Predictive modeling of patient pathways using process mining and deep learning

Hugo de Oliveira

N° d'ordre NNT: 2020LYSEM021

THÈSE de DOCTORAT DE L'UNIVERSITÉ DE LYON
opérée au sein de
l'École des Mines de Saint-Étienne

École Doctorale N° 488
(Sciences, Ingénierie, Santé)

Spécialité de doctorat: Génie Industriel

Soutenue publiquement le 26/11/2020, par:
Hugo De Oliveira

Modélisation prédictive des parcours de soins à l'aide de techniques de Process Mining et de Deep Learning

—

Predictive Modeling of Patient Pathways using Process Mining and Deep Learning

---

Devant le jury composé de :

M. Emmanuel Bacry	Directeur de recherche CNRS, CEREMADE, Ecole Polytechnique	Président du Jury
Mme. Clarisse Dhaenens	Professeure des universités, CRISTAL, Université de Lille	Rapporteure
Mme. Frédérique Laforest	Professeure des universités, LIRIS, INSA Lyon	Rapporteure
Mme. Laurence Watier	Chargée de recherche Inserm (HDR), Institut Pasteur	Examinatrice
M. Vincent Augusto	Maitre de recherche, LIMOS, Mines Saint-Etienne	Directeur de thèse
M. Xiaolan Xie	Professeur, LIMOS, Mines Saint-Etienne	Co-directeur de thèse
M. Ludovic Lamarsalle	Dirigeant (PharmD, MSc), HEVA	Co-encadrant
M. Martin Prodel	Senior Data Scientist (PhD), HEVA	Co-encadrant

Spécialités doctorales	Responsables :	Spécialités doctorales	Responsables
SCIENCES ET GENIE DES MATERIAUX	K. Wolski Directeur de recherche	MATHEMATIQUES APPLIQUEES	O. Roustant, Maître-assistant
MECANIQUE ET INGENIERIE	S. Drapier, professeur	INFORMATIQUE	O. Boissier, Professeur
GENIE DES PROCEDES	F. Gruy, Maître de recherche	SCIENCES DES IMAGES ET DES FORMES	JC. Pinoli, Professeur
SCIENCES DE LA TERRE	B. Guy, Directeur de recherche	GENIE INDUSTRIEL	N. Absi, Maitre de recherche
SCIENCES ET GENIE DE L'ENVIRONNEMENT	D. Graillot, Directeur de recherche	MICROELECTRONIQUE	Ph. Lalevée, Professeur

EMSE : Enseignants-chercheurs et chercheurs autorisés à diriger des thèses de doctorat (titulaires d'un doctorat d'État ou d'une HDR)

ABSI	Nabil	MR	Génie industriel	CMP
AUGUSTO	Vincent	CR	Image, Vision, Signal	CIS
AVRIL	Stéphane	PR2	Mécanique et ingénierie	CIS
BADEL	Pierre	MA(MDC)	Mécanique et ingénierie	CIS
BALBO	Flavien	PR2	Informatique	FAYOL
BASSEREAU	Jean-François	PR	Sciences et génie des matériaux	SMS
BATTON-HUBERT	Mireille	PR2	Sciences et génie de l'environnement	FAYOL
BEIGBEDER	Michel	MA(MDC)	Informatique	FAYOL
BLAYAC	Sylvain	MA(MDC)	Microélectronique	CMP
BOISSIER	Olivier	PR1	Informatique	FAYOL
BONNEFOY	Olivier	PR	Génie des Procédés	SPIN
BORBELY	Andras	MR(DR2)	Sciences et génie des matériaux	SMS
BOUCHER	Xavier	PR2	Génie Industriel	FAYOL
BRODHAG	Christian	DR	Sciences et génie de l'environnement	FAYOL
BRUCHON	Julien	MA(MDC)	Mécanique et ingénierie	SMS
CAMEIRAO	Ana	MA(MDC)	Génie des Procédés	SPIN
CHRISTIEN	Frédéric	PR	Science et génie des matériaux	SMS
DAUZERE-PERES	Stéphane	PR1	Génie Industriel	CMP
DEBAYLE	Johan	MR	Sciences des Images et des Formes	SPIN
DEGEORGE	Jean-Michel	MA(MDC)	Génie industriel	Fayol
DELAFOSSE	David	PR0	Sciences et génie des matériaux	SMS
DELORME	Xavier	MA(MDC)	Génie industriel	FAYOL
DESRAYAUD	Christophe	PR1	Mécanique et ingénierie	SMS
DJENIZIAN	Thierry	PR	Science et génie des matériaux	CMP
BERGER-DOUCE	Sandrine	PR1	Sciences de gestion	FAYOL
DRAPIER	Sylvain	PR1	Mécanique et ingénierie	SMS
DUTERTRE	Jean-Max	MA(MDC)		CMP
EL MRABET	Nadia	MA(MDC)		CMP
FAUCHEU	Jenny	MA(MDC)	Sciences et génie des matériaux	SMS
FAVERGEON	Loïc	CR	Génie des Procédés	SPIN
FEILLET	Dominique	PR1	Génie Industriel	CMP
FOREST	Valérie	MA(MDC)	Génie des Procédés	CIS
FRACZKIEWICZ	Anna	DR	Sciences et génie des matériaux	SMS
GARCIA	Daniel	MR(DR2)	Sciences de la Terre	SPIN
GAVET	Yann	MA(MDC)	Sciences des Images et des Formes	SPIN
GERINGER	Jean	MA(MDC)	Sciences et génie des matériaux	CIS
GOEURIOT	Dominique	DR	Sciences et génie des matériaux	SMS
GONDRAN	Natacha	MA(MDC)	Sciences et génie de l'environnement	FAYOL
GONZALEZ FELIU	Jesus	MA(MDC)	Sciences économiques	FAYOL
GRAILLOT	Didier	DR	Sciences et génie de l'environnement	SPIN
GROSSEAU	Philippe	DR	Génie des Procédés	SPIN
GRUY	Frédéric	PR1	Génie des Procédés	SPIN
HAN	Woo-Suck	MR	Mécanique et ingénierie	SMS
HERRI	Jean Michel	PR1	Génie des Procédés	SPIN
KERMOUCHE	Guillaume	PR2	Mécanique et Ingénierie	SMS
KLOCKER	Helmut	DR	Sciences et génie des matériaux	SMS
LAFOREST	Valérie	MR(DR2)	Sciences et génie de l'environnement	FAYOL
LERICHE	Rodolphe	CR	Mécanique et ingénierie	FAYOL
MALLIARAS	Georges	PR1	Microélectronique	CMP
MOLIMARD	Jérôme	PR2	Mécanique et ingénierie	CIS
MOUTTE	Jacques	CR	Génie des Procédés	SPIN
NAVARRO	Laurent	CR		CIS
NEUBERT	Gilles			FAYOL
NIKOLOVSKI	Jean-Pierre	Ingénieur de recherche	Mécanique et ingénierie	CMP
NORTIER	Patrice	PR1	Génie des Procédés	SPIN
O CONNOR	Rodney Philip	MA(MDC)	Microélectronique	CMP
PICARD	Gauthier	MA(MDC)	Informatique	FAYOL
PINOLI	Jean Charles	PR0	Sciences des Images et des Formes	SPIN
POURCHEZ	Jérémy	MR	Génie des Procédés	CIS
ROUSSY	Agnès	MA(MDC)	Microélectronique	CMP
ROUSTANT	Olivier	MA(MDC)	Mathématiques appliquées	FAYOL
SANAUR	Sébastien	MA(MDC)	Microélectronique	CMP
SERRIS	Eric	IRD		FAYOL
STOLARZ	Jacques	CR	Sciences et génie des matériaux	SMS
TRIA	Assia	Ingénieur de recherche	Microélectronique	CMP
VALDIVIESO	François	PR2	Sciences et génie des matériaux	SMS
VIRICELLE	Jean Paul	DR	Génie des Procédés	SPIN
WOLSKI	Krzystof	DR	Sciences et génie des matériaux	SMS
XIE	Xiaolan	PR0	Génie industriel	CIS
YUGMA	Gallian	CR	Génie industriel	CMP

Mise à jour : 30/08/2018

Predictive Modeling of Patient Pathways using Process Mining and Deep Learning

Hugo De Oliveira

26/11/2020

Abstract

Initially created for a reimbursement purpose, non-clinical claim databases are exhaustive Electronic Health Records (EHRs) which are particularly valuable for evidence-based stud- ies. The objective of this work is to develop predictive methods for patient pathways data, which leverage the complexity of non-clinical claims data and produce explainable results. Our first contribution focuses on the modeling of event logs extracted from such databases. New process models and an adapted process discovery algorithm are introduced, with the objective of accurately model characteristic transitions and time hidden in non-clinical claims data. The second contribution is a preprocessing solution to handle one complexity of such data, which is the representation of medical events by multiple codes belonging to different standard coding systems, organized in hierarchical structures. The proposed method uses auto-encoders and clustering in an adequate latent space to automatically produce relevant and explainable labels. From these contributions, an optimization-based predictive method is introduced, which uses a process model to perform binary classifica- tion from event logs and highlight distinctive patterns as a global explanation. A second predictive method is also proposed, which uses images to represent patient pathways and a modified Variational Auto-Encoders (VAE) to predict. This method globally explains predictions by showing an image of identified predictive factors which can be both frequent and infrequent.

Résumé

Les bases de données médico-administratives sont des bases de données de santé particulièrement exhaustives. L'objectif de ce travail réside dans le développement d'algorithmes prédictifs à partir des données de parcours patients, considérant la complexité des données médico-administratives et produisant des résultats explicables. De nouveaux modèles de processus et un algorithme de process mining adapté sont présentés, modélisant les transitions et leurs temporalités. Une solution de prétraitement des journaux d'événements est également proposée, permettant une représentation des évènements complexes caractérisés par de multiples codes appartenant à différents systèmes de codage, organisés en structures hiérarchiques. Cette méthode de clustering par auto-encodage permet de regrouper dans l'espace latent les événements similaires et produit automatiquement des labels pertinents pour le process mining, explicables médicalement. Un premier algorithme de prédiction adapté aux parcours est alors proposé, produisant via une procédure d'optimisation un modèle de processus utilisé pour classifier les parcours directement à partir des données de journaux d'événements. Ce modèle de processus sert également de support pour expliquer les patterns de parcours distinctifs entre deux populations. Une seconde méthode de prédiction est présentée, avec un focus particulier sur les événements médicaux récurrents. En utilisant des images pour modéliser les parcours, et une architecture d'auto-encodage variationnel modifiée pour l'apprentissage prédictif, cette méthode permet de classifier tout en expliquant de manière globale, en visualisant une image des facteurs prédictifs identifiés.

“Tu veux savoir c’qui m’effraie ?
C’est pas c’que j’ignore,
Mais tout c’que je sais qui n’est pas vrai.”

Médine, Global

Remerciements

Je tiens à remercier tout d'abord Vincent et Xiaolan, mes directeurs de thèses, pour leur encadrement et leurs retours avisés. En particulier, un grand merci à toi Vincent, qui m'a fait confiance pour mon premier stage à Montréal en 2016, et sans qui je n'aurai surement pas eu la chance de pouvoir aboutir à ce travail. Merci à mes camarades-doctorants, notamment pour ces pauses café et ces belles soirées stéphanoises. En particulier, merci à Cyriac et Nilson, avec qui j'ai eu la chance de partager mon bureau lors de ces trois années au Centre Ingénierie Santé.

Merci à toutes les personnes que j'ai pu côtoyer dans cette belle entreprise qu'est HEVA. J'ai eu la chance d'énormément apprendre au sein de cette structure pluridisciplinaire à taille humaine, pleine de talentueuses personnes et de bonne humeur. Merci à vous Alexandre et Ludovic pour votre confiance et votre regard innovant sur l'écosystème des données de santé. Merci à l'équipe Data Science, pour tous ces moments, au bureau ou à distance pendant cette étrange période de confinement. Merci pour ces échanges enrichissants, ces rires, et pour ces petites phrases qui, je l'espère, continueront d'être retranscrites. Un grand merci à toi Martin, qui a suivi mon travail avec beaucoup d'intérêt. Les choses ont évolué depuis mon arrivée en avril 2017, mais l'équipe actuelle est à l'image de ce que tu as pu insuffler depuis le début : une bonne ambiance, de la curiosité, et de bienveillance.

Merci à mes amis qui m'ont, de près ou de loin, accompagnés durant ces trois années. Merci aux AURAsiens, en particulier Bassel, Bastien, Jules et Victor. Bastien, tu m'as très souvent offert le gîte ainsi que le couvert, je t'en suis très reconnaissant. Merci aux Lamas pour tous ces joyeux échanges de spams qui auront égayés bien des journées. Merci à Alexandre, Benjamin, Clothilde, Eliott, Florence, Marion et Rémi, pour votre amitié qui, depuis l'Ecole, ne s'est point altérée. Vous revoir fût à chaque fois une fête, que ce soit lors d'aller-retours éclairs sur Paris, en week-end ou bien de l'autre côté de l'Atlantique. Il y a des liens qu'il faut savoir chérir, et ceux-là en font partie.

J'ai une pensée toute particulière pour ma grande famille, dont je suis très fier. Grandir auprès de vous aura façonné une grande partie de moi. Merci à mes parents, Isabel et Jaime, qui ont toujours tout fait pour moi, sans jamais douter. Je vous dois tout.

Enfin, un grand merci à toi Jasmin, mein Lieblingsmensch, la meilleure supportrice. Merci pour tes nombreuses relectures, pour ta bienveillance, pour tes encouragements incessants et ton amour. J'ai de la chance.

Contents

Note to the reader

This manuscript presents a thesis organized as a collection of research articles (Chapters 2-5). To improve its readability and overall coherence, each article is preceded by an introduction including the motivation behind the article, a brief summary and conclusion.

Glossary

AE AutoEncoder. 30, 31, 40

AI Artificial Intelligence. 17, 18, 19, 20, 25, 27, 32

CépiDc the center for epidemiology of medical causes of death. 19

CNN Convolution Neural Network. 30, 31

EHR Electronic Health Record. 18, 27, 29, 30, 31, 32, 37, 38

GAN Generative Adversarial Network. 30

HDH Health Data Hub. 20

HES Hospital Episode Statistics. 68, 112

HMM Hidden Markov Model. 29, 38

ICU Intensive Care Unit. 28, 32

KDE Kernel Density Estimation. 42

LOS Length Of Stay. 27, 28, 31, 84

LSTM Long-Short Term Memory. 30, 31, 32

PMSI the national hospital discharge database. 19, 20, 37, 39, 109, 111, 112

RNN Recurrent Neural Network. 30, 32

SNIIRAM the national health insurance information system. 19, 20, 31, 94, 108, 110, 111, 112

VAE Variational AutoEncoder. 21, 40, 70, 94, 110

XAI eXplainable Artificial Intelligence. 32, 109

Introduction

Over the past years, a new vision regarding healthcare has emerged, advocating for a more predictive, preventive, personalized and participatory (P4) medicine [1], [2]. Considering medicine as an information science, this vision is widely adopted today [3]. The use of scientific and technological innovations constitutes a real opportunity to improve health systems, by providing more cost-effective disease care, reducing incidence of diseases, and constantly improving health systems through a global learning process [4]. The digital revolution takes a significant part in the emergence of technological innovations in health-care. Recent advances in information and computer science have empowered countries, administrations, hospitals and health companies, by giving them the tools to pursue the path of P4 medicine.

Artificial Intelligence

Artificial Intelligence (AI) was formally introduced as a scientific discipline in 1955. The assumption laying the foundation of this new discipline is that "every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it" [5]. In the past 30 years, through the fast evolution of compu- tational power, the discipline has dramatically extended. The milestones and advances in AI have been marked by the achievement of very complex tasks. In 1997, IBM's Deep Blue algorithm defeated World Chess Champion Garry Kasparov in a six-game match [6]. In 2011, IBM's Watson participated in *Jeopardy!*, an American television game show in which questions have to be identified based on the answers. In a real-time two-game competition, Watson defeated the best participants of *Jeopardy!* [7]. In 2016, Google Deep Mind's Al- phaGo [8] defeated Lee Sedol, the winner of 18 world titles and considered the greatest Go player of the past decade. Improvements in computer vision [9], [10] and natural language processing [11]–[13] further contributed to the large development of the field. In parallel to these advancements, the democratization of AI facilitated the practical deployment of AI algorithms across various industries and companies. These developments were fueled by a large amount of open-source scientific research and practical resources.

The discipline of AI regroups different trends and methodologies. Strengthened by an increasing availability of data, many advances in AI have been data driven. *Machine learning*, a subset of AI, enables "computers to tackle problems involving knowledge of the real world and make decisions that appear subjective" [14]. Data constitutes this "knowledge", which can be of various formats, more or less structured. The representation of data is a crucial machine learning task, referred to as *representation learning* [15]. As learning

an accurate representation of data is a complex task, *deep learning* constitutes a solution in which a complex representation is expressed in terms of other simpler representations [14]. By using successive layers of simple non-linear functions, deep learning methods are able to learn a more complex function with a high level of abstraction. The two mathematical pillars of deep learning are linear algebra and probability. Optimization is also part of the scientific brick of deep learning, as the training of deep architecture is performed through gradient-based optimization. Figure 1 positions deep learning among the previously introduced disciplines of AI.

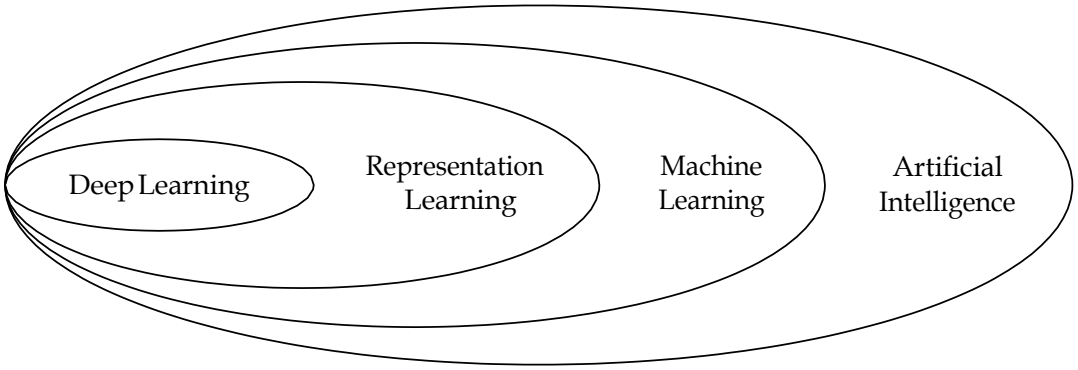


Figure 1: Positioning of deep learning in AI (based on Goodfellow et al. [14]).

Electronic health records

Electronic Health Records (EHRs) can be defined as a longitudinal collection of electronic health information for individual patients and populations [16]. The initial motivation for the deployment of EHRs was to document patients' care for reimbursement [17]. However, the data contained in EHRs is a gold mine for research, which constitutes a major secondary use of EHRs. In the context of *evidence-based medicine* [18], the use of these data serves to validate assumptions and draw conclusions from quantitative information and observed medical practices. As detailed by Yadav et al. [17], EHR data have used to derived research assumptions, analyze quantitative information and observe medical practices. Descriptive information about a diseases is useful to understand the evolution of disease for patients at scale (from a subgroup of patients to an entire population). The understanding of patients' medical trajectories, as well as the analysis of related comorbidities give insights into understanding a given pathology and practical care. Cohort analysis, involving two similar groups of patients with and without a given outcome, can be performed at a very large scale due to EHR data being widely available. In a next step, risk prediction can be performed based on the results obtained from cohort analysis. Another application of data analysis using EHR data is the quantification of drug or surgery effects. One particularity of EHRs is their ability to store information in various formats. Structured data is commonly the most encountered data format. Examples of structured data are patient information (age, gender, address), or medical event characteristics (medical codes regarding diagnosis or medical procedures, description of the medical unit, tests results). Unstructured data such as free text, images or sensor signals can also be found in EHRs. The unstructured data format is useful to store precise medical information, but is more complex to process.

The French national health data

In France, the national health insurance information system (SNIIRAM) was created in 1999 in order to improve the overall management of the national health insurance. Among the objectives were the improvement of health care policies and care quality as well as the provision of useful information to health practitioners. The data is collected from reimbursements which are registered by the national health insurance. In 2006, almost all French citizens became part of the SNIIRAM, leading to colossal quantity of information (66 million inhabitants in 2015 [19]). Based on reimbursements, it contains *non-clinical claims data* [20]. This database contains individual data used for billing outpatient healthcare consumption, and is linked to the national hospital discharge database (PMSI) by using a unique anonymous identifier for each patient, derived from the social security number. Since 2016, medical causes of death are transmitted to the national health insurance by the center for epidemiology of medical causes of death (CépiDc), and are linked to the SNIIRAM using the same unique anonymous identifier.

Figure 2 shows a schematic representation of these databases. The SNIIRAM includes outpatient healthcare consumption (along with private clinics healthcare expenditures). The PMSI includes data about hospital stays in short-stay wards, rehabilitation units, home care units and psychiatric institutions. At the end of each stay, an anonymous discharge summary is produced, with information such as the duration of stay, month and year of discharge, source and destination before and after the stay, and multiple information about diagnosis, related comorbidities and care (medical procedures, drugs, medical devices). The CépiDc includes the date, place, and the medical causes of death. More details can be found in Appendix A.

The complexity of these databases is a challenge: an extensive number of tables, based on reimbursements and with complex relations. Also, the lack of precise medical information (such as test results, imaging reports, or vital signs) can be a limiting factor for certain studies. However, the main advantage of the SNIIRAM is that it forms an exhaustive database as all patients' characteristics, hospitalizations, and outpatient information are recorded at the scale of an entire population.

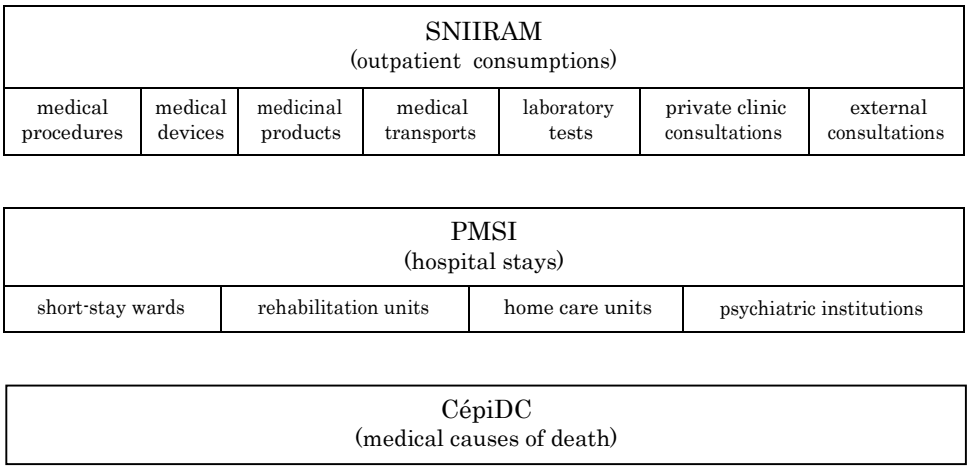


Figure 2: Schematic representation of the SNIIRAM, the PMSI and the CépiDc.

More recently, a governmental initiative in favor of the development of AI has promoted the application of AI for the analysis of health data. Mandated by the French Prime

Minister, the mathematician and deputy Cédric Villani was asked to conduct a study and report on the French and European strategy regarding the deployment of AI methods and technologies. Delivered in October 2018, this report recommends priority investments and proposes future developments [21]. Among discussions on interdisciplinary topics such as economy, research and ethics, a focus on healthcare was presented. One recommendation related to healthcare was the creation of a platform for access and sharing of health data, to facilitate research and innovation. This platform, referred to as the Health Data Hub (HDH)[1], is built on one main principle: because the national health data are financed by the national solidarity, the data should be shared. Created in 2019, the motivation of the HDH is to promote unified and secure access to the national health data, for all stakeholders, while strictly respecting ethical guidelines and citizen rights. As a result, each request to access data from the HDH need to be precisely detailed in a study protocol. Each submission is then examined by a multidisciplinary assembly of experts which evaluates the quality of the protocol, the aim of the project and its benefits for the common good. Today, the access to the PMSI or to extractions of the SNIIRAM database are possible through the HDH.

Scientific objectives

In accordance of the development of a more preventive and predictive medicine, the general aim of this work is the predictive modeling of patient pathways. In terms of the methods used, the positioning of this work falls into two disciplines. The first is *process mining* [22] with multiple recent applications in health care [23], [24]. The second discipline, often used to prediction but not only, is *machine learning*, and more precisely *deep learning*. The data analyzed in this work are part of the the French national health data. Due to the challenges arising from the practical deployment of predictive methods, this work puts special emphasis on the explainability of predictions. The predictive models and methods developed in this work are designed to highlight predictive factors extracted during the training. This achieves transparency which in turn allows for the discovery of new patterns hidden in the data and facilitates the discussion of predictive results with medical experts and decision makers. This transparency may be useful for practical applications, in order to discover new patterns hidden in the data and discuss it with medical experts and decision makers.

Thus, the scientific objectives of the presented work are twofold, each one formulated as a research question:

1. **How to properly model patient pathways information for descriptive and predictive data analysis?** Extracted from non-clinical claims data , the complex representation of such information is multifaceted. Two axes are explored in this work, namely the modeling of *time* (causal and numerical), and the complexity of *medical events* (macro events described by multiple codes from varied coding systems, each one having its own hierarchy structure).

2. **How to perform predictions from the analysis of complex patient pathways while including explainability?** In this work, *process mining* and *deep learning* methods are used to propose adapted predictive modeling methods for patient pathways.

[1] https://www.health-data-hub.fr/

Thesis outline

The rest of this manuscript is organized as follows.

- Chapter 1 presents a literature review of the topics covered by this work, starting with a general overview of recent innovations involving health data. As predictive modeling is the main objective of this work, the related literature is reviewed. Recent advances of process mining are also exposed, with a particular focus on applications in healthcare.

- A new process mining framework adapted to time modeling is presented in Chapter 2. This framework includes two new process models, an optimization procedure to perform process discovery from event logs, and a list of adapted descriptors for both event logs and process models.

- Chapter 3 introduces a preprocessing method for event log data, useful for process mining applications in healthcare. Focused on complex macro medical events, this deep learning-based method serves to assign synthetic labels to events through clustering in an adapted latent space, learned using autoencoding. The decoding learned serves to explain and medically interpret the synthetic labels created. This contribution can be used to preprocess medical events before performing process discovery such as the one introduced in Chapter 2.

- An optimization-based predictive algorithm is presented in Chapter 4, constructed using the process mining framework presented in Chapter 2. By performing classification of traces from event logs using an adapted learned process model, the method is explainable thanks to the characteristics of the resulting process model.

- A second predictive method is presented in Chapter 5. Based on a Variational AutoEncoder (VAE), an adapted representation of patients and their medical events over the time is introduced. The advantage of this method is to be able to identify both frequent and infrequent predictive factors, while explaining the learning by producing an image of risk patterns.

- Finally, the main contributions and results are summarized and discussed. Perspectives for future works are also proposed.

A schematic representation of the relations between the technical chapters is presented on Figure 3.

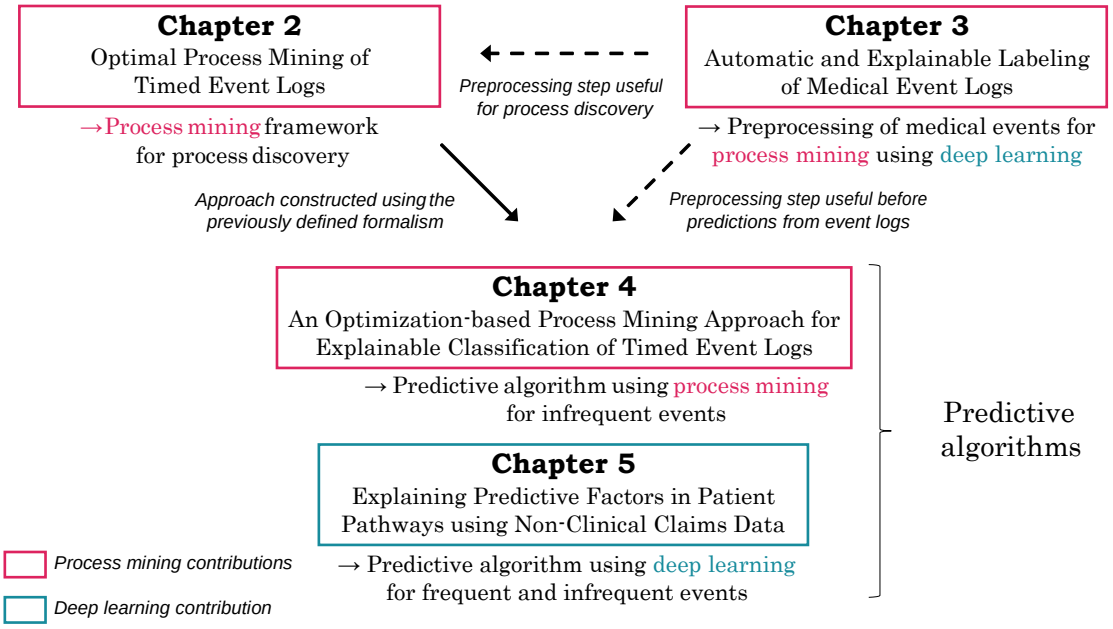


Figure 3: Schematic representation of the thesis outline.

Chapter 1

Literature Review

Contents of the chapter

Abstract

This chapter presents a literature review of data in healthcare applications, especially with a predictive perspective. A general overview of recent data-driven innovation in healthcare is presented. The topic of predictive analysis in healthcare is then addressed, by giving the different targets of prediction noticed in the literature and presenting the evolution of the methods used. Particular attention is given to the modeling of patient pathways and the explainability of results. In a next step, recent developments in the field of process mining are reviewed, particularly applications of process mining in healthcare. Finally, the research findings that motivated this thesis are presented and the objectives of this work are positioned amid the literature reviewed.

1.1 Data in Healthcare

Data analytics has been strongly developed over the past years. These developments have been reinforced by a constant increase in data availability and the constant growth of computational power. As a result, the following progress can be observed: performance improvement of existing methods, introduction of new algorithms, reinforcement and growing of technical communities, multiplication of available knowledge and teaching resources, and deployment in new field of applications. Among the recent improvements in healthcare, the use of data and data-related algorithms took a non-negligible place in various industrial applications and axis of research. This section presents a general overview of recent scientific advances in the field of healthcare data. In order to map recent data-driven innovations in healthcare, each of the following sections focuses on a particular field in which the generation of data has guided scientific contributions in a new direction.

Hospital resource management

Hospital resource management and the organization of care is a crucial issue. With limited resources, it is challenging to find the right balance between adequate care and economic efficiency. For the patient, it is important to receive comprehensive care. For the organization, it is important to treat patients in a reasonable time and at reasonable costs. For the medical staff, proper working conditions are a necessity. In some extreme situations, working conditions may worsen, leading to risks for health practitioners, such as mental health risks [25]. Taking all these parameters into account, it can be observed that the organization of care is a complex problem for national and regional health services, hospitals and medical units. With the development of information technology (IT) systems, data analytics has proven useful to improve the organization of care. The field of operation research (OR) regroups a set of algorithms, methods and tools which are deployed in hospital resource management. Some examples of application are operating room planning and scheduling [26] and emergency department optimization [27]. For these applications, methods identified by the authors are various: mathematical programming (linear, mixed integer programming, column generation), improvement heuristic, queuing theory and simulation (discrete-event, Monte Carlo). The use of simulation is a widespread practice in order to test different scenarios, evaluate and quantify the sensitivity of these changes and find an optimal policy [28]. In this context, quantitative measures extracted from real-world data serve as input parameters for the simulation.

Medical imaging

For many pathologies, the use of medical imaging is central for diagnosis and computer-aided diagnosis has become an important field of applied research in medical imaging and diagnostic radiology [29]. Automated image analysis showed promising advances in the past decade, particularly using deep learning [30]. One widespread application is the automated classification of images. This is the case for medical disciplines such as dermatology for skin cancer detection [31], or ophthalmology using retinal fundus photographs for diabetic retinopathy detection [32]. Furthermore, automated classification of ultrasounds is used to identify breast lesions, and pulmonary nodules identification is derived from computed tomography (CT) scans [33]. The segmentation of medical images is possible, for magnetic resonance imaging (MRI) images of the knee cartilage [34], or for positron emission tomography (PET) scans [35].

Connected devices

Nowadays, the use of monitoring devices is democratizing, leading to an increasing quantity of generated data, such as the position, behavior or user's vital signs. Sensors, which produce such data, are found in many different devices. Wearable devices, such as smartphones and smart watches, can monitor movements and deduce activities [36]. In addition, smart watches are able to recognize the user's activity, detect sleep patterns and measure the heart beat [37]. Ambient-assisted living for healthcare consist in created a connected environment using sensors, databases and applications, in order to assist the patients at their home [38]. These connected devices can estimate physical activity, detect behavioral changes and potential accidents or emergencies [39]. Medical devices, which are more and more connected, can produce data reports over the time. An analysis of resulting data is crucial for risk identification and prevention. The same applies to implantable medical devices such as implantable cardioverter-defibrillators, where an effective follow-up enables a rapid evaluation of critical events by a physician [40]. Regular monitoring may improve the quality of care for all patients. The monitoring of activity for patients suffering from chronic diseases such as diabetes is one example where monitoring can improve the quality of care by personalizing the approach [41]. Moreover, the monitoring of vital signs after a patient has been discharged from the hospital, has the potential to improve patient follow-up after surgery by early detection of risks [42]. At the hospital, a constant control of patients' vital signs is essential: automated monitoring can improve patient outcomes, by detecting a deterioration of health which allows for a fast intervention by a physician [43]. Among the monitoring exams, electroencephalography (EEG) and electrocardiography (ECG) provide complex data which can be automatically analyzed. In fact, recent developments in this area are the use of EEG data for automated medication classifications [44], the use of ECG data for automated sleep stage scoring [45] and for diagnosis of arrhythmia [46], [47].

Natural medical language

Natural language processing (NLP) is a research topic in the field of AI with multiple possible applications such as speech recognition, natural language understanding and natural-language generation. In healthcare, the comprehension of clinical notes is one example of automated processing and understanding of free text data. As shown by the systematic review of Sheikhalishahi et al. [48], a majority of the research in the area of chronic diseases centers around identification of risk factors and the classification of diseases. A significant increase in the use of machine learning compared with rule-based methods is noted by the authors. The generation of free text is another research topic which could be useful in healthcare applications. As an example, Xiong et al. propose a deep architecture to automatically generate discharge summary in Chinese [49]. Also, Lee proposed in 2018 a deep neural network architecture to generate synthetic chief complaints from variables found in electronic health records (EHR) data, such as the age category, gender, and diagnosis [50].

Bioinformatics

The field of bioinformatics focuses on biological processes at a molecular level. The sequencing of the human genome constituted an initial step for further analysis. The Human Genome Project was an innovative project, which produced a first version of the sequence of the human genome in the beginning of this century [51], [52]. By initiation an innovative movement, the results of this project motivated the recent developments in the

field of computational biology [53]. Personal medicine, through personalized sequencing, can be useful to optimize individual treatments [54]. Today, the cost of the sequencing of a human genome is less than US$1,000 [53], which makes it affordable compared to the first sequencing methods. The three main applications in the field of bioinformatics are the prediction of biological processes, the prevention of diseases and the production of personalized treatments [55].

Drug discovery

Machine learning has the potential to improve drug discovery processes at all stages, from target discovery to post approval [56]. However, the quality and quantity of such data are critical challenges. Pharmaceutical companies have large data sets which dates back to the 1980s [57]. As these data are proprietary, sharing these data is impossible because of competitive interests. In this context, the use of federated learning is promising. The main idea of federated learning is to leave the data in silos and make the learning algorithm transit between databases in order to learn from the entire available data. As a result, multiple actors can build a common machine learning model without sharing data. For example, the MELLODDY[1] project aims to use federated learning to capitalize from the data of ten pharmaceutical companies and to advance drug development. Regarding molecular research, the space of all small molecules is very large (theoretically around 10^{60}) [57]. As a result, innovative computational approach can help in searching promising molecule candidates. Quantum computing is a set of methods which are promising regarding drug discovery. Quantum simulation for example is an alternative to quantum chemistry methods for the characterization of molecular systems [58]. Also, quantum machine learning methods may, in the future, replace the classical machine learning approach in early phases of drug discovery. [58].

Discussion

Although recent data-driven healthcare innovations can be useful, their practical deployment is complex. The complexity is mainly due to a large variety of stakeholders who are involved in the deployment of such an innovative approach. As explained by Hood and Friend in the context of P4 cancer medicine, one requirement for a successful deployment is to align the objectives of all stakeholders (patients, health practitioners, industries) [3]. Trust in a new technology by all the stakeholders is an indispensable condition. This trust can be built throughout a lengthy process consisting of discussions, demonstrations, explanations and teaching. During the deployment, severe errors must be limited to avoid a quick collapse of the slowly constructed trust. A common fear regarding innovation is that an innovative method may lead to the loss of jobs or the loss of a complete medical profession. Medical imaging is an example of a discipline in which medical professionals are already confronted with this issue. Due to the high performance achieved by medical imaging algorithms, some tasks can be automated, resulting in an increased deployment of these algorithms. Knowledge regarding these methods should be integrated in the training of diagnostic radiographers, to promote acceptance and to facilitate the collaboration be- tween radiographers and their "virtual colleagues", as referred to by Lewis et al. [59]. As these innovations are often data-driven, another common concern is data security which makes guarantees in terms of data privacy mandatory. Recent European initiatives go in

[1] https://www.imi.europa.eu/projects-results/project-factsheets/melloddy

that direction. The General Data Protection Regulation[2] (GDPR) imposes "data protection as a pillar of citizens' empowerment and the EU's approach to the digital transition" [60]. The right to obtain, reuse and delete personal data (portability), is a powerful manner to empower patients and make it easier for them to allow the use of their data for the public good [60]. Lastly, equality is a major concern when discussing the benefits of AI. Facial recognition has become one of the most controversial applications of AI in terms of discrimination and inequality. Training the algorithm on an insufficient amount of data of minorities, will result in a performance gap of the algorithm, making it more susceptible to errors when it comes to recognizing the faces of a minority group. One example is the use of facial recognition software by the police in the United States, leading to serious errors towards minorities [61]. Unfortunately, AI in healthcare is not spared from an equality deficit. Some recent examples show how healthcare related algorithms can discriminate minorities if the data used to train these algorithms are patchy or biased [62].

1.2 Predictive Analysis of Health Data

In this section, the focus is on predictive analysis using EHR data. Recently, the interest in this research topic has rapidly grown due to an increased availability of these types of data. In the following, general objectives, methods, recent advances and current challenges regarding predictive analysis of EHRs data are presented.

1.2.1 Machine Learning

Initially, in order to identify at-risk patients, medical scores have been introduced in the literature. These scores make it possible to classify the condition of a patient on a mostly one-dimensional scale. One widespread example is the Charlson index [63], which predicts the one-year mortality for a patient who may have a range of comorbidities. In the context of automatically defining such scores, statistical and more recently machine learning methods are relevant. The most common machine learning algorithm design considers a future medical condition of patients as a prediction target and tries to fit a learning algorithm on previously observed scenarios. Starting from previous cases targets and features, these methods are able to find relevant relations, in order to predict for new cases. In comparison to manual assessment of medical scores, the machine learning algorithm assigns predictions based on the knowledge learned. In the literature, many different studies have applied machine learning to health data. Examples of these studies, detailed in the following, differ regarding, for example, the target of prediction, the type of data and data features, the type of algorithm used, or the pathology studied.

Length of stay

The prediction of the Length Of Stay (LOS) is one of the possible machine learning prediction targets. The ability to identify patients with a risk to stay for a long period of time in the hospital is useful for an organizational purpose. This problem can be addressed as a regression problem, or as a classification problem in which thresholds are specified to define different categories of LOS such as short and long stays for example. An examples of such a study was presented by Morton et al. [64], which predicts the LOS for diabetic

[2]Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC - OJ L 119, 4.5.2016, p. 1-88

patients using binary classification. Different algorithms such as random forest or support vector machine were used to identify long stays. Daghistani et al. [65] also studied LOS for cardiac patients as a classification problem, using random forest, multi-layer perceptron and bayesian networks. An example of treating LOS prediction problem as a regression is the work of Turgeman et al. [66], which uses cubist trees for patients with congestive heart failure.

Mortality

Another example of predictive analysis using machine learning is the prediction of mortality. Multiple binary classification algorithms can be used to predict the mortality of high risk patients. For example, multivariate statistical analysis was used by Billings et al. [67] and logistic regression by Di Marco et al. [68]. Using data extracted from the MIMIC-II data set, Salcedo et al. [69] performed a benchmark study of predictive algorithms such as logistic regression, multi-layer perceptron, *k*-nearest neighbors and decision tree, conclud- ing that the multi-layer perceptron and logistic regression with radial kernel models were the most adapted. In addition, the configuration of algorithms parameters was identified as a key determinant in performances.

