risque 2023 – 05/12/2023

SEMINAR AND CONFERENCES PARTICIPATION:

- *Towards tree-based time-to-event models with longitudinal data, Literature review and applications to churn analysis* at Chaire DIALog's technical seminar, 05/11/2021
- *Longitudinal trees: an overview of tree-based models for longitudinal analysis with time-varying covariates* - Workshop de l'EDSEG 2022 – 11/04/2022
- *MLISTRAL Conference at CIRM, Including Customer Lifetime Value in tree-based lapse management strategy*, 26-30/09/2022
- *Tree-based models, "longitudinal covariates and customer behaviour predictions* at Chaire DIALog's technical seminar, 21/10/2022
- *Journée 100% Actuaire 100% Data Science, Handling Longitudinal Data*, 17/11/2022
- *Ma thèse en 180 secondes (3 minutes thesis), regional final, 2nd place.*
- *Longitudinal analysis: contributions in life insurance and applications* at Chaire DIALog's technical seminar, 15/11/2023
- *Title to be determined – « Data science et lois comportementales » Galea and associates Data Lab* 08/02/2024

INTERNATIONAL PUBLICATIONS AND SUBMISSIONS

- *Mathias Valla, Xavier Milhau, Anani Ayodélé Olympio. Including individual Customer Lifetime Value and competing risks in tree-based lapse management strategies. European Actuarial Journal, 2023, (10.1007/s13385-023-00358-0). (hal-03903047v4)*
- *Mathias Valla. A longitudinal framework for lapse management in life insurance. 2023. (hal-04178278) - Submitted to the Annals of Actuarial Science – In review*
- *Mathias Valla. Time-penalized trees (TpT): a new tree-based data mining algorithm for time-varying covariates. 2023. (hal-04178282) - Submitted to the Annals of Mathematics and Artificial Intelligence – In review*

RESEARCH DATA MANAGEMENT

As part of this doctoral research in Machine Learning, I utilized valuable data sourced from CNP, a respected French insurance company. This dataset was accessed via a highly secure data platform, ensuring compliance with GDPR. The information provided by CNP consisted of anonymized life insurance data, accessed through a meticulously secured private Jupiter environment.

Before incorporating this data into my research, I conducted extensive consistency checks and financial controls on the datasets obtained from CNP's databases. These rigorous checks validated the cleanliness and consistency of the data, confirming its alignment with accounting realities. The dataset comprises a comprehensive repository of over 200,000 distinct policies and policyholders procured between 1997 and 2019. Two primary datasets were meticulously compiled: the cross-sectional dataset, denoted as D_{last} , and the longitudinal dataset, termed D_{long} . D_{last} represents a unique policy/insured individual pair in each row, identified by a unique ID, while D_{long} encapsulates the longitudinal history of each subject present in D_{last} .

This extensive longitudinal dataset delineates the comprehensive journey of each policy and policyholder, encompassing all payment records, surrenders, charges, profit-sharing, or discount rates, from the policy's inception to the latest available information. Basic demographic covariates such as gender and age at the time of underwriting were also included for analytical purposes. However, due to the strict adherence to data privacy and confidentiality

regulations, the sensitivity and confidentiality of the information contained within these datasets preclude their sharing or public dissemination.

The proprietary nature of the data, coupled with the inherent privacy concerns, necessitates strict confidentiality and prohibits the sharing of these datasets beyond the confines of this research endeavor.

The original publications integrated within the thesis provide basic statistical descriptions of the final datasets. They are the only public and shareable documentation about these datasets and offer insights into key aspects such as data distributions, summary statistics, and other relevant statistical measures essential for comprehending the nature and characteristics of the datasets used in this doctoral research. These descriptions serve as valuable references for readers seeking a deeper understanding of the datasets while maintaining strict adherence to the confidentiality and privacy constraints governing the non-disclosure of proprietary data beyond the original publications within the thesis.

PLAGIARISM CHECK

Submitting the manuscript for a plagiarism check

ADDITIONAL PHD ACTIVITIES

- Participation to the competition “*Ma these en 180 secondes (MT180s)*” or *3-minute thesis* on the 21st of March 2023 – 2nd place at the regional final
 - Participation to the online competition *Summer of Math Exposition (Some2)* that aims at encouraging more people to put out explainers of math online. Entry available at <https://www.youtube.com/watch?v=J0fpXEIAzwk>
 - Participation to the research program SHAPE-Med@Lyon which aims to develop transdisciplinary health research based on a "One Health" approach. Its objectives are to promote transdisciplinarity and mobility between companies and the academic world, to set up training courses for future healthcare professions, and to accelerate the digital transformation of healthcare research.
 - Member of the Société Française de Statistique (SFdS) - *French Statistical Society*
 - Member of the Société Mathématique de France (SMF) - *French Mathematical Society*
-

TEACHING

2020-2021: 64 hours of teaching:

- 28h as teaching assistant for “Fondamentaux des mathématiques 1” – *Mathematics Fundamentals 1* - Freshman year
- 36h as teaching assistant and examiner (khôlles) for “Fondamentaux des mathématiques 2” – *Mathematics Fundamentals 2* - 2nd semester

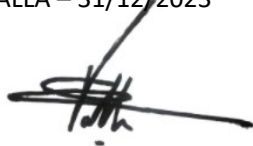
2021-2022: 64 hours of teaching:

- 64h as teaching assistant and examiner for “Fondamentaux des mathématiques 2” – *Mathematics Fundamentals 2* - Freshman year
- Participation to interviews for students’ professional project course

2022-2023: 64 hours of teaching:

- 18h as teaching assistant in “Licence L1 - Math-Info- Algèbre 1” - Freshman year
- 18h as teaching assistant in “Licence L2 Informatique- Math-Info - Statistiques pour l’informatique” - Freshman year
- 28h as teaching assistant and examiner (kholles) for “Licence L1 - Prépa - Analyse 1 et Algèbre 1” - Freshman year

Mathias VALLA – 31/12/2023



By signing this doctoral booklet, the supervisory committee evaluates the progress that the student made as sufficient, and allows her/him to enroll for the academic year 2023 – 2024

Katrien ANTONIO and Xavier Milhaud –:



AFDELING

Adres

3000 LEUVEN, België

tel. + 32 16 00 00 00

fax + 32 16 00 00 00

@kuleuven.be

www.kuleuven.be