Admission and readmission

The prediction of patients' admission represents another use case for predictive analysis using machine learning techniques. Rahimian et al. [70] used a random forest and gradient boosting classifier to predict unplanned emergency admissions and to compare to the results of a cox proportional hazards (CPH) model. They found that the machine learning methods outperformed the CPH model. Moreover, they included time-related features such as durations or the time since some given events, which increased the performances. In addition, by assessing the probability of the patient's short-term readmission prior to being discharged from the hospital, a decline of the patient's health outside the hospital can be prevented. For this purpose, logistic regression was used by Ben-Assuli et al. [71]. Futoma et al. [72] compared various machine learning methods, showing that random forest and penalized logistic regression outperformed support vector machine. Desautels at al. [73] predicted unplanned Intensive Care Unit (ICU) readmissions, with the particularity of using transfer learning between two databases to improve prediction performances. Other type of methods have been developed to predict readmission. For example, Lee et al. introduced an optimization based method, the DAMIP [74]. This method uses discrim- inant analysis with swarm optimization for feature selection. This work was extended by Hooijenga et al. [75], by implementing a tabu search-based feature selection method and preventing the model from classifying patients with uncertain prediction results.

Discussion

The use of machine learning methods for predictive analysis of health data has been discussed in the literature. Based on the preceding literature review, a few interesting point can be concluded. The first observation is that medical outcomes are in many cases pre- dicted as a binary classification problem. The machine learning algorithms used in this context are often part of a list of conventional algorithms. Furthermore, it can be observed that the input data used to train these machine learning algorithms are data features cre- ated from raw health data. These features can be demographic (e.g. age, gender), related to hospitalization (e.g. vital signs, test results) or to the patient overall health condition

(e.g. comorbidities). The creation and formatting of these features is a challenging task relying on expert knowledge, and often depends on the database available and the pathology studied. Finally, it can be noted that patient pathway information (e.g. the successions of medical events and time in-between) is rarely considered when creating features for machine learning algorithms. When considering patient pathway history structured in event logs, the considered binary classification problem is a sequence classification problem.

1.2.2 Sequence classification and modeling

Xing et al. [76] identified three main solutions to solve the binary classification problem of sequences. The first solution is feature-based classification: by extracting features directly from an event log, common machine learning algorithms can be applied. However, the construction of such features and the modeling of time is complex. The second solution is distance-based classification, were the similarity of sequences is analyzed in order to classify a new sequence. This approach is commonly used in bioinformatics for deoxyribonucleic acid (DNA) alignment, often by using the method developed by Needleman and Wunsch [77] or the one of Smith and Waterman [78]. The third solution is model-based classification, which uses statistical models as Hidden Markov Models (HMMs) (see for example Yoon [79] or Blasiak and Rangwala [80]).

The problem of extracting predictive patterns from sequential data has been discussed in the literature, with and without consideration of the temporal aspect. The work of Klema et al. [81] studies the development of associated health conditions and risk factors for patients suffering from atherosclerosis. To extract the predictive patterns, three different sequential mining approaches were used: windowing, inductive logic programming and episode rules. To perform sequence classification based in extracted patterns, Zhou et al. [82] tested two methods: one based on association rules, the other based on a ranking system which used the previously defined rules and the actual element to classify. In their work, Bose and van der Aalst [83] define "characteristic patterns that can be used to discriminate between desirable and undesirable behavior" as signatures, another well-known term for patterns. They presented a methodology to mine such patterns using a feature-based representation of events, prior to apply decision tree and association rules to classify traces while finding patterns. They applied the method to discover signatures in event logs of X-ray diagnoses. Signature mining was also addressed by Wang et al. [84] in the context of longitudinal heterogeneous event data extracted from EHRs. A useful 2-dimensional representation of patients was proposed to model event data. In addition, a framework for mining signatures is presented, which is based on convolution and has the particularity to be shift-invariant (discrete time-invariant). Vandromme et al. [85] presented a hybrid model for classification tasks which handles both non-temporal attributes and sequences, using various types of data such as numeric, binary, symbolic or temporal. Based on a heuristic approach, the method extracts classification rules. A case study using the French national hospital database shows accurate performances. Among the possible extension discussed, a better handling of recurrent events and taxonomy of events are mentioned.

1.2.3 Deep Learning

As presented before, machine learning methods use patients' features which are often extracted and structured from raw data following expert knowledge. A recent evolution is the use of deep learning for predictive modeling, with two main advantages. The first one is the

use of deep architectures, allowing a high degree of abstraction in the learning process. In the case of difficult tasks, this abstraction can strongly improve performances. The second advantage is that in deep learning methods the creation of data features from raw data is automated. When constructing features for machine learning, a simplified representation of data is performed. But the representation of the data can strongly impact performances of machine learning methods [15]. Thus, by learning a representation from raw data during the training, deep learning methods avoid the feature creation step and let the algorithm learn an optimal representation. The shift from manually defined features to deep learning had an significant impact, particularly in healthcare, as shown in the following.

Recent reviews of deep learning from health data

In many studies, deep learning has been applied on health data and particularly EHR data, using various methods and architectures for different purposes. In 2018, Ravi et al. [55] presented a review focused on the development of deep learning methods applied to health informatics, which regroups multiple applications such as medical informatics, sensing, bioinformatics, imaging and public health. According to Ravi et al., between 2010 and 2015, a rapid increase in terms of the number of publications in health informatics occurred. This is particularly the case for publications in medical imaging and public health. In terms of deep learning methods deployed, the authors identified deep neural networks, Convolution Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) as the most commonly applied methods. Regarding health informatics, the quantity and the variety of data contained in EHRs are noted as factors of success (from previous diagnoses, exams, and medications, to precise information such as radiology images or sensors multivariate time series). The quality, variability and complexity of the data are challenges for practical deployment of analytical methods. Long term dependencies and interpretabilty are also important issues that need to be addressed. The authors further underlined that the systematic use of deep learning in healthcare can be problematic, as it could restrain the development of other methods which are less resource-greedy and more interpretive. A similar review was presented by Miotto et al. [86] in 2018. Possible use cases for deep learning methods are identified such as clinical imaging, genomics and data analysis of sensors and mobile devices. According to the authors, challenges of deep learning on EHR data were the high volume, a lack of quality and the complexity of the data, the latter being particularly relevant for the healthcare domain. The author highlights two other limitations of deep learning methods: the temporality and the interpretability. The first one is still a challenge today, as many existing methods use fixed vector-based input data. This formalization is not optimal to represent the linear characteristic of time. The second limitation, the interpretability, is particularly important for medical experts to facilitate the practical deployment of a method. Two other publications of 2018 focused on describ- ing the application of deep learning methods on EHRs data. In their review, Shickel et al.
[87] highlighted the recent increase in terms of publications in the field of deep learning applied to EHR data. Among the areas of application, the authors identified prediction and representation as the most frequent. Moreover, unsupervised learning and deep architectures such as CNNs, RNNs, Long-Short Term Memorys (LSTMs) and AutoEncoders (AEs) were commonly used learning algorithms. The systematic review of Xiao et al. [88] also identified five common areas of application for deep learning in healthcare: disease detection, sequential prediction, concept embedding, data augmentation and data privacy. The main architectures used were CNNs, RNNs, AEs embedding methods and Generative

Adversarial Networks (GANs). In terms of evaluation metrics, the authors found accuracy and the area under the curve (AUC) to be the most commonly used methods. Challenges noted when working on EHR data were the temporality (the complex long- and short-term relations between events), the multi-modality (the use of heterogeneous data), the lack of labels (even if labeling using a given medical code is used in practice), and last but not least the interpretability.

Prediction

Identified as one of the main topics when applying deep learning on EHR data, relevant studies performing predictions are described in the following. Doctor AI [89] was introduced by Choi et al. in 2016. In their paper, they presented a deep learning framework which was able to perform differential diagnosis using EHR data. The same year, Miotto et al. [90] presented Deep Patient. Using a stack of denoising AEs, they learn an unsupervised representation of patients from EHR data. A random forest algorithm was trained over encoded patients in order to predict the probability of disease appearance. This method shown better performances than the original representation or other dimension reduction methods. An exhaustive study was conducted by Rajkomar et al. [91] in 2018. With a focus on scalability, different models were used to perform various predictive tasks, such as in-hospital mortality, 30-day unplanned readmission, prolonged LOS or patient's final discharge diagnoses. Zhang et al. [92] focused on limited data samples predictions and proposed MetaPred for clinical risk prediction. LSTM was used by Ashfaq et al. [93] to predict readmission using human and machine-derived features from EHRs. Choi et al [94] introduces graph convolutional transformer (GCT) to learn hidden structures of EHRs and perform predictions. Studies using data from the SNIIRAM database for predictive tasks are not numerous. In 2018, Janssoone et al. [95] compared multiple models to predict medication non-adherence using this database. Recently, Kabeshova et al. [20] presented ZiMM ED, a predictive model for the long-term prediction of adverse events. However, the performances of deep recurrent models have been tempered by Min et al. [96] in the context of readmission risk prediction after a hospitalization for chronic obstructive pulmonary disease (COPD). In their studies, machine learning methods using knowledge-driven and data-driven features achieved the best performances.

Embedding

As reported by recent reviews on the subject [87], [88], embedding applied on EHR data is also a topic of interest. Embedding for patient representation is widely tackled in the literature. Henriksson et al. [97] used multiple semantic spaces to learn a representation of free-text clinical notes and clinical codes such as diagnosis, drugs and measurements. They tested their method to provide input data to random forest, reducing sparsity and improving performances in adverse drug event detection tasks. Zhu et al. [98] focused on measuring patient similarities, and proposed a deep architecture using CNNs to learn a representation for patients which preserves the temporal ordering properties. In order to perform patient clustering, Landi et al. [99] recently used a convolutional AE to learn a latent representation of patients. Medical concept embedding is also useful to construct meaningful labels directly from data. Notable examples of medical concept embedding are Med2Vec [100], GRAM [101], and more recently Cui2vec [102].

1.2.4 Explainability

Identified as one of the main challenges regarding deep learning applied in health data, the explainability of deep learning methods is an important topic. But this research track is neither limited to healthcare nor to deep learning, and general works on the subject can be referred to one field: eXplainable Artificial Intelligence (XAI).

Overview

Explainability and interpretability in AI has become critical, motivating the expansion of the XAI field. As shown by Barredo Arrieta et al. [103], the number of publications related to the subject strongly increased in the past few years. On the one hand, interpretability is defined as "the ability to explain or to provide the meaning in understandable terms to a human" [103]. On the other hand, explainability is "associated with the notion of explanation as an interface between humans and a decision maker that is, at the same time, both an accurate proxy of the decision maker and comprehensible to humans" [103]. Explainability is more an active characteristic for the model, in opposite to interpretability which is a passive and inherent component of the method. Without being focused on deep learning methods, model-agnostic explainable frameworks exist in order to explain black-box model predictions. Examples are LIME [104], which uses linear models to approximate local behaviors, and SHAP [105], which uses Shapley values for both global and local interpretability. The main noted limitation of these model-agnostic methods is the need to run multiple evaluations of the model to provide interpretations, the use of such methods in practice being time consuming. Moreover, a recent comment by Rudin in 2019 [106] arbitrates for the use of intrinsically interpretative models for high stakes instead of trying to explain black box models.

Explainability of deep learning for EHRs

In some fields like healthcare, the chance of adoption of a method that relies on machine learning may be seriously limited [107]. When deployed on health data, the explainability of deep learning models becomes a promising research topic in recent years, as suggested by recent reviews [55], [87], [88], [90]. For example, Choi et al. presented RETAIN [108] in 2016. This two-level neural model is designed for an interpretation purpose, while keeping comparable performances. Interpretations are provided for a given patient, by giving importance of each element of its history. In 2017, Suresh et al. [109] benchmarked LSTM and CNNs for the prediction of clinical intervention using the MIMIC-III database. They used feature-level occlusion for LSTM and filter/activation visualization for CNN to explain predictions. RoMCP [110] was introduced by Xu et al. in 2018. This representation of clinical pathways has the advantage to capture both diagnostic information and temporal relations. Moreover, interpretations are provided using a top-k medical activities when predicting. The use of attention mechanism to provide explainability is also an actual topic of research. Patient2Vec [111], proposed by Zhang et al. the same year, is a framework which learns a deep representation of longitudinal EHR data which is interpretable, and personalized for each patient. RNNs are used to capture the relations between medical events and attention mechanism for personalized representation. LSTM with attention mechanisms was used by Kaji et al. [112] in 2019 to predict from ICU data. Personalized attention maps were presented in order to show how explainability can be presented, as the scale of a patient and form several prediction targets. Yin et al. [113] introduced DG-RNN in 2019 a model which incorporates medical knowledge graph information into LSTM, using

attention. Interpretations are provided by showing the contribution of medical events over time.

1.3 Process Mining

When working with processes, the most commonly encountered data format is event logs. These event logs regroup instances called traces, for which time-ordered events are listed. Each event carries single or multiple information, of various formats. To study such event logs, a field grew in the past 20 years, bridging process sciences and data sciences: process mining [22]. In this first section, an overview of main concepts and contributions of process mining is presented.

1.3.1 Recent Reviews

Recently, exhaustive mappings of the process mining field have been proposed. As an example, the systematic mapping of Maita et al. [114] depicts the main trends, topics and applications of process mining. 705 papers published from 2005 to 2014 are analyzed. The study shows an increasing dynamic in terms of the number of publications, year after year. "Business process discovery" is identified as the most frequent area of process mining (504 studies), in front of "Business process conformance" (259) and "Business process enhancement" (120). Regarding the tools used, ProM is undoubtedly the preferred choice, with 295 papers. In comparison, "in-house frameworks", which are the second most used category of tools, gather only 51 studies. However, in a non-negligible number of studies (304), no particular tool is mentioned. Regarding applied sectors where process mining methods are deployed, "Entreprise", "Medicine and Healthcare" and "Manufacture industry" are the three most represented in this systematic mapping (with 61, 59, and 48 papers, respec- tively). 104 papers are also identified as "Pure or theoretical research: algorithm", with no particular field of application. To the best of our knowledge, the most recent systematic mapping of process mining techniques and applications is the one of Garcia et al. [115]. 1, 278 papers published between 2002 and 2018 were selected to perform the systematic mapping. A useful labeling in terms of categories of process mining is presented. In addi- tion to the three widespread areas of process mining, "Supporting Area" was used to gather publications related to projects, applications and tools. 528 papers analyzed are part of the latter category, in front of "Discovery" (480), "Enhancement" (306) and "Conformance" (247). These results align with the first aim of the IEEE Task Force on Process Mining, which is to "promote the application of process mining" [116]. In the detail of application domains, 6 major area categories regroup almost 80% of the papers: "Healthcare" (162), "Information and communication technology" (95), "Manufacturing" (77), "Education" (61, also noticed by Maita et al. [114] as an emerging application), "Finance" (37), "Logistic" (27).

1.3.2 Process Discovery

Process discovery is, according to the previously presented reviews, the main process mining category. In the following, features of process discovery are discussed, starting with existing process discovery algorithms.

Process discovery algorithms

As mentioned before, performing process discovery is the most encountered task in the literature. Many process discovery algorithms exist, starting, according to Garcia et al. [115], in 1995 by Cook and Wolf [117].

In 2004, the α-algorithm [118] is introduced, being one of the pioneer process discovery algorithms, able to deal with concurrency [22]. Many improvements of this algorithm exist, developed in the past few years. Examples are the α^{+}-algorithm [119] introduced in 2004 to deal with repeated occurrences (short loops), or the α^{++}-algorithm [120] of 2007 (to model causal dependencies). In 2010, the $\alpha^{\#}$-algorithm [121] extent the α-algorithm by detecting invisible tasks from event logs (tasks that are not directly observable in the event log). The 2015 $\alpha^{\$}$-algorithm [122] is introduced to mine process models with invisible tasks but also dependencies. Heuristic mining algorithms deal with a major drawback of the previous described α-family algorithms, which is to take frequency into account [22]. In fact, considering the frequency of events and transitions in order to clean infrequent path from the resulting process model is valuable for practical use. It avoids the discovery of "spaghetti-like" process models, particularly when applying process discovery on real noisy event logs resulting from unstructured processes. The "HeuristicsMiner" [123] and its updated version the "FlexibleHeuristicsMiner" [124] are practical examples of such algorithms. The Fuzzy Miner [125], introduced by Gunther and van der Aalst in 2007, is another technique which deals with the frequency directly when constructing the dependency graph, being also able to extract hierarchical models [22]. The Genetic Miner [126] algorithm is a process discovery approach based on an evolutionary algorithm. Process discovery algorithms of this category are flexible, robust, but not very efficient for large process models and logs [22]. In 2012, the Evolutionary Tree Miner (ETM) is introduced by Buijs et al. [127]. The method uses an evolutionary algorithm and process trees to only consider sound process models as solutions, reducing the research space and improving performances. Moreover, the four quality dimensions are used to evaluate possible solutions. The Inductive Miner [128], introduced in 2013, also uses process trees to discover a directly-follows graph from an event log. This method is flexible, robust and do not suffer from scalability problems [22]. Even if some limitations have been noticed, contributions to leverage issues of the Inductive Miner have been proposed [129]–[131] (infrequent, incompleteness, directly-follows based). WoMan [132], a framework based on First order Logic, was introduced by Ferilli in 2014. The particularity of this framework is its ability to learn but also refine a process model, which makes it suitable for dynamic environments. Activity prediction is also possible to use this framework [133]. Chapela-Campa et al. introduced WoMine-i [134], which focuses on mining infrequent behaviors from event logs. In 2019, Augusto et al. introduced the Split Miner [135]. The method uses a directly-follows graph but analyzes loops and concurrency relations prior to the filtering step, the later allowing a balance between fitness and precision.

Thus, many process discovery algorithms have been introduced since 1995. In the recent years, improvements of existing algorithms but also new methods were proposed. In terms of practical deployments on real case studies, Garcia et al. noted that Heuristic miner and Fuzzy miner were the most frequently used [115]. Benchmarks of these algorithms were proposed by De Weerdt et al. [136] in 2012, and Augusto et. al. [137] in 2019. In the later, performances of actual state-of-the-art methods were compared on 24 real-life event logs. Conclusions drawn are that the Inductive Miner and the Evolutionary Tree Miner are able to achieve good performances regarding fitness, precision, and complexity. Also, the

Split Miner produces good results in terms of F-score. An observation made regarding very complex event logs is that the use of a filtering method prior to the discovery is necessary.

Modeling languages

A majority of process discovery algorithm used in practice are procedural, using common process modeling languages. For example, transition system is a notation which is composed of two main elements: states and transitions. Petri nets are the most investigated language used to model processes, which deal with concurrency. WorkFlow nets (WF-nets) are particular Petri Nets with a source to start, a sink to end, and for all nodes, a direct path from the start and to the end. Another useful and widespread language for business processes is the Business Process Modeling Notation (BPMN). When directly used to perform process discovery, a majority of the previous formalism are unsound (they carry undesirable properties which are independent of the input event log). Process trees is a notation useful to represent a category of models which is sound by construction: the block-structured models. Process trees are used by inductive process discovery techniques, as well as the Evolutionary Tree Miner. Causal nets (C-net) are representations which are less constraint in their formalism, making them suitable for many process discovery algorithm. Another family of modeling methods uses declarative statements to represent processes. These methods are referred to as Declarative Process Mining methods. In such languages like Declare for example, everything is possible unless explicitly forbidden [22].

Quality measures

In order to quantitatively measure the quality of a process model, four quality criteria are used [22]: fitness, simplicity, precision and generalization. The fitness measures the ability of a process model to represent a given event log. The simplicity characterizes the ability of the model to explain the event log while being simple. The precision is introduced to clarify that an acceptable model needs to be restricted to traces actually present in the log. The generalization states that the model should not focus too much to all the traces actually present in the considered event log. A parallel can be drawn between machine learning and process mining, with, on the one hand, precision/generalization, and, on the other hand, underfitting/overfitting. Thus, an opposition exists between fitness and simplicity, as it also exists between precision and generalization. As a result, one challenge of process discovery is to deal with the four aspects of the quality, which can depend on the case study and the intended purpose.

Readability of process models

As the aim of process discovery is the output of a visual representation of processes, the notion of readability and interpretation is a key issue in practice. The size of resulting models is one of the most notable parameters, but other factors exist. In their study, Reijers et al. [138] analyses the impact of personal and model factors using a questionnaire submitted to students regarding various process models. The results show a clear influence of personal factors (theory, practice, education) on understanding, much more explaining the understanding that other model-related factors. However, the two most impacting model factors were the average connector degree and the density. A recent contribution of van der Aalst [139] warns the practitioners on the use of directly-follows graph and frequency-based simplification. Such simplifications can produce different models, leading to different conclusions.

1.3.3 Augmented Process Mining

As process mining is presented by van der Aalst to be a bridge between process science and data science [22], trends in terms of combining process mining with other methods emerged. As an example, the use of simulation can turn process mining into a more forward-looking method [140]. In their work [141], Augusto et al. proposed a methodology which automatically creates a simulation model from raw event logs using process mining techniques. The method, tested on patient pathways extracted from the French national hospital database, appears to be useful to test different scenarios and policies. A similar methodology was used by Phan et al. [142], applied on the same type of data. Machine learning methods could also be useful when combined with process mining. An example is the work of Prodel et al. [143], where decision trees are fitted on patients' features in order to refine the process mining-based simulation. Predictive tasks in process mining are also practical perspectives, useful for many applications. In their work, Maquez-Chamoro et al. [144] present a survey on predictive monitoring of business processes. Prediction tasks identified were classified as numerical predictions (e.g. the time before the next event of the cost) or categorical predictions (e.g. the risk category or the next event) and a third more recent task which is the prediction of the next activities. A split in the results is presented, by considering the process-awareness of the methods. They noted that process-aware methods were more frequently used for regression. Results of the study show that a majority of the methods use the sequence of events for the prediction. Generally, an encoding of the sequences of events was noted by the authors, in order to produce vectors of features. The main reason is the use of well-known machine learning methods, which strictly use this kind of data to perform predictions. The history of events also need to be considered when encoding events. Finally, the lack of interpretability for predictive models is noticed, where the majority of the studies do not mention relevant explanation factors.

1.3.4 Process Mining and Healthcare

Process mining was largely applied on health data, for different case studies and various purposes.

Recent reviews

In a literature review, Rojas et al. [23] selected 74 papers focused on health processes (excluding clinical pathways related studies) and published before February 8, 2016. The majority of the studies were related to process discovery (control flow), often using Heuris- tic Miner and Fuzzy Miner. The choice of these methods mainly is driven by the noise and the less structured processes encountered in health data. Oncology, surgery and car- diology were the main medical fields of applications. One of the improvement tracks noted by the authors is the lack of a good visualization of the process models. This could be useful, particularly when working on the complex and less structured processes found in the healthcare domain. Moreover, a great amount of reliance on process mining experts is noted, motivating future efforts on developing more straightforward solutions. A sys- tematic review focused on healthcare was conducted by Erdogan and Tarhan in 2018 [24]. They selected 172 studies, between 2005 and 2017, with 93 on healthcare processes and 59 on clinical pathways. Generally, a dramatic increase in the volume of publication is noted in the past years. Process discovery is also the main objective of these studies (156 papers). The most frequently used process mining algorithms are the Heuristic Miner (39)

and Fuzzy Miner (28). But no more than 46 papers introduced new process discovery techniques. This may illustrate the need for custom-made methods for healthcare process mining, motivated by the complexity and the challenges of healthcare data. Oncology, surgery, emergency departments, neurological diseases and cardiovascular diseases were the most represented healthcare specialty. Finally, a few studies were deployed in multiple departments or multiple hospitals. This is a challenge pointed out by the authors, to be addressed in the future. In their general review on process mining [115], Garcia et al. dedicated a section on healthcare. Most contributions noted were related to clinical pathways, using conformance checking or process discovery. The topics of interest are numerous, but mostly centered on resource utilization, the identification of bottlenecks and the point- ing of potential improvements in processes. The high variability in health event logs is identified as a major challenge. Moreover, many context-specific information is needed to successfully conduct process discovery on health data. Thus, mobilizing medical experts in order to retrieve specific knowledge is necessary, but may also be challenging in some organizations. Two literature reviews focused on two healthcare specialties: oncology and cardiology. In the first review, Kurniati et al. [145] identified 37 papers published between 2008 and 2016. One suggestion noted is the need to develop new mining techniques to obtain understandable and high-level information. In the review of Kusuma et al. [146], 32 papers were selected. In these two studies, major limitations were related to the data (missing values, noise, dimensionality and complexity).

Clinical pathways

To use the definition given by Yang et al. [147] and inspired by Ireson [148]: "Clinical pathways (CP) is a structured, multidisciplinary, patient care plan in which diagnostic and therapeutic interventions performed by physicians, nurses, and other staff for a particular diagnosis or procedure are sequenced on a timeline". Yang et al., in their literature review of process mining studies applied on clinical pathways, selected 37 studies [147]. Results of their works are multiple. Firstly, the identification of variants is a key point to adapt the clinical pathway (leading to its adjustment or its redesign). Related to adjustment, the self-learning improvement of the clinical pathway is also necessary. Moreover, the diversity of events and the complexity of chronic disease in clinical pathways are challenging. These complexities make traditional process mining algorithms not practically suitable for clinical pathways [147]. Finally, the entire medical process needs to be considered when studying clinical pathways in hospital. The integration of prevention and pre-hospitalization (before) and rehabilitation (after) is necessary to fully consider the patient journey. In fewer details, Baker et al. [149] used routinely collected EHR data from a UK oncology center. After eight iterations over nine months, a final process model was produced, resulting in a gap between real processes and assumed care pathways. The case study depicted in the paper of Prodel et. al. [160] show how process mining can model the clinical pathway of patients being implemented of an implantable cardioverter defibrillators. An event log extracted from the PMSI was used. Opportunities and challenges regarding clinical guidelines and process mining have been discussed in the work of Gatta et al. [151], where 12 research centers were included in a collegial discussion. According to the authors, one role of process mining for physicians in this context is to check how the patients conform to clinical guidelines but also to identify patients who do not conform in order to identify the reasons and the implications. This problematic is complex, as described by Gatta et al.: "we need to provide models formal but understandable; complete, but usable; standard, but adaptable;

specific, but flexible; general, but personalized..." [151]. In their study, Rovani et al. [152] used declarative process mining to analyze medical treatment processes. By using, on the one hand, real process data, and, on the other hand, clinical guidelines, they discovered deviations and created a compromise model. The conducted case study concerned patients affected by cryptorchidism, the data being extracted from the urology department of the Isala hospital. Referring to "patient trajectories", Mannhardt et al. [153] analyzed the stay of sepsis patients within a Dutch hospital. By using procedural process mining, they show that this approach could still by applied in healthcare context. Many practical challenges were noticed, as the absence of flexibility when applying an end-to-end process mining project on a particular sub-group of patients. Also, the first obtained model was not totally recognized by the people working in the process. An iterative work with experts seems to be inevitable to produce accurate results.

Disease trajectories

The study of disease trajectory is another topic of interest when working on health processes. In such studies, the goal is to identify relations between disease and their progressions in time. In 2014, Jensen et al. [154] studied disease correlations and temporal disease progressions at a very large scale. Using the Danish national hospital database, they studied the data of 6.2 million patients, recorded for 14.9 years, and identified 1, 171 trajectories. Pathology-centered clusters were created, using ICD-10 codes to characterize the diseases. Relative risks (RR) were computed to measure diagnosis pair correlation. In 2020, Kusuma et al. [155] show how the use of process mining could be valuable to study disease trajectories. By using EHR data, they extracted an event log of disease by selecting the first occurrence for each patient. This feasibility study shows an example of a recent topic where process mining could be useful.

Health processes challenges

As the main noted challenges when deploying process mining on health data is the data itself, some studies addressed some data-related issues. In the work of Alharbi et al. [156], the problem of repeated event selection and removing was tackled. By using an interval-based selection method, the authors reduced the number of repeated events, deleting outliers. The method was applied on a case study regarding diabetes, using the MIMIC-III database. As pointed out by Kaymak et al. [157], when working on health processes, the data need to be reprocessed in order to keep the correct level of granularity. The work of Hompes et al. [158] shows how the analysis of a causality matrix serves to group similar events and improve the labeling of event logs. Alharbi et al. [159] used HMMs to help in preprocessing events for process mining. By automatically detecting hidden patterns, the method is a useful preprocessing step which does not need domain experts. In their work, Prodel et al. [160] used an automatic approach to deal with the hierarchy of ICD-10 codes directly during the process discovery. Considering outpatient clinic's appointment, Martin et al. [161] propose an interactive approach for process mining data cleaning. The data-based cleaning (missing or overlapping timestamps, time ordering violation, appointment using the same resource) on the one hand, and the discovery-based cleaning (mainly anomalies regarding expert knowledge of the process) on the other hand, are considered. Other challenges such as working with text and unstructured data were also tackled. An example is the work of Weber et al. [150], where free text and web scrapping were used to label events of the log.

1.4 Context and Positioning

Starting in 2017 from our knowledge about the literature, preliminary predictive analyses were conducted. The first objective was to perform predictions using machine learning algorithms applied on the PMSI. In the literature, a shortlist of machine learning algorithms for binary predictions was identified, without having one method clearly outperforming the others. Moreover, hyperparameter tuning was identified as a challenge in deploying these methods in practice for healthcare. As a result, a benchmark of 7 machine learning algorithms was performed, on 3 case studies where features were defined using medical expert knowledge. Efficient Global Optimization (EGO) [162] algorithm was used for hyperparameter tuning. According to the results obtained, hyperparameter tuning is mandatory in order to avoid overfitting, particularly for tree-based methods. Moreover, random forest was identified as the best method on the 3 case studies. These results gave inputs for future studies deployed on the PMSI database. These results are published and were presented at the 2018 *IEEE International Conference on Systems, Man and Cybernetics* [163].

However, no real pathway information was used in this study. In order to verify that patient history was valuable to improve performances, a case study about predicting readmission after a stroke episode was conducted. Patients with a stroke episode in 2015 were selected, using data extracted from the PMSI. 3, 787 binary sparse features were created, from patient information, diagnosis, medical procedures, and medical unit information. These features were created with and without considering patients' history (all the previous medical stays between 2010 and the stroke episode for each patient). As the case study was really unbalanced, two balancing methods were compared. The first one was a random undersampling strategy. In the second one, training elements of the majority class were shuffled and grouped in batches of the size of the minority class. Then, for each batch, a classifier was fitted using majority class observations from the batch and all observations from the minority class. Finally, the classification of patients from the test set is used by taking the majority vote of all the classifiers. The classifier used was a simple decision tree in order to perform explainable predictions. The results show that the balancing method based on a majority vote generally outperformed the simple undersampling method. Moreover, adding the history of the patient improves the predictions, which motivates the future works on modeling patient history when training predictive methods. This work has been presented at the 2018 *European Conference on Operational Research*[3].

Starting from these results, the modeling of history appears as an essential challenge for prediction. This information, contained in patient history, is mainly structured as a patient pathway when working on non-clinical claims data. As a result, some challenges were identified in order to properly model patient pathways:

- The modeling of time, in order to capture logic transitions between events, but also modeling the time between two particular medical events;
- The complexity of macro-medical events, as it consists of multiple codes from different hierarchy systems;
- The need of constructing methods which are interpretative or explainable, in order to facilitate discussions with all the stakeholders, to identify potential biases or inequalities, and also facilitating practical deployment.

[3]Presentation material is accessible here: http://doi.org/10.13140/RG.2.2.26336.10249

Thus, the use of process mining to perform prediction appeared as a first track. As patient pathways can be seen as processes, and process mining bridges process science and data science according to van der Aalst [22], the starting point of our work focused on developing predictive modeling using process mining. Starting from the previous work of Prodel et al. [141], [143], [160], we introduce new process models which accurately model and capture time patterns during the optimization process of discovery. This work is presented in Chapter 2.

The second challenge is the modeling of complex medical events such as the ones found in non-clinical claims data. As these events were either manually defined by experts rules or partial information, the help of domain experts to improve the quality of events in event logs is necessary, but can be expensive and time consuming [159]. As a result, Chapter 3 presents a preprocessing step which automatically define events prior to the use of process discovery from event logs. As noted by Mannhardt et al. [153], working on health data to perform process mining studies is an iterative process. Thus, this contribution serves as a tool to facilitate such iterative discussions with medical experts, based on AEs to perform latent space clustering. The use of AEs to preprocess event logs were also used by Nguyen et al. in 2019 [164] in order to improve the quality of event logs by dealing with anomalies or missing values.

As Chapters 2 and Chapters 3 introduce contributions in modeling complex sequential data, Chapter 4 introduces an optimization-based predictive methods. By using the process models presented in Chapter 2, starting from event logs of positive and negative classes, the optimization process outputs a process model which characterizes the positive class while badly representing the negative one. Predictions are provided for a new trace by computing its fitness regarding the previously fitted process model. As a result, patterns observed on the latter process model highlights distinctive patterns of the positive class, in terms of events, transitions, and time characteristics. These patterns constitute a support for global explainability.

This method, based on process models, suits well for macro medical events. Unfortunately, performances are impacted when working with more frequent events. To also deal with frequent events, another method to model, predict and explain from patient pathways is presented in Chapter 5. By using a VAE, an end-to-end methodology is presented, Starting from a raw event log, time and codes' hierarchy are models in order to keep all the relevant information. Predictions are provided for an individual patient pathway, and a global explanation is deduced from the proposed model. This contribution is an alternative to widespread, interpretable but simple algorithms (such as decision tree), high performing but deep complex networks based on attention mechanisms, or to the use of model-agnostic explainable frameworks (i.e. SHAP or LIME).

Chapter 2

Optimal Process Mining of Timed Event Logs

Contents of the chapter

2.1 Motivation

The field of process mining proposes methods and tools to work on patient pathways data. As part of process mining, process discovery serves to mine patterns directly from event logs. The initial motivation of this work is the idea of using process mining for explainable predictive analysis of patient pathways. For the purpose of prediction, the extraction of distinctive patterns from pathways is performed first. Starting from the work of Prodel et al. [160], which proposes a process discovery algorithm adapted to patient pathways, some modeling limitations were identified. Patterns such as the repetition of a given event but also the time separating two events were noticed as being neither explicitly modeled nor embedded in the optimization process. Moreover, readability of a process model in the context of patient pathways appears to be impaired by self-loops and backward edges cutting the linearity of pathways observed in practice.

2.2 Summary

In this chapter, a new problem is introduced, which is the problem of determining the optimal process model of an event log of traces of events with temporal information. To solve this problem, two new process models are introduced: the *grid process model* and the *time grid process model*. The first one is reminiscent of Petri net unfolding, and is a graph

with multiple layers of labeled nodes and arcs connecting lower to upper layer nodes. The second one is an extension of the first, which assigns a time interval to each arc, with the particularity to have multiple edges connecting the same nodes. The property is useful when defining characteristic time patterns in the model and incorporate such information directly during the optimization process. Moreover, the two previously defined process models address the problem of impaired readability described before. The fitness measure is adapted and a new replayability score is defined. The optimization procedure produces a process model which maximizes the replayability score, under the constraints of the maximal number of nodes and edges. As demonstrated in this chapter, the replayability game for defined process models does not depend on the choice of edges. As a result, for a given node configurations, optimal edges can be selected under the size constraint, which reduces the search space by only selecting accurate node configurations. In order to capture time characteristic patterns, Kernel Density Estimation (KDE) is used to model time transition distributions and automatically define adequate intervals from the event log before the beginning of the optimization. Experiments are conducted on synthetic and noisy event logs, in order to compare performances of the tabu search with other heuristics. The results validate the good performances of the method, particularly when the property of independence of the replayability regarding the edges is used. Finally, a real-life case study is presented, to analyze the pathways of diabetic patients before the occurrence of certain complications.

2.3 Optimal Process Mining of Timed Event Logs

H. De Oliveira, V. Augusto, B. Jouaneton, L. Lamarsalle, M. Prodel and X. Xie, "Optimal process mining of timed event logs", *Information Sciences*, vol. 528, pp. 58-78, 2020, https://doi.org/10.1016/j.ins.2020.04.020.

Optimal Process Mining of Timed Event Logs

Hugo De Oliveira[a,b], Vincent Augusto[a], Baptiste Jouaneton[b], Ludovic Lamarsalle[b], Martin Prodel[b], Xiaolan Xie[a,c]

[a]*Mines Saint-Etienne, Univ Clermont Auvergne, CNRS, UMR 6158 LIMOS, Centre CIS, F - 42023 Saint-Etienne France*
[b] *HEVA, 186 avenue Thiers, F-69465, Lyon, France*
[c]*Antai College of Economics and Management, Shanghai Jiao Tong University, China*

Abstract

The problem of determining the optimal process model of an event log of traces of events with temporal information is presented. A formal description of the event log and relevant complexity measures are detailed. Then the process model and its replayability score that measures model fitness with respect to the event log are defined. Two process models are formulated, taking into account temporal information. The first, called grid process model, is reminiscent of Petri net unfolding and is a graph with multiple layers of labeled nodes and arcs connecting lower to upper layer nodes. Our second model is an extension of the first. Denoted the time grid process model, it associates a time interval to each arc. Subsequently, a Tabu search algorithm is constructed to determine the optimal process model that maximizes the replayability score subject to the constraints of the maximal number of nodes and arcs. Numerical experiments are conducted to assess the performance of the proposed Tabu search algorithm. Lastly, a healthcare case study was conducted to demonstrate the applicability of our approach for clinical pathway modeling. Special attention was paid on readability, so that final users could interpret the process mining results.

Keywords: process mining, event log, time modeling, tabu search, healthcare data, patient pathways.

1. Introduction

The digital revolution affects all industries and businesses and produces a huge amount of data. Numerous decision aid analytical methods and tools are available to take advantage on relevant data. Machine Learning methods have been widely used. Supervised and unsupervised learning for knowledge discovery, when applied to matrix data where each row is an observation characterized by features in columns, have spread over.

Knowledge discovery from healthcare data, such as patient lifetime hospitalization history is presently not optimal yet crucial. Therefore, we want to find common patterns, process models, or care pathways of hospitalization histories for cohorts honed to a specific time periods. Such studies are important to detect relevant "causal" relationships or transitions, to check the conformance of practice to guidelines, etc. An example of a causal relation could be "most patients of a given surgery had the same prior underlining condition or past medical event".

Traditional machine learning techniques are not well suited to generate process models from data. Process mining is well suited for this purpose, and has been first formalized in 2004 [14], followed by developments in various fields [7] including healthcare [1]. Temporal information such as the time between two events and the number of repetitions of a given event in the past are particularly important in process modeling and prediction. In healthcare, a second hospital visit shortly after the first, unfortunately is likely to be an undesirable complication or result of the earlier admission. A patient having been hospitalized several times for a given disease or had a much longer prior hospital stay before recovery is more likely to need a non-standard surgery than a patient having no or just one past hospitalization. Unfortunately, such features are rarely taken into account by process mining approaches. For example, repeated events are prohibited for the sake of visibility and time is not considered during model construction. A previous study on the understanding of process models found that the average connector degree and density are two identified factors which induce negative effects on comprehension (at a fixed size) [11].

Starting from these limitations regarding time consideration and comprehension, an extension of existing optimal process mining theory is presented. Thus, the main scientific contribution of this paper is the mathematical formalization of a new ascending and time-dependent process mining approach, structured on a grid for a better representation and understanding. Reminiscent of Petri net unfolding [2], ascending representation forbids loops and confusing backward transitions, whereas the time-dependent feature considers time patterns of event logs during optimization. Descriptors are introduced, characterizing event logs complexity and process models structure. The replayability indicator [10] is slightly modified and acts as a key performance indicator to evaluate the resulting process models. Finally, a new Tabu search procedure for process mining optimization is presented, tested and validated through a series of designed experiments as well as a real-life healthcare case study.

The rest of this paper is organized as follows. A literature review focused on recent work in process mining with a particular focus on healthcare is presented in Section 2. General definitions of event log representation are given in Section 3. A mathematical formulation of the new process models and optimization problems is presented in Section 4. Section 5 describes the process discovery methods involved and Section 6 details in depth computational experiments designed to test methods on different simulated event logs. A case study based on real data is presented in Section 7. Finally, conclusion and perspectives are given in Section 8.

2. Literature Review

The raison d'être of Process mining is to discover, monitor and ameliorate actual processes as they occur by extracting knowledge from event logs readily available in I.T. systems. Process mining is situated between Big Data and Data Mining on one side, and Business Process Modeling and Analysis on the other. The field of process mining can be divided in 3 main areas: process discovery, conformance checking and extension of a model [13].

Recently, two systematic studies mapped out Process Mining [7, 1], which are valuable to describe the scope and the dynamics of this field. Maita et al. regrouped in their study 705 papers about process mining from 2005 to 2014 [7]. The number of publications addressing process mining has significantly increased from 2005 to 2014. "Discovery" is the main purpose for the use of process mining (71% of papers), with "graph structure-based techniques" being the most common intersection. For studies mentioning a specific application domain, "Medicine and Healthcare" is the second most frequent domain just after the overall sector "Enterprise". Moreover, "clinical analysis pathway" ranks third as data sources used for case studies or experiments. Importantly, the majority of these studies does not mention any process mining tool being used. For the rest, `ProM` is the most often mentioned with in-house frameworks a distant second.

A similar work [1] on healthcare studies considers 172 papers from 2005 to 2017. Observations detail a rapid expansion of the field, giving new opportunities for further research and practice. Process discovery appears as the most important activity of process mining when applied to healthcare. Studies on "Healthcare process" (93) are more common than "Clinical pathway" (59). Furthermore, studies corresponding to "Multiple hospital" data are less frequent (14) compared to "Single hospital" data (130). The same observation is true for studies regarding "Multiple department" data (13) and "Single department" data (122). Finally, the number of studies which propose a new tool, model or metric is low (17%). This limited body of relevant papers in our field seems to imply the originality of our contribution.

Since the later systematic review on healthcare, new process mining contributions applied to healthcare have been proposed. Kusuma et al. compiled a literature review of process mining in cardiology [5]. Promising opportunities to assist medical experts in care analysis were shown, although few formal process mining methodologies were included. Litchfield et al. [6] introduced a study protocol to apply process mining to primary care in the UK for the first time. The use of orthodox process mapping in addition to data-driven process mining is presented as useful to identify differences and similarities.

Similar works on patient pathways mining were published using discrete optimization [8] and simulation [9]. The most related paper to our work is of Prodel et al. in 2018, where a mathematical formulation of the problem was presented, along with a Tabu search optimization process to search for best process model. In addition, to reduce the computation time for large-scale data sets, they used a Monte Carlo sampling

method. As healthcare data often contains hierarchical structure (ICD-10 codes for example), this data characteristic for labels was considered during the optimization process [9]. The large scale problematic and the hierarchical structure of labels were also addressed.

There are two limitations. First, qualitative representation of pathways with repeated events is not readily understood by decision makers and clinicians. Indeed, a repeated event pattern in the event log will be represented by a loop, which does not take into account the linear characteristic of patient pathways. More importantly, the time is not considered during the optimization process. The including of time as a parameter could be beneficial to highlight hidden time patterns contained in the event log.

Therefore, two extended process mining approaches are presented in this paper. New descriptors to characterize data sets and newly formulated process models are also presented. The Tabu search used by Prodel et al. [10] is tested and compared to other methods. Moreover, an improved version of the Tabu Search algorithm adapted to new process model is suggested.

3. Event, Trace and Log

This section provides a formal description of data involved including events, traces and event logs.

Definition 1. *(Event). Each event denoted e is a couple (a, t) where $a \in A$ is an element of a finite set A of labels corresponding to the event class of e, and $t \in T$ with $T = \mathbb{N}$ or $\mathbb{R}$ corresponding to the event time also called time-stamp. An event e is then equivalently defined by the two following functions:*

- $label(e) = a$ *called labeling function;*
- $time(e) = t$ *called timing function.*

Definition 2. *(Trace). A trace is a sequence of events denoted as $\sigma = e_1, e_2, \ldots, e_m$ with $m \in \mathbb{N}^*$ such that $e_k \in A \times T$ and $time(e_k) < time(e_{k+1})$. A trace leads to the following functions:*

- $trace(e_k)$ *denoting the trace ID of each event;*
- $position(e_k) = k$ *denoting the order of the event in the trace;*
- $|\sigma| = m$ *denoting the length of the trace.*

Definition 3. *(Event log). An event log is a set of traces denoted as $L = \{\sigma_1, \sigma_2, \ldots\}$ σ_n with $n \in \mathbb{N}^*$. An event log contains all input data of this paper. Without loss of generality, we assume that each label appears at least once in the event log L, i.e. $\forall a \in A : \exists \sigma = e_1, \ldots, e_m \in L, \exists e \in \{e_1, \ldots, e_m\} | e = (a, t)$. The set of possible combinations of labels and positions in L is:*

$$A_{lab,pos} = \bigcup_{\sigma = e_1, \ldots, e_m \in L} \{(label(e_1), 1), \ldots, (label(e_m), m)\}$$

Definition 4. *(Causal relations). For each trace $\sigma = e_1, \ldots, e_m$ in L, all pairs of labels $(label(e_k), label(e_l))$ such that $1 \leq k < l \leq m$ are called causal relations and pairs $(label(e_k), label(e_{k+1}))$ are called direct causal relations. Let T^C be the set of all causal relations and T^{DC} be the set of direct causal relations.*

Definition 5. *(Timed causal relations). All triplets $(label(e_k), label(e_l), time(e_l) - time(e_k))$ such that $(label(e_k), label(e_l)) \in T^C$ (or T^{DC}) are called timed causal relations (or direct timed causal relations). Let T^{tC} and T^{tDC} be the set of all timed causal relations and the set of direct timed causal relations.*

Definition 6. *(Diversity measures). The event log L has the following diversity measures:*

- *event diversity* $div_e = |A|$;
- *event-position diversity* $div_{e,p} = |A_{lab,pos}|$;
- *causal relation diversity* $div_{causal} = |T^{DC}|$;
- *timed causal relation diversity* $div_{t-causal} = |T^{tDC}|$.

Event log definitions
Label set: A
Event: $e_k = (a, t)$
$a \in A$, $t \in T$ with $T = \mathbb{N}$ or $\mathbb{R}$
Trace: $\sigma = e_1, \ldots, e_m$
Log: $L = \{\sigma_1, \ldots, \sigma_n\}$
Label: $label(e_k) = a$
Time: $time(e_k) = t$
ID: $trace(e_k)$
Position: $position(e_k) = k$
Relation sets
Causal relations set: $T^{C} = \{(label(e_k), label(e_l))\}_{\substack{1 \le k < l \le m \\ \sigma = e_1, \ldots, e_m \in L}}$
Direct causal relations set: $T^{DC} = \{(label(e_k), label(e_{k+1}))\}_{\substack{1 \le k < m \\ \sigma = e_1, \ldots, e_m \in L}}$
Timed causal relations set: $T^{tC} = \{(label(e_k), label(e_l), time(e_l) - time(e_k))\}_{\substack{1 \le k < l \le m \\ \sigma = e_1, \ldots, e_m \in L}}$
Timed direct causal relations set: $T^{tDC} = \{(label(e_k), label(e_{k+1}), time(e_{k+1}) - time(e_k))\}_{\substack{1 \le k < m \\ \sigma = e_1, \ldots, e_m \in L}}$
Descriptors
Traces length description: $\lvert\sigma\rvert_{mean}$, $\lvert\sigma\rvert_{std}$, $\lvert\sigma\rvert_{max}$, $\lvert\sigma\rvert_{min}$
Event diversity: $div_e = \lvert A \rvert$
Event-position diversity: $div_{e,p} = \lvert A_{lab,pos} \rvert$
Causal relation diversity: $div_{causal} = \lvert T^{DC} \rvert$
Timed causal relation diversity: $div_{t-causal} = \lvert T^{tDC} \rvert$

Table 1: Event log notations.

Proposition 1. If $A \neq \emptyset$ and $\exists \sigma$ $\lvert\sigma\rvert > 1$ for at least one trace σ, then $1 \le div_e \le div_{e,p} \le \sum_{\sigma \in L} \lvert\sigma\rvert$ and $1 \le div_{causal} \le div_{t-causal} \le \sum_{\sigma \in L} (\lvert\sigma\rvert - 1)$

Proof. Trivial by definitions. □

The diversity measures characterize the log's complexity: a high diversity means an increased number of different elements in terms of labels and causal relations. Also, the distribution of trace lengths is a valuable predictor to assess event logs' complexity (let $\lvert\sigma\rvert_{mean}$, $\lvert\sigma\rvert_{std}$, $\lvert\sigma\rvert_{max}$ and $\lvert\sigma\rvert_{min}$ be its mean, standard deviation, maximum and minimum values, respectively). Notations of event logs are summarized in Table 1.

Example 1. *Table 2 shows a short event log related to patient hospitalization pathways. Each row is a hospitalization event. Events with the same ID are ordered by increasing time stamp and represent a trace. Each trace is a patient's hospitalization history. Each first event of a patient has a timestamp set to* 0*. By clustering hospitalization by main diagnosis, the set of labels is as follows :*

- $A_{lab} = \{I500, I44.2, I621, G935, E149, I272\}$.

Alternative event clustering is possible by taking into account more detailed information such as the duration and secondary treatments during the hospitalization.

Thus, event-position set is as follows:

- $A_{lab,pos} = \{(I500, 1), (I44.2, 2), (I621, 3), (G935, 4), (E149, 2), (I272, 3), (I500, 4), (I44.2, 5)\}$.

Basic and timed transitions are:

- $T^{C} = \{(I500, I44.2), (I500, I621), (I500, G935), (I44.2, I621), (I44.2, G935), (I621, G935), (I500, E149), (I500, I272), (I500, I500), (E149, I272), (E149, I500), (E149, I44.2), (I272, I500), (I272, I44.2))\}$

ID	time-stamp	duration	main diagnosis	DRG
0	0	8	I500	05M092
0	30	0	I44.2	05C142
0	72	1	I621	01M311
0	103	3	G935	01M131
1	0	2	I500	05M092
1	5	0	E149	10M02T
1	93	2	I272	05M172
1	145	1	I500	05M092
1	180	8	I44.2	05C142

Table 2: An example of an event log of patient pathways.

- T^{DC} = {(I500, I44.2), (I44.2, I621), (I621, G935), (I500, E149), (E149, I272), (I272, I500))};
- T^{tC} = {(I500, I44.2, 30), (I500, I621, 72), (I500, G935, 103), (I44.2, I621, 42), (I44.2, G935, 73), (I621, G935, 31), (I500, E149, 5), (I500, I272, 93), (I500, I500, 145), (I500, I44.2, 180), (E149, I272, 88), (E149, I500, 140), (E149, I44.2, 175), (I272, I500, 52), (I272, I44.2, 87), (I500, I44.2, 35))};
- T^{tDC} = {(I500, I44.2, 30), (I44.2, I621, 42), (I621, G935, 31), (I500, E149, 5), (E149, I272, 88), (I272, I500, 52), (I500, I44.2, 35))}.

Finally, diversity descriptors are:

- $div_e = 6$; $div_{e,p} = 8$; $div_{causal} = 6$; $div_{t-causal} = 7$.

4. Grid Process Model Optimization Problem

This section is dedicated to discover process models that best fit the event logs given in Section 3. For this purpose, first the concepts of grid process models and time grid process models are introduced. Subsequently, the fitness measure is introduced to measure how well a process model captures the causal relations of an event log. We terminate by a formal definition of the process model optimization problem.

4.1. GridProcess Models

Definition 7. *(Grid process model). A grid process model of a given log L is a triplet $G\text{-}PsM = (N, E, L)$ where:*

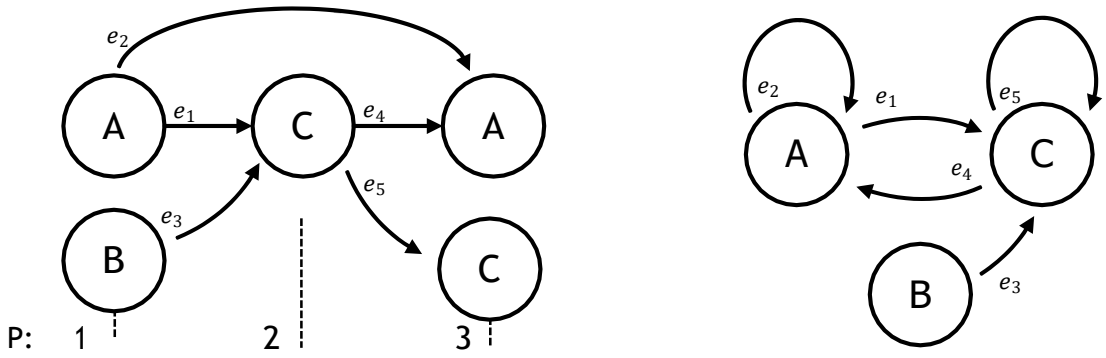
- *N is a set of nodes partitioned into K disjoint subsets called layers, i.e. $N = N_1 \cup \cdots \cup N_k$, $N_k \cap N_l = \emptyset$;*
- *$E \subset N \times N$ is a set of edges such that $(x, y) \subset E$ with $x \in N_k$, $y \in N_l$ implies $k < l$, i.e. the process model is acyclic with edges going from lower layers to higher layers;*
- *$L : N \rightarrow A$ is the labeling function of the nodes.*

From the above definition, one can also define the position function $P(x) = k$, if $x \in N_k$. As a result, $(x, y) \subset E$ implies $P(x) < P(y)$.

The main difference from the previous process model definition in [10] is the possibility for an event to appear at various positions in a trace. For example, an event happening both at the beginning and at the end of a patient pathway could now be described in the process model by two nodes, one with low and the other with a high position. In this case, loops on the same node and backward edges are not allowed anymore. An example of a *G-PsM* and its process model equivalent are given in Figure 1.

Example 2. *The Figure 1a shows a grid process model $G\text{-}PsM = (N, E, L)$ with:*

$$N = (n_1, n_2, n_3, n_4, n_5) \quad E = (e_1, e_2, e_3, e_4, e_5)$$



(a) Grid process model. (b) Equivalent classic process model.

Figure 1: Example of a grid process model and its classic process model equivalent.

$$L(n_1) = A \; P(n_1) = 1 \; e_1 = (n_1, n_3)$$
$$L(n_2) = B \; P(n_2) = 1 \; e_2 = (n_1, n_4)$$
$$L(n_3) = C \; P(n_3) = 2 \; e_3 = (n_2, n_3)$$
$$L(n_4) = A \; P(n_4) = 3 \; e_4 = (n_3, n_4)$$
$$L(n_5) = C \quad P(n_5) = 3 \quad e_5 = (n_3, n_5)$$

An equivalent classic process model is presented in Figure 1b, without duplicated label nodes, allowing loops and backward transitions.

4.2. Time Grid Process Models

To the best of our knowledge, in existing process mining approaches, time-related information is added after model discovery and remains descriptive. Significantly, we provide a new approach to include time-related information within the optimization process of building a process model.

Definition 8. *(Time grid process model). A time grid process model of a given log L is a four-uplet T G-PsM = (N, E, L, T) where:*

- (N, E, L) *is a grid process model with eventually multiple edges between nodes;*
- $T : E \to T \times T$ *associates a time interval* $[a_{(x,y)}, b_{(x,y)}]$ *to each edge* $(x, y) \in E$.

As shown in Figure 2, using this definition, previous unique edges between two nodes in *G-PsM* are replaced by multiple possible edges in T *G-PsM* , each of them having its own time interval. Definition 8 ensures that a given causal relation, with a given time value between two events, will be characterized by a unique possible edge with same starting and ending node in the process model. The uniqueness of characterization will be useful for graph construction and replayability defined thereafter.

Example 3. *The Figure 2 shows a time grid process model T G-PsM = (N, E, L, T) with:*

$$N = (n_1, n_2, n_3, n_4) \quad E = (e_1, e_2, e_3, e_4, e_5, e_6, e_7)$$

$$\begin{array}{llll} L(n_1) = A & P(n_1) = 1 & e_1 = (n_1, n_2) & T(e_1) = [10, 24] \\ L(n_2) = B & P(n_2) = 2 & e_2 = (n_1, n_2) & T(e_2) = [30, 35] \\ L(n_3) = C & P(n_3) = 2 & e_3 = (n_1, n_2) & T(e_3) = [45, 62] \\ L(n_4) = A & P(n_4) = 3 & e_4 = (n_2, n_4) & T(e_4) = [0, 5] \\ & & e_5 = (n_2, n_4) & T(e_5) = [5, 25] \\ & & e_6 = (n_3, n_4) & T(e_6) = [2, 50] \\ & & e_7 = (n_3, n_4) & T(e_7) = [75, 100] \end{array}$$

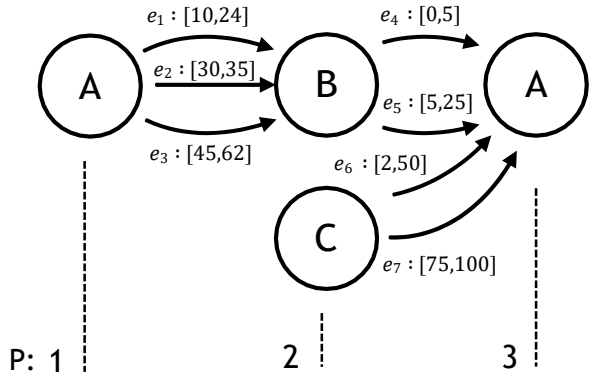


Figure 2: Example of a time grid process model.

G-PsM	TG-PsM
$G\text{-}PsM = (N, E, L)$	$TG\text{-}PsM = (N, E, L, T)$
$N = N_1 \cup \cdots \cup N_k$	$N = N_1 \cup \cdots \cup N_k$
$L(x) \in A, x \in N$	$L(x) \in A, x \in N$
$P(x) = k, x \in N_k$	$P(x) = l, x \in N_l$
	$T((x, y)) = [a_{(x,y)}, b_{(x,y)}], (x, y) \subset E$
$Div_N = \lvert\{L(n)\}_{n \in N}\rvert$	$Div_N = \lvert\{L(n)\}_{n \in N}\rvert$
	$Div_E = \lvert\{(L(x), P(x), L(y), P(y))\}_{(x,y) \subset E}\rvert$

Table 3: Process model related notations.

4.3. Process Model Complexity Characterization

The larger a process model is, the better it represents traces from an event log, but the drawback is that the more convoluted it becomes for someone to understand. Building a process model while controlling its complexity is crucial. A process model's complexity is described by its number of nodes $|N|$ and edges $|E|$. Process model diversities are defined similarly to event log diversities (Definition 6).

Definition 9. *(Node and edge diversities). For a process model G-PsM or T G-PsM, the node diversity is* $Div_N = \lvert\{L(n)\}_{n \in N}\rvert$. *For a time grid process model T G-PsM, the edge diversity is* $Div_E = \lvert\{(L(x), P(x), L(y), P(y))\}_{(x,y) \subset E}\rvert$.

Diversity descriptors characterize the variety of nodes and edges of a process model. A high diversity means that only few nodes (or edges) have the same label, whereas a low diversity indicates many similarly labeled nodes (or edges). For instance, in the *G-PsM* of Figure 1a, $|N| = 5$, $Div_N = 3$ and $|E| = 5$, and in the *T G-PsM* presented in Figure 2, $|N| = 4$, $Div_N = 3$, $|E| = 7$ and $Div_E = 3$. This simple example highlights the increase of $|E|$ for *T G-PsM* compared with that of *G-PsM*. Notations of the process models are summarized in Table 3.

4.4. Replayability

To evaluate the capacity of a process model to represent a trace, a new replayability score has been devised to match the newly defined grid process models. Initially, preliminary definitions are required.

Definition 10. *(Replayability). An event e is said replayed by a process model G-PsM if* $label(e) = L(x)$ *for some node x of G-PsM. A causal relation* (e_k, e_l) *is said basic replayed by G-PsM if* $label(e_k) = L(x)$ *and* $label(e_l) = L(y)$ *for some edge* (x, y) *of G-PsM. A timed causal relation* $(e_k, e_l, time(e_l) - time(e_k))$ *is said time-replayed by an T G-PsM if* $label(e_k) = L(x)$, $label(e_l) = L(y)$ *and* $time(e_l) - time(e_k) \in [a_{(x,y)}, b_{(x,y)}]$ *for some edge* (x, y) *of the TG-PsM.*

Algorithm 1 Grid and Time Grid Replayability Games.

1: **Initialization**
2: $z \leftarrow 0, \delta \leftarrow 0, \varphi \leftarrow 0, m \leftarrow 0$
3: Find c_m the first replayed event of σ
4: If no event of σ replayed:
5: **return** z, δ, φ
6: Else:
7: Set N_{actual} as the node which replayed c_m
8: $z \leftarrow z + 1$
9: **Trace crossing**
10: While $m < |\sigma|$:
11: If c_{m+1} can be replayed in *G-PsM* by a node N_{next} with $P(N_{actual}) < P(N_{next})$:
12: $z \leftarrow z + 1$, $N_{actual} \leftarrow N_{next}$ (with the lowest position)
13: If no edge exists between N_{actual} and N_{next} (or if the transition between (c_m, c_{m+1}) cannot be time replayed *(Time grid replayability game)*):
14: $\varphi \leftarrow \varphi + 1$
15: $m \leftarrow m + 1$
16: **Skipped elements analysis**
17: If at least one element of σ has been skipped (not been replayed between replayed elements):
18: $\delta = 1$
19: **Conclusion**
20: **return** z, δ, φ

Remark 1. For any transition in a trace we have, by definition, {time replayability} $\Rightarrow$ {basic replayability} but {basic replayability} *' {time replayability}.

To calculate the replayability of a trace, an algorithmic procedure is presented, named **Grid replayability game** (or **Time grid replayability game**) (Algorithm 1). The grid replayability game starts from c_m, with m being the index of the first event of σ replayed in *G-PsM* (line 3). If c_m is replayed by several nodes of *G-PsM*, the node with the lowest position is chosen (line 7). The next event c_{m+1} of σ possibly replayed by *G-PsM* is sought, with a strictly superior node position than the previous node which replayed c_m (lines 11–12). If the transition (c_m, c_{m+1}) is not basic-replayed by *G-PsM*, the transition is said **strongly-forced**. If the transition $((c_m, c_{m+1}), t_{m,m+1})$ is not time-replayed by *T G-PsM*, the transition is said **time-forced** (lines 13–14). This process is repeated until the last replayable event is reached (line 10). If at least one event of σ has not been replayed while being in between two replayed events, it is said **skipped** (lines 17–18).

The strictly ascending condition of Definition 7 ensures that transitions are achieved by increasing positions during the replayability game. Thus, some events of σ might not be replayed during the game, even if they would have been according to Definition 10. Based on these new replayability games, adapted replayability score functions are introduced.

Definition 11. *(Replayability score). Considering the Grid replayability game, the replayability score of a sequence σ in a G-PsM or T G-PsM is defined as follows:*

$$R(\text{G-PsM}, \sigma) \text{ or } R(\text{TG-PsM}, \sigma) = \left(\frac{z}{|\sigma| - a * \delta - \beta * \frac{\varphi}{|\sigma|}}\right)^{+}$$

where:

- *z is the number of events of σ replayed by G-PsM;*
- *δ a binary variable equal to 0 if no event of σ is skipped;*
- *φ is the number of (timed) strongly-forced transitions;*
- *a, β are weighting factors.*

Proposition 2. For a given trace σ, a process model *T G-PsM* and a *G-PsM* obtained with nodes and simple edges of *T G-PsM*:

$$R(\text{TG-PsM}, \sigma) \leq R(\text{G-PsM}, \sigma) \quad (1)$$

Proof. Proposition 2 translates the strictest character of the time grid replayability compared to grid replayability, for fixed coefficients. □

4.5. Problem Formulation for Process Model Discovery

Let L be an event log, the process model optimization problem consists in determining an optimal grid process model *G-PsM* defined on *L*, maximizing the replayability and under some process model complexity constraints.

$$\text{(GridOpt)} \quad \max_{\text{G-PsM}=(N,E,L)} R(\text{G-PsM}, L) \tag{2}$$

with $R(\text{G-PsM}, L) = \frac{1}{|L|} \sum_{\sigma \in L} R(\text{G-PsM}, \sigma)$

subject to

$$E \subseteq N \times N \tag{3}$$

$$max(P(x)_{x \in N}) \leq p_{max} \tag{4}$$

$$|N| \leq U_N \tag{5}$$

$$|E| \leq U_E \tag{6}$$

where $R(\text{G-PsM}, \sigma)$, $R(\text{G-PsM}, L) \in [0, 1]$, $p_{max} \in \mathbb{N}^*$ is the maximum position for the process model construction, $U_N \in \mathbb{N}^*$ and $U_E \in \mathbb{N}$ are the process model node and edge complexity bounds, respectively.

The problem of determining an optimal time grid process model, denoted as *TimeGridOpt* is similar, with the following supplementary constraint:

$$\forall e \in E, T(e) \in T_E(e) \tag{7}$$

where $T_E(e) = ([a_j, b_j])_{j \in \mathbb{N}^*}$ is a pre-defined set of disjointed intervals, with $\forall e \in E$, $\forall I = [a, b] \in T_E(e)$, $a, b \in \mathbb{N}$ and $a < b$.

The purpose of this constraint is to have accurate but specific values of time intervals for each edge. Thus, we first define judicious intervals for each edge, before optimization of the time grid replayability score. For example, let x and y be two nodes of a time grid process model $T\,G\text{-}PsM$. After looking at all possible causal transitions $(L(x), L(y))$ in event log L, we obtain D the distribution of corresponding time values. Let m^- and m^+ be the maximum and minimum values of D, respectively. An optimal solution in terms of complexity and replayability is to take the single edge with the full interval $[m^-, m^+]$. Inconveniently, this does not highlight time particularities and specific intervals. This is why dividing the interval $[m^-, m^+]$ in relevant sub-intervals $\{[x_i, y_i]\}_i$ where $\min_i x_i = m^-$, $\max_i y_i = m^+$ and $\bigcup_i [x_i, y_i] = \emptyset$ would prove more advantageous.

For both problems, an optimal process model has to be found in terms of nodes (with their labels and positions) and (time-) edges (basic or with a specific time interval). Complexity being a hard constraint, it can be useful to first define an optimal solution without complexity constraints, to illustrate the complexity of a potential optimal model.

Proposition 3. For any log L with $p_{max} = max_{\sigma \in L} |\sigma|$, $U_N = div_{e,p}$ and $U_E = \sum_{\sigma \in L} (|\sigma| - 1)$, there exists a process model *G-PsM* such that $R(G\text{-}PsM, L) = 1$.

Proof. All traces are perfectly replayed in a process model with $A_{lab,pos}$ as the set of nodes and arcs connecting nodes of position p to position $p + 1$ corresponding to direct causal relations from events of position p in some trace. □

The Figure 3 shows an example of the process models described previously, with $A_{lab} = \{A, B, C, D, E, F\}$ and $|\sigma_{max}| = 6$.

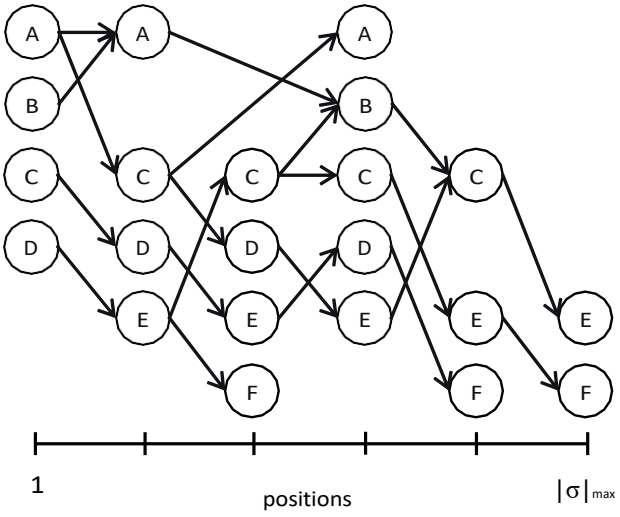


Figure 3: Example of a process model described in Proposition 3.

4.6. Property of the Replayability Game

Proposition 4. (Optimized edges configuration property) For any given process model (*G-PsM* or *T G-PsM*) and for any trace σ, the nodes reached during the replayability game are independent of the edges of the process model.

Proof. Trivial as, at each step of the replayability game, the closest event of σ with label equal to the label of a node of higher position gives the next node. This is independent of arcs. □

5. Optimization and Process Discovery

This section presents five algorithms : Random Search (RS), Frequent Model (FM), Spring Search (SS), Tabu Search (TS) and Tabu Search with Optimal Edges (TSOE). These algorithms are used to solve the optimization problems *GridOpt* and *TimeGridOpt* (defined through Equation 2). Among them, one heuristic (SS) and two meta-heuristics (TS and TSOE) are tested, modifying a current graph solution (or creating a neighborhood of new solutions). Transformation of the current solution is done by achieving some moves of nodes and edges. Before presenting these methods, the data preparation process is detailed in the following.

5.1. Data Preparation

To select nodes or edges to add (or delete) during the optimization process, a function $f_n : A \times [1, p_{max}] \rightarrow \mathbb{N}$ is defined. For each tuple (l, p) with $l \in A$ and $p \in [1, p_{max}]$, a value corresponding to its number of appearances within the event log is assigned. An event's position is either its real position in its trace if $|\sigma|_{max} \leq p_{max}$, or a rescaled value to ensure all positions to be between 1 and p_{max} if $|\sigma|_{max} > p_{max}$. Similarly, a function $f_e : T^C \rightarrow \mathbb{N}$ is defined regarding appearance of transitions in event log. An extension for *T G-PsM* is also proposed, considering transitions and time intervals to compute f_e values. In the following, "*promising* " is employed to characterize a node $n = (l, p)$ or an edge $e = (x, y)$ (with $T(e)$ for *T G-PsM*) with a high value for $f_n(l, p)$ or $f_e(L(x), L(y))$. These functions are useful to select *promising* nodes or edges to add during the optimization process.

Algorithm 2 Tabu Search with Optimized Edges (TSOE) for *GridOpt*.

Step 1 – Initialization
  Select an initial random unconnected solution: s_0^*
  Create the best connected solution s_0 from s_0^*
    Compute Grid replayability game for each patient
    Identify among possible edges the most frequently used
    Add these edges to the final model: s_0
  Compute replayability for the initial solution: $R(s_0, L)$
  Update the best known solution: $s_{best} \leftarrow s_0$
  Create initial Tabu list of unconnected solutions : $TL = [s_0^*]$
Step 2 – Iteration
  Generate from current unconnected solution s^* a neighborhood of unconnected non-tabu solutions : $N_H^*(s^*)$
  Generate best connected solutions from $N_H^*(s^*)$: $N_H(s^*)$
  Compute replayability of each element of $N_H(s^*)$
  Select the new solution as the neighbor with the highest value of replayability: s_{new}
  Update current solution : $s \leftarrow s_{new}$
  Update Tabu list: $TL \leftarrow TL + [s_{new}^*]$
    If $R(s_{best}, L) < R(s, L)$:
      $s_{best} \leftarrow s$
Step 3 – Repeat step 2 until a stopping criterion is reached

5.2. Spring Search (SS) for GridOpt

A simple greedy heuristic, called Spring search (iterative jumps) is initially utilized to solve the *GridOpt* problem. During the search, new solutions are proposed by iteratively increasing and decreasing the size of a solution, which brings diversity Each iteration consists of four steps: (1) delete all edges of the current solution, (2) delete K *non-promising* nodes, (3) add K new *promising* nodes, and (4) add U_E edges (the edge complexity bound), selected from the subset of possible edges (depending on nodes present in the *G-PsM*). At the end of an iteration, the obtained *G-PsM* becomes the current solution.

5.3. Tabu search (TS) for GridOpt

Similarly to [10], a Tabu search is implemented to solve *GridOpt*. Because the strictly ascending condition increases the dependence of edges towards nodes, only one move is used to generate neighborhoods. At each iteration, a neighborhood is made of non-tabu neighbors which are obtained from the current solution in 4 steps : (1) delete a *non-promising* node and its surrounding edges, (2) consider the number of deleted edges as a budget X to reassign, (3) add a *promising* new node, and (4) add X new *promising* edges respecting the strictly ascending condition. Each neighbor is evaluated by computing replayability, and the best neighbor is kept as current solution.

5.4. Tabu Search with Optimized Edges (TSOE) for GridOpt

According to Proposition 4, edges do not intervene in the choice of the next node reached during the replayability game. Considering a solution without edges *G-PsM* * , the replayability $R(G\text{-}PsM^*, \sigma)$ of a trace $\sigma \in L$ will have all possible transitions from a node x to a node y forced during the replayability game.

If an edge from x to y is added to the solution, the replayability score $R(G\text{-}PsM^*, L) = \frac{1}{|L|} \sum_{\sigma \in L} R(G\text{-}PsM^*, \sigma)$ will increase, by a coefficient $c_{(x,y)} = \frac{\beta}{|L|} \sum_{\sigma \in L} \frac{f_\varphi(\sigma,(x,y))}{|\sigma|}$, with $f_\varphi(\sigma, (x, y)) = 1$ if transition (x, y) has been forced during the replayability game of σ, 0 otherwise. Thus, by computing $c_{(x,y)}$ for every possible transition (x, y), keeping top-U_E transitions produces the best-edge configuration from *G-PsM**, respecting constraint (6) of Equation 2.

From these observations, an adapted version of Tabu Search is defined, consisting in searching only for solutions without edges, and then for every solution in evaluating the best edges to add for an optimized replayability score. This method's advantage drastically reduces the search space to graphs without edges. Tabu Search with Optimized Edges is further detailed in Algorithm 2.

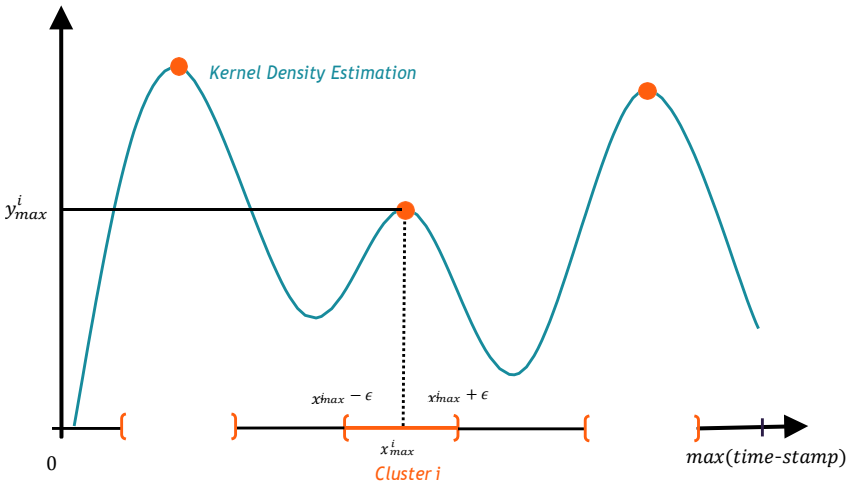


Figure 4: Illustration of KDE clustering.

5.5. TSOE for TimeGridOpt

Before solving *TimeGridOpt*, accurate but specific values of time intervals $T_E(e)$ for possible edges $e \notin E$ need to be defined. First, causal transitions of event logs are analyzed to determine corresponding time distributions. Thus, a 1-D clustering method based on Kernel Density Estimation (KDE) is used to construct the set of possible edges to add during the process model optimization. KDE is the construction of an estimate of the density function from observed data, using a kernel function [12]. Applied to each time distribution, the resulting function will be used to define clusters by considering local maxima of the function as centers of clusters. For each maximum, an interval $[x_{max} - E,\ x_{max} + E]$ is defined, as shown in Figure 4. A low value of E gives small and precise clusters; a high value decreases precision, replays more transitions and increases the global replayability score.

Finally, the approach is the same as for TSOE *GridOpt*, except that edges to add come with the most suited time interval as found by 1-D clustering. For an unconnected solution, the algorithm will choose the top-U_E timed-edges to get the final process model solution.

6. Computational Experiments

6.1. Log Generation

The following design of experiments, to test and compare the performances of the previously defined algorithms on various event logs, is presented here. Event logs of various sizes were generated to match real-life variability in data sets. All experiments were performed on an Intel Core i7 processor (2.8 GHz), 16 GB RAM, and Windows 10 OS. The algorithms were coded in Python 3.6.

6.1.1. Log Generation for GridOpt (without time)

Event logs are generated from process models, which must be created first. Three parameters are required to create a *G-PsM* : (i) a maximum position p_{max}, (ii) an event diversity div_e, and (iii) an event-position diversity $div_{e,p}$. For a given combination of these three, a fully connected *G-PsM* is randomly created with all the edges respecting the strictly ascending condition. Then, traces are generated by selecting a graph's node (with higher probability for nodes at lower positions) and following a path in the model until a terminal node is reached (a node without any outgoing edge). A number $N_{noise} = \mathcal{N}(Z, \sigma)$ of noisy random elements (not in the graph) is added to traces, at random positions. We arbitrarily set $Z = 0.1$ (10% of noise in event logs). Resulting traces make a log.

6.1.2. Log Generation for TimeGridOpt (with time)

Logs are generated in the same way as for *GridOpt*, except that edges of generated models are now split in two categories:

- Without time pattern (each transition resulting from this type of edge will have a time between events respecting uniform law $U(a, b)$);
- With time patterns (a set $\{N(\mu_i, s_i^2)\}_{i \in \mathbb{N}^*}$ of Gaussian distributions is defined, each transition following one of these laws, randomly selected).

The goal of the optimization is now to highlight temporal patterns in the discovered model with multiple timed edges. The following distributions are used for the log generation:

1. No time pattern $U(0,400)$;
2. Simple time pattern ($\{N(200, s^2)\}$;
3. Double time pattern ($\{N(100, s^2), N(300, s^2)\}$;
4. Triple time pattern ($\{N(100, s^2), N(200, s^2),$ $N(300, s^2)\}$);

with s the standard deviation. The allocation of edges to patterns (1-4) is uniform. For double or triple time patterns, the choice among distributions is equiprobable. The value of s is set to 25, to test the robustness of the method for time with variability while keeping patterns identifiable. The edge constraint U_E is set to 80 ($4 \times U_N$) to allow the model to add multiple edges. Other parameters are set as for previous design of experiments. For KDE clustering, $E = 0.05 \times max(time\text{-}stamp)$ to have a precise time interval for edges.

6.2. Design of Experiments

For each of 15 combinations of ($div_{e,p}$, div_e, p_{max}) used to create logs, 10 random *G-PsM* are created. From each *G-PsM* , a log of 1,000 traces is generated. 5 methods are then applied to solve the problem: Random search (RS), simple Frequency model (FM) obtained by only taking most frequent nodes and edges, Spring search (SS), Tabu search (TS) and Tabu Search with Optimized Edges (TSOE). Figure 5 summarizes the design of experiments at hand. For the search algorithms, stopping criteria are the maximum number of iterations (x = 250) or the number of iterations without improvement (x = 25). The neighborhood's size for TS and TSOE is empirically set to 15, based on previous tests showing small variability in the best obtained solution's replayability. Similarly, the Tabu list's size is set to 15. The size constraints (number of nodes, number of edges and maximal position) are constant throughout the entire experimental design to always get an interpretative and comprehensive model. Constant parameters and constraint values are summarized in Table 4. Configurations of the design of experiments are listed in the left part of Table 5. For *TimeGridOpt*, configurations 2, 4 and 15 are tested.

6.3. GridOpt Results

The Figure 6 shows the evolution of replayability during the optimization process for each method, specifically applied to configurations 2 and 15. Graphs used to generate event logs are also tested on noised data to compare results of different methods with the initial model used for trace generation, without size limitations ("ROOT" in the figure). Median replayability among 10 event logs for each configuration is presented versus the number of iterations (maximum value for the number of iterations of each method is set to the minimum stopping criterion among 10 replications). ROOT and FM models are obtained by non-iterative methods, their means are displayed by horizontal lines on the same figure for comparison purposes. For RS, an improving solution is rarely obtained, and the stopping criterion is more quickly reached compared to other iterative methods. The margin for improvement during search is small for complex data sets as visible by comparison of Figure 6b and Figure 6a. Furthermore, the gap between ROOT model and search solution strongly increases from Figure 6a to Figure 6b.

Event log description for *GridOpt*, design of experiments and resulting replayability of the best mined models are given in Table 5. Computation times are presented in Table 6. Neighborhood searches (TS and TSOE) systematically outperform other methods (including the heuristic SS and the frequency model FM).

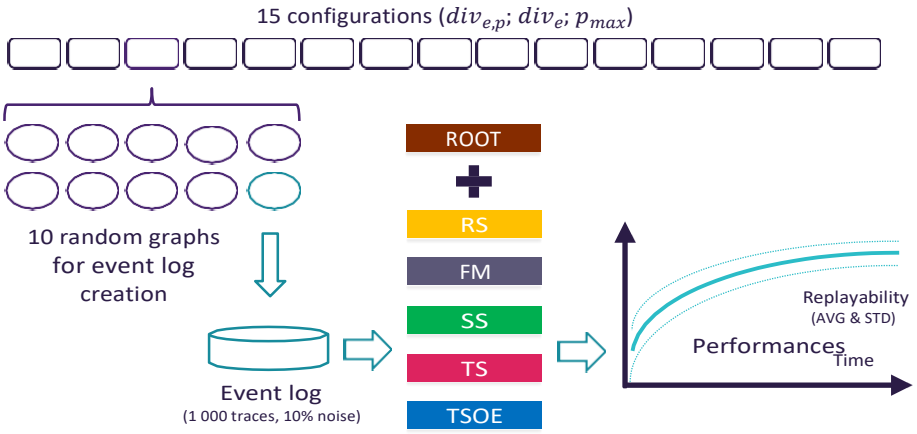


Figure 5: Schematic representation of the design of experiments.

Replayability parameters	
α	0.1
β	0.1
Search parameters	
K (SS)	2
Neighborhood's size (TS and TSOE)	15
Size of Tabu list	15
Max. number of iterations	250
Max. number of iterations without improvement	25
Constraints	
Number of nodes U_N	20
Number of edges U_E	40 $(2\times U_N)$
Maximal position p_{max}	$\min(10, \lvert\sigma\rvert_{max})$

Table 4: Search parameters and constraints used for design of experiments.

TSOE outperforms TS on 9 out of 15 data configurations, especially when $div_e = 5$. Otherwise, TSOE and TS perform equally. TSOE scores ranges from 0.26 to 0.90. Lower values (< 0.30) are obtained for complex data configurations ($div_{e,p} = 300$ and $div_e = 100$), due to the model size constraints. The unconstrained model used for event log generation (ROOT, where $|N| = div_{e,p}$) systematically scores at 0.90 ± 0.01.

Visual representations of the best models mined by TSOE are presented in Figure 7. Visualization of a process model is possible via a tablet application developed by the company HEVA for that purpose. Each graph is read from left to right, increasing positions. Circles represent nodes of the model, and flux from circles represents edges. The size of nodes and edges are proportional to the number of traces replayed by them during the replayability game. The first qualitative observation is the repetition of events with the same label (Figure 7a), with Label 1 or Label 4. The strong decrease in replayability score from Figure 7a to Figure 7b is visible in the decrease in node and edge size, as fewer traces are well represented. If we focus on edges, Figure 8 highlights this strong decrease. Within the optimization for edges, the leeway in replayability is reduced because of the decrease in the number of patients going through edge pathways (172 vs 38 patients in the example of Figure 8). For this reason, the effect of edge optimization in TSOE is less visible in more complex data sets as Figure 7b compared to Figure 7a.

6.4. TimeGridOpt Results

Results are presented in Table 7. For each data configuration, the mean number of incoherent edges (edges which do not correspond to any defined pattern through design of experiments, by not containing

	Data			RS		FM		SS		TS		TSOE		ROOT	
	$div_{e,p}$	div_e	p_{max}	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD
1	25	5	5	0.33	0.08	0.87	0.00	0.86	0.00	0.87	0.00	**0.90**	0.00	*0.91*	*0.00*
2	25	5	25	0.15	0.09	0.77	0.02	0.74	0.02	0.81	0.02	**0.84**	0.02	*0.90*	*0.01*
3	100	5	25	0.24	0.04	0.70	0.01	0.70	0.02	0.75	0.01	**0.77**	0.01	*0.90*	*0.01*
4	100	5	50	0.14	0.08	0.66	0.01	0.65	0.02	0.72	0.01	**0.74**	0.01	*0.89*	*0.01*
5	100	50	5	0.19	0.04	0.34	0.01	0.35	0.02	**0.44**	0.01	**0.44**	0.02	*0.90*	*0.01*
6	100	50	25	0.15	0.04	0.34	0.03	0.33	0.03	**0.42**	0.02	**0.42**	0.02	*0.90*	*0.01*
7	100	50	50	0.11	0.02	0.34	0.04	0.33	0.03	0.41	0.02	**0.42**	0.02	*0.89*	*0.01*
8	100	100	5	0.13	0.02	0.23	0.04	0.25	0.03	**0.36**	0.03	**0.36**	0.02	*0.90*	*0.01*
9	100	100	25	0.13	0.04	0.29	0.03	0.27	0.04	0.37	0.02	**0.38**	0.02	*0.90*	*0.01*
10	100	100	50	0.10	0.03	0.30	0.02	0.29	0.03	**0.38**	0.03	**0.38**	0.02	*0.89*	*0.01*
11	300	50	25	0.15	0.02	0.26	0.02	0.26	0.01	**0.31**	0.01	**0.31**	0.02	*0.89*	*0.01*
12	300	50	50	0.13	0.02	0.24	0.01	0.24	0.02	0.30	0.01	**0.31**	0.01	*0.89*	*0.01*
13	300	100	5	0.13	0.02	0.20	0.01	0.21	0.01	**0.26**	0.01	**0.26**	0.01	*0.89*	*0.01*
14	300	100	25	0.11	0.01	0.21	0.02	0.21	0.02	0.27	0.02	**0.28**	0.02	*0.89*	*0.01*
15	300	100	50	0.10	0.02	0.21	0.02	0.21	0.02	0.27	0.02	**0.29**	0.02	*0.89*	*0.01*

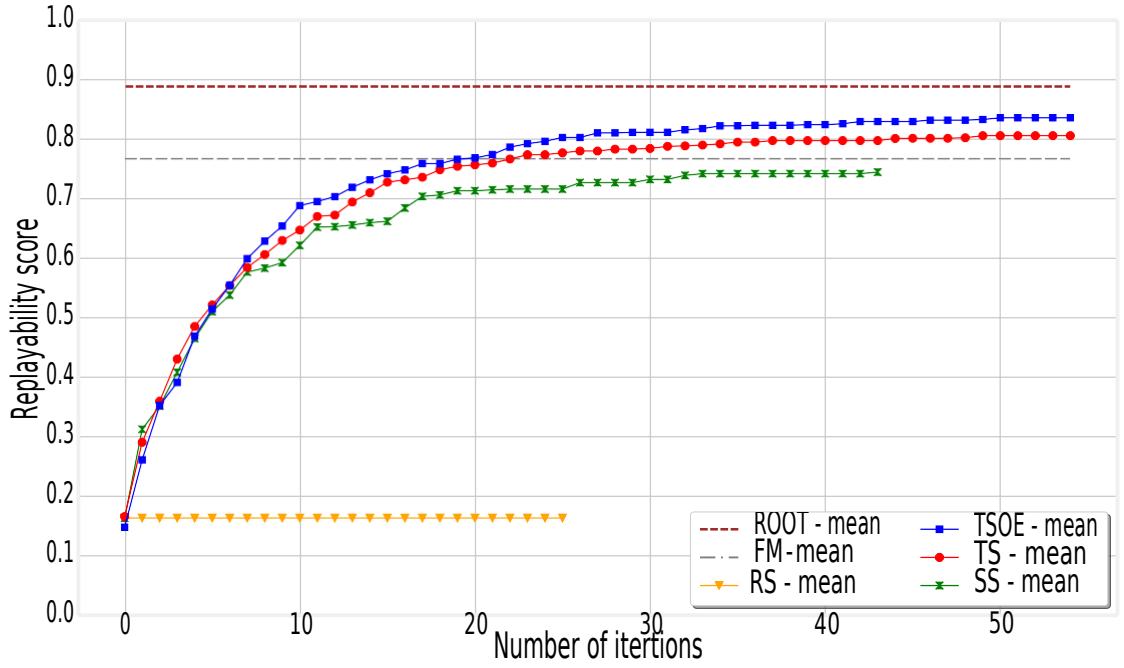
Table 5: The replayability score of the best models mined by different methods: average and standard deviation.

Data	RS		SS		TS		TSOE	
	AVG	STD	AVG	STD	AVG	STD	AVG	STD
1	21	9	39	9	622	151	1479	392
2	15	1	33	6	731	155	1825	419
3	22	7	40	11	665	134	1591	619
4	19	5	38	6	873	234	1930	598
5	34	12	47	16	878	363	1828	362
6	18	5	39	12	711	191	1463	321
7	18	5	33	7	726	162	1541	407
8	15	1	40	12	679	184	1182	335
9	15	<1	33	10	678	162	1513	398
10	15	<1	35	8	689	150	1416	454
11	19	5	38	8	639	99	1353	411
12	18	2	38	8	745	182	1404	334
13	15	<1	37	6	785	213	1354	347
14	16	1	43	8	656	132	1406	253
15	16	<1	36	7	674	173	1497	257

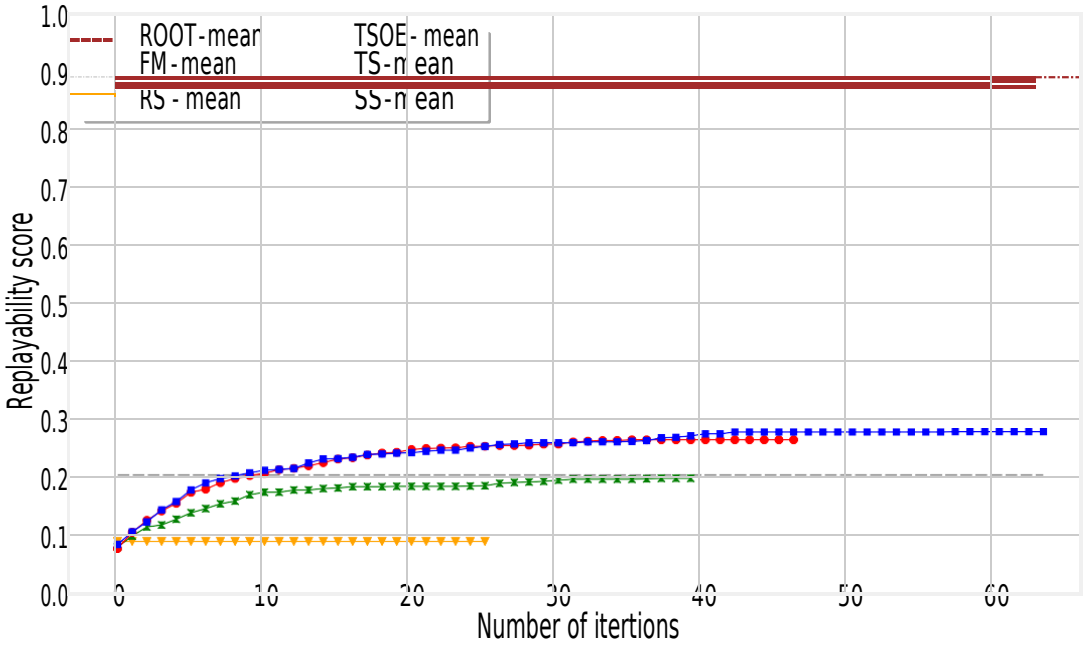
Table 6: Computation time (in seconds) of each method: average and standard deviation.

Data	Replayability		Time		Incoherent edges
Config.	AVG	STD	AVG	STD	AVG
2	0.81	0.02	9820	3048	5.6%
4	0.72	0.01	7965	2547	7.0%
15	0.28	0.02	10999	3688	6.0%

Table 7: Best models mined by TSOE for *TimeGridOpt* : replayability, time (in seconds) and percentage of incoherent edges.

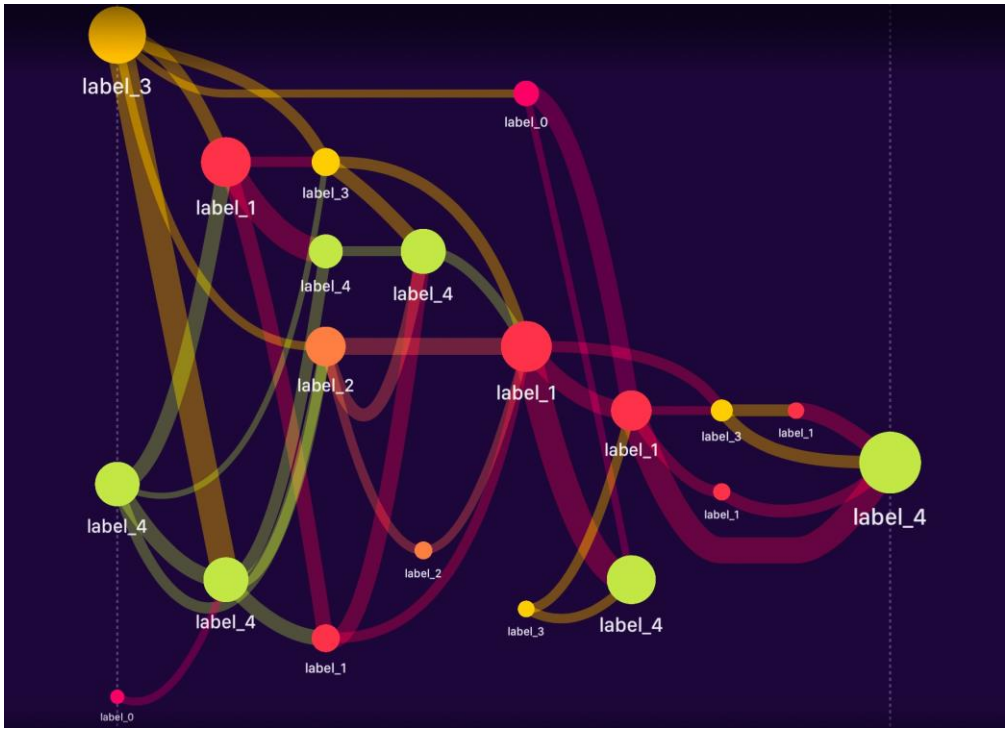


(a) Configuration 2 - GridOpt

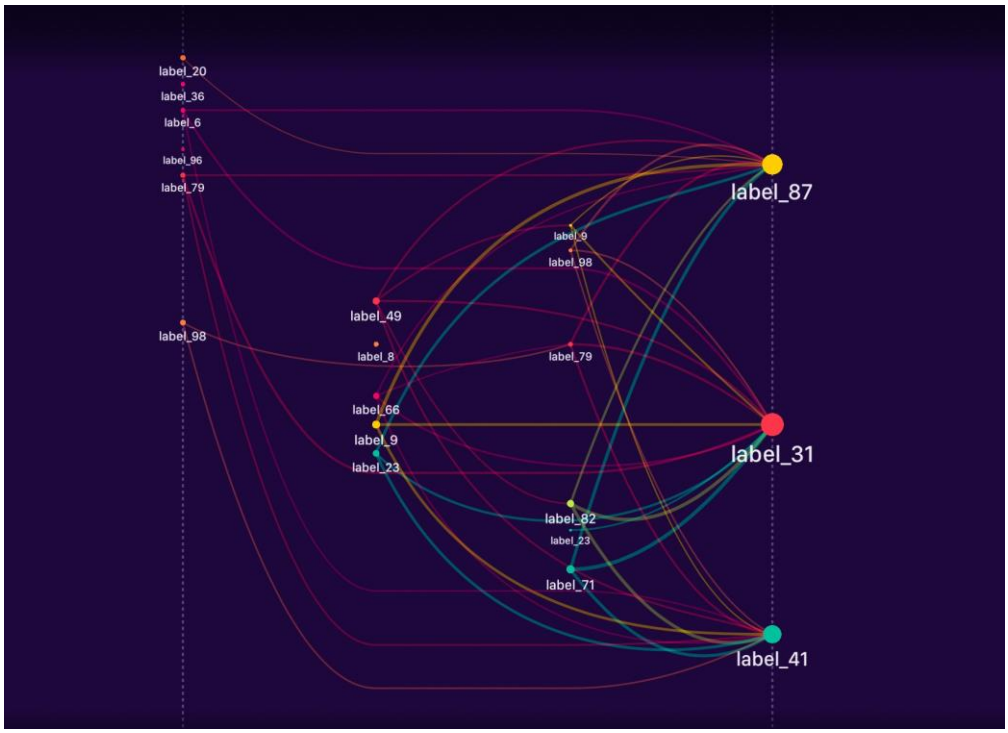


(b) Configuration 15 - GridOpt

Figure 6: Replayability versus the number of iterations: 6 different methods applied to three logs for the *GridOpt* problem; log of configuration 2 (Fig. 6a) and configuration 15 (Fig. 6b).



(a) Configuration 2



(b) Configuration 15

Figure 7: Examples of the best *G-P sM* models mined by the TSOE algorithm.

(a) Configuration 2

(b) Configuration 15

Figure 8: Focus on the edges for *G-P sM* models.

time values 100, 200 or 300) is also given. The replayability of the best mined models with TSOE for *TimeGridOpt* is slightly inferior to TSOE for *GridOpt* on the same event log. The number of incoherent edges, i.e. the edges not respecting time patterns, is low (5.6%, 7.0% and 6.0%) and thus is encouraging for the methodology presented. These incoherent edges characterize traces with transitions generated from no time pattern edges (25% of transitions following $\{(0, 400)\}$) or noisy transitions obtained after adding noise to the event log. Visual representations of the best models are shown in Figure 9 for configuration 2. The general shape of the process model obtained by solving *TimeGridOpt* is similar to previous *G-PsM* graphs (Figure 9a). The time-focused representation of *TG-PsM* highlights the type of edges obtained after the optimization, corresponding to the amount of timed edges it contains. According to the simulated event log, the time pattern edges could be of 3 types: simple (one interval centered in 200), double (2 intervals centered in 100 and 300) or triple (centered in 100, 200 and 300), as shown in Figure 9b.

7. Real-life Case Study

7.1. Diabetes Mellitus

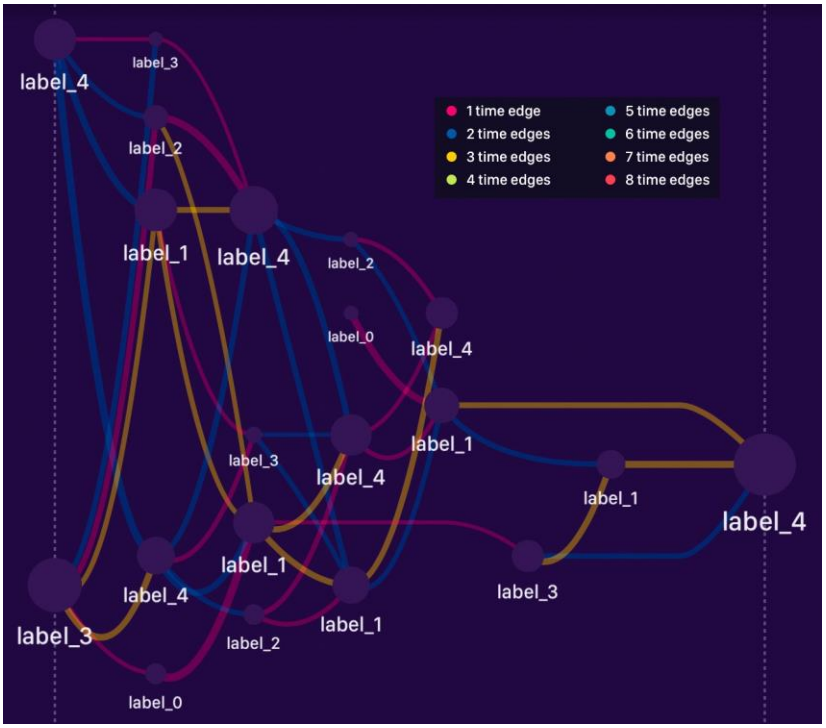
Diabetes Mellitus (DM) is a group of metabolic disorders, resulting in chronic hyperglycemia due to unregulated insulin secretion and/or action. Common forms of DM are type-1 and type-2. Type-2 diabetes is the most common form (90-95% of patients). It is mainly characterized by insulin resistance and relates to the lifestyle, physical activity, dietary habits and heredity. Type-1 diabetes is less frequent (5-10%) and is due to destruction of β cells of the pancreas [4]. Data Mining methods have been widely applied to DM data and supervised learning prevails (85% of studies). Moreover, clinical data sets were the most used [3]. Our method, unsupervised Process Mining, adds a new angle and diversity to existing approaches in DM research. This real-life case study shows how the newly developed approach helps to analyze patient pathways before the appearance of four identified complications.

7.2. Data and Methodology

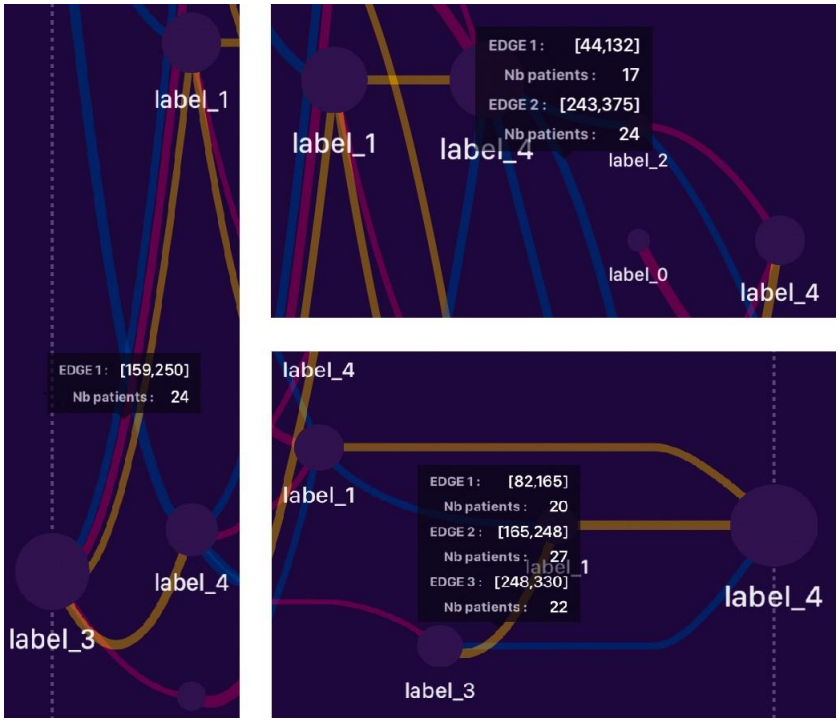
Data originates from the French National Health Insurance (CNAM), where a group of 50,000 patients suffering type-1 or type-2 diabetes in 2008 was constructed. Within this population, 5,714 patients developed at least one of the following complications until 2016: stroke, amputation, infarctus or TCKD (Terminal Chronic Kidney Disease). For each complication and for each patient, a 2-year period of medical history was analyzed. A time grid process model was built for each complication over these 2-year periods. TSOE algorithm was used with the following parameters : models' size is $|N| = 20$, $|E| = 4 \times |N|$, and otherwise as in Table 4. Events of different categories were available:

- Hospitalizations (diabetes, cardiovascular, surgery...);
- Complications (stroke, amputation, infarctus, TCKD);
- Other medical events (dialysis , insulin, emergency without hospitalization).

Other follow-up exams, much more frequent in patient pathways (around 70% of the events), were also available: general practitioner visits, glycated hemoglobin tests (HBA1C), glycemia tests, creatinine tests, etc. Discussions with medical experts led to the non-consideration of these exams as key nodes for the process model. Instead of studying their sequence and successions in the pathway, they were simply and usefully quantified within the period between two nodes (i.e. on an edge). The quantification of such events was performed on the final mined model: for each patient crossing an edge during the replayability game, a list of frequent exams is computed, and median values for each frequent exam are printed on the edges. An unconnected grid process model with best time grid process model's nodes was created first. Then, a number of edges $|E|$, equal to the Div_E, are used to connect the grid process model using optimized edges.



(a) Configuration 2



(b) Focus on the edges for *TGPSM* models.

Figure 9: Examples of the best *TGPSM* models mined by the TSOE algorithm

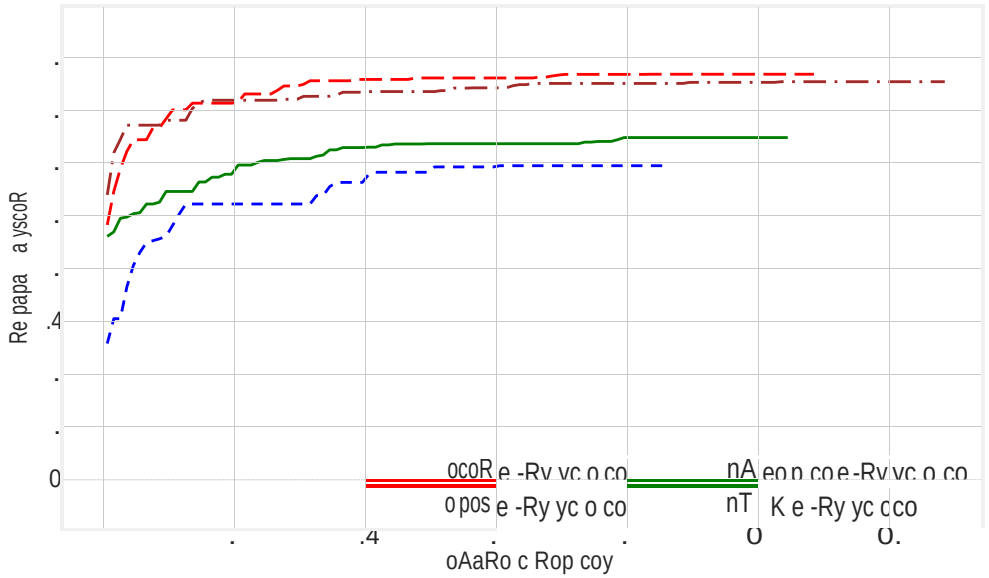


Figure 10: Replayability versus iterations for the four complication event logs.

Complication	$\|L\|$	div_e	$div_{e,p}$	$\|\sigma\|_{q_1,m,q_3}$	R^G	R^{TG}
Amputation	695	15	232	3/5/8	0.76	0.75
Stroke	2152	15	222	1/3/4	0.89	0.87
Infarctus	2913	15	253	2/3/5	0.88	0.86
TCKD	421	13	225	4/6/9	0.72	0.70

Table 8: Diabetes event log descriptors and replayability score.

7.3. Results

Descriptors and replayability performances are presented in Table 8. The evolution of the optimization for each event log is presented in Figure 10. Event log analyses using the descriptors show pathway differences between stroke, infarctus, amputation and TKCD. Indeed, the first two complications are characterized by shorter traces ($|\sigma|_{q_1,m,q_3}$: (1, 3, 4) and (2, 3, 5) vs. (3, 5, 8) and (4, 6, 9)), that is to say short and unstructured pathways compared to the other two. This difference between complications is also highlighted by the best final replayability scores: amputation and TCKD have lower scores (R^{TG} : 0.75 and 0.70) compared with those of stroke and infarctus (0.87 and 0.86) because longer and more complex pathways are less easily replayed in a graph than shorter ones. These observations are illustrated by Figure 11. An example of frequent events' information can be seen in Figure 11b where a pattern of diabetes hospitalization before stroke is highlighted. For 206 patients concerned, time between events was 242 days on average. As an example of frequent exams, the median number of general practitioner visits is displayed ("MG : 5"). The grid structure, which allows duplicate labels in a process model, is particularly suitable in this case study. As shown in Figure 11a, a high number of "Other hospitalizations excluding surgery" and "Cardiology hospitalizations" are interesting patterns revealed by the grid process model. Time pathway analysis gives further opportunities for understanding patient pathways. As an example, the process model relating to the complication "amputation" (Figure 11a) shows globally unique short time pathways. On the opposite, the process model relating to the complication "stroke" (Figure 11b) presents diverse time pathways, with not only short duration transition.

8. Conclusion and Future Research

An extended methodology to create suitable process models for healthcare applications has been presented. Its Scientific contributions are multiple. New process models considering a grid structure and time patterns were mathematically defined. We formulated a set of descriptors to characterize the structure of such a process model and event log complexity. The establishment of a new property for grid process models leads to a novel search algorithm to mine optimized process models. The search incorporates the grid structure and includes time patterns upon construction of the process model. Computational experiments validate the overall performance of this approach. The interest of neighborhood-based searches to solve the problem was quantitatively shown: Tabu Search with Optimized Edges is more efficient for a small event diversity. A qualitative observation was made regarding the grid structure, representing with more fidelity the linearity of patient pathways over time. This improves the visualization of repeated events. The advantage of considering time within optimization was also spotlighted. In addition, the applicability of the method and the interest in patient pathways analysis is demonstrated by a case study. The grid structure, the time patterns and the display of certain frequent events on edges provide interpretative highlights for medical staff and decision makers.

Three opportunities for future work come to mind. Firstly, a focus on optimization performances for complex data sets will be made, by considering less strict constraints (nodes and edges). During the experiments presented in this work, constraints were specifically set to obtain an overall comprehensible model capable of being simply visually interpreted. However, increasing the complexity of the process model can be achieved if interactive tools permit the exploration of the final model wherein key elements are able to be clearly discerned. Secondly, studying the relation between event log descriptors, graph constraints and replayability of the best models minded is of important interest as well. Any results rendered will be useful for the calibration of constraints, particularly for the third research axis. Eventually, future research should focus on creating a methodology to perform supervised learning with traces as input data, whereas current state-of-the-art classification methods only take “flattened data” as input (vectors of features). A process model optimized for classification purposes will produce an explainable predictive model.

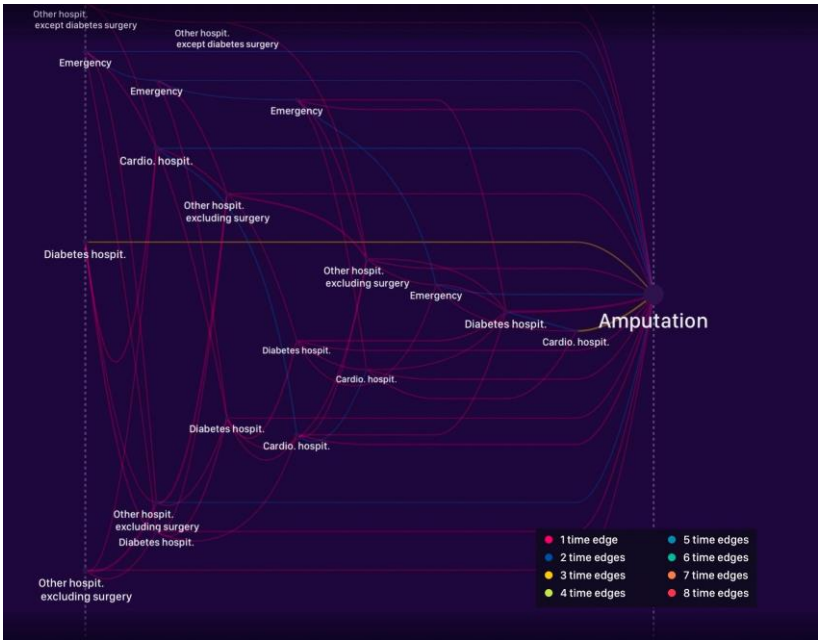
Acknowledgement

The authors wish to thank Chris Yukna for his help in proofreading.

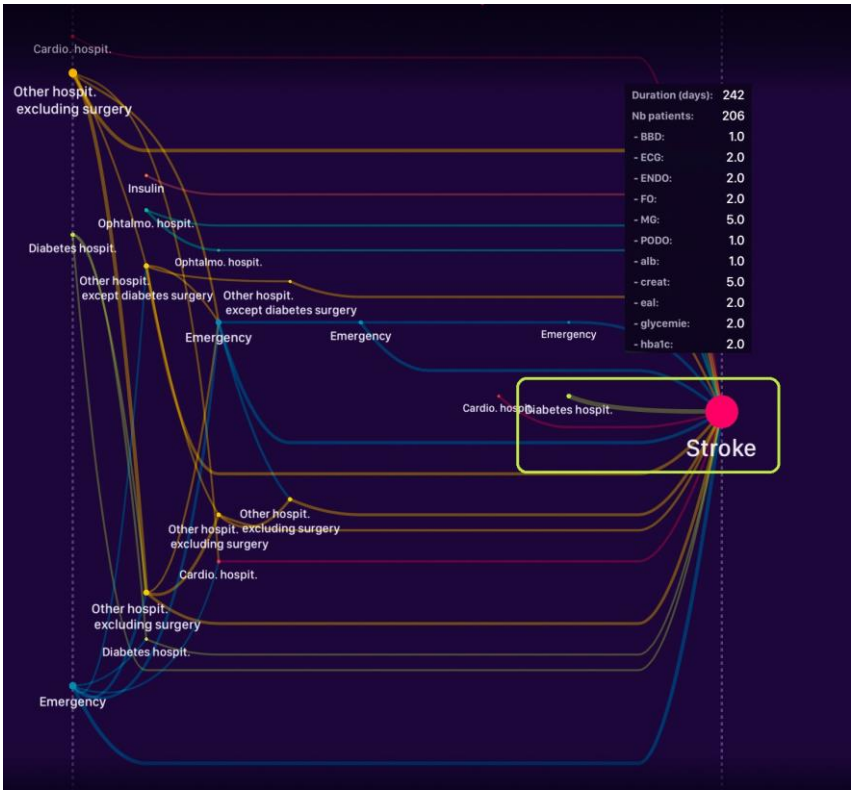
References

[1] T. G. Erdogan and T. Ayca. Systematic mapping of process mining studies in healthcare. *IEEE Access*, 6:1–1, 2018.
[2] A. Giua and X. Xie. Control of Safe Ordinary Petri Nets Using Unfolding. *Discrete Event Dynamic Systems*, 15(4): 349–373, 2005.
[3] I. Kavakiotis, O. Tsave, A. Salifoglou, N. Maglaveras, I. Vlahavas, and I. Chouvarda. Machine learning and data mining methods in diabetes research. *Computational and Structural Biotechnology Journal*, 15:104 – 116, 2017.
[4] A. T. Kharroubi and H. M. Darwish. Diabetes mellitus: The epidemic of the century. *World journal of diabetes*, 6: 850–867, 2015.
[5] G. P. Kusuma, M. Hall, C. Gale, and O. Johnson. Process mining in cardiology: A literature review. *International Journal of Bioscience, Biochemistry and Bioinformatics*, 8(4):226–236, 2018.
[6] I. Litchfield, C. Hoye, D. Shukla, R. Backman, A. Turner, M. Lee, and P. Weber. Can process mining automatically describe care pathways of patients with long-term conditions in uk primary care? a study protocol. *BMJ Open*, 8(12), 2018.
[7] A. R. C. Maita, L. C. Martins, C. R. L. Paz, L. Rafferty, P. C. K. Hung, S. M. Peres, and M. Fantinato. A systematic mapping study of process mining. *Enterprise Information Systems*, 12(5):505–549, 2018.
[8] M. Prodel, V. Augusto, X. Xie, B. Jouaneto, and L. Lamarsalle. Discovery of patient pathways from a national hospital database using process mining and integer linear programming. In *CASE*, pages 1409–1414, 2015.
[9] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie. Evaluation of discovered clinical pathways using process mining and joint agent-based discrete-event simulation. In *Proceedings of the Winter Simulation Conference 2016*, 2016.
[10] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie. Optimal process mining for large and complex event logs. *IEEE Transactions on Automation Science and Engineering*, 15(3):1309–1325, 2018.
[11] H. A. Reijers and J. Mendling. A study into the factors that influence the understandability of business process models. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 41(3):449–462, 2011.

[12] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Springer US, 1986.
[13] W. M. P. van der Aalst. Introduction. In *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, pages 1–25. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
[14] W. M. P. van der Aalst, T. Weijters, and L. Maruster. Workflow mining: discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, 16(9):1128–1142, 2004.



(a) Amputation



(b) Stroke (with an example of frequent events display)

Figure 11: Example of time grid process models resulting from the case study.

2.4 Conclusion

This chapter presented a new process mining setting, including event log complexity measures, process models, replayability game, replayability score and optimization method. The method is able to perform process discovery from event logs, with the advantage of mining temporal characteristics during the optimization process. A preliminary version of the presented work contributes to a second process mining case study on sepsis, using the Hospital Episode Statistics (HES) database [165].

Some limitations of the proposed method should be considered. As presented in the first case study, some medical events such as lab tests are very frequent. First discussions with medical experts motivated the suppression of such frequent events from labeling. The frequent elements were reintroduced on the final obtained process model, by including distribution parameters on edges. This visualization is useful to discover patterns and analyze pathways. However, the reintroduced information is not directly used in the optimization process, such distinctive patterns are not identified by the resulting process model. For the purpose of performing prediction, this may lead to unconsidered information.

Another limitation of this work is the choice of labels for an event. The method takes a labeled event log as input data. In practice, these labels are manually defined by using expert knowledge. In many cases, only partial information is used, such as the main diagnosis code, at a certain level of hierarchy. But when studying macro events like hospitalization, the internal patient pathway is often summarized with many different codes, representing diagnostics, medical procedures, drugs or medical devices. As the preprocessing of such data without medical knowledge is challenging, the next chapter addresses this problem.

Chapter 3

Automatic and Explainable Labeling of Medical Event Logs

Contents of the chapter

3.1 Motivation

When deploying process mining in practical studies related to patient pathways, the labeling of medical events was identified as a challenging preprocessing step. Generally, one type of activity occurring during the event is selected as the label used to describe each event (i.e. the code related to the main diagnosis for a hospital stay). However, when working on patient pathways, macro medical events are frequently described using multiple activities. One the one hand, the selection of a sole activity to describe macro medical events may decrease the precision of the labels and lead to less pertinent results. On the other hand, considering all of the activities will create labels that are too specific as hospital stays are rarely exactly the same all activities considered. Another challenge in labeling of medical events is that activities are mainly characterized by medical codes from various coding systems. These coding systems are often organized in hierarchical structures and the selection of a convenient aggregation level is complicated and dependent of the case study. In practice, the use of expert knowledge to manually define the labels using activities is the most trustworthy methodology. The studied pathology, the population and, more generally, the context of the study guides the experts in choosing the relevant labels. However, when expert knowledge is not available, a preprocessing step for automatic labeling of event logs can be a valuable contribution. Such a method could also be useful as a knowledge discovery tool. But a key condition for the use of such methods by

non-experts is the explainability of the resulting labels, which is required to facilitate the discussions with clinical teams. An advantage of using such a data-oriented preprocessing step is the absence of prior knowledge, minimizing the risk of including bias.

3.2 Summary

This chapter introduces a methodology developed to account for the previously described complexity of events in medical event logs, and automatically create relevant labels directly from the data. The core of the method is the sparse representation of stays with multiple activity codes with hierarchical structure, and the use of deep autoencoding in order to learn a useful representation of stays. As deep autoencoders learn a representation of stays in the latent space which maximize the reconstruction of data, representation of stays in the latent space is a compact representation of stays from the initial space. Based on this method, accurate labels are created by clustering similar events in latent space. Moreover, the explanation of created labels is provided by decoding the corresponding events. Several deep autoencoding architectures, as well as direct clustering using sparse data, were tested on synthetic events. Results show the ability of the method to find hidden clusters, as well as to accurately explain created labels, in particular when using VAE. In order to validate the method on real data, a case study is presented, where the pathways of patients having an incisional hernia after a laparotomy operation were analyzed. Both manual (using prior knowledge on the pathology) and automatic labeling of the event logs were performed. The process mining method presented in Chapter 2 was used to perform process discovery using both manually and automatically labeled event logs. The analysis of resulting process models shows strong similarities between the automatic labels and the manually defined ones. The results presented in this chapter are available online, through an interactive dashboard which was created to present the results and facilitate discussions:

- Link: https://artemis-emse-laparo.hevaweb.com/
- Login: laparotomy
- Password: P38D8P35f6

3.3 Automatic and Explainable Labeling of Medical Event Logs

H. De Oliveira, V. Augusto, B. Jouaneton, L. Lamarsalle, M. Prodel and X. Xie, "Automatic and Explainable Labeling of Medical Event Logs with Autoencoding", *IEEE Journal of Biomedical and Health Informatics (J-BHI)*, 2020, https://doi.org/10.1109/JBHI.2020.3021790.

Automatic and Explainable Labeling of Medical Event Logs with Autoencoding

Hugo De Oliveira, Vincent Augusto, Baptiste Jouaneton, Ludovic Lamarsalle, Martin Prodel and Xiaolan Xie

***Abstract*—Process mining is a suitable method for knowledge extraction from patient pathways. Structured in event logs, medical events are complex, often described using various medical codes. An efficient labeling of these events before applying process mining analysis is challenging. This paper presents an innovative methodology to handle the complexity of events in medical event logs. Based on autoencoding, accurate labels are created by clustering similar events in latent space. Moreover, the explanation of created labels is provided by the decoding of its corresponding events. Tested on synthetic events, the method is able to find hidden clusters on sparse binary data, as well as accurately explain created labels. A case study on real healthcare data is performed. Results confirm the suitability of the method to extract knowledge from complex event logs representing patient pathways.**

***Index Terms*—process mining; event log; healthcare data; patient pathways; autoencoding;**

I. Introduction

Data analytic regroups an extensive number of methods to investigate data produced in various systems such as industry, software engineering and healthcare. Knowledge extraction from such data is a lever to improve performances, to predict or simply to describe the reality of facts. Among different types of data, event logs are challenging to analyze because of the presence of time, the high variability of events, and the complex relations between events. Thus, the use of widespread data mining algorithm may not be fully straightforward for some applications. A wise preprocessing step to capture meaningful information may be necessary. Describing processes, these data are present in the manufacture industry, in software engineering and in healthcare [1]. To analyze event logs, a data-driven approach named *process mining* has been proposed [2]. Between data mining and process modeling, event logs are impartially used to extra [3].

The French national health insurance database (SNIIRAM) is a non-clinical claim database. Containing healthcare reimbursements of almost all French citizens, the amount of data is colossal. 66 million inhabitants were part of this database in 2015 [4]. Among all reimbursement information contained in the SNIIRAM, patients' hospitalizations are provided. However, no precise medical information such as test results, imaging reports, or vital signs are available. Nevertheless, such a database is useful to map patient pathways [5]–[8], perform medical data clustering [9] and prediction tasks [10], [11]. Regarding healthcare processes, the complexity is multi-fold. Illustrations are, but not limited to, the presence of free text, the granularity of events analyzed, and the occurrence of multiple event simultaneously, leading to multiple codes describing a given event. These codes, representing medical activities of different types, could be numerous and often inherit of hierarchical structures [6]. Even if the hierarchy could be useful to simplify codes and reduce the overall cardinality, the choice of the accurate level in order to produce meaningful events is not obvious, depends on the pathology or the health process studied and often requires a clinical expertise. This aspect of complexity is one of the main challenge regarding non-clinical claim database, such as the SNIIRAM.

Therefore, the main contribution of the present paper is a new methodology to analyze the complexity of events and produce meaningful labels. Using autoencoding and clustering, the proposed method creates artificial labels from initial data. These labels are assigned to events, transforming the raw event log by reducing the overall variability of events. The method provides transparency for practitioners by giving an interpretation for each created artificial label. In practice, the contribution consists of a preprocessing methodology to treat this particular complexity of events. As a result, available process mining tools[1] can be used starting from event logs obtained via the proposed methodology.

This paper is organized as follows. An overview of related works is given in Section II. Preliminary notations are presented in Section III. Section IV introduces the problem addressed in this paper. The proposed methodology is described in Section V. To validate the method, a design of experiments is presented in Section VI, followed by a case study based on real-life healthcare data in Section VII. Finally, conclusions and perspectives are given in Section VIII.

II. Literature review

Healthcare data analysis constitutes a large field to test and apply a wide spectrum of analytic methods. Among them, machine learning and more recently deep learning methods have been largely deployed. Electronic Health Records (EHR) have permitted the development of new models and methods, boosting the field publication activity [12], [13]. Among the tasks addressed by deep learning, supervised learning and concept embedding emerge for a majority of studies in

H. De Oliveira, V. Augusto and X. Xie are with Mines Saint-Étienne, Univ Clermont Auvergne, CNRS, UMR 6158 LIMOS, Centre CIS, F - 42023 Saint-Étienne France (e-mails: hdeoliveira@hevaweb.com; augusto@emse.fr; xie@emse.fr).

H. De Oliveira, B. Jouaneton, L. Lamarsalle and M. Prodel are with HEVA, 186 avenue Thiers, F-69465, Lyon, France (e-mails: bjouaneton@hevaweb.com; llamarsalle@hevaweb.com; mprodel@hevaweb.com).

X. Xie is also with the Antai College of Economics and Management, Shanghai Jiao Tong University, China.

[1]Such as `ProM`, `Disco`, `PM4Py` or `bupaR`.

healthcare [14]. In spite of the need to develop high performing predictive models, explainability has been highlighted as a key issue for future model developments [14], [15]. Valuable for practitioners and experts of the medical field, the explanation of predictive results has already been addressed by deep learning studies [16], [17].

In addition to supervised learning and concept embedding, process mining is also a promising analytic method. By performing process discovery on event log data, process mining algorithms produce graphical and interpretative representations of occurring processes. The field is active and recent publications are numerous [1], particularly in healthcare [18]. Initiatives of the research community to improve practices and knowledge sharing illustrate the field activity.[2] Recent topics addressed are, but not limited to, privacy [19], clinical guideline [20], and data cleaning [21]. Regarding process discovery, an optimal procedure to construct process models from raw data bases has been proposed by Prodel et. al. in 2018 [6]. Applied to healthcare data event logs, the proposed method has been adapted to take into account temporal information during the optimization process [8].

Patients' pathways analysis based on real-life healthcare data is valuable to represent and understand how patients' care occurs in real-life. Even if deep representation has been applied on clinical pathways [22], process mining use graphical process models as a support for representation. Thus, it makes the method suitable to discover patients' pathways from raw data when the focus lies on interpretation. However, medical data can be complex to analyze, due to the variety of different medical codes used in claims databases (e.g. diagnoses, procedures and drugs). As a result, the labeling of events is a challenging step in data processing. A commune practice is the definition of labels by hand, based on expert knowledge [8]. The detection of hidden healthcare sub-processes has been proposed, using Hidden Markov Models (HMM) [23]. The presented method allows a reduction of complexity by the enrichment of the log with HMM-derived states, reducing complexity and saving experts time. A practical solution proposed by Prodel et al. in 2018 [6] is the creation of labels during the optimization, using the hierarchy of events from one type of codes, such as main diagnosis. However, when multiple codes from multiple coding systems characterize the events of a process, the selection of the right aggregation level and the combination of codes is not, to the best of our knowledge, a treated problem.

Therefore, the main contribution of this paper is the proposal of a general methodology to treat complex events such as multiple medical activities, in order to apply process mining. Following the definition of Lenz and Reichert [24], the proposed method identifies the activities of the *medical treatment process* by analyzing the coded events of the *organizational processes*. The core of this methodology is based on recent work in representation learning. A widely used method in representation learning is *autoencoding* [25]. The general idea behind autoencoding is the learning of a structure which can encode and decode information while minimizing the loss of information. By compressing the data, a transformation of the input representation is performed.

Thus, before formally introducing the problem and the proposed methodology, preliminary notations are presented in the following.

III. Preliminaries

A formal description of the data involved is provided in the following, including events, traces and event logs.

Definition 1: Event. Each event denoted e is a couple (c, t, a) where:

- c is the related *case ID* of the event, with the id function $id(e) = c$ returning the case ID of event e;
- $t \in T$ with $T = \mathbb{N}$ or $\mathbb{R}$ corresponds to the event time also called *time-stamp*, with the event time function $time(e) = t$ returning the time-stamp of event e;
- a is a nonempty set called *activity set*, each element $a_i \in a$ being an *activity*.

Definition 2: Trace. A trace is a sequence of events denoted as $\sigma = e_1, e_2, \ldots, e_m$ with $m \in \mathbb{N}^*$ such that $time(e_k) < time(e_{k+1})$ and $\forall e, e' \in \sigma$, $id(e) = id(e')$. The *size* of the trace $|\sigma|$ is defined as the number of events in σ.

Definition 3: Event log. An event log is a set of traces denoted as $L = \{\sigma_1, \sigma_2, \ldots, \sigma_n\}$ with $n \in \mathbb{N}^*$. The *size* of the event log $|L|$ is defined as the number of traces in L. Its *length* $len(L)$, defined as $len(L) = \sum_{\sigma \in L} |\sigma|$ gives the total number of events composing L.

According to the previous definitions, an event log is a group of traces, each trace being a succession of events characterized by activities occurring at a given time-stamp, composing an activity set.

Definition 4: Log activity set. Let L be an event log. The *log activity set* of L is the nonempty set A defined as:

- $A = \{a_i | \exists \sigma \in L, \exists e = (c, t, a) \in \sigma, a_i \in a\}$
- $\forall \sigma \in L, \forall e = (c, t, a) \in \sigma, \forall a_i \in a, \exists! A_j \subset A | a_i \in A_j$.

Thus, the log activity set is composed of activities encountered in L, with every single activity of event log L belonging to a unique subset A_j of A.

Proposition 1:

$$A = \bigcup_j^n A_j \text{ and } \bigcap_j^n A_j = \emptyset \tag{1}$$

Proposition 2:

$$card(A) = \sum_j^n card(A_j) \tag{2}$$

Definition 5: Activity vector. Let $e = (c, t, a)$ be an event, the *activity vector* x of e is defined such that $x \in E_X = \{0, 1\}^d$, E_X being the *activity vector space*, with $d = card(A)$. A mapping function $vect()$ is also introduced, defined as:

$$vect : A \to E_X$$
$$a_1 \to vect(a) = x$$

with its inverse $vect^{-1} : E_X \to A$.

The previously defined functions allow a mapping between an activity set and its corresponding activity vector. The

[2] http://pods4h.com/

case id	time-stamp	medical activities
0	0	{Z511; ZZNL053}
0	10	{Z511}
0	20	{Z5101; 9261771}
1	0	{Z511; 9261110}
1	5	{Z511; 9261110}
1	20	{Z511; 9261110}
1	50	{Z511 ; 9261771; ZZNL053}

(a) Activity set representation.

case id	time-stamp	Z511	Z5101	ZZNL053	9261771	9261110
0	0	1	0	1	0	0
0	10	1	0	0	0	0
0	20	0	1	0	1	0
1	0	1	0	0	0	1
1	5	1	0	0	0	1
1	20	1	0	0	0	1
1	50	1	0	1	1	0

(b) Activity vector representation.

TABLE I: Example of an event log of patient pathways.

activity vector x of an event is the equivalent representation of an activity set using 1-of-k coding from all possible activities from A to the activity vector space E_X. Therefore, a given event e could be defined as $e = (c, t, a)$ or $e = (c, t, x)$ knowing $vect$ and $vect^{-1}$ without any meaning loss regarding the log activity set.

Definition 6: Activity matrix. The *activity matrix* of an event log L is defined as $M_X = (x_i)_{i \in [1, len(L)]}$ with $dim(M_X) = len(L) \times d$.

The activity matrix M_X of an event log L gives a binary representation of activity sets, which is a common representation in machine learning.

Example 1: Let us define an event log $L = \{\sigma\}$ with one trace $\sigma_1 = e_1, e_2$ having two events such that $e_1 = (c_1, t_1, a_1)$ and $e_2 = (c_2, t_2, a_2)$ with:

- $A = A_1 \cup A_2$ with $A_1 = \{A_1, B_1\}$ and $A_2 = \{A_2, B_2, C_2\}$;
- case ids $c_1 = c_2 = c$;
- time-stamps $t_1 = 0$, $t_2 = 10$;
- activities $a_1 = \{A_1, B_2\}$, $a_2 = \{A_1, B_1, A_2, C_2\}$.

According to the previous definitions:

- $d = card(A_1) + card(A_2) = 5$;
- $x_1 = (1, 0, 0, 1, 0)$ and $x_2 = (1, 1, 1, 0, 1)$.

Example 2:

Table I presents a short event log related to patient pathways analysis using data as found in claims database. Each row is a hospitalization event. Events with the same case ID are ordered by increasing time stamp and represent a trace, which is a patient's hospitalization history. Medical activities are of three different categories: diagnosis, medical procedures and drugs, coded using standard notations ICD-10 (International Classification of Diseases 10^{th} revision), CCAM (*Classification Commune des Actes Médicaux*) and UCD (*Unite´ Commune de Dispensation*), respectively. According to previous definitions, $A = A_{diag} \cup A_{med.proc.} \cup A_{drugs}$, with $A_{diag} = \{Z511; Z5101\}$, $A_{med.proc.} = \{ZZNL053\}$ and $A_{drugs} = \{9261771; 9261110\}$. Moreover, $d = card(A_{diag}) + card(A_{med.proc.}) + card(A_{drugs}) = 5$. Representation using activity set and its equivalent using activity vector are presented in Table Ia and Table Ib, respectively.

Thus, event log notations as well as activity vector and matrix formalization have been introduced. The following definitions introduce the problem setting.

IV. Problem definition

Definition 7: Label function and set. Given an event log L, a *label function* λ is a function such that:

$$\lambda : A \rightarrow L \tag{3}$$

with L being the *label set*.

The label function maps activities of an event log L to L, a set of possible labels for each activity vector.

Definition 8: Explaining function. Given an event log L and its label set L, an *explaining function* η on L is a function:

$$\eta : L \rightarrow A \tag{4}$$

The explaining function allows a mapping of a label l to interpretative elements a from the log activity set : $a \subset A$

Definition 9: Activity clustering problem. Lets L be an event log with its log activity set A and its activity matrix M_X, with $dim(M_X) = len(L) \times d$. The *activity clustering problem* on L is defined as the search of the triple (L, λ, η) such that:

- L is a label set, $L = \emptyset$;
- $K_{min} \leq card(L) \leq K_{max}$, with $K_{min}, K_{max} \in \mathbb{N}^*$;
- $K_{max} \ll d$;
- λ is a label function;
- η is an explaining function.

The main objective here is to find an accurate triple (L, λ, η) for the considered event log L. This problem can be seen as a clustering problem, with L being the set of cluster labels and λ the clustering function. The particularity here is for the input data M_X to be sparse and of high dimensionality d in terms of features. The label set L should be finite, its cardinality (number of elements) being reasonable. A high cardinality of L will induce difficulties in process mining readability, also encouraging overfitting regarding A. This consideration leads to the proposed upper bound condition $K_{max} \ll d$. But a small cardinality for the label set will lead to a lack of information and medical meaning, motivating the lower bound K_{min}. Moreover, as explainability is an essential constraint when dealing with medical pathways analysis, elements of L should keep a medical meaning, justifying the search of the explain function η. Given these considerations, it is assumed in this paper that there exist some relevant clusters hidden in medical activities of event log L. This assumption suggests that some combinations of elements of A which could characterize well a sufficient number of events e could be found.

In the following, a methodology based on autoencoding is proposed to find a relevant label set and a label function from a given event log.

V. Proposed methodology

A. Overview

To solve the activity clustering problem for an event log L and find an accurate triple (L, λ, η), an autoencoding method is proposed. The idea is to transform space data

from E_X to a *latent space* E_Z of reduced dimensionality, where similar elements are close to each other. This transformation is done using an *autoencoding* architecture, with an *encoder* $f : E_X \rightarrow E_Z$ and a decoder $g : E_Z \rightarrow E_X$. In the latent space, because the dimensionality is reduced, applying clustering methods is meaningful. *Latent clustering* is proposed to assign for each element z a label through a function $h : E_Z \rightarrow L$. Thus, cluster labels defined through clustering in latent space will constitute the label set L defined in Definition 7:

$$\lambda : A \xrightarrow{vect(a)} E_X \xrightarrow{f(x)} E_Z \xrightarrow{h(z)} L \tag{5}$$

To construct the explaining function η, the function $h^{-1} : L \rightarrow E_Z$ first returns for each cluster label l, the set of corresponding vectors in latent space Z_l. Then, each activity $z \in Z_l$ can be decoded from E_Z to E_X using the decoding function g learned during the autoencoder training phase. This results in a set of activity vectors X_l, from which an average vector $\bar{X}_l$ is computed. Finally, $\bar{X}_l$ is interpreted as a set of activities using $vect^{-1}$ function:

$$\eta : L \xrightarrow{h^{-1}(l)} E_Z \xrightarrow{g(z_l)} E_X \xrightarrow{vect^{-1}(\bar{x}_l)} A \tag{6}$$

Consequently, the proposed method is composed of three steps: (1) autoencoder training, (2) latent space clustering, and (3) clusters' related activities decoding. These steps are presented in more detail in the following sections.

B. Autoencoder training

To perform clustering on sparse, binary, high-dimensional activity vectors, the data is transformed into a new space where variables are continuous and the dimensionality is lower. This transformation is performed using an autoencoder, trained on an activity matrix. In this paper, three methods for autoencoder training are investigated.

1) Autoencoder: An autoencoder (AE) is composed of two functions f and g, named the *encoding* and *decoding* functions, respectively. The encoder transforms a vector x from the input space into a new vector z from the *latent space*, its dimensionality being drastically reduced. The decoder takes the vector z from the latent space and decode it back to the input space, resulting in a new vector x'. The training of a classic autoencoder is done by minimizing the reconstruction error, usually the binary cross entropy loss function. The dimensionality reduction allows a concentration of information while keeping the useful information in latent space for the reconstruction of input data. In this paper, encoder and decoder are constructed using symmetric feed-forward, fully connected neural networks.

2) Denoising autoencoder: A denoising autoencoder (DAE) is constructed using the same architecture as for an AE. A noisy vector $\tilde{x}$ is created from x, which is encoded and then decoded. The loss function remains unchanged, while the goal of training is to be robust against artificially added noise and to keep useful information to decode data without noise.

3) Variational autoencoder: A variational autoencoder (VAE) is a particular autoencoder where the learned variables are parameters of a distribution. The encoder f is an *inference network* $q(x|z)$ and the decoder g a *generative network* $p(z|x)$. The *reparameterization trick* makes the training of the network possible using gradient descent optimization. The loss function here is the inverse of the expected lower bound $ELBO$ defined as $ELBO = \mathsf{E}_{q(z|x)} \log \frac{p(x,z)}{q(z|x)}$. In practice, the single sample estimate $\log p(x|z) + \log p(z) - \log q(z|x)$ with z sampled from the inference network is optimized [26].

C. Latent space clustering

Once an autoencoder is trained using activity matrix, a representation of every activity vector in latent space E_Z can be obtained. In E_Z, observations are characterized by a reasonable number of continuous features. As a result, applying clustering in this space is meaningful. *K*-means algorithm is used to learn the function $h : E_Z \rightarrow L$ and create K clusters of similar observations in latent space. The parameter K corresponds to the final number of labels in L : $K = card(L)$, respecting $K_{min} \leq K \leq K_{max}$ (Definition 9). To find such an accurate value of K, one possible criterion could be to maximize the *mean silhouette score* [27], defined as:

$$S = \frac{1}{len(L)} \sum_{z \in M_Z} \frac{b_z - a_z}{max(a_z, b_z)} \tag{7}$$

with

- a_z : the mean distance between z and all other points of the same cluster;
- b_z : the mean distance between z and all other points of the nearest next cluster.

D. Clusters' activities decoding

Once the latent space clustering has constructed a function $h : E_Z \rightarrow L$, the label function $\lambda : A \rightarrow L$ is fully defined. A label is assigned to each activity set of event log L, which can be used in the final process model as a node label. To construct the explaining function $\eta : L \rightarrow A$, a methodology based on the decoding of each cluster's activities is proposed. The hypothesis formulated here is that for each cluster defined in latent space E_Z, averaging the output of its activities decoding gives an overview of the cluster's meaning. In practice, the $vect^{-1}$ function needs a threshold to convert an activity vector into decoded activities. The analysis of activity vectors can lead to a judicious choice for the threshold, as presented in Remark 1, Section VI.

To summarize this section, the presented methodology allows for the analysis of activity sets from event logs and defines a set of labels that can be used for process mining. Each event of the event log has a label defined through the label function, λ, and every label is interpreted by the explaining function η. A summary of preliminary notations is proposed in Table II.

name	notation	name	notation
activity	a_i	log activity set	A
activity set	a	activity vector	x
time-stamp	t	activity matrix	M_x
event	e	activity vector space	E_x
trace	σ	encoder	f
event log	L	decoder	g
label set	L	latent vector	z
label function	λ	latent matrix	M_z
explaining function	η	latent space	E_z

TABLE II: Notations summary.

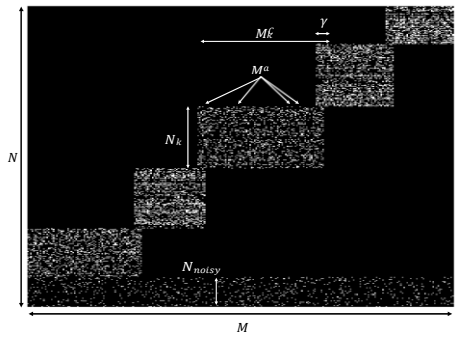


Fig. 1: Example of generated data. Here, $\kappa = 5$, $a = 0.05$, $\beta = 2$ and $\gamma = 0.1$, leading to $\tilde{S} = 0.9$ with each white pixel being a value of 1 and black pixels being a value of 0.

VI. Design of experiments

In this section, a design of experiment on synthetic data is presented. The objectives of such an experiment are multiple: (1) to verify the accuracy of the methodology in identifying hidden patterns (clusters) in event logs; (2) to demonstrate the veracity of the decoder in explaining clusters' labels; (3) to compare performances of autoencoding methods with direct clustering on sparse data; and (4) to benchmark autoencoding methods with one another.

A. Data description

The input data used in this paper are event logs, composed of events for which accurate labels are searched, according to the problem defined in Definition 9. Thus, labels hidden in the data were represented using groups of activities, among which vectors will be generated. Such a construction was motivated by data experts observations regarding non-clinical claims data: similar hospitalizations are often described by codes from a same "set", which can be approximated using clinicians knowledge. As an example, a stay for a given operation can be fill in the database using one code representing the operation, another for medical imaging procedures, some codes related to particular diagnosis related to complications of the operation or to the medical condition of the patient. Thus, there exist similarities between two stays for the same operation, because some codes are issued from a same "set", but they are rarely identical.

Considering these remarks, synthetic data were generated representing an activity matrix as defined in Definition 6, where each row represented an activity vector and each column a 1-of-k representation of activities. The number of different labels hidden in the data (i.e. the number of clusters to find) is referred to as κ. For each label $k \in [\![1, \kappa]\!]$, N_k vectors were generated such that $N_k = \mathcal{N}(\mu_N, \frac{\mu_N}{5})$, $\mu_N = \frac{2500}{\kappa}$. The number of *characteristic activities* for each label k is referred to as $M_k^C \in \mathbb{N}^*$, constructed such that $M_k^C = \mathcal{N}(\mu_C, \frac{\mu_C}{5})$ with $\mu_C = a \times \beta \times \mu_N$. The number of all different activities involved is referred to as M. For each activity vector of a given label k, a number $M^a = a \times \mu_N \in \mathbb{N}^*$ of activities was randomly chosen to construct it. The number M^a corresponds to the number of activities randomly chosen among characteristic activities. For these activities, the corresponding attribute value was set to 1, keeping 0 otherwise. An overlapping ratio γ is also introduced, representing the quantity of activities of a label shared with the nearest one. Moreover, a number $N_{noisy} = 250$ of noisy events was generated. These noisy events were composed of activities among all possible activities in the constructed data set, with no particular pattern related to a given hidden label k. As a result, the total number of traces (i.e. the number of rows) is $N = N_{noisy} + \sum_{k=1}^{\kappa} N_k$. The parameters that have been chosen in the design of experiments lead to an approximated sparsity $\tilde{S}$ between 0.9 and 0.99. An example of synthetic data generated is presented in Figure 1.

B. Experiment description

The autoencoder methods AE, DAE, and VAE, presented in Section V-B, were implemented and compared in terms of performances. Neural networks used are feed-forward networks, composed of four fully connected layers of size $10 \times d_{latent}$, $5 \times d_{latent}$ and d_{latent}, the latter being the dimensionality of the latent space defined as $d_{latent} = 8$. For DAE, noise was defined as randomly selecting elements in vectors and changing their values (from 0 to 1 or 0 to 1). 1% of noise is added in every vector. For VAE, d_{latent} couple of parameters of Gaussian distributions were learned as latent variables. The inverse single sample Monte Carlo estimate of the *ELBO* was used as the loss function to minimize during training. For each parameter combination of the data, autoencoder training was done using a symmetric architecture between the encoder and the decoder. Dropout and L-2 regularization were used for each layer to prevent the training from overfitting. The chosen optimizer for training was *Adam*, with mini-batch of size 32. The total number of epochs was set to 1000. Of the overall data, 80% was used for training, while 20% was kept to evaluate validation error for early stopping (with a patience of 25 iterations). After autoencoder training, K-mean clustering in latent space was applied using all training and validation data, K being fixed by maximizing mean silhouette score for $K \in [K_{min}, K_{max}]$ with $K_{min} = 2$ and $K_{max} = 15$.

Performances were evaluated regarding clustering on the one hand, and explainability on the second hand. To evaluate clustering performances, an automatic procedure constructed a confusion matrix between hidden labels and found clusters, maximizing the accuracy (the sum of the diagonal) by permuting columns only (found clusters) to align proposed clusters with potential corresponding ones in hidden ones. The

accuracy of resulting confusion matrix was then computed, defined as the ability of a method to accurately assign the right label to each event. To evaluate the ability of the method to explain found clusters, the *explaining F -score* F_η is introduced. Let $k \in \{1, \ldots, K^{true}\}$ be the label of the cluster hidden in data, and $c : k \to c(k) \in \{1, \ldots, K^{pred}\}$ being a function returning the corresponding cluster label according to previously described confusion matrix optimization. The average of decoded elements from the cluster $c(k)$ is computed, and its activity set $a^{pred}_{c(k)}$ is compared to characteristic activities of the corresponding label a^{true}_k. To quantitatively analyze the decoding performances, the *explaining F -score* is defined such as: $F_\eta = \frac{2 \times R_\eta \times P_\eta}{R_\eta + P_\eta}$ with $R_\eta = \frac{\sum_{k \in K^{true}} card(a^{true}_k \cap a^{pred}_{c(k)})}{\sum_{k \in K^{true}} card(a^{true}_k)}$ the *explaining recall* and $P_\eta = \frac{\sum_{k \in K^{pred}} card(a^{true}_{c(k)} \cap a^{pred}_k)}{\sum_{k \in K^{pred}} card(a^{pred}_k)}$ the *explaining precision*. These expressions are analogous to classical binary classification metrics. A high explaining recall means a high ability of the explaining function to find corresponding activities with hidden activities of the identified labels. Furthermore, a high explaining precision corresponds to a decoding that keeps interesting activities without being too general. Ideally, each discovered label corresponds to a hidden one. This is not necessarily the case, as the number of discovered and hidden clusters could be different. Fewer predicted than hidden clusters will impact the explaining recall, and more predicted than true clusters will impact explaining precision.

Remark 1: As mentioned in Section V-D, the threshold will have significant impact in the previously defined metrics. In the design of experiments conducted here, an automatic approach was used. For a cluster l, a list of all decoded values from the average decoding vector $\bar{X}_l$ was constructed. All elements of this list were ordered in descending order. By differentiation of this curve another list of values was obtained. The minimum value of the resulting curve was used to automatically define a judicious threshold for keeping activities in the explaining set of the related cluster.

For every combination of parameters, 10 data sets were constructed. Columns (activities) were shuffled, right before the shuffling of rows (events). The proposed method was applied using the previously described autoencoders (AE, DAE and VAE) as well as a direct K-mean clustering without autoencoding step, tested as a baseline (referred to as BASIC). Performances were analyzed through mean and standard deviation of clustering and explaining metrics. All algorithms and experiments were conducted using Python 3.7 and Tensorflow 1.14. A schematic description of experiments is presented in Figure 2.

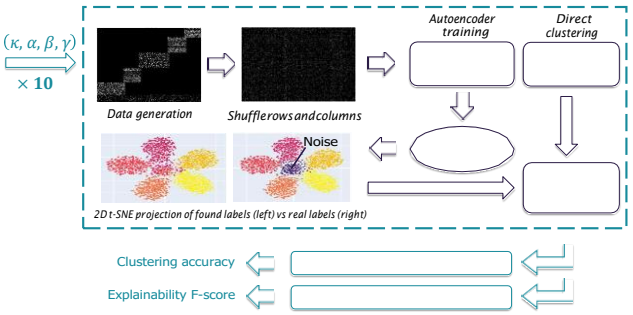


Fig. 2: Schematic representation of the design of experiments.

C. Results

Results[3] are summarized in Table III. A total of 24 experiments of increasing difficulty were conducted. For each combination of parameters, the accuracy and the F_η score of the tested methods are presented. Results show that the autoencoding methods outperform the direct clustering in sparse high-dimensional space (except regarding experiments 15 and 21, where DAE performances are inferior). Therefore, autoencoding plays an important role in data transformation for the proposed methodology. Furthermore, the results highlight that VAE always outperforms the other methods regarding both accuracy and F_η. Event though the standard deviation increases for difficult experiments, VAE shows a lower variation compared to the other methods. As a conclusion, results motivate the choice of VAE as part of the proposed method to obtain accurate clusters and explain them.

VII. Case study

After proving the accuracy of the method on synthetic data, this section is dedicated to demonstrate the relevance of the method on real healthcare data. In that purpose, a medical case study is presented, were process mining was deployed to extract knowledge about patients' hospital pathways.

A. Overview

A laparotomy is an abdominal surgery consisting of a large incision of the abdomen, sometime necessary to investigate abdominal pain. Incisional hernia (IH) is one of the possible complications following laparotomy. These complication consists in a protrusion of the tissues of the abdomen through the abdominal muscle. The repair on an IH is a common surgery, which can lead to chronic pain and decreased quality of life. Colorectal surgeries, bariatric surgeries and abdominal aortic aneurysm are laparotomy surgeries that may lead to IH [7]. In this case study, we focused on patients developing an IH after a laparotomy operation. The objective was to apply the previously defined methodology to label raw medical event logs before applying process mining. A process mining study using manual labeling was also performed, to illustrate the relevance of the methodology to automatically define interesting labels.

B. Methods

The data[4] were extracted from the SNIIRAM database. All anonimized patients with a first laparotomy operation in 2010,

[3]Detailed results regarding the design of experiments of Section VI, as well as the case study described in Section VII can be found as supplementary materials on the following website: https://artemis-emse-laparo.hevaweb.com/

[4]Access to an extraction of the SNIIRAM database was provided by the French CNIL under the agreement number DR-2019-147.

	Exp.				ACCURACY								F_η							
					BASIC		AE		DAE		VAE		BASIC		AE		DAE		VAE	
	κ	α	β	γ	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD
1	5	0.05	2	0.00	0.55	0.06	0.66	0.16	0.71	0.05	**0.90**	0.01	0.71	0.13	0.80	0.27	0.95	0.11	**1.00**	0.00
2	5	0.05	2	0.10	0.55	0.09	0.66	0.16	0.71	0.08	**0.89**	0.01	0.68	0.17	0.82	0.32	0.95	0.14	**1.00**	0.00
3	5	0.05	2	0.25	0.56	0.07	0.68	0.16	0.72	0.04	**0.89**	0.03	0.71	0.17	0.82	0.25	0.92	0.12	**0.97**	0.09
4	5	0.05	5	0.00	0.38	0.06	0.59	0.16	0.61	0.18	**0.85**	0.03	0.37	0.14	0.73	0.27	0.77	0.31	**1.00**	0.00
5	5	0.05	5	0.10	0.37	0.04	0.68	0.14	0.57	0.18	**0.86**	0.03	0.37	0.11	0.83	0.26	0.67	0.37	**1.00**	0.00
6	5	0.05	5	0.25	0.43	0.06	0.59	0.19	0.63	0.15	**0.85**	0.01	0.47	0.13	0.67	0.35	0.80	0.29	**1.00**	0.00
7	5	0.10	2	0.00	0.51	0.07	0.69	0.13	0.70	0.07	**0.88**	0.01	0.65	0.13	0.86	0.23	0.91	0.15	**1.00**	0.00
8	5	0.10	2	0.10	0.52	0.07	0.63	0.16	0.70	0.08	**0.89**	0.01	0.63	0.15	0.80	0.27	0.93	0.16	**1.00**	0.00
9	5	0.10	2	0.25	0.54	0.09	0.68	0.16	0.70	0.12	**0.88**	0.01	0.69	0.13	0.84	0.25	0.90	0.20	**0.94**	0.12
10	5	0.10	5	0.00	0.39	0.07	0.57	0.16	0.64	0.15	**0.82**	0.02	0.40	0.08	0.68	0.26	0.77	0.25	**1.00**	0.00
11	5	0.10	5	0.10	0.36	0.05	0.60	0.18	0.69	0.13	**0.83**	0.01	0.36	0.07	0.72	0.25	0.90	0.25	**0.97**	0.10
12	5	0.10	5	0.25	0.45	0.09	0.64	0.17	0.63	0.17	**0.84**	0.02	0.41	0.16	0.79	0.27	0.80	0.28	**1.00**	0.00
13	10	0.05	2	0.00	0.50	0.06	0.61	0.18	0.62	0.15	**0.87**	0.04	0.62	0.10	0.72	0.29	0.84	0.24	**0.98**	0.06
14	10	0.05	2	0.10	0.54	0.06	0.62	0.22	0.66	0.14	**0.88**	0.02	0.71	0.09	0.73	0.37	0.88	0.25	**0.97**	0.10
15	10	0.05	2	0.25	0.53	0.14	0.61	0.22	0.38	0.23	**0.86**	0.02	0.72	0.20	0.76	0.35	0.44	0.37	**0.94**	0.11
16	10	0.05	5	0.00	0.22	0.05	0.45	0.24	0.70	0.04	**0.82**	0.04	0.22	0.10	0.48	0.37	0.93	0.09	**0.95**	0.06
17	10	0.05	5	0.10	0.23	0.07	0.39	0.21	0.52	0.23	**0.83**	0.02	0.24	0.16	0.36	0.33	0.61	0.35	**0.96**	0.05
18	10	0.05	5	0.25	0.22	0.04	0.31	0.20	0.42	0.23	**0.83**	0.02	0.17	0.08	0.24	0.30	0.41	0.38	**0.93**	0.05
19	10	0.10	2	0.00	0.46	0.08	0.58	0.19	0.66	0.04	**0.86**	0.03	0.58	0.11	0.65	0.26	0.89	0.13	**1.00**	0.00
20	10	0.10	2	0.10	0.50	0.05	0.58	0.17	0.61	0.17	**0.86**	0.02	0.72	0.08	0.75	0.30	0.81	0.27	**0.97**	0.10
21	10	0.10	2	0.25	0.51	0.11	0.53	0.22	0.44	0.22	**0.86**	0.02	0.69	0.21	0.68	0.36	0.52	0.38	**0.94**	0.12
22	10	0.10	5	0.00	0.20	0.02	0.32	0.19	0.53	0.19	**0.82**	0.02	0.16	0.05	0.31	0.27	0.65	0.33	**0.95**	0.05
23	10	0.10	5	0.10	0.21	0.03	0.32	0.18	0.49	0.24	**0.82**	0.02	0.17	0.07	0.30	0.24	0.61	0.41	**0.92**	0.06
24	10	0.10	5	0.25	0.24	0.06	0.33	0.17	0.50	0.20	**0.80**	0.04	0.21	0.10	0.28	0.31	0.56	0.31	**0.89**	0.09

TABLE III: Clustering and explainability performance, measured by accuracy and F_η, respectively. For all parameters combinations, average and standard deviation over 10 replications are presented. Best values are highlighted in bold.

followed by an IH within 5 years after the operation were selected. This resulted in a total number of 7, 906 patients included in the study, for which all hospitalization information was extracted. Each patient's hospitalization was transformed into a trace of his ordered medical activities. Thus, the activity set was structured as follows:

$$A = A_{\mathrm{MD}}^{1} A_{\mathrm{AD}}^{1} A_{\mathrm{MP}}^{1} A_{\mathrm{D}}^{1} A_{\mathrm{TAD}} \quad (8)$$

where:

- A_{MD} is the set of main diagnoses, the reasons of the hospitalization (MD, using ICD-10 coding system);
- A_{AD} is the set of additional diagnoses (AD, using ICD-10 coding system);
- A_{MP} is the set of medical procedures (MP, using French CCAM coding system);
- A_{D} is the set of delivered drugs (D, using French UCD coding system linked to ATC - Anatomical Therapeutic Chemical - classes);
- A_{TAD} is the set of drugs under temporary authorization for delivery in French hospitals (using French LPP - *Liste des Produits et Préstations* - coding system).

Moreover, for each activity code, hierarchical knowledge (codes of upper levels in the hierarchy) was added as part of the corresponding activity set. This procedure enables relations between activity codes of a same group during autoencoding. It also enriches the explainability of clusters, by providing hierarchical knowledge and setting the level of precision in coding depending on clusters, as shown in the following results.

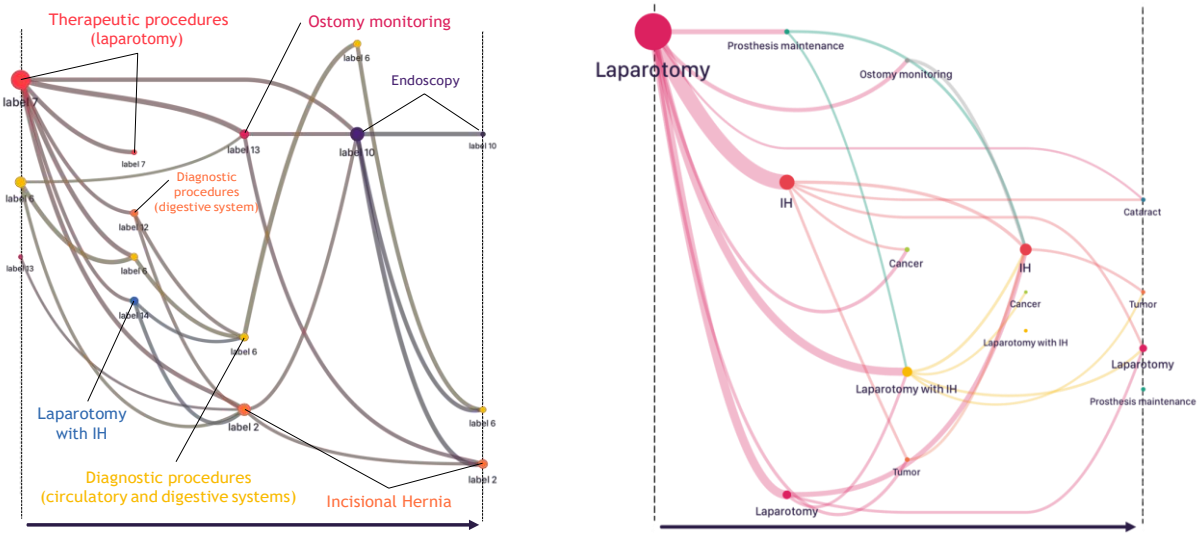
Stays related to dialyses or chemotherapy, which are known to appears very frequently, have been filtered. Codes appearing less than 50 times were also filtered, resulting in keeping 95.0% of codes while decreasing the size of the log activity set by 85.7%. The final event log constructed for the study contained, for 7906 traces (patients), 57533 events (stays) and 2228 unique activity codes. The previously defined methodology was conducted on the resulting activity matrix (of size 57533×2228), using VAE as autoencoder. The number of clusters used was $K = 15$, which qualitatively appears as a wise trade-off between explainability of clusters and final process model readability. The process mining framework used was the one proposed in [8], designed for application to medical event logs. The maximum number of nodes, edges and positions for process model optimization was fixed to 15, 25 and 5, respectively.

Cluster label	Activity label	Set (level)
2	Postoperative venting of the anterior abdominal wall	MP (lvl. 4)
	Ventral hernia	MD (lvl. 2)
6	Diagnostic acts on the circulatory system	MP (lvl. 2)
	Diagnostic acts on the digestive system	MP (lvl. 2)
7	Therapeutic acts on the digestive system	MP (lvl. 2)
10	Endoscopy of the alimentary canal	MP (lvl. 3)
12	Radiography of the digestive system	MP (lvl. 3)
13	Encounter for attention to artificial openings	MD (lvl. 2)
	Therapeutic acts on the colon	MP (lvl. 3)
14	Therapeutic acts on digestive system	MP (lvl. 2)
	Therapeutic acts on the anus	MP (lvl. 3)
	Therapeutic acts on the abdominal wall	MP (lvl. 3)

TABLE IV: Explanation of clusters appearing on process model: relevant decoded activities with corresponding activity set and level in the hierarchy.

C. Results

Results obtained by automatic labeling (Figure 3a) were compared with a process model of the same dimensionality, constructed with the same process mining procedure but start-



(a) Automatic labeling.

(b) Manual labeling.

Fig. 3: Comparison of process models obtained using automatic labeling (left) vs manual labeling (right).

ing from an event log with manually defined labels according to authors' prior knowledge about the pathology (Figure 3b). The process models read from left to right, the size of nodes and edges being, for each process model, proportional to the number of patients represented. Explanation results regarding clusters obtained and visualized in Figure 3a are detailed in Table IV. According to Figure 3, similarities are observed between the process models. Most frequent medical procedure codes related to laparotomy are subcategories of the "Therapeutic acts on the digestive system" code in the hierarchy. Thus, the related cluster (label 7) appears at the beginning of the pathways. Label 2, which contains codes related to IH, appears in the following of the pathway, as well as label 14 (combining codes related to laparotomy and IH). Also, stays related to diagnostic procedures (labels 6 and 12) and more precisely to an endoscopy of the alimentary canal (label 10) occupy a significant place in the process analyzed. These stays were not considered during manual labeling but were pointed out by the automatic labeling procedure based on raw data. They may be related to patient's medical control or investigation regarding suspicion of complication after operation. This example illustrates that other interesting information can be extracted from raw data with minimal initial input from the user. However, by comparing the replayability score (which gives a quantitative fitness measure of the resulting process models), a gap was observed between automatic (42, 7%) and manual (77.4%) labeling. The main explanation may be provided by the first laparotomy node (representing 3, 802 vs 7, 906 patients, respectively). In practice, 549 different medical procedure codes, from different chapters of the hierarchy, were selected by medical experts to identify laparotomy in the database. Even if most of the codes are gathered in label 7, remaining laparotomy stays were grouped in other less frequent clusters, which do not appear in the final process model because of the size constraint in optimization. Thus, even if a quantitative replayability gap remains between the two presented methods, the qualitative interpretation resulting from the pathway analysis remains similar, as most of the interesting events were pointed out. Moreover, the final process model and the explanation of clusters furnish an interesting base for discussion with medical experts.

VIII. Conclusion

In this paper, a methodology to handle the complexity of event logs regarding activities was presented. Based on autoencoding, artificial labels to characterize these events are created, which can be used to apply process mining. Explainability of each label is possible through decoding, which allows the practical application of this method in fields like healthcare where transparency is essential. A design of experiments was presented, designed to mimic non-clinical claims databases regarding authors' knowledge. The ability of the method to both create relevant clusters and explain them accurately was demonstrated. In particular, the Variational Auto-Encoder shows better performances than others tested autoencoders, motivating the use of such learning methods for further applications. Finally, a case study has been presented, illustrating the potential of the methodology when applied on real healthcare data. The presented method sounds promising as a preprocessing solution for process mining, to handle the complexity of medical activities in non-clinical claims databases and of other similar databases.

Further work will focus on the deployment of the method in new case study, to experience the method and generalize its utilization. Particularly, a focus on providing interactive tools to explore the results and facilitate discussions with clinical experts will be made. As the complexity of medical event logs depends on the database, other complexities could be considered in further studies. As an example, the integration of free text with structured medical information to create pertinent medical labels could be interesting to consider in future

works. As noted by Helm et al [28], a lack of sufficient coding in existing case studies remains. Therefore, addressing such complexities in the method will be interesting for practical uses on case studies with insufficiently coded data. Also, the present methodology is a preprocessing step, applied before deploying a process discovery algorithm (in our example, based on optimization). An interesting subject could be the fusion of both steps, by integrating the labeling step directly during the optimization procedure of creating the final process model. Also, as the K-mean algorithm was used to perform clustering in the latent space, future work will focus on testing other clustering algorithms. On a larger scope, future works will focus on the use of the proposed methodology to perform supervised learning on complex event logs. The proposal of a transparent classification algorithm is of interest, particularly for patients pathways data. Furthermore, the bridging of process mining and deep learning is an interesting research track. The use of recent advancements in deep learning for process analysis and prediction seems promising, in particular if process mining is used as an interface between model learning and human understanding.

REFERENCES

[1] A. R. C. Maita, L. C. Martins, C. R. L. Paz, L. Rafferty, P. C. K. Hung, S. M. Peres, and M. Fantinato, "A systematic mapping study of process mining," *Enterprise Information Systems*, vol. 12, pp. 505–549, May 2018.

[2] W. van der Aalst, T. Weijters, and L. Maruster, "Workflow mining: discovering process models from event logs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, pp. 1128–1142, Sept. 2004.

[3] W. v. d. Aalst, *Process Mining: Data Science in Action*. Berlin Heidelberg: Springer-Verlag, 2 ed., 2016.

[4] P. Tuppin, J. Rudant, P. Constantinou, C. Gastaldi-Ménager, A. Rachas, L. de Roquefeuil, G. Maura, H. Caillol, A. Tajahmady, J. Coste, C. Gissot, A. Weill, and A. Fagot-Campagna, "Value of a national administrative database to guide public decisions: From the système national d'information interrégimes de l'Assurance Maladie (SNIIRAM) to the système national des données de santé (SNDS) in France," *Revue d'Épidémiologie et de Santé Publique*, vol. 65, pp. S149–S167, Oct. 2017.

[5] M. Prodel, V. Augusto, X. Xie, B. Jouaneton, and L. Lamarsalle, "Stochastic simulation of clinical pathways from raw health databases," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, pp. 580–585, Aug. 2017.

[6] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie, "Optimal Process Mining for Large and Complex Event Logs," *IEEE Transactions on Automation Science and Engineering*, vol. 15, pp. 1309–1325, July 2018.

[7] R. Phan, V. Augusto, D. Martin, and M. Sarazin, "Clinical Pathway Analysis Using Process Mining and Discrete-Event Simulation: an Application to Incisional Hernia," in *2019 Winter Simulation Conference (WSC)*, (National Harbor, MD, USA), pp. 1172–1183, IEEE, Dec. 2019.

[8] H. De Oliveira, V. Augusto, B. Jouaneton, L. Lamarsalle, M. Prodel, and X. Xie, "Optimal process mining of timed event logs," *Information Sciences*, vol. 528, pp. 58–78, Aug. 2020.

[9] M. Vandromme, J. Jacques, J. Taillard, L. Jourdan, and C. Dhaenens, "A Scalable Biclustering Method for Heterogeneous Medical Data," in *Machine Learning, Optimization, and Big Data* (P. M. Pardalos, P. Conca, G. Giuffrida, and G. Nicosia, eds.), Lecture Notes in Computer Science, (Cham), pp. 70–81, Springer International Publishing, 2016.

[10] M. Vandromme, J. Jacques, J. Taillard, A. Hansske, L. Jourdan, and C. Dhaenens, "Extraction and optimization of classification rules for temporal sequences: Application to hospital data," *Knowledge-Based Systems*, vol. 122, pp. 148–158, Apr. 2017.

[11] H. De Oliveira, M. Prodel, and V. Augusto, "Binary Classification on French Hospital Data: Benchmark of 7 Machine Learning Algorithms," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1743–1748, Oct. 2018.

[12] D. Rav`ı, C. Wong, F. Deligianni, M. Berthelot, J. Andreu-Perez, B. Lo, and G.-Z. Yang, "Deep Learning for Health Informatics," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, pp. 4–21, Jan. 2017.

[13] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi, "Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, pp. 1589–1604, Sept. 2018.

[14] C. Xiao, E. Choi, and J. Sun, "Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review," *Journal of the American Medical Informatics Association*, vol. 25, pp. 1419–1428, Oct. 2018.

[15] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics*, vol. 19, pp. 1236–1246, Nov. 2018.

[16] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor AI: Predicting Clinical Events via Recurrent Neural Networks," in *Machine Learning for Healthcare Conference*, pp. 301–318, Dec. 2016.

[17] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, "RETAIN: An Interpretable Predictive Model for Healthcare using Reverse Time Attention Mechanism," in *Advances in Neural Information Processing Systems 29* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds.), pp. 3504–3512, Curran Associates, Inc., 2016.

[18] T. G. Erdogan and A. Tarhan, "Systematic Mapping of Process Mining Studies in Healthcare," *IEEE Access*, vol. 6, pp. 24543–24567, 2018.

[19] A. Pika, M. T. Wynn, S. Budiono, A. H. M. ter Hofstede, W. M. P. van der Aalst, and H. A. Reijers, "Towards Privacy-Preserving Process Mining in Healthcare," in *Business Process Management Workshops* (C. Di Francescomarino, R. Dijkman, and U. Zdun, eds.), vol. 362, (Cham), pp. 483–495, Springer International Publishing, 2019.

[20] R. Gatta, M. Vallati, C. Fernandez-Llatas, A. Martinez-Millana, S. Orini, L. Sacchi, J. Lenkowicz, M. Marcos, J. Munoz-Gama, M. Cuendet, B. de Bari, L. Marco-Ruiz, A. Stefanini, and M. Castellano, "Clinical Guidelines: A Crossroad of Many Research Areas. Challenges and Opportunities in Process Mining for Healthcare," in *Business Process Management Workshops* (C. Di Francescomarino, R. Dijkman, and U. Zdun, eds.), Lecture Notes in Business Information Processing, (Cham), pp. 545–556, Springer International Publishing, 2019.

[21] N. Martin, A. Martinez-Millana, B. Valdivieso, and C. Fernández-Llatas, "Interactive Data Cleaning for Process Mining: A Case Study of an Outpatient Clinic's Appointment System," in *Business Process Management Workshops* (C. Di Francescomarino, R. Dijkman, and U. Zdun, eds.), vol. 362, (Cham), pp. 532–544, Springer International Publishing, 2019.

[22] X. Xu, Y. Wang, T. Jin, and J. Wang, "Learning the Representation of Medical Features for Clinical Pathway Analysis," in *Database Systems for Advanced Applications* (J. Pei, Y. Manolopoulos, S. Sadiq, and J. Li, eds.), Lecture Notes in Computer Science, (Cham), pp. 37–52, Springer International Publishing, 2018.

[23] A. Alharbi, A. Bulpitt, and O. A. Johnson, "Towards Unsupervised Detection of Process Models in Healthcare," *Studies in Health Technology and Informatics*, vol. 247, pp. 381–385, 2018.

[24] R. Lenz and M. Reichert, "IT support for healthcare processes – premises, challenges, perspectives," *Data & Knowledge Engineering*, vol. 61, pp. 39–58, Apr. 2007.

[25] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, Massachusetts: MIT Press, Nov. 2016.

[26] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *arXiv:1312.6114 [cs, stat]*, May 2014.

[27] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, Nov. 1987.

[28] E. Helm, A. M. Lin, D. Baumgartner, A. C. Lin, and J. Küng, "Towards the Use of Standardized Terms in Clinical Case Studies for Process Mining in Healthcare," *International Journal of Environmental Research and Public Health*, vol. 17, Feb. 2020.

3.4 Conclusion

In this chapter, a preprocessing method for the labeling of complex event logs is presented. In the context of complex macro-medical events, this method is able to learn a useful representation of stays including all available codes and relative hierarchical information. The latent representation of stays is useful to perform clustering, and shows satisfying performances in finding hidden clusters represented by sparse vectors. A case study on laparotomy was presented, validating the use of the method in a real case study. An interactive tool was also developed as a proof of concept, which is available online. Such an interactive tool can be useful throughout all phases of case study, in order to discuss primary results with medical experts. The proposed preprocessing method as well as the proposed interactive tool are not designed to perform full end-to-end studies but to enrich discussions with medical experts by presenting data-driven results.

A limitation regarding the integration of this preprocessing method with the previously defined process mining method should be discussed. In Chapter 2, the process mining approach uses an event log with predefined labels and performs process discovery using an optimization process. Automatic labeling as a preprocessing step can be well integrated into the method. However, future work could study the integration of automatic labeling directly during the optimization process, as performed by Prodel et al. when choosing the right hierarchical level during optimization [160]. Furthermore, future work should focus on deploying the method for future process mining studies using health data to test the robustness of this methodology.

It can be concluded that the presented work improves event representation and information processing for process mining. As performing prediction is the main objective of this thesis, the next chapter introduces an optimization-based method for classification which is constructed using the formalism defined in Chapter 2.

Chapter 4

An Optimization-based Process Mining Approach for Explainable Classification of Timed Event Logs

Contents of the chapter

4.1 Motivation

The main objective of the work presented in this chapter is the design of a predictive model for patient pathway data in the form of event logs. When data is organized in event logs, process mining provides a set of tools useful to model this input data. This is particularly the case when considering explainability as a necessity for predictive models, as the result-ing process model could serve as a support to explain predictions. In Chapter 2, grid and time grid process models were introduced to model patient pathways and incorporate time during process discovery optimization. In Chapter 3, a method for the labeling of complex medical events with multiple activities was presented. As a result, the previous chapters leverage challenges in time modeling and complex medical labeling. These contributions were first demonstrated in the context of process discovery. The present chapter employs the previous contributions in the context of of predictive modeling of event logs, more precisely the classification of traces. The motivation here is to extract pathway-related information which may influence the occurrence of a particular outcome. Such patterns are the occurrence of certain events, the transition from one event to another, or a specific time period occurring between two events. Using the process models introduced in Chap-

ter 2, such distinctive patterns are highlighted in the formalism and in the optimization process.

4.2 Summary

A new problem is presented in the following work, which is the problem of supervised classification of timed event logs of two classes: positive and negative population. In healthcare, this problem is recurrent in the literature, to detect patients with a given risk, to identify long LOS patients, or to predict readmission. Moreover, emphasis is put on the explainability of such predictions. In order to predict and explain the results of learning at the same time, the main idea is to determine a process model that agrees well with the positive traces and poorly with the negative ones. To do so, event log classification is introduced as an optimization problem for the determination of a process model that maximizes its replayability for the positive population and minimizes its replayability for the negative population. Based on the process discovery objective function introduced in Chapter 2, a new objective function is presented, adapted for binary classification. The same optimization process using a tabu search is used to solve this problem. An evaluation of the proposed method is performed on unbalanced synthetic data of various complexity. Performances are compared to different machine learning methods applied to features extracted from the event log. Results show the ability of the proposed method to accurately classify unbalanced data, outperforming other machine learning methods combined with an oversampling strategy and hyperparameters tuning. Moreover, qualitative results were analyzed, using the resulting process models which produce a graphic representation of distinctive patterns extracted from the positive traces.

The work presented was first presented as a poster at the *2019 Data Science Summer School (DS3)*. This poster is presented in Appendix C.

4.3 An Optimization-based Process Mining Approach for Explainable Classification of Timed Event Logs

H. De Oliveira, V. Augusto, B. Jouaneton, L. Lamarsalle, M. Prodel and X. Xie, "An optimization-based process mining approach for explainable classification of timed event logs", *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, pp. 43-48, 2020, https://doi.org/10.1109/CASE48305.2020.9216841.

An optimization-based process mining approach for explainable classification of timed event logs

Hugo De Oliveira[1,2], Vincent Augusto[1], Baptiste Jouaneton[2], Ludovic Lamarsalle[2], Martin Prodel[2] and Xiaolan Xie[1,3]

***Abstract*— This paper addresses the problem of supervised classification of time event logs of two classes: positive and negative population. The key idea of this paper to explain classification is to determine some process model that fits well the positive event logs and poorly the negative ones. More specifically, we introduce formal definitions of event logs, process models and a replayability score that measures the fitness of a process model for a given event log. We then set the event log classification as an optimization problem for the determination of a process model that maximizes its replayability for the positive population and minimizes its replayability for the negative one. A tabu search algorithm is then proposed to solve this problem. The proposed algorithm is compared with three state-of-the-art classification algorithms on test cases of various complexity. It is shown to provide superior performances and a graphic representation of the process model of the positive event logs.**

I. Introduction

Data is a powerful resource. Different structures of data can be found, within a large spectrum of complexity. In the field of supervised learning, machine learning algorithms for classification have been widely used. The paradigm for state-of-the-art classification algorithms is matrix-shaped input data: each observation (row) is a vector of features (column). Once trained, classifier's predictions for new observations are based on feature similarity with training observations.

However, when data is structured in event logs, each observation is an ordered list of events and no longer a single vector of features. Distinctive characteristics (patterns) can be of different types, as for example a special event's occurrence, an event preceding another or a typical time between two events. Data engineering exists in order to transform an event log into a feature matrix ("flattening" process). This data preprocessing step is challenging because potential distinctive patterns of the event log data need to be kept for the classifier. Furthermore, it might lead to high dimension and sparse matrices, especially when considering time between events.

Even if predictive performance is the predominant criterion for model approval, human understanding is a key lever for acceptance and practical application of decisions. This is the case in healthcare, where the identification of patients with pathways being suspicious of developing future medical complications is valuable. It offers the opportunity to identify at-risk patients and to respond with personalized medical care and prevention. However, ensuring a high predictive performance is challenging, as patients' pathways are complex: high number of different medical events, variability of pathways' length, variability of time between events... The imbalance between groups of patients with and without a given complication is also a recurrent problem. Even if some preprocessing methods such as over/under-sampling exist in the literature, performances can be substantially impacted.

To tackle these scientific challenges, we propose in this paper an explainable method for classification of time event logs of two classes: positive and negative population. The main idea to predict and explain is to construct a process model that fits well the positive event logs and poorly the negative ones. The proposed framework which relies on *time grid process models* [1], has the following characteristics: (1) designed for event log data; (2) robust to imbalanced classes; (3) explainable through the obtained process model, which represents the knowledge extracted from the event log of the positive class.

This paper is organized as follows. Section II presents a brief literature review related to classification using event logs. Important definitions and notations are presented in Section III. In Section IV, the problem settings and the proposed methodology is introduced. Section V presents a design of experiments on simulated data to assess the proposed methodology performances. Finally, a conclusion and future perspectives are given in Section VI.

II. Literature review

Collecting real-life process data results in time-dependent event logs. Many fields are concerned such as healthcare, manufacturing industry, software engineering or telecommunication [2]. The use of unsupervised methods on event logs is helpful to extract knowledge from data. For that purpose, process mining has become state-of-the-art. The primary objective of Process Mining is to do process discovery, i.e. to represent a summarized model of the event log [3]. It has been used in healthcare to map care processes and clinical pathways [4]. For example, Prodel et al. [5] used linear integer programming to discover patients' pathways from hospital data. In 2018, authors presented a meta-heuristic to perform optimal discovery of clinical pathways [6]. An

[1]H. De Oliveira, V. Augusto and X. Xie are with Mines Saint- Étienne, Univ Clermont Auvergne, CNRS, UMR 6158 LIMOS, Centre CIS, F - 42023 Saint-Étienne France (e-mails: hugo.de-oliveira@emse.fr; augusto@emse.fr; xie@emse.fr).

[2]H. De Oliveira, B. Jouaneton, L. Lamarsalle and M. Prodel are with HEVA, 186 avenue Thiers, F-69465, Lyon, France (e-mails: bjouaneton@hevaweb.com; llamarsalle@hevaweb.com; mprodel@hevaweb.com).

[3]X. Xie is also with the Antai College of Economics and Management, Shanghai Jiao Tong University, China.

enhancement has been proposed by De Oliveira et al. [1] to introduce time in optimal process models.

The use of process mining frameworks to perform predictions has been already presented in the literature. Examples are numerous, as for example the prediction of the time before the occurrence of an event in a process, or the probability of a given task to be performed [7]. Each of these tasks could be performed using different methods. In the case of next activity prediction, Ferilli et al. presented two methods based on the WoMan framework [8]. The implementation of a predictive model in each node of a process model to predict future steps taking into account patients' characteristics has also been proposed in [9].

Binary classification applied on event logs data has been addressed in the literature. If time is neglected and only a succession of events is analyzed, the classification of event log data problem is similar to sequence classification. For that, three types of studies can be identified [10]: (1) featured-based classification (extracting features from an event log to create a matrix input for a classifier model); (2) distance-based classification (using a similarity measure between sequences); (3) model-based classification. The two first types of approaches are used in bioinformatics for DNA (deoxyribonucleic acid) alignment, and based on models defined by [11], [12] in particular. Improvements of these methods are explored in the literature [13]. The third type gathers statistical models as Hidden Markov Models [14], [15].

For supervised prediction on event log data, a data preprocessing phase is generally needed: applying existing prediction algorithms on event logs is not straightforward, as the event log needs to be transformed into a feature matrix. This transformation is done automatically using features extraction or using experts' knowledge. Healthcare is a field of interest, as patients' pathways are defined by succession of medical events, time between events being a key indicator of care. State-of-the-art machine learning approaches have been widely used for prediction in healthcare. Case studies found in the literature are of various type [16], as prediction of diseases [17], mortality [18], prevention tests [19] and readmission [20], [21]. Medical features are generally selected by experts, but the longitudinal structure of patients' pathway is either lost during feature extraction or leads to a sparse representation of data [22]. Moreover, algorithms like Decision Tree or Logistic Regression are preferred by practitioners due to their explainability.

As a result, to the best of our knowledge, no classification algorithm has been designed for event log data, with a particular focus on learning explainability. This focus carries potential applications, such as healthcare and particularly patient's pathway constitutes the initial motivation of the development of such a methodology. The explainability of predictions for medical experts and decision makers is essential, especially when time is a possibly distinctive feature.

III. Preliminaries on event logs and process models

A. Event log

Definition 1: (Event). An event denoted e is defined as a couple (a, t) where $a \in A$ is an element of a finite set A of labels corresponding to the event class of e, and $t \in T$ with $T = \mathbb{N}$ or $\mathbb{R}$ is the event time or time-stamp. An event e is also defined by the labeling function $label(e) = a$ and the timing function $time(e) = t$.

Definition 2: (Trace). A trace is a sequence of events $\sigma = e_1, e_2, \ldots, e_m$ with $m \in \mathbb{N}^*$ such that $e_k \in A \times T$ and $time(e_k) < time(e_{k+1})$.

Definition 3: (Event log). An event log is a set of traces $L = \{\sigma_1, \sigma_2, \ldots, \sigma_n\}$ with $n \in \mathbb{N}^*$. An event log contains all input data of this paper. It is assumed that each label appears at least once in the event log L, i.e. $\forall a \in A : \exists \sigma \in L, e \in \sigma \mid e = (a, t)$.

Definition 4: (Event diversity). The event diversity div_e is defined as $div_e = |A|$. This descriptor gives information about the variability of the event log in terms of labels.

B. Process models

Definition 5: (Time grid process model). A time grid process model of a given log L is a four-uplet $TG\text{-}PsM = (N, E, L, T)$ where:

- N is a set of nodes partitioned into K disjoint subsets called layers, i.e. $N = N_1 \cup \cdots \cup N_k$, $N_k \cap N_l = \emptyset$;
- $E \subset N \times N$ is a set of edges such that $(x, y) \subset E$ with $x \in N_k$, $y \in N_l$ implies $k < l$, i.e. the process model is acyclic with edges going from lower layers to higher layers;
- $L : N \rightarrow A$ is the labeling function of the nodes.
- $T : E \rightarrow T \times T$ associates a time interval $[a_{(x,y)}, b_{(x,y)}]$ to each edge $(x, y) \in E$.

Interesting properties of such a process model are as follows. A same label can appear at different positions in the model. Constraints for edges link lower positions to strictly higher ones. This produces oriented process models, with no backward edge and possibly a same label found in lower an higher positions. Moreover, multiple edges can be found between two nodes, each edge having a different time characteristic. This time characteristic on edges serves to consider time during the optimization process.

In the following, all process models are supposed to be time grid process models, as defined in Definition 5.

C. Replayability

Definition 6: (Replayability). The replayability function is denoted R, and returns the replayability score:

$$R(\text{TG-PsM}, \sigma) \in [0, 1] \quad (1)$$

which is the representativeness of the trace σ by the process model $TG\text{-}PsM$. By extension, the replayability score distribution of an event log L is the set of replayability score values for each trace in L:

$$R(\text{TG-PsM}, L) = (R(\text{TG-PsM}, \sigma))_{\sigma \in L} \quad (2)$$

The replayability is used in [5], [6], [1] to evaluate the ability of a process model to represent a given trace. The result of the procedure is a replayability score between 0 and 1, where 1 corresponds to the best possible representation of a trace by a process model. The following elements are positively taken into account in the replayability: (1) nodes matching trace's events; (2) edges matching event transitions; (3) time characteristic of edges matching time-stamp of event logs; (4) no central event of the trace skipped. As the replayability score measures the ability of a process model to represent a given trace, the analysis of the replayability score distribution points out the representativeness of a process model regarding the entire event log. Further details about the replayability for time grid process models are given in [1].

IV. Process model-based classification of event logs data

The proposed approach is an optimization-based method, at the crossroad between machine learning, process mining and operational research. In this section, we formally define the problem settings and describe the methodology, with a particular focus on the optimization process involved.

A. Problem setting

The problem here consists of having two labeled event logs L_0 and L_1, and learn patterns from this data in order to predict for new, unlabeled traces. In other words, the problem addressed here is a binary classification problem, with data involved being event log of traces (and not sets of labeled vectors described by features as in classic binary classification).

Lets define a binary classed event log:

$$L^{train} = (L_0^{train}, L_1^{train}) \quad (3)$$

where traces from L_k^{train} are of class k for $k \in \{0, 1\}$. For a given process model $TG\text{-}PsM$, let

$$R_0^{train} = R(\text{TG-PsM}, L_0^{train}) \quad (4)$$

and

$$R_1^{train} = R(\text{TG-PsM}, L_1^{train}) \quad (5)$$

be the replayabilities of traces of L_0^{train} and L_1^{train}, respectively. Resulting replayability distributions can be visualized on a single plot, as shown in Figure 1. Supposing that the $TG\text{-}PsM$ better represents traces from the positive class than traces from the negative one, replayability scores from R_1^{train} will be generally higher than replayability scores from R_0^{train}. The process of training a $TG\text{-}PsM$ consists in creating such a process model.

B. Process model-based classification

The process model-based classification algorithm is composed of 2 steps:

1) **Train**: construct a $TG\text{-}PsM$ from L^{train} to get replayability distributions R_0^{train} and R_1^{train};

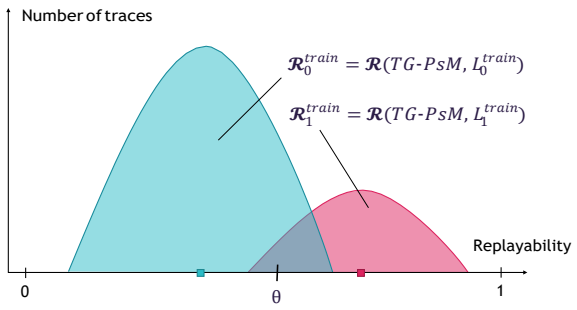


Fig. 1: Replayability graph of a process model $TG\text{-}PsM$ on training data $(L_0^{train}, L_1^{train})$ with threshold θ for classification.

2) **Predict**: for a new trace σ, compute its replayability $R(TG\text{-}PsM, \sigma)$ and return the corresponding predicted class by comparing it to a given threshold θ.

The choice of the threshold θ is a widespread issue for binary classifiers, to predict in practice for every individuals. To find the best split between the two replayability distributions, the threshold which minimizes the gini impurity is chosen. One can infer that the construction of the process model (the training of $TG\text{-}PsM$) is the key to improve classification performances. The main idea here is to build a process model which produces distinct distributions for both training classes on the replayability graph. A Tabu search is used, motivated by previous work [6], [1]. Before detailing the search algorithm, two objective functions are described.

C. Objective functions for process model optimization

Two objective functions are presented, each involving a different measure of process model quality. The average replayability function is denoted as:

$$\bar{R}(\text{TG-PsM}, L) = \frac{1}{|L|} \sum_{\sigma} R(\text{TG-PsM}, \sigma) \quad (6)$$

1) RepOpt: The first objective function consists in searching a final solution which maximizes the mean replayability of the event log L_1^{train} (positive class):

$$\bar{R}(\text{TG-PsM}, L_1^{train}) \quad \text{(RepOpt)}$$

One can notice that elements of L_0^{train} stay unused during the optimization process. This objective function was used in process discovery for unsupervised process mining [1].

2) DiffOpt: Instead of maximizing the replayability of the traces of the positive class, we maximize the difference between the means of the two classes. The idea is to construct a graph that best replays traces of L_1^{train} and that replays badly traces of L_0^{train}.

$$\bar{R}(\text{TG-PsM}, L_1^{train}) - \bar{R}(\text{TG-PsM}, L_0^{train}) \quad \text{(DiffOpt)}$$

Expectations with this objective function is the evacuation of shared patterns between positive and negative classes, while keeping those specific to the positive one.

The two previously defined objective functions (RepOpt) and (DiffOpt) constitute the core of the local search procedure used to create optimal process models. This procedure is detailed thereafter.

D. Tabu search for process model optimization

The proposed methodology to fit a process model *T G-PsM* on train data L^{train} is an optimization process based on a local search. Starting from a random solution (i.e. a process model), a neighborhood of solutions is created. Each neighbor is a slightly modified version of the current process model (2 possible moves: add a new promising node or randomly delete a node). Each neighbor is then evaluated by computing the objective function (RepOpt) or (DiffOpt). The best neighbor is kept as the current solution and added to a fixed sized first-in-first-out list of tabu solutions. Tabu solutions cannot be selected when creating a neighborhood. This process is iterated until a stopping criterion is reached (a total maximum number of iterations or a maximum number of iterations without any improvement). Resulting process model is the best evaluated solution encountered during the entire search.

Required parameters for the optimization procedure are the constraints (the maximum number of nodes U_N , the maximum number of edges U_E and the maximum position in the process model p_{max}) and search parameters (the neighborhood size, the tabu list size, and the stopping criteria). The set of time intervals for edges is also an input parameter. Pertinent time intervals are constructed using Kernel Density Estimation, previously proposed in [1].

V. Numerical Experiment

The classification methodology is validated through the following design of experiments. Event logs $(L_0, L_1) = (L_0^{train}, L_0^{test}, L_1^{train}, L_1^{test})$ are build with different hidden patterns, the objective being to learn from training event logs and accurately predict for test ones.

A. Data generation

Two graphs G_0 and G_1 are constructed, constituted of nodes arranged in layers having a maximum number of identical positions equal to p_m. For each position, the corresponding layer is composed of $n = div_e \in [1, p_m]$ different nodes. Then, a proportion of shared patterns is removed, by deleting $c_{pat} * N$ randomly chosen nodes from G_1 and corresponding edges. An illustration for G_0 and G_1 is shown on the left of Figure 2.

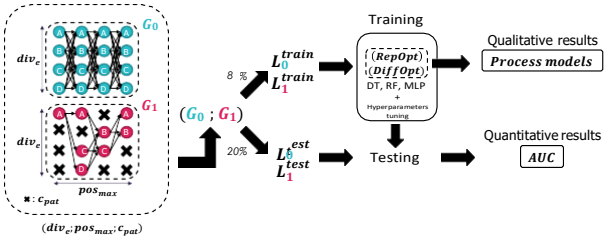


Fig. 2: Schematic representation of the design of experiments.

A trace σ is created by starting from the lower position of the graph, and by adding to the trace an event with the same label as a randomly selected node of the next increasing position. The time-stamp of the new event is added to the previous time-stamp; it is a time value randomly selected for L_0 and L_1 following respectively $U(0, 400)$ and $N(200, 25)$. At each step the process can be stopped with a probability $p = \frac{P(n_{current})}{p_m}$, where $n_{current}$ is the current node of G, corresponding to the last addition to σ. Such event log construction process ensures the presence of a pattern in G_1, in terms of labels, transitions and time. The probability of stopping the construction process ensures a variability in traces' lengths. The higher c_{pat} is, the smaller G_1 and the more specific the process model will be. Event logs dimensions are noted $N = |L_0^{train}|$ and $P = |L_1^{train}|$. The design of experiments consists in testing different configurations for div_e, p_m and c_{pat}. A summary of parameters for the design of experiment is presented in Table I.

TABLE I: Search parameters and constraints used for design of experiments.

Data parameters	**Value**
Number of traces	$N = 1800$ and $P = 200$
Diversity of events	$div_e \in [10, 50, 100]$
Maximum length of generated traces	$p_m \in [10, 25, 50]$
Event pattern coefficient	$c_{pat} \in [0.90, 0.75]$
Time transition patterns	G_0: $U(0, 400)$ G_1: $N(200, 25)$
Graph parameters	**Value**
Maximum number of nodes	$U_N = 0.2 \times div_e \times p_{max}$
Maximum number of edges	$U_E = 2 \times U_N$
Maximum number of positions	$p_{max} = \lvert\sigma\rvert_{max}$
Tabu search parameters	**Value**
Neighborhood size	15
Size of Tabu list	10
Max. number of iterations	500
Max. number of iterations without improvement	15

B. Evaluation metrics

ROC (Receiver Operating Characteristic) curve is the true positive rate (tpr) in function of false positive rate (fpr). This curve is obtained by varying the threshold for prediction (θ for process model classifier). The AUC (Area Under the Curve) is chosen as the performance measure, justified by the presence of imbalanced classes.

C. Benchmark of binary classifiers

The process model-based classifier is compared with three state-of-the-art machine learning algorithms for binary classification: Decision Tree (DT), Random Forest (RF) and feed-forward Multi-layer Perceptron (MLP). These methods expect matrix-shaped input data, so a "flattening" prepro- cessing is applied to the event log: features are created by combining every possible event's labels with corresponding time-stamps encountered in the event log. For each trace having a given event at a given time stamp, the corresponding

feature value is set to 1, 0 otherwise. The advantage of this preprocessing approach is to provide the 3 machine learning models with the most accurate data possible. The inconvenience is the high dimension and sparsity of input data. Because imbalanced classes are an issue for binary classification algorithms, data balancing has been applied before fitting DT, RF and MLP. An oversampling of the minority class was applied using the SMOTE algorithm [23]. Moreover, a high-dimension grid of hyperparameters was defined, and a random search on it was performed. A 3-fold cross-validation was used on the training set to determine the best hyperparameter combination for each data set. The previously described design of experiment is summarized in Figure 2. Calculations were done in `python 3.6`, using `scikit-learn` library for DT, RF and MLP methods.

D. Results

1) Quantitative results: For each descriptor combination, median and standard deviation of AUC on test sets for 10 replications are presented in Table II. The best average AUC score is highlighted in bold.

Our method with objective function (DiffOpt) globally outperforms DT, RF and MLP in most settings. The gap between proposed methods and state-of-the-art algorithms increases with the increase of div_e for $c_{pat} = 0.90$. When patterns in event logs of the positive class are less specific ($c_{pat} = 0.75$), the general performances decreases and variability increases. (DiffOpt) seems robust regarding the increase in diversity of events (div_e) and the increase in traces' size (p_m). Other methods are negatively impacted by the increase in diversity and trace size which results in reduced AUC performances.

TABLE II: Benchmark of AUC for 5 methods: average and standard deviation.

Data			DT		RF		MLP		(RepOpt)		(DiffOpt)	
c_{pat}	div_e	p_m	AVG	STD	AVG	STD	AVG	STD	AVG	STD	AVG	STD
0.90	10	10	0.96	0.01	0.96	0.01	0.99	0.01	0.99	0.01	**1.00**	0.00
		25	0.95	0.01	0.95	0.01	**1.00**	0.00	0.99	0.02	0.99	0.02
		50	0.96	0.01	0.96	0.01	**1.00**	0.00	0.99	0.02	0.98	0.01
	50	10	0.95	0.02	0.95	0.02	0.97	0.02	0.98	0.01	**1.00**	0.00
		25	0.95	0.02	0.95	0.02	0.97	0.01	0.97	0.01	**0.99**	0.00
		50	0.95	0.03	0.95	0.03	0.98	0.02	0.97	0.01	**0.99**	0.00
	100	10	0.95	0.01	0.95	0.01	0.96	0.01	0.98	0.01	**0.99**	0.00
		25	0.92	0.05	0.92	0.05	0.97	0.01	0.98	0.02	**0.99**	0.00
		50	0.90	0.07	0.90	0.07	0.97	0.02	0.97	0.01	**0.99**	0.00
0.75	10	10	0.88	0.05	0.88	0.05	0.94	0.03	0.95	0.05	**0.97**	0.02
		25	0.89	0.04	0.90	0.04	0.95	0.04	0.95	0.06	**0.96**	0.02
		50	0.85	0.06	0.86	0.06	0.93	0.04	**0.94**	0.04	0.91	0.06
	50	10	0.88	0.03	0.88	0.03	0.86	0.06	0.90	0.03	**0.95**	0.02
		25	0.87	0.04	0.85	0.06	0.87	0.05	0.91	0.04	**0.94**	0.01
		50	0.88	0.03	0.86	0.06	0.85	0.08	0.87	0.03	**0.94**	0.02
	100	10	0.77	0.10	0.78	0.06	0.86	0.03	0.87	0.05	**0.93**	0.02
		25	0.65	0.06	0.64	0.07	0.80	0.11	0.81	0.04	**0.92**	0.02
		50	0.64	0.07	0.63	0.06	0.84	0.05	0.72	0.05	**0.86**	0.07

2) Qualitative and explainable results: The interpretability is a crucial motivation in this study. Two examples of process models obtained after training (using (DiffOpt) objective function) are displayed in Figures 3 and 5. Process models are read from left to right, following increasing node positions. Circles represent nodes of the model, and flux from circles represent edges. Node size and edge size are proportional to the number of traces from L^{train} replayed during the replayability game. Each obtained process model graphically highlights distinctive patterns, mined during the training optimization. Thus, for simulated event log with high pattern coefficient ($c_{pat} = 0.9$) and narrow dimensions ($div_e = 10$ and $p_{max} = 10$), the resulting process model is simple (Figure 3). However, its power to distinct traces is strong, as highlighted by AUC performances (AUC $= 1.00 \pm 0.00$).

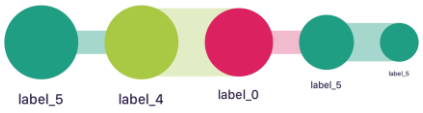


Fig. 3: Example of process model obtained using (DiffOpt), with $c_{pat} = 0.9$, $div_e = 10$ and $p_{max} = 10$.

To illustrate the prediction method, an example is presented in the following.

Example 1: An event log containing 2 traces σ_A and σ_B is presented in Figure 4. We want to predict if these traces are of class 0 or 1, according to the process model of Figure 3. After training on L_1^{train} and L_0^{train}, the process model TG-PsM maximizes (DiffOpt). Thus, the mean replayability of traces of L_1^{train} (0.98) is much higher than the mean replayability of traces of L_0^{train} (0.32). The threshold minimizing the gini impurity on the two training replayability distributions is $\theta = 0.40$. By computing the replayabilities of both traces, it appears that σ_A is well replayed (0.80), while σ_B replayability is pretty low (0.25). After a comparison to the threshold θ, class 1 and class 0 are attributed to σ_A and σ_B, respectively.

id	time-stamp	event
A	0	label_5
A	12	label_4
A	25	label_0
A	28	label_5
A	31	label_8
B	0	label_8
B	15	label_9
B	42	label_0
B	51	label_4

$$\bar{\mathcal{R}}(TG\text{-}PsM, L_0^{train}) = 0.32$$
$$\bar{\mathcal{R}}(TG\text{-}PsM, L_1^{train}) = 0.98$$

Predictions ($\theta = 0.40$)

$$\mathcal{R}(TG\text{-}PsM, \sigma_A) = 0.80 > \theta \rightarrow 1$$
$$\mathcal{R}(TG\text{-}PsM, \sigma_B) = 0.25 < \theta \rightarrow 0$$

Fig. 4: Event log of σ_A and σ_B (left) and predictions (right).

A more complex pattern extraction is presented in Figure 5, with $c_{pat} = 0.75$, $div_e = 10$ and $p_{max} = 50$. The process model is characterized by two central events ("*label_6*" and "*label_9*"), surrounded by other spread out and less specific ones. As the process model definition carry time characteristics on edges, potential distinctive time patterns are also extracted. Time-transitions for the class 1 followed $\mathcal{N}(200, 25)$ (and $\mathcal{N}(0, 400)$ for class 0). Thus, examples of time interval retained by the model (for example $[88, 264]$ and $[96, 289]$ in Figures 5), validate the ability of the method to display hidden time patterns.

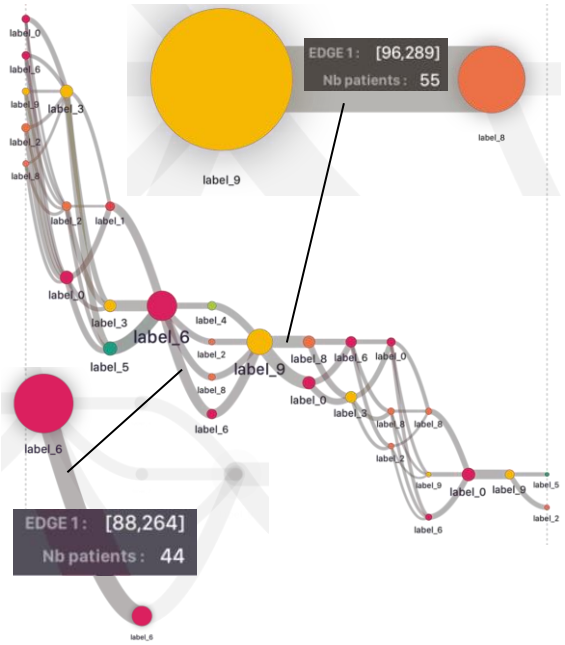


Fig. 5: Example of process model obtained using (DiffOpt), with $c_{pat} = 0.75$, $div_e = 10$ and $p_{max} = 50$.

VI. Conclusion

In this article, a new method for binary classification on timed event logs is proposed. Numerical experiments on synthetic data are presented, the robustness of the method being tested on event logs of increasing complexity. Quantitative results demonstrate the ability of the (DiffOpt) method to give outstanding performances in terms of AUC. Comparisons with state-of-the-art machine learning methods show the competitiveness of the proposed binary classifier when directly applied on imbalanced event logs with no use of over/under-sampling on training data. As process models carry distinctive patterns discovered during the training process, displaying them graphically illustrate future predictions.

Limitations and opportunities for future work are the following. Multi-class classification is not directly treated in this paper, but one can switch from binary to multi-class classification through "one-versus-all" settings. The current model cannot be updated with new traces batch. It must be entirely rebuilt. However, starting a new optimization process with already trained model as the initial solution could be a good strategy. The simulated event logs used here were designed to mimic patterns which will be interesting to found in clinical pathways extracted from claim databases. As the presented methodology is promising on synthetic data, future work will focus on practical healthcare case studies.

References

[1] H. De Oliveira, V. Augusto, B. Jouaneton, L. Lamarsalle, M. Prodel, and X. Xie, "Optimal process mining of timed event logs," *Information Sciences*, vol. 528, pp. 58 – 78, 2020.

[2] A. R. C. Maita, L. C. Martins, C. R. L. Paz, L. Rafferty, P. C. K. Hung, S. M. Peres, and M. Fantinato, "A systematic mapping study of process mining," *Enterprise Information Systems*, vol. 12, no. 5, pp. 505–549, 2018.

[3] W. M. P. van der Aalst, "Introduction," in *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, pp. 1–25, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[4] T. G. Erdogan and T. Ayca, "Systematic mapping of process mining studies in healthcare," *IEEE Access*, vol. 6, pp. 1–1, 04 2018.

[5] M. Prodel, V. Augusto, X. Xie, B. Jouaneto, and L. Lamarsalle, "Discovery of patient pathways from a national hospital database using process mining and integer linear programming," in *CASE*, pp. 1409–1414, 2015.

[6] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie, "Optimal process mining for large and complex event logs," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1309–1325, 2018.

[7] W. M. P. van der Aalst, *Operational Support*, pp. 301–321. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016.

[8] S. Ferilli and S. Angelastro, "Activity prediction in process mining using the WoMan framework," *Journal of Intelligent Information Systems*, vol. 53, pp. 93–112, 2019.

[9] M. Prodel, V. Augusto, X. Xie, B. Jouaneton, and L. Lamarsalle, "Stochastic simulation of clinical pathways from raw health databases," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, pp. 580–585, Aug 2017.

[10] Z. Z. Xing, J. Pei, and J. K. Eamonn, "A brief survey on sequence classification," *SIGKDD Explorations*, vol. 12, pp. 40–48, 11 2010.

[11] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of Molecular Biology*, vol. 48, no. 3, pp. 443 – 453, 1970.

[12] T. Smith and M. Waterman, "Identification of common molecular subsequences," *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195 – 197, 1981.

[13] B. Chowdhury and G. Garai, "A review on multiple sequence alignment from the perspective of genetic algorithm," *Genomics*, vol. 109, no. 5, pp. 419 – 431, 2017.

[14] Y. Byung-Jun, "Hidden markov models and their applications in biological sequence analysis," *Current Genomics*, vol. 10, no. 6, pp. 402–415, 2009.

[15] S. Blasiak and H. Rangwala, "A hidden markov model variant for sequence classification.," *IJCAI International Joint Conference on Artificial Intelligence*, pp. 1192–1197, 01 2011.

[16] H. De Oliveira, M. Prodel, and V. Augusto, "Binary classification on french hospital data: Benchmark of 7 machine learning algorithms," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1743–1748, 10 2018.

[17] M. Nilashi, O. Ibrahim, H. Ahmadi, and L. Shahmoradi, "An analytical method for diseases prediction using machine learning techniques," *Computers & Chemical Engineering*, vol. 106, 06 2017.

[18] A. Salcedo-Bernal, M. Villamil-Giraldo, and A. Moreno-Barbosa, "Clinical data analysis: An opportunity to compare machine learning methods," *Procedia Computer Science*, vol. 100, pp. 731 – 738, 2016. International Conference on Health and Social Care Information Systems and Technologies 2016.

[19] N. Herazo-Padilla, V. Augusto, B. Dalmas, X. Xie, and B. Bongue, "Screening a portfolio of pathologies by subject profiling and medical test rationing," in *2019 15th IEEE Conference on Automation Science and Engineering (CASE)*, pp. 424–430, Aug 2019.

[20] O. Ben-Assuli, R. Padman, M. Leshno, and I. Shabtai, "Analyzing hospital readmissions using creatinine results for patients with many visits," *Procedia Computer Science*, vol. 98, pp. 357 – 361, 2016. The 7th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2016).

[21] D. Hooijenga, R. Phan, V. Augusto, X. Xie, and A. Redjaline, "Discriminant analysis and feature selection for emergency department readmission prediction," in *IEEE Symposium Series on Computational Intelligence, SSCI 2018, Bangalore, India, November 18-21, 2018*, pp. 836–842, 2018.

[22] M. Vandromme, J. Jacques, J. Taillard, A. Hansske, L. Jourdan, and C. Dhaenens, "Extraction and optimization of classification rules for temporal sequences: Application to hospital data," *Knowledge-Based Systems*, vol. 122, pp. 148 – 158, 2017.

[23] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, p. 321–357, Jun 2002.

4.4 Conclusion

The present chapter introduces a methodology to perform explainable classification of timed event logs. This optimization-based approach is constructed using the formalism previously defined in Chapter 2. The results show promising classification performances on unbalanced synthetic data.

One limitation of the method is its ability to use only the occurrence and not the absence of a characteristic pattern for prediction. If the absence of a given pattern of the pathway is related to the occurrence of a given complication, the process model cannot identify it. Future deployment of this method should focus on this limitation, for example by using two optimization processes, one for each class, and looking at both distributions and both process models in order to predict and explain, respectively.

Also, future work will focus on the deployment of this method in the context of a real-data case study, constituting an opportunity to also deploy the automatic preprocessing methodology introduced in Chapter 3. As initial use case, the method has been tested on a case study related to sepsis. Sepsis relapse was predicted by considering patient pathways before the first sepsis hospitalization. The replayability score was added as a patient feature together with other patient centered characteristics such age gender, age or identified comorbidities. Using a decision tree algorithm to predict relapse, results show that adding the replayability score improved the model performances, with the replayability being the first feature used to split the population when constructing the decision tree. These results were presented as a poster at the *2019 Conference of the European Working Group on Operations Research Applied to Healthcare Services (ORAHS)*. This poster is presented in Appendix D. Finally, as described in Section 2.4, frequent medical events were not correctly embedded in the previously defined model, which should also be addressed. In the next chapter, another predictive method is presented, which considers the previously introduced challenges of patient pathways in order to perform explainable predictions.

Chapter 5

Explaining Predictive Factors in Patient Pathways using Non-Clinical Claims Data

Contents of the chapter

5.1 Motivation

In Chapter 4, a predictive methodology to perform explainable classification of timed event logs was presented. Based on the formalism introduced in Chapter 2, the methodology uses process models to perform prediction by comparing replayability scores and explains the results by showing the resulting process model after optimization. The complexity of medical events, mainly represented by multiple activities, was addressed in Chapter 3 by introducing a preprocessing step for event logs. However, some challenges still remain. One of them is the modeling of very frequent events in pathways. When focusing on hospitalization events, the latter are medically meaningful and in general not really frequent. As a result, process mining suits well to model such pathways. But when considering the entire patient pathway, some events occur very frequently. The modeling of such frequent events for process mining was already addressed in the case study of Chapter 2, but in a rather descriptive way by adding distributions of frequent elements on edges. Even if these frequent events have a lesser impact on the state of health of a patient, the information from these events may be useful to consider while constructing predictive models. Examples of such events in non-clinical claims data are general practitioner visits, specialist visits,

or biological tests. Addressing the challenges of frequent events, this chapter presents a method to model, predict and explain from patient pathways data. This methodology:
(1) starts from raw event logs; (2) models time and codes' hierarchy; (3) predicts from individual patient pathways; and (4) globally explains the results. This chapter focuses on identifying predictive factors which can be both frequent and infrequent.

5.2 Summary

In this last chapter, an end-to-end method to perform binary classification while identifying predictive factors using non-clinical claims database is presented. The first contribution is an adapted data processing method. By transforming an event log into a collection of images, the information in time, ordering of events, activities and their hierarchical structure is conserved. This allows for the extraction of relevant patterns by a classification method. The second contribution is the proposition of VPAE (Variational and Predictive AutoEncoder). Based on a VAE architecture, VPAE is trained to reconstruct positive elements from the data, while reconstructing a zero matrix for the negative elements. The classification is done for a given individual by encoding and decoding its representation and computing a score based on the decoding (the higher the score, the higher the probability of the patient to be of the positive class). A comprehensive image representing the extracted patterns is obtained by averaging the encoded-decoded representation of the positive population, providing a global explanation of the learning. To validate the performances of the method, experiments were conducted on synthetic event log data were a more or less notable pattern was hidden. Results validate the competitiveness of VPAE compared to state-of-the-art classifiers when the hidden pattern is notable. A case study on health data is also presented. Using data extracted from the SNIIRAM, the short-term mortality risk after the implementation of an Implantable Cardioverter-Defibrillator was predicted. The method provides accurate and explainable predictions, as predictive factors are highlighted. These factors were identified as infrequent events related to hospitalizations, but also as frequent events occurring throughout the medical history of patients.

5.3 Explaining Predictive Factors in Patient Pathways using Non-Clinical Claims Data

H. De Oliveira, M. Prodel, C. Leboucher, L. Lamarsalle, V. Augusto and X. Xie, "Explaining Predictive Factors in Patient Pathways using Non-Clinical Claims Data", *(to be submitted)*.

Explaining Predictive Factors in Patient Pathways using Non-Clinical Claims Data

Hugo De Oliveira, Martin Prodel, Claire Leboucher, Ludovic Lamarsalle, Vincent Augusto and Xiaolan Xie

***Abstract*—Event logs extracted from non-clinical claims databases are challenging, mainly because of the need to properly model time and medical codes with hierarchical structures. This paper presents an end-to-end methodology to predict and identify predictive factors in patient pathways using such event logs. The first contribution is an adapted data processing method able to properly model time and medical codes in order to make the extraction of relevant patterns possible. The second contribution is the introduction of `VPAE` (Variational and Predictive Auto-Encoder), a method which relies on a VAE architecture to perform binary classification using processed event logs. The method is trained so as to produce a comprehensive image representing the extracted patterns, in order to provide a global explanation of the learning. A design of experiments validates the competitiveness of `VPAE` compared to state-of-the-art classifiers when a notable pattern is hidden on synthetic event log data. Finally, a case study is presented, in which the short-term mortality after the implementation of an Implantable Cardioverter-Defibrillator is predicted.**

***Index Terms*—EHR, non-clinical claims data, patient pathways, explainability, variational auto-encoder**

I. Introduction

A. Context

Electronic Health Record (EHR) systems have been firstly created to improve care delivery. Their number strongly increased in the past few years, and so did the secondary use of these databases. Among EHR databases, non-clinical claims databases are promising but challenging. The French national health insurance database (SNIIRAM) is one of these. It contains healthcare reimbursements of almost all French citizens since 2006. The amount of data is colossal: in 2015, 66 million inhabitants were part of it [1]. The main interest of this database is its exhaustiveness as all patients' hospitalizations, medical visits and drug prescriptions are recorded. Despite the inherent complexity of the data (an extensive number of tables, centered on reimbursement and with complex relations) and even if precise medical information is not present (such as test results, imaging reports, or vital signs) the available structured databases are resourceful.

In this paper, we focus on the problem of predicting and identifying predictive factors regarding a particular future outcome. Examples of such binary classification tasks in healthcare include, but is not limited to, the prediction of relapse, the occurrence of a surgery, and the mortality within a period of time. This problem has been largely tackled in the literature, where the application of deep learning methods skyrocketed [2].

However, some challenges remains, particularly regarding *trust*: (i) trust in the administrations which use personal health data, but also (ii) trust in the algorithms when it comes to predictions. For the first aspect, the use of data with a very low risk of patient identification is valuable. A particularity of the present study is to focus only on pathways data, without any other patient-centered information such age, gender, ethnicity or localization. For the second aspect of trust, regarding the algorithm, the production of interpretive predictions is a key challenge for actual and future work [2]–[4]. Naturally, quantitative performances are a necessary condition for the validation of deep learning-based predictive tools. But the explanation of the predictions is essential (i) to simplify the practical deployment of such a novelty at a national level, (ii) to help the comprehension of hidden patterns discovered by a model, and (iii) to enable knowledge discovery regarding patient pathways. Such promising discoveries could be the causalities between a medical event and a selected outcome, the early detection of drug side effects, or the highlight of compliance failures.

Another challenge is the data describing patient pathways and medical events in such databases. Regarding patient pathway information which can be extracted from non-clinical claims databases, medical events are often provided using **standard medical codes**. These codes give inputs regarding events such as hospitalizations or medical visits, with precise information related to diagnostics, medical procedures, devices or drugs. Mainly taken from widely used classification systems, such as the International Classification of Disease (ICD) for diagnoses or the Anatomical Therapeutic Chemical (ATC) classification system for drugs, these codes are organized in a hierarchical structure, with different levels of aggregation. In practice, the selection of the right aggregation level depends on the pathology studied and often necessitate experts' input. The modeling of **time** in predictive models is also crucial. If a given medical event may influence future outcomes, the time between these events (for example long- or short-term before inclusion) but also the repetition during the history (for example a single event or the multiple repetition of such event in time) may influence the prediction. As a result, in this paper we focus in particular on codes and time when modeling medical event logs.

H. De Oliveira, V. Augusto and X. Xie are with Mines Saint-Étienne, Univ Clermont Auvergne, CNRS, UMR 6158 LIMOS, Centre CIS, F - 42023 Saint-Étienne France.

H. De Oliveira, C. Leboucher, M. Prodel and L. Lamarsalle are with HEVA, 186 avenue Thiers, F-69465, Lyon, France.

X. Xie is also with the Antai College of Economics and Management, Shanghai Jiao Tong University, China.

B. Scientific contributions

Regarding these previous considerations, the present paper introduces an end-to-end methodology to predict and identify predictive factors in patient pathways using non-clinical claims data. A particular focus is given on:

1) dealing with the hierarchical structure of codes;
2) modeling time to make the extraction of relevant patterns possible;
3) providing a graphical explanation of what has been learned by the model.

Thus, the first contribution of this paper is a methodology to model and transform such complex event logs. This transformation keeps the pathway information in terms of events, transitions, time but also the hierarchy information carried in medical codes.

The second contribution is the introduction of `VPAE` (Variational and Predictive Auto-Encoder). `VPAE` is method which relies on a VAE architecture to perform binary classification from processed event logs. The modification in the loss function when training the model allows the production of a global explanation of the learning process, through a comprehensive image representing the extracted patterns.

The rest of this paper is organized as follows. Related works are presented in Section II. Notations on event logs and the adapted modeling methodology are introduced in Section III After having formally defined the problem, `VPAE` is introduced in Section IV, along with computational experiments to test it on various simulated event logs. A case study using the SNIIRAM database is presented in Section V. Finally, discussion and conclusion are presented in Section VI and VII, respectively.

II. Related Work

EHR data are valuable resource to understand the natural history of disease, quantify the effect of an intervention, construct evidence-based guidelines or detect adverse events [5]. Even if the performances of deep recurrent models have been tempered by Min et al. [6] in the context of readmission risk prediction after a hospitalization for COPD, recent studies mainly rely on deep learning methods. Widely applied on EHR data, the main idea of deep learning is to switch from expert-defined to data-driven feature creation [4]. `Doctor AI` [7] has been presented in 2016 to perform differential diagnosis from EHR data. Miotto et al. [3] presented `Deep Patient` in 2016, an unsupervised method to encode patient representation from EHR. To predict the probability of disease appearance, a Random Forest algorithm was trained over encoded patients, giving better performances than the original representation or other dimension reduction methods. A global study focused on scalablility was performed by Rajkomar et al. [8], were various targets and models were used. Representation learning for medical concept is also a current research topic. Medical concept embedding, such as `Med2Vec` [9], `GRAM` [10], and more recently `Cui2vec` [11] are notable examples. In order to perform patient clustering, Landi et al. used a convolutional auto-encoder to learn a latent representation of patients [12].

Studies deploying predictive tasks using extractions of the SNIIRAM database are not numerous. In 2018, Janssoone et al. compared multiple models to predict medication non-adherence using this database [13]. Recently, Kabeshova et al. presented `ZiMM ED` [14], a predictive model for the long-term prediction of adverse events. To the best of our knowledge, no other paper presented prediction methods on this database.

The interpretation of predictions becomes an interesting research topic in recent years. In 2017, Suresh et al. [15] compared performances of Long-short term memory (LSTM) and convolutional neural networks (CNN) for the prediction of clinical intervention using the MIMIC-III database. A specific focus was done on interpretability, by using feature-level occlusion for LSTM and filter/activation visualization for CNN to explain predictions. In 2018, `RETAIN` was presented by Choi et al. [16]. This two-level neural model is designed for an interpretation purpose, while keeping comparable performances. Interpretations are provided for a given patient, by giving importance of each element of its history. Not focused on deep learning methods, some model-agnostic explainable frameworks have been introduced to explain black-box models. `LIME` [17] and `SHAP` [18] are examples of such models. The first one uses linear models to approximate local behaviors. The second one uses Shapley values for both global and local interpretability. A limitation of such frameworks is the need to run multiple evaluation of the model to provide interpretations, which in practice can be time consuming. Moreover, a recent comment by Rudin in 2019 [19] arbitrates for the use of intrinsically interpretative models for high stakes instead of trying to explain black box models.

Finally, the explanation of temporal patterns remains an interesting research track. The representation introduced by Wang et al. in 2013 is one example [20]. The two-dimensional representation proposed has been successfully used to mine signatures from patient pathways. Another support for temporal visualization of event logs is process mining. As an example, Prodel et al. [21] proposed an algorithm for raw event logs processing. Applied on a case study using patient pathways, a particular focus on integrating the hierarchy of codes during the optimization process was presented. In order to properly model time, an improvement of the previously mentionned work has been recently proposed [22].

As a result, most recent studies used complex embedding and deep architectures to process health data. These methods have been accurate and successful in predictive tasks. However, the challenge of explaining predictive results while using complex event logs remains. Event if the consideration of time has been treated in the literature, such representation in the context of explainability is still a lead. Moreover, a focus on modeling widely used medical codes in this context seems an actual challenge. Taken all these consideration into account, and in addition to explain the predictions for a given patient, providing a general visualization of patterns influencing a given outcome could be valuable for the field.

III. Complex Medical Event Logs Modeling

In the following, Definitions 1-3 describe the notations related to event log data, the input data type used through

this paper.

Definition 1 (Event): Each event denoted e is a couple (a, t) where:

- a is a nonempty set called activity set, each element $a_i \in a$ being an activity;
- $t \in T$ with $T = \mathbb{N}$ or $\mathbb{R}$ corresponds to the event time also called time-stamp.

Definition 2 (Trace): A trace is a sequence of events denoted $\sigma = e_1, \ldots, e_m$ with $m \in \mathbb{N}^*$ such that $\forall k \in [1, m-1], t_k < t_{k+1}$.

Definition 3 (Event log): An event log is a set of traces denoted as $L = \{\sigma_1, \ldots, \sigma_n\}$ with $n \in \mathbb{N}^*$. The size of the event log $|L|$ is defined as the number of traces in L.

Definitions 4 and 5 present a matrix representation of the data which is adapted to machine learning methods beside raw event logs.

Definition 4 (Trace matrix): Let σ be a trace, the trace matrix $x = (x_{i,j})$ of σ is a 2-dimensional array with:

- $dim(x) = l \times w$ where l is the label dimension and w is the time window dimension;
- $\forall i \in [1, l], \forall j \in [1, w], x_{i,j} \in \{0, 1\}$.

The conversion function $mat(.)$ converts a trace σ into a trace matrix x.

The trace matrix is similar to the *Temporal Event Matrix Representation (TEMR)* introduced by Wang et al. [20], with the notion of time windows to modify the time scale for long-time follow-up studies. It gives a representation of a trace where each row is an activity label and each column is a time window. Thus, for a trace σ and its trace matrix x, if $x_{i,j} = 1$ then the activity i occurred at least once during the time window j. Each row vector corresponds to a 1-of-k coding of a given activity, ordered and regrouped in time windows (columns). The function $mat(.)$ converts a trace from an event log to a sparse matrix representation. Different choices in conversion will influence the parameters l and w, as detailed in the following.

Definition 5 (Event log array): By extension of Definition 4, the event log array X of an event log $L = \{\sigma_k\}$ is defined as $X = mat(L) = (x^k_{i,j})$, where $\forall k \in [1, |L|], x^k = mat(\sigma_k)$ is the trace matrix of trace σ_k. Moreover, $dim(X) = |L| \times l \times w$.

The two key elements of the transformation defined by the $mat(.)$ function are the selection of activity labels, and the the granularity of the time window. As the presented work focuses on medical activities which are often represented using medical codes from universal coding systems, a medical activity is inherently completed by its hierarchical knowledge. To incorporate the hierarchy information in the modeling, these levels can be integrated as a given row of x. Thus, adding more than one level of the hierarchy to describe an activity will increase the generalization in the coding, but will also increase the dimension l. Example 1 depicts a complete example of the data-related concepts.

Example 1:

Let σ be the trace illustrated in Figure 1. The possible activities are $\{$A0, A1, B0, B1$\}$, where $\{$A0, A1$\}$ and $\{$B0, B1$\}$ inherit from chapter A and B, respectively (like ICD-10 chapters for example). The matrix $x = mat(\sigma)$ is also displayed in Figure 1. Black (resp. white) squares correspond to a value of 1 (resp. 0). $dim(x) = l \times w$, with $l = 6$ and $w = 9$ (time window is daily). A broader time window (e.g. weekly) would have decreased the precision of x by regrouping activities in only 2 columns. Moreover, not using the hierarchical structure of codes would lead to $l = 4$ by deleting higher chapter rows (A and B).

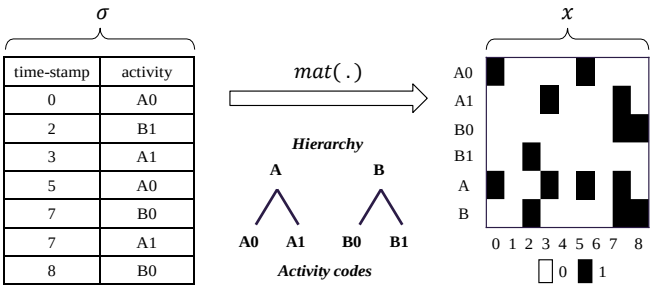


Fig. 1: A complete data example: from a raw trace to its matrix representation.

IV. VPAE: Variational and Predictive Auto-Encoder

A. Problem definition

The problem addressed in this paper is the binary classification of traces. Let $L = (L_0, L_1)$ be a set of two event logs, where traces from L_c are of class c with $c \in \{0, 1\}$. The class 1 is referred to as the *positive class* in the following. We intend to create a binary classifier with good predictive performances over new traces. A major attention is given to the explainability of the predictive model. For that, we highlight the differentiating patterns among the two classes, as found during the learning process. The problem consists in finding two functions:

- a predictive function $\lambda : \sigma \mapsto s$, which attributes a score $\lambda(\sigma)$ of being of class 1 to each trace σ;
- an explanation function η, which provides a key global explanation about the differentiating patterns learned during the construction of λ.

The following section proposes a methodology to construct both λ and η.

To build the predictive function λ and the explanation function η, we introduce VPAE (Variational and Predictive Auto-Encoder). VPAE is a Variational Auto-Encoder to which we incorporate prior knowledge from known classes for training event log.

B. Variational Auto-Encoder

A widely used method in representation learning is *autoencoding* [23]. Autoencoders generally consists of 2 parts: an encoder and a decoder. In this paper, the encoder $f : x \mapsto z$ vectorizes a trace matrix x in a latent space of a smaller dimension. The decoder $g : z \mapsto x'$ reconstructs x as x' from this vector. Autoencoders are designed to only learn useful properties of a data set, thanks to a constrained reconstruction process. The restriction comes from the reduced latent space

dimension, which forces the network to prioritize information during the training process. A Variational Auto-Encoder (VAE) is an autoencoder where the learnt variables are parameters of a distribution. Introduced by Kingma et. al. in 2014, the method shows high performances regarding data generations [24]. The encoder f is an *inference network* $q(x|z)$ and the decoder g a *generative network* $p(z|x)$. The training process consists in maximizing the expected lower bound $ELBO$ defined as $ELBO = \mathbb{E}_{q(z|x)}\left[\log \frac{p(x,z)}{q(z|x)}\right]$. In practice, the single sample estimate $\log p(x|z) + \log p(z) - \log q(z|x)$ with z sampled from the inference network is optimized. A previous work on autoencoders shows that the choice of such method applied on non-clinical claims data was relevant comparing to other autoencoders [25].

C. Class-dependent lower bound

The autoencoder architecture used in this paper is a VAE with a modification when computing the lower bound. Let $C(.)$ be a function which returns, for a trace matrix x of class k, its class $C(x) = k$. Then, the class-dependent lower bound $CD\text{-}ELBO$ is defined as:

$$CD\text{-}ELBO = L\left(x^*, g(f(x))\right) + \log p(z) - \log q(z|x) \quad (1)$$

where $L(.,.)$ is the sigmoid cross-entropy function, and:

$$x^* = \begin{cases} 0_x, & \text{if } C(x) = 0 \\ x, & \text{if } C(x) = 1 \end{cases}$$

where 0_x is a zero matrix of the same dimensions.

Thus, the term $\log p(x|z)$, which represents the reconstruction error in practice, is replaced by $L(x^*, g(f(x)))$ here. For a given training trace matrix x, the reconstruction target depends on its class $C(x)$. If x is of the positive class, the loss function remains the same as in a regular VAE implementation. Otherwise, x is not compared with its encoded and decoded version $x' = g(f(x))$, but only with a zero matrix. Training the VAE in such a way ensures that it only considers patterns related to the positive class for the reconstruction. In contrast, other patterns, which can only be used for the reconstruction of negative class elements, are deleted. This idea is the core of the proposed prediction methodology, which is detailed in the following.

D. Predictive and explanation functions

1) Predictive function: The decoding performance of a new trace determines its predicted class: if the decoding is accurate, then the input trace is considered to contain patterns of the positive class. If not, no patterns are kept during the decoding, and the trace is likely not to have any pattern of the positive class. This architecture enables to deduce a predictive function λ and an explanation function η.

Once the VAE is trained, the class prediction of a new trace σ requires to encode and decode its trace matrix x. If most of the trace matrix is well decoded, with a significant number of elements $x'_{i,j}$ being activated, the trace has a strong probability of belonging to the positive class. Otherwise, as for elements of class 0 during the training, the trace matrix is poorly decoded, and a majority of zeros constitutes the output. The predictive function λ can be defined as follows:

$$\lambda : \sigma \xrightarrow{mat(.)} x \xrightarrow{f(.)} z \xrightarrow{g(.)} x' \xrightarrow{sum(.)} s \quad (2)$$

with $s = sum(x') = \sum_{i,j} x'_{i,j}$ being the sum of all elements of x'. The encoding of x results in the parameters of Gaussian distributions. z is a vector of the mean values. Then, z is decoded in x', and s is finally computed. The higher s is, the more likely for a trace to be of the positive class. This score function is used to compute binary classification metrics such as the area under the receiver operating characteristic curve (AUC).

2) Explanation function: Using training data, the `VPAE` architecture allows for the deduction of global explanation patterns from the predictive model. From traces of L_1^{train} an event log array X_1^{train} is created, which is encoded and decoded. From the decoded event log array $X_1'^{train}$ an average trace matrix $\bar{x}_1'^{train}$ is computed. The visualization of this average trace matrix highlights and explains the characteristic patterns of the positive class. The explanation function η is then defined as follows:

$$\eta : L_1^{train} \xrightarrow{mat(.)} X_1^{train} \xrightarrow{g(f(.))} X_1'^{train} \xrightarrow{mean(.)} \bar{x}_1'^{train} \quad (3)$$

with the mean function $mean(.)$ returning the average element-wise trace matrix of an event log array.

E. Tests on synthetic event logs

The proposed method is tested on synthetic event logs. It requires two logs, one for each class of patients, $(L_0, L_1) = (L_0^{train}, L_0^{test}, L_1^{train}, L_1^{test})$. The construction of these two event logs is done such that different hidden patterns are created depending on their class. Parameters to generate event logs are (div_e, p_m, c_{pat}), which are the number of different events, the maximal trace length and the coefficient of patterns, respectively. The higher c_{pat} is, the more specific and homogeneous traces of L_1 are. The objective is to learn from training event logs and accurately predict for test ones. During the entire experiment, the latent dimension d_{latent} is empirically set to 50. The training of the network is done by maximizing the $CD\text{-}ELBO$ defined in Equation 1. A validation set (20% of the training set) is isolated at the beginning of the optimization. The $ADAM$ [26] algorithm with batch normalization (each batch being of size 128) is used to minimize the opposite of the $CD\text{-}ELBO$. The maximal number of epochs is set to 5000. An early stopping criterion is applied, with a waiting patience of 50 iterations without improvement of the validation loss function. The proposed method is compared to three state-of-the-art binary classification algorithms: Decision Tree (DT), Random Forest (RF), and Feed-forward Neural Network (NN). For DT and RF, trace matrices are flattened before being used for training and testing. Hyperparameters are optimized using 50 iterations of the Efficient Global Optimization (EGO) algorithm [27], maximizing the 5-fold cross validation AUC. The coding is in `python 3.7` with the `scikit-learn 0.21` library for

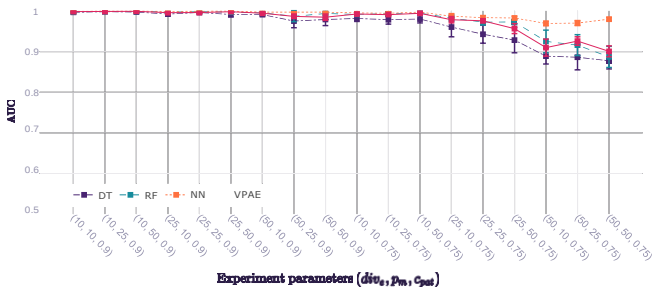


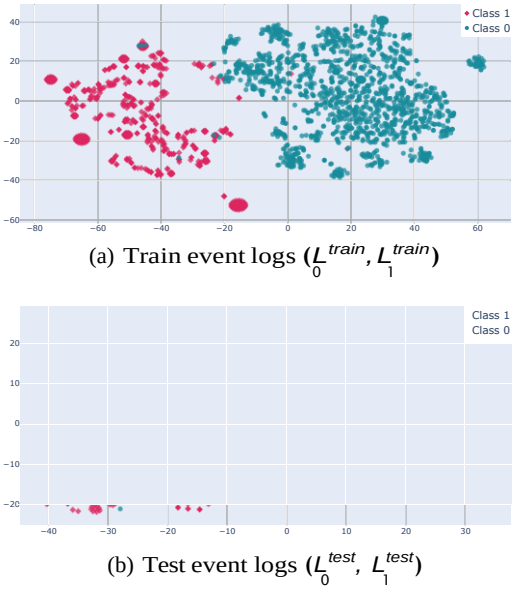
Fig. 2: AUC score on the test data of 4 methods (mean and 95% confidence interval), for 18 (div_e, p_m, c_{pat}) combinations.

DT and RF. The EGO algorithm is coded using `bayes-opt`. For NN, the same architecture and training parameters as the encoder part of the `VPAE` are used. The only modification is the last layer, being a 1-dimensional output with a sigmoid activation function for the binary prediction. All deep learning frameworks are implemented using `tensorflow 1.14`.

1) Quantitative results: The classification performance is measured with the AUC on both test logs (L_0^{test}, L_1^{test}). For each combination of the experiment parameters, and for each method, the mean AUC score and 95% confidence interval over 10 generated event logs are computed. Results are presented in Figure 2. Our `VPAE` method shows interesting performances in accurately classifying event log data with strong characteristic pattern for the positive class ($c_{pat} = 0.9$). `VPAE` globally outperforms DT and reaches competitive performances compared to RF, which in practice is as a black box model. By comparing `VPAE` to NN, performances are competitive for the clear pattern ($c_{pat} = 0.9$). That is not longer the case for $c_{pat} = 0.75$ and $div_e = 50$, where the outperforming of NN overall compared methods is clear. NN seems to be an upper bound regarding `VPAE` in terms of performance, as it is a black-box model of similar dimension without an explainable constraint (decoding) in training. The constraint on representation learning in `VPAE`, which permits the global explanation via the decoding, is affecting predictive performances in the case of complex hidden patterns. This experiment on synthetic data highlights a trade-off in complex cases between prediction performances and explanation.

To summarize, the proposed method reaches interesting classification performances when a notable pattern is hidden in synthetic event log data. Notable patterns are the ones which can be explicitly presented to the user, humanely understood, and carrying a sufficient predictive power.

2) Qualitative results: The explanation process is exemplified using one of the experiments ($c_{pat} = 0.9$, $div_e = 10$, $p_m = 10$). After training on (L_0^{train}, L_1^{train}), the `VPAE` is able to properly reconstruct a trace of the positive class, while dismissing other traces (close to zero matrix). As a result, traces are encoded differently depending on their structure. This can be visualized by looking at training traces distribution in latent space for both classes. For that purpose, t-SNE [28] is used to visualize the distribution of traces in latent space. Figure 3 presents a 2-dimensional t-SNE projection of



(a) Train event logs (L_0^{train}, L_1^{train})

(b) Test event logs (L_0^{test}, L_1^{test})

Fig. 3: 2-dimensional t-SNE projection of L_0 and L_1 in latent space.

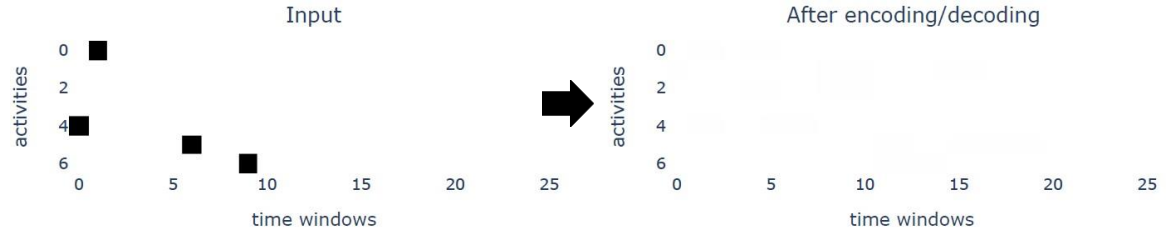
training and testing traces of both classes. The representation of training traces (3a) shows that a separation appears in latent space: the training is successful. By representing testing traces (3b), the same separation is highlighted, which confirms a generalization in pattern learning.

The difference between a trace x and its encoded/decoded version x' is interesting as presented in Figure 4 for test traces. For unobserved data, the generalization allows an accurate reconstruction of positive class elements (4b), whereas no particular information is reconstructed for elements of class 0 (4a). These remarks are at the core of the explainability process. After encoding and decoding all the positive training traces, the average trace matrix is computed. A visualization of main patterns retained by the `VPAE` is done, as shown in Figure 5: hidden patterns appear in terms of activities (rows) and their temporal appearance (columns).

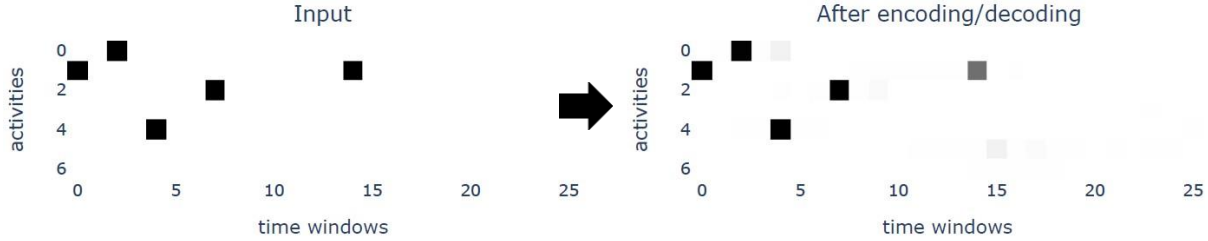
V. Case Study

A. Overview and context

Among all deaths due to cardiovascular diseases, no less than 60% are caused by sudden cardiac death (SCD) [29]. About 3/4 of SCDs are related to ventricular tachycardia. The treatment consists in a cardiopulmonary resuscitation, combined with an electric impulse provided by an automated external defibrillator. For high-risk patients, Implantable Cardioverter-Defibrillators (ICDs) are used to prevent cardiac arrest. Once implanted, the ICD sends electric impulses to stimulate the heart in response to a potentially lethal ventricular arrhythmia. Three types of ICD exist, depending on the number of leads connecting the generator to the heart (single-lead: single chamber; two-lead: dual chamber; and three-lead: biventricular). An ICD replacement is usually necessary after several years. Possible replacement causes are a complication,



(a) Example of a trace of class 0 (left): absence of patterns after decoding (right).



(b) Example of a trace of class 1 (left): strong similarities after decoding (right).

Fig. 4: Examples of test traces before and after being encoded/decoded.

Fig. 5: Pattern explanation: visualization of $\bar{x}_1^{'train}$.

a malfunction or the naturally limited durability of the device. The replacement is qualified as *short-term* if it occurs 6 to 8 years after the implantation, depending on the type of ICD.

The problem addressed here is the identification of patients with a risk of post-implantation mortality within the short-term replacement period. Moreover, the goal is to identify predictive factors in medical event logs extracted from the SNIIRAM database[1], considering time and hierarchy structure in medical codes, and without patient-centered information. In this context, this case study serves as a proof of concept for automatic predictive factor discovery and at-risk patient identification.

B. The Data

The study included all adult patients who had an ICD implantation between 2008/01/01 and 2011/02/28[2]. Among the 19,408 patients selected, 730 (3.8%) were excluded due to insufficient follow-up. Thus, 18,678 patients were included in the study (5,448, 5,216 and 8,014 patients having a single-, two- or three-lead ICD, respectively). The follow-up of patients was done until 2018/12/31 to identify potential deceases. According to medical experts' recommendations, a replacement was considered short-term if it occurs within the 8, 7, 6 years after implantation for a single-, two- or three-lead ICD, respectively. Among the population, 7,551 patients deceased during the short-term replacement period (40.4%). For each patient, 2 years of medical history prior to the ICD implantation were collected. Patient pathways during this period were made into an event log, which constituted the input data for the prediction. Patient-centered information, such as age, gender, living localization, were not used in order to focus on the analysis of patient pathways for prediction. Based on the medical history prior to the ICD implantation, the prediction was made at the implantation discharge.

After extracting and processing the data, an event log of all patient's medical history was created with 959,931 events and more than 4,876,336 activities. The extraction regrouped the different medical events occurring over the course of the two years. Regarding hospitalizations, the reason for admittance (main diagnosis), associated diagnosis (comorbidities) and performed health care services (medical procedures and devices) were included. Other care episodes regrouped activities as consultations, biological tests and other medical procedures not performed as inpatient care. Each activity was identified by a medical code, mostly organized hierarchically: (1) ICD-10: diagnoses and comorbidities (2 levels of hierarchy); (2) CCAM: medical procedures (3 levels of hierarchy); (3) medical devices (3 levels of hierarchy); (4) biological tests (1 level in the hierarchy); (5) consultations (the code related to the type of consultation). The population was split in train (80%) and test (20%) sets, respecting the ratio of the two prediction classes. The activities and the related hierarchy levels were filtered to discard infrequent elements in the train event log (threshold: 500 occurrences). The filtering evacuated non-representative codes, while keeping widely represented

[1]CEREES number: TPS 347167, CNIL authorization number: DR-2019-122.

[2]Medical procedure codes (French CCAM): DELA004, DELF013, DELF016, DELF900, DELF014, DELF020, DELA007.

TABLE I: AUC evaluated on train and test data for the 4 compared methods.

	DT	RF	NN	VPAE
TRAIN	0.74	0.71	0.83	0.88
TEST	0.64	0.69	0.71	0.70

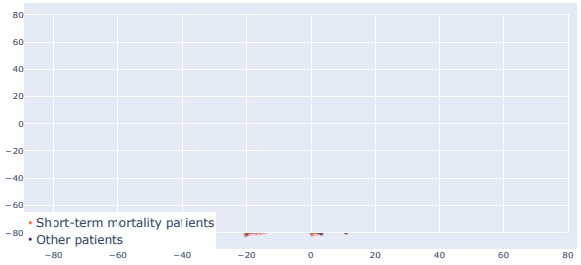
ones and higher levels in the hierarchy. It resulted in $I = 928$ different labels. Next, a transformation of all traces into trace matrices was performed (see Example 1 of Section III). A time window of 15 days was used, leading to a time window dimension $w = 49$.

C. Results

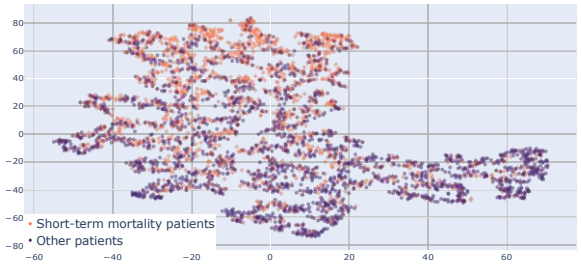
To compare the prediction performance, the same setting as the design of experiments (Section IV-E) was used. The method was compared to DT, RF and NN. The only change was the latent dimension for VPAE which was increased to 100, due to the dimensions of input data. This change also impacted NN by increasing the size of its layers. Performances were evaluated in predicting mortality in the short-term replacement period. Parameters of DT and RF were finely tuned using global optimization and routine deep learning procedures. Early stopping, dropout and regularization were used to tune NN and VPAE. Results regarding AUC are presented in Table I. For the train set, VPAE reached 0.88 and outperformed DT (0.74), RF (0.71) and NN (0.83). For test sets, the difference between NN and VPAE was small. Both methods outperformed DT, and results were similar with RF. For all methods, a gap between train and test performances was observed, highlighting a general overfitting. Consequently, we assume that, due to the variability of patient pathways, all methods would require more training data reduce overfitting. Figure 6 shows a 2-dimensional t-SNE projection of en- coded patients with and without short-term mortality. A horizontal separation between both classes is visible for the training data (6a). This separation also appears for the test data (6b), even if less explicitly. However, as suggested by the slight shading of Figure 6b and the performances on test data, some patterns which are useful for prediction are learned.

Figure 7 shows the encoding and decoding of test traces from both classes, selected among the best predicted results. For the sake of readability, only the first 400 activities are displayed. The badly reconstructed trace of class 0 (7a) demonstrates the accurate prediction of the negative class. The faithful reconstruction of a positive class is also observed (7b), highlighting the desired behavior. Only a partial vision of the matrix is reconstructed here, sufficient to predict and highlight discriminant features.

Regarding explainability, Figure 8 presents a subset of the average trace matrix. Here, two patterns emerge from the average trace matrix: (1) continuous horizontal lines over the 2 years; and (2) particular events occurring in the last 15 days before implantation. This implies that both some long-term and recurrent medical events, but also punctual and last-week events, have an influence on the prediction target. To verify these assumptions, relative risks (RRs) are computed among the entire population (train and test). Firstly, a top-100



(a) Train data.

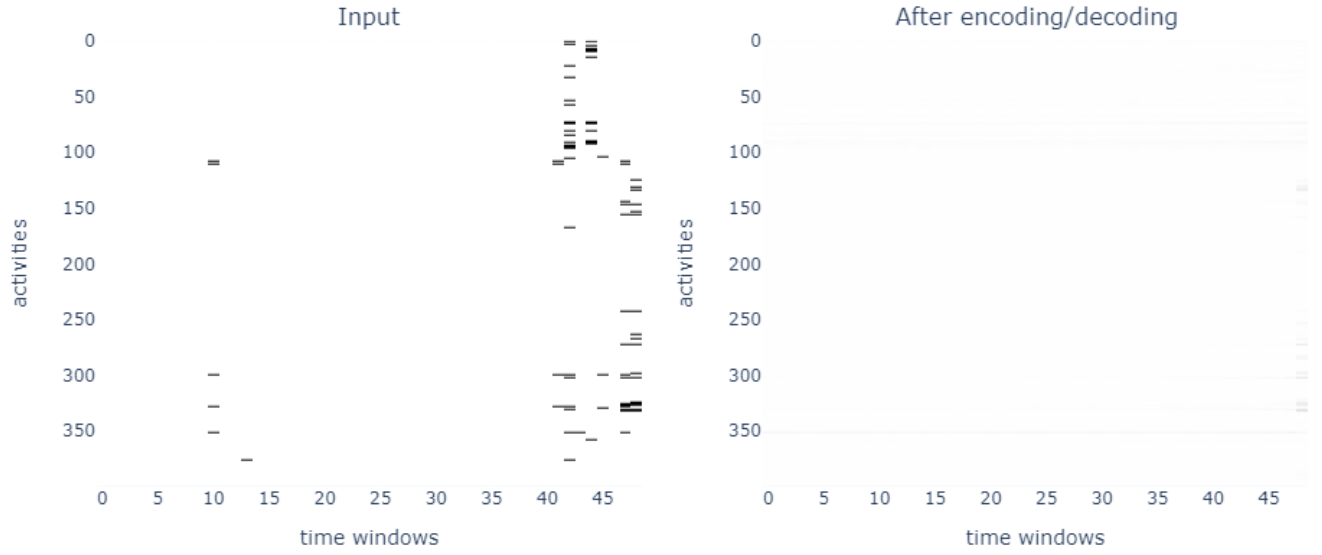


(b) Test data.

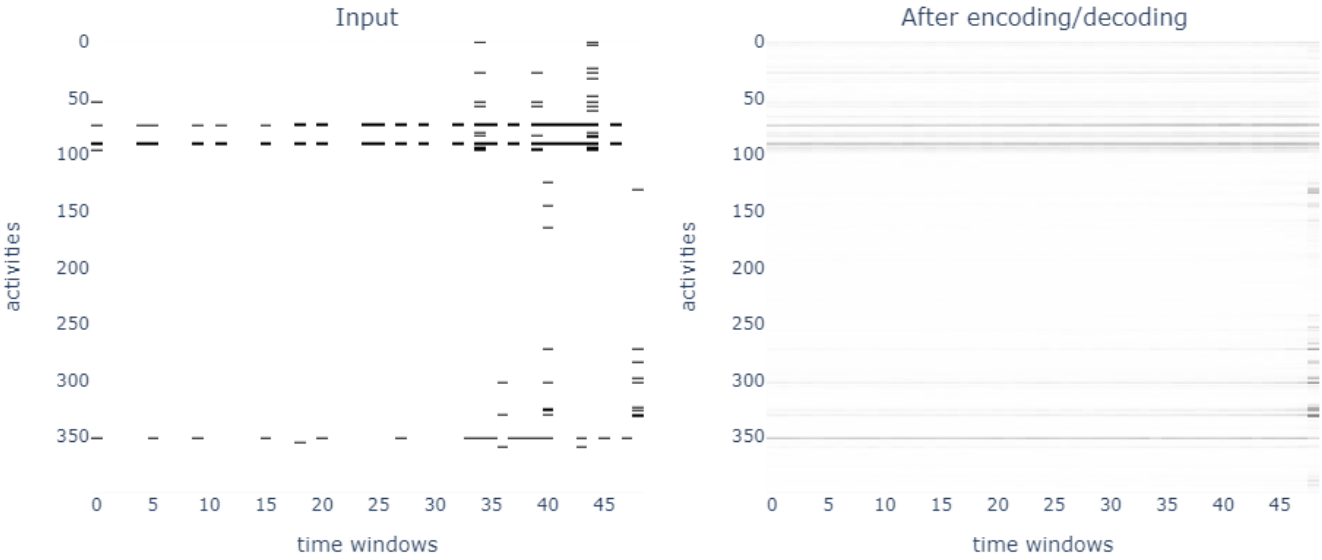
Fig. 6: 2-dimensional t-SNE projection of both patient classes in latent space.

of the most decoded activities is computed, regarding patient pathways strictly before the last time window. Then, for each of the most decoded activities, patients are split in two groups: those with the reoccurring activity (activated in more than 24 time windows, i.e. more than 50% of the time) versus others. Finally, RRs are computed regarding these groups. The same process is applied for the last time window (15 days) before implantation. Therefore, activities are filtered in Figure 8. This filtering allows for visualizing most decoded activities (top-100) for recurrent or last-week events while improving readability. Among them, activities with a significant RR are highlighted. For these impacting activities, RRs are displayed in Figure 9, with a 95% confidence interval. Regarding recurrent events during the past 2 years (9a), results shows that recurrent biological tests, frequent general practitioner visits and repeated respiratory disease device prescriptions have a significant impact on the prediction target (RR $>$ 1). For blood tests, the impact is even higher as the RRs rise to almost 2. Regarding the last time window (9b), it is found that comorbidities related to the genitourinary system, atrial fibrillation and flutter, or disease of the respiratory system increase the risk.

As a result, the presented case study illustrates the ability of the method to identify predictive factors from event logs extracted from a non-clinical claims database. Starting from anonimyzed event logs, without patient-centered information, the method is able to model the data to keep hierarchical code structure and meaningful temporal information. Prediction performances are competitive with a similar black-box model, and some predictive factors are identified, with an interesting difference between infrequent short-term events and frequent



(a) Trace matrix of a class 0 patient.



(b) Trace matrix of a class 1 patient.

Fig. 7: Example of test patient's traces before and after being encoded/decoded.

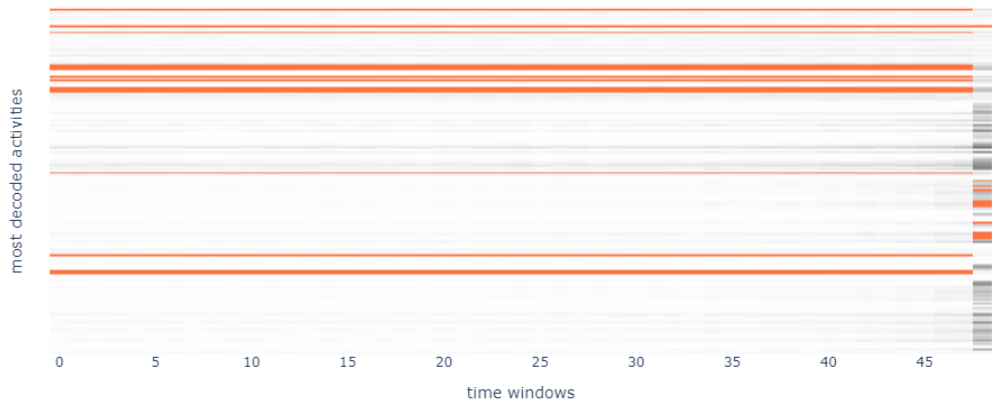
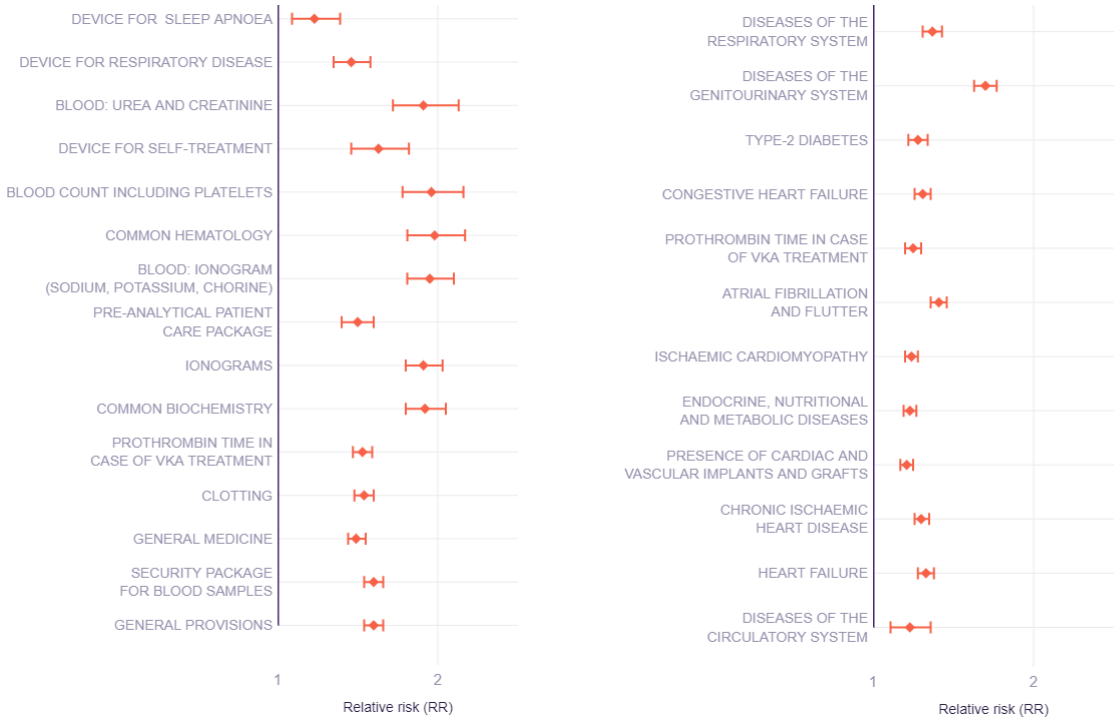


Fig. 8: Pattern explanation: most decoded activities after filtering (train set). Activities with a significant RR are highlighted in orange.



(a) Frequent events in medical history.

(b) Last time window (15 days).

Fig. 9: Relative risks regarding the prediction target.

long-term ones. A verification of the implication of such factors with the observed outcome was also presented, to validate the ability of the method to discover such factors.

VI. Discussion

The main strength of the presented method is the ability to predict and explain by extracting knowledge from raw medical event logs, dealing with the inherent complexity of such data and without a priori information. However, some methodological limitations should be discussed. Choices like the label dimension l and the time window dimension w may impact general performances. These parameters have been set experimentally, limited by computing hardware in the data security environment provided to conduct the study. As long as the dimensions of the autoencoder increases, more precision may improve performances, and such assumption need to be experimentally validated. The simplicity of the architectural structure of the encoder and the decoder also need to be noticed. The use of convolution and transpose convolution layers have been set aside, as the property of the images used to model patient pathways are not following the same property as real images (a given organization of pixels in one region of the image has not the same meaning if it appears in another region of the image). But the use of recurrent neural network architectures to encode the data seems to be promising. The limitation will be the decoder, which needs to sequentially reproduce the image under the constraints embedded in the loss function, and may be a challenge to rise.

VII. Conclusion and Future Work

In this paper, a method to model and transform complex medical event logs is presented, with a focus on time and hierarchy codes modeling. The `VPAE` method is introduced, performing explainable binary classification on such patient pathways modeling. Numerical experiments on synthetic data validate the competitiveness of the method compared to state-of-the-art classifiers if an identifiable pattern is present in the input data. A case study on real data extracted from the SNIIRAM database is detailed. The short-term mortality after the implementation of an ICD is predicted, reaching performances which are close to the values observed in the literature. Moreover, the ability of the method to provide explainable predictions is illustrated.

In terms of future applications, other medical case studies will be conducted. This will challenge the adaptability of the method but also prove the value of the methodology as part of the deployment of prevention policies. In fact, the explainability in this context will produce knowledge directly from evidence-based patient pathway analysis. This can be beneficial to generalize guidelines from identified predictive factors, in order to for perform early at-risk patient detection. At a national scale, such methodology could motivate the deployment of targeted prevention policies. Also in more exploratory studies, weak signals could be detected by following this approach. These could lead to the formulation of a hypothesis to test and loop on the data using more humanly comprehensive indicators, like descriptive statistics and relative risks.

REFERENCES

[1] P. Tuppin, J. Rudant, P. Constantinou, C. Gastaldi-Ménager, A. Rachas, L. de Roquefeuil, G. Maura, H. Caillol, A. Tajahmady, J. Coste, C. Gissot, A. Weill, and A. Fagot-Campagna, "Value of a national administrative database to guide public decisions: From the système national d'information interrégimes de l'Assurance Maladie (SNIIRAM) to the système national des données de santé (SNDS) in France," *Revue d'Épidémiologie et de Santé Publique*, vol. 65, pp. S149–S167, Oct. 2017.

[2] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi, "Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, pp. 1589–1604, Sept. 2018.

[3] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley, "Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records," *Scientific Reports*, vol. 6, p. 26094, May 2016.

[4] C. Xiao, E. Choi, and J. Sun, "Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review," *J Am Med Inform Assoc*, vol. 25, pp. 1419–1428, Oct. 2018.

[5] P. Yadav, M. Steinbach, V. Kumar, and G. Simon, "Mining Electronic Health Records (EHRs): A Survey," *ACM Comput. Surv.*, vol. 50, pp. 85:1–85:40, Jan. 2018.

[6] X. Min, B. Yu, and F. Wang, "Predictive Modeling of the Hospital Readmission Risk from Patients' Claims Data Using Machine Learning: A Case Study on COPD," *Scientific Reports*, vol. 9, p. 2362, Feb. 2019.

[7] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor AI: Predicting Clinical Events via Recurrent Neural Networks," in *Machine Learning for Healthcare Conference*, pp. 301–318, Dec. 2016.

[8] A. Rajkomar, E. Oren, K. Chen, A. M. Dai, N. Hajaj, M. Hardt, P. J. Liu, X. Liu, J. Marcus, M. Sun, P. Sundberg, H. Yee, K. Zhang, Y. Zhang, G. Flores, G. E. Duggan, J. Irvine, Q. Le, K. Litsch, A. Mossin, J. Tansuwan, D. Wang, J. Wexler, J. Wilson, D. Ludwig, S. L. Volchenboum, K. Chou, M. Pearson, S. Madabushi, N. H. Shah, A. J. Butte, M. D. Howell, C. Cui, G. S. Corrado, and J. Dean, "Scalable and accurate deep learning with electronic health records," *npj Digital Medicine*, vol. 1, pp. 1–10, May 2018.

[9] E. Choi, M. T. Bahadori, E. Searles, C. Coffey, M. Thompson, J. Bost, J. Tejedor-Sojo, and J. Sun, "Multi-layer Representation Learning for Medical Concepts," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (San Francisco California USA), pp. 1495–1504, ACM, Aug. 2016.

[10] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, "GRAM: Graph-based Attention Model for Healthcare Representation Learning," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (Halifax NS Canada), pp. 787–795, ACM, Aug. 2017.

[11] A. L. Beam, B. Kompa, A. Schmaltz, I. Fried, G. Weber, N. Palmer, X. Shi, T. Cai, and I. S. Kohane, "Clinical Concept Embeddings Learned from Massive Sources of Multimodal Medical Data," *Pac Symp Biocomput*, vol. 25, pp. 295–306, 2020.

[12] I. Landi, B. S. Glicksberg, H.-C. Lee, S. Cherng, G. Landi, M. Danieletto, J. T. Dudley, C. Furlanello, and R. Miotto, "Deep representation learning of electronic health records to unlock patient stratification at scale," *npj Digit. Med.*, vol. 3, p. 96, Dec. 2020.

[13] T. Janssoone, C. Bic, D. Kanoun, P. Hornus, and P. Rinder, "Machine Learning on Electronic Health Records: Models and Features Usages to predict Medication Non-Adherence," *arXiv:1811.12234 [cs, stat]*, Nov. 2018.

[14] A. Kabesho va, Y. Yu, B. Lukacs, E. Bacry, and S. Gaïffas, "ZiMM: a deep learning model for long term and blurry relapses with non-clinical claims data," *arXiv:1911.05346 [cs, stat]*, Mar. 2020.

[15] H. Suresh, N. Hunt, A. Johnson, L. A. Celi, P. Szolovits, and M. Ghassemi, "Clinical Intervention Prediction and Understanding with Deep Neural Networks," in *Machine Learning for Healthcare Conference*, pp. 322–337, Nov. 2017.

[16] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, "RETAIN: An Interpretable Predictive Model for Healthcare using Reverse Time Attention Mechanism," in *Advances in Neural Information Processing Systems 29* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds.), pp. 3504–3512, Curran Associates, Inc., 2016.

[17] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, (San Francisco, California, USA), pp. 1135–1144, Association for Computing Machinery, Aug. 2016.

[18] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems 30* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), pp. 4765–4774, Curran Associates, Inc., 2017.

[19] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, pp. 206–215, May 2019.

[20] F. Wang, N. Lee, J. Hu, J. Sun, S. Ebadollahi, and A. F. Laine, "A Framework for Mining Signatures from Event Sequences and Its Applications in Healthcare Data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 272–285, Feb. 2013.

[21] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie, "Optimal Process Mining for Large and Complex Event Logs," *IEEE Transactions on Automation Science and Engineering*, vol. 15, pp. 1309–1325, July 2018.

[22] H. De Oliveira, V. Augusto, B. Jouaneton, L. Lamarsalle, M. Prodel, and X. Xie, "Optimal process mining of timed event logs," *Information Sciences*, vol. 528, pp. 58–78, Aug. 2020.

[23] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, Massachusetts: MIT Press, Nov. 2016.

[24] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *arXiv:1312.6114 [cs, stat]*, May 2014.

[25] H. De Oliveira, V. Augusto, B. Jouaneton, L. Lamarsalle, M. Prodel, and X. Xie, "Automatic and explainable labeling of medical event logs with autoencoding." Submitted to the IEEE Journal of Biomedical and Health Informatics, 2020.

[26] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv:1412.6980 [cs]*, Jan. 2017.

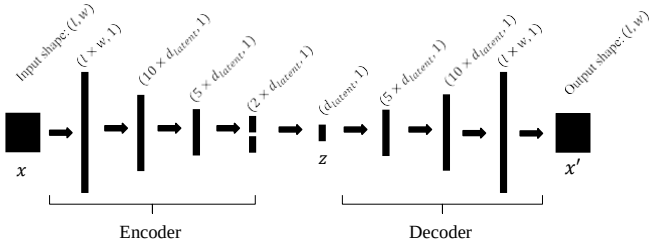
[27] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient Global Optimization of Expensive Black-Box Functions," *Journal of Global Optimization*, vol. 13, pp. 455–492, Dec. 1998.

[28] L. V. D. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2625, 2008.

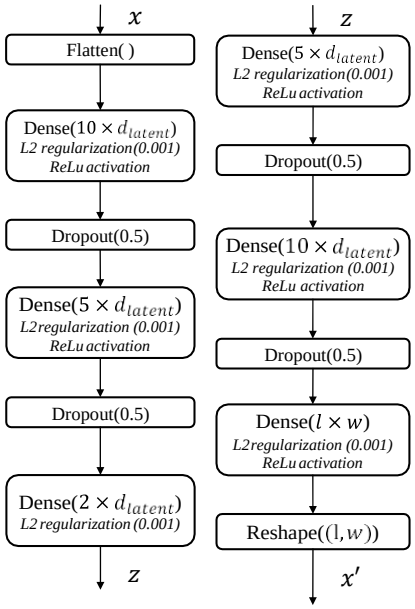
[29] A. S. Adabag, R. V. Luepker, V. L. Roger, and B. J. Gersh, "Sudden cardiac death: epidemiology and risk factors," *Nature Reviews Cardiology*, vol. 7, pp. 216–225, Apr. 2010.

APPENDIX

An overview of the VPAE architecture is detailed in Figure 10. The encoder (Figure 10b) consists of a 3-layer neural network. After flattening the trace matrix x, the dimension is gradually reduced to twice the latent dimension $\times 2\ d_{latent}$. One set is used as the mean values for a learned distribution, the other one for standard deviations. Based on the learned distribution, a vector z is generated in latent space. This element serves as input for the decoder (Figure 10c), to compute x' in the initial space of dimension $l \times w$.



(a) VPAE architecture description with a focus on successive shapes.



(b) Encoder details. (c) Decoder details.

Fig. 10: VPAE architecture overview.

The proposed VPAE method is tested on synthetic event logs $(L_0, L_1) = (L_0^{train}, L_0^{test}, L_1^{train}, L_1^{test})$. The construction of these event logs is done such that different hidden patterns are created depending on their class. Details about this construction are presented in the following.

A. Data generation

Two graphs G_0 and G_1 are constructed, one related to each class. The graphs consist of nodes arranged in layers having a maximum number of identical positions equal to p_m. For each position $p \in [1, p_m]$, the corresponding layer is composed of n = div_a different nodes. Each of these nodes carries an activity label, the event diversity div_e being the total number of different activities in the final event log. Then, a proportion of shared patterns is removed by deleting $c_{pat}N*|\ |$ randomly chosen nodes from G_1 and corresponding edges. An illustration for G_0 and G_1 is shown in the left part of Figure 11.

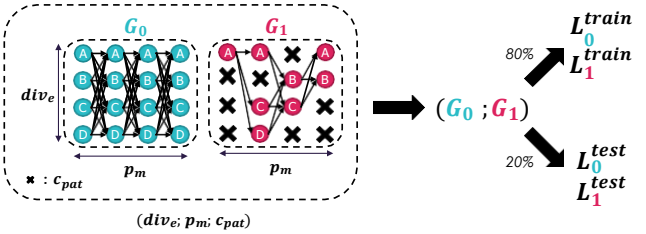


Fig. 11: Schematic representation of the design of experiments.

A trace σ is created by crossing the graph from left to right. Starting from the lowest position of the graph (left), a random node is selected, being the actual node. The label a of this node is added to the trace as the activity of its first event $e_1 = (a_1, 0)$. Then, a new node of the next position is reached, by using possible edges starting for the actual node and going to the next layer. The new event $e_2 = (a_2, t)$ is added to the trace, the time-stamp t of this new event being computed by adding a random value. This value is randomly selected for L_0 and L_1 following $\mathcal{U}(0, 100)$ and $\mathcal{N}(\mu, 10)$, respectively, where $\mu \in [0, 100]$ is selected once for each edge of G_1. By crossing the graph layer by layer, the trace is created. At each step, the process of adding events to the trace can be stopped with a probability $p = \frac{P(n_{current})}{p_m}$, where $n_{current}$ is the current node of G, corresponding to the last addition to σ. The probability of stopping the construction process ensures variability in trace lengths. Such an event log construction process ensures the presence of a pattern in G_1, in terms of labels, transitions and time. On the contrary, elements generated from G_0 follow no particular pattern. The higher c_{pat} is, the smaller G_1 and the more specific the process model will be. The design of experiments consists in testing different configurations for c_{pat}, div_e and p_m. For each combination, 10 different couples (G_0, G_1) are produced, leading to 10 event logs $L = (L_0, L_1)$. The number of traces simulated is 1000 for the positive class and 2000 for the class 0. The unbalancing of the data illustrates what is found in healthcare case studies when a particular complication is predicted, which generally concerns a sub-group of the population. From these traces, 80% are used for training $(L_0^{train}, L_1^{train})$, while 20% are isolated as a test set (L_0^{test}, L_1^{test}). The transformation of traces into trace matrices using the previously defined $mat(.)$ function is done with l being the total number of labels identified in the event log L_1^{train} (div_e being an upper bound). The time window dimension w is set in order to have time windows of size 15. A summary of parameters for the design of experiments is presented in Table II.

B. Hyperparameters tuning

In order to set the hyperparameters of DT and RF, a global optimization algorithm is used. Efficient Global Optimization

Parameters	Values
Number of traces	class 0: 2000 and class 1: 1000
Event pattern coef.	$c_{pat} \in [0.90, 0.75]$
Diversity of events	$div_e \in [10, 25, 50]$
Max length of traces	$p_m \in [10, 25, 50]$
Time transition patterns	$G_0 : U(0, 100)$ and $G_1 : N(\mu, 10)$ with $\mu \in [0, 100]$

TABLE II: Parameters of the event logs constructed for experiments.

(EGO) [27] is used to search for optimal parameters. Here, 5 random iterations are applied, followed by 45 steps of optimization. Table III presents the hyperparameter grids used to search for such optimal parameters. The number patient is $n = |L_0^{train}| + |L_1^{train}|$, and the number of features obtained after flattening is $c = l \times w$. The same setting is used to optimize hyperparameters of DT, RF and NN in the case study presented in Section V.

TABLE III: Hyperparameters description for Decision Tree and Random Forest.

model	parameter	values
DT	max_depth	$[2, n]$
	max_features	$[1, c]$
	min_samples_leaf	$[2, n]$
	class_weight	[None, balanced]
RF	class_weight	[10, 100]
	max_depth	$[2, n]$
	max_features	$[1, c]$
	min_samples_leaf	$[2, n]$
	class_weight	[None, balanced]

C. Results

Exp.			DT		RF		NN		VPAE	
c_{pat}	p_m	div_e	AVG	CI	AVG	CI	AVG	CI	AVG	CI
0.90	10	10	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00
		25	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00
		50	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00
	25	10	0.99	0.00	1.00	0.00	1.00	0.00	1.00	0.00
		25	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00
		50	0.99	0.01	1.00	0.00	1.00	0.00	1.00	0.00
	50	10	0.99	0.00	1.00	0.00	1.00	0.00	0.99	0.00
		25	0.98	0.02	0.99	0.01	1.00	0.00	0.99	0.01
		50	0.98	0.01	0.99	0.00	1.00	0.00	0.98	0.01
0.75	10	10	0.98	0.01	0.99	0.00	1.00	0.00	0.99	0.00
		25	0.98	0.01	0.99	0.01	0.99	0.00	0.99	0.00
		50	0.98	0.01	0.99	0.00	1.00	0.00	0.99	0.00
	25	10	0.96	0.02	0.98	0.01	0.99	0.01	0.98	0.01
		25	0.94	0.02	0.97	0.01	0.98	0.00	0.98	0.00
		50	0.93	0.03	0.97	0.01	0.98	0.00	0.96	0.01
	50	10	0.89	0.02	0.93	0.03	0.97	0.01	0.91	0.02
		25	0.89	0.03	0.92	0.03	0.97	0.01	0.93	0.01
		50	0.88	0.02	0.89	0.03	0.98	0.00	0.90	0.01

TABLE IV: AUC score on the test data of 4 methods (mean and 95% confidence interval), for 18 (c_{pat}, div_e, p_m) combinations.

5.4 Conclusion

This last chapter introduced an end-to-end methodology to perform binary classification using patient pathway information. A representation of patient pathways which model time, infrequent and frequent medical events, with the hierarchy of coding systems is also described. A method which relies on autoencoding to perform binary classification while producing a global explanation of predictive factors is also introduced. The short-term mortality risk after the implementation of an Implantable Cardioverter-Defibrillator was predicted using data extracted from the SNIIRAM. Predictive factors were highlighted, the VPAE method being competitive with other methods tested. Both frequent and infrequent factors, at various levels of the hierarchy of codes, were identified.

However, the proposed method can be improved by considering some limitations. For instance, the performance gap between neural network and VPAE is a point of attention. When patterns are complex and difficult to identify, the proposed methodology has difficulties to provide competitive performances. To leverage this challenge, the use of different architectures for the encoder and the decoder should be considered in future research to improve data processing while keeping the explainability of the image representation. The ability of the method to use only the occurrence and not the absence of a characteristic pattern for prediction is also an issue for the presented method. The VPAE method identifies elements whose occurrence is related to the positive class. An architecture which consists in two auto encoders for both positive and negative class will also identify elements for which the absence is a predictive factor, potentially improving prediction performances. While this work is focused on patient pathways, the VPAE method also has the potential to be used to encode patient characteristics or other medical information (e.g. image, vital signs, free text, etc.) which could be further processed using deep learning architectures. The explainability of such results should also be considered in future research.

Conclusion

Summary

This work proposes multiple contributions for improving predictive modeling of patient pathways, with a focus on the particularities of non-clinical claims data, such as time, complex macro medical events and frequent events. To this end, contribution to the process mining field were introduced (Chapter 2 and 3). In addition, two predictive methods were proposed: one process mining-based (Chapter 4), and another deep learning-based method (Chapter 5). For both predictive models developed in this work, explainability was considered as a necessity. The reader can referred to the schematic representation of Figure 3 presented in the introduction for an overview of these contributions.

In more details, the literature review presented in Chapter 1 provides an overview of current health data applications. A discussion of the challenges regarding the use of health data was presented, highlighting the need for more trust between all stakeholders (particularly regarding the future of medical practices, data protections and equality). Regarding predictive modeling, a review of the subject was presented, with a focus on longitudinal health data, and detailing different predictive algorithms and various targets of these algorithms. A trend was observed that automated feature construction by means of deep learning has been used in the majority of recent studies on predictive modeling of health data. XAI was presented, with a focus on its application to health data. The field of process mining was presented, and the use of pathway modeling for application in healthcare was discussed. Finally, the preliminary results of predictive modeling with machine learning methods, applied to the PMSI data base, were introduced. These preliminary results have motivated the use of medical history data such as patient pathways to improve predictions.

For the purpose of improving pathway modeling, a new process mining framework was proposed in Chapter 2. The proposed methodology is a general contribution to process mining which includes descriptors, two new process models adapted to time modeling, along with a process discovery algorithm. Based on a tabu search, this optimization procedure identifies a process model which maximizes its fitness, the mean replayability score, for an event log. The performance of the process discovery algorithm was tested on synthetic data, validating the ability of the algorithm to mine representative processes compared to other heuristics. The process mining framework was used in a real case study on diabetes, working with manually labeled event logs extracted from non-clinical claims data of the French national health insurance. By using the proposed framework, patient pathways before four identified diabetes complications were visualized.

Since manually labeling event logs extracted from non-clinical claims data is a complex preprocessing task, an automatic labeling procedure based on clustering was proposed in Chapter 3. Deep autoencoding architectures are used to learn an adapted representation of medical events. By using *k*-mean clustering in the latent space, labels are automatically created. Moreover, the previously trained decoder serves to characterize these labels, providing medically meaningful explanation. Several deep autoencoding architectures, along with direct clustering on a sparse representation, were applied to synthetic data and compared. The performances of the autoencoders were evaluated in terms of their accuracy in finding hidden clusters, but also regarding their ability to provide meaningful explanations. Results showed that autoencoding methods generally outperformed direct clustering, with VAE outperforming all other autoencoding methods. The proposed method was deployed as a preprocessing task, prior to the discovery of processes from an extraction of the SNIIRAM database. In this study, pathways of patients having a laparotomy operation followed by an incisional hernia were analyzed, using both a manual and an automatic labeling procedure. The qualitative analysis of the obtained process models showed great similarities between the two labeling methods, making the automatic labeling contribution valuable for practical deployments. Moreover, an interactive dashboard was deployed in order to visualize the results. It was concluded that this method, coupled with an interactive tool such as the proposed dashboard, could facilitate the exchange with medical expert in future case studies.

The initial motivation for developing both the process discovery framework of Chapter 2 and the preprocessing method of Chapter 3 was to improve the modeling of patient pathways. Considering both the modeling of time and the complexity of medical events, Chapter 4 proposed a predictive modeling method based on process mining. The framework of Chapter 2 has been adapted in order to perform binary classification, directly from event logs. Using the same optimization procedure, a novel objective function was introduced to mine a process model which represents positive traces well while poorly representing negative ones. Classification is performed by computing the replayability of the new trace regarding the previously obtained process model, and by comparing the replayability score with an adapted classification threshold. The performances of this process mining-based approach were evaluated on synthetic event logs of variable complexities. The event logs were generated with an imbalanced configuration where the traces of the positive class were in a minority. Common algorithms such as decision tree, random forest or multilayer perceptron were also tested. Since these algorithms are not able to directly process traces, sparse features were created in a preprocessing step, representing events, transitions and time, along with an oversampling procedure. Results showed that the proposed methodology performed well in classifying new traces, particularly when using the newly introduced objective function (DiffOpt). Moreover, the process model which is used to classify contributes to the global explainability of the proposed methodology. Distinctive elements such as events, transitions and characteristic time are extracted and visualized on the process model.

Chapter 5 introduced a second predictive methodology adapted to non-clinical claims data, which is based on deep learning. This method was created to capture predictive frequent outpatient events such as consultations, laboratory tests or drug deliveries. A 2-dimensional representation of patient pathways is proposed, modeling events, time and hierarchy of codes. In addition, VPAE, a new predictive algorithm based on a modification of the VAE architecture, is introduced. The training of VPAE is performed by accurately reconstructing patient's representations of the positive class, while reconstructing a zero

matrix for patients of the negative class. Global explainability of the results is achieved by decoding and averaging elements of the positive class, resulting in an image which displays the characteristic patterns over the time. A design of experiments on simulated data was conducted, and performances of the proposed methodology were compared to decision tree, random forest and a deep feed-forward neural networks (NN) architecture. The results showed competitive performances of VPAE, outperforming decision tree, being competitive with random forest, but also with NN when the pattern clearly appears in the data. However, the performance gap between VPAE and NN increased with the complexity of the data, illustrating that the reconstruction constraint in the VPAE loss function is restrictive regarding classification. A case study was presented, where the short-term mortality after the implementation of a cardioverter defibrillator was predicted. Results showed a good performance of the proposed methodology for the prediction of short-term mortality. The predictive factors identified by the model were frequent medical events during the history (such as general practitioner visits or biological test), but also punctual hospitalizations shortly before implantation (related to the genitourinary or to the respiratory system). The computation of relative risks of these events regarding the studied outcome validates the significance of identified factors.

Future work

To summarize, the work presented in this thesis results in two predictive models: one based on process mining (Chapter 4), and another based on deep learning (Chapter 5). In the following, discussions and leads for future researches are presented.

The process mining-based predictive model is based on the process mining framework presented in Chapter 2. As a result, this method is adapted to pathways with punctual medical events such as hospitalization. The deployment of this method in a real case study, for example with data extracted from the PMSI, is part of future work. Thus, the preprocessing method proposed in Chapter 3 will be tested in a predictive context. But when automatically defining event labels for prediction, event logs of positive and negative class need to be considered differently. As a result, a modification of the loss function may be necessary in order to create labels adapted to the positive class. The use of the class-dependent lower bound defined in Chapter 5 could be implemented and tested. Another lead for research is the extension of the process mining-based method to handle frequent events. In the case study of Chapter 2, frequent events were described in edges only after the optimization. In order to incorporate this knowledge during the optimization, the re-playability game needs to consider frequent events as real events. But the strictly ascending condition imposed on grid process models for the sake of readability is problematic when considering frequent events: with frequent events, the length of traces will increase, leading to a lower replayability for similar size constraints. A new formalization of process models which allows loops for example, could facilitate the use of frequent events for predictions. The deep learning-based predictive model, presented in Chapter 5, is capable of han- dling frequent outpatient medical events (as found in the SNIIRAM database). Research on deploying other layers for the VPAE's architecture (particularly recurrent layers both for encoding and decoding) could improve performances. Another valuable perspective would be to improve the interactivity of VPAE's explainability results, particularly for future case study applications.

One discussion concerns the arbitration between the two predictive methods for practical deployment. The deep learning-based model is able to deal with both frequent and

infrequent events, and the explainability of results by means of an image is an added value. When working only with infrequent medical events, the representation of pathways using process mining is also advantageous. The interactive process model, along with the visual explanation, is valuable for discussion with medical experts.

A comparison of the two predictive models proposed in this work is also recommended as future work. Moreover, the use of other types of deep explainable architectures should be investigated (e.g. attention mechanisms) and a benchmark of deep explainable architectures should be performed in order to validate the performances of both proposed methods. In this context, the use of different layers for the VPAE architecture (e.g. particularly recurrent layers) should also be investigated and could lead to performance improvements. In terms of explainability, both predictive methods proposed in this work can generate a global explanation of the results. Patterns which are common to the entire positive class are captured and displayed. A clustering analysis amid the positive class observations could show potentially independent sub-patterns. As a result, partial subgroup explanation for both methods is an interesting track for future research (e.g. using multiple process models for the process mining-based method, or using latent space clustering for the deep learning-based method). Moreover, individual explainability (i.e. explaining distinctive factors at the scale of a patient) should be investigated for both methods as well.

The work presented here only focuses on patient pathway data. Individual patient information such as age, gender or geographical code was not used for prediction. Incorporating such information into both methods presented here could improve prediction performances. However, the risk of identifying a patient based on individual patient information is higher compared to using strictly event-based medical information contained in event logs. For the process mining-based predictive method, an attempt has been made to incorporate patients' features into the model. Based on HES data, a decision tree was fitted on patient features to predict sepsis relapse. Pathway information was used for prediction using the process mining-based approach. The replayability score was then considered as a feature: for each patient, his replayability score characterizes how similar his pathway is to that of patients with sepsis relapse. The use of this feature increased prediction performances and was the first used feature when constructing the decision tree (i.e. first split). These results were presented as a poster at the *2019 Operations Research Applied to Healthcare Services (ORAHS) conference*, available in Appendix D.

As stated in the introduction, non-clinical claims data bases such as PMSI or SNIIRAM only include reimbursement data. As a result, some particular health conditions may be quite complicated to identify. This is the case when characteristic medical procedures or drugs are related to multiple pathologies. Compared to non-clinical claims data, cohort data are oftentimes more extensive and more precise due to regular follow-ups. Matching between the SNIIRAM and cohort data is possible (e.g. probabilistic or direct using a unique anonymous identifier) to identify such patients in the SNIIRAM database. However, these patients only constitute a sample of the entire population. In this context, a wise modeling and representation learning of this information could be useful to identify other patients by similarity, thus improving the inclusion of patients in studies. Lastly, in a more general perspective, both of these contributions may serve for other applications involving timed event logs. The hierarchical codes found in non-clinical claims data may also be found in other fields, such as retail sales where products may be classified in hierarchical structures. In a more technical perspective, the recent advances of deep learning are powerful and may be a benefit to the field of process mining. In complex tasks such as process discovery where a representation of a process hidden in event logs is searched, deep

learning may prove to be efficient.

It is concluded that the most important perspective for non-clinical claims data is to deploy solutions such as the ones presented in this work in practice. This is the only approach to properly evaluate their benefits for the health system, in terms of *data-driven targeted prevention policies*. Due to the possibilities offered by explainability, patterns and new knowledge could be extracted from such evidence-based analysis. This could be beneficial to generalize guidelines based on identified predictive factors and to detect at-risk patients.

Appendix A

Details on the French national health databases

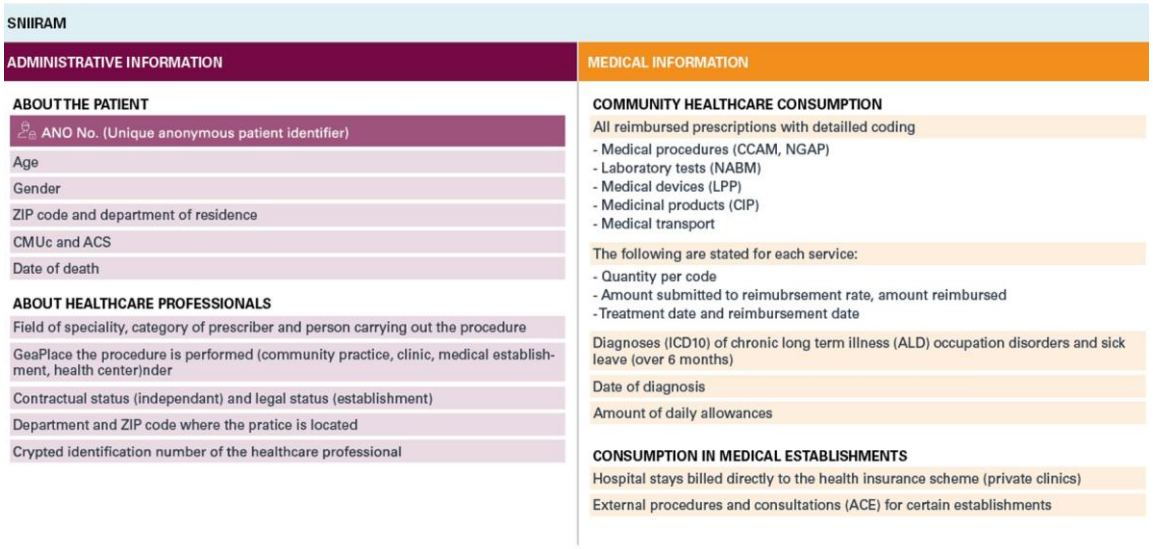


Figure A.1: Details on the SNIIRAM database.

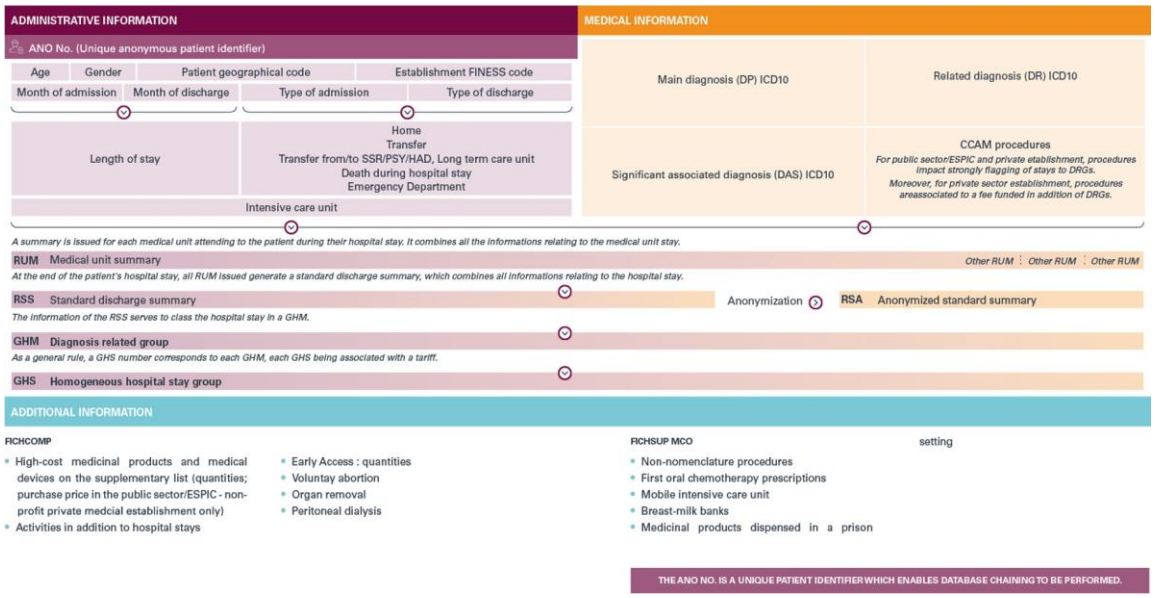


Figure A.2: Details on the PMSI database (for short-stay wards information).

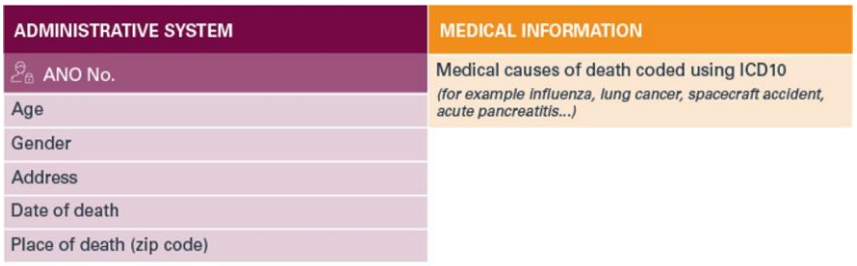


Figure A.3: Details on the CépiDC database.

Appendix B

Details on replayability parameters

The replayability $R(\text{PsM}, \sigma)$ measures the ability of a (time) grid process model PsM to characterize a given trace σ. As detailed in Chapter 2, two weighting parameters exist in theformulation:

- α, which influence the penalization of skipped elements during the replayability game;
- β, which influence the strongly (or time) forced transitions.

In order to evaluate the sensitivity of the replayability score obtained regarding these parameters, an experiment was conducted, detailed in the following. As for the numerical experiments presented in Chapter 2, 10 event logs containing 1, 000 traces were generated from 10 graphs with parameters $div_{e,p} = 500$, $div_e = 50$ and $p_{max} = 10$. For each event log L, process discovery was conducted on a grid of parameters $\alpha \in [0, 1]$ and $\beta \in [0, max(\{|\sigma|\}_{\sigma \in L}) - 1]$. The TSOE method was used for the optimization.

Figure B.1 shows the mean and standard deviation of the final replayability score obtained over the 10 replications, in function of parameters α and β. Parameters chosen for descriptive and predictive tasks in Chapters 2-4 are highlighted. As observed when looking at the mean values, the final replayability score can be strongly impacted by the parameter β. On the contrary, α seems to having fewer impact on the final score obtained. However, the variability observed over the replications shows that the standard deviation can me minimized by wisely selecting α and β.

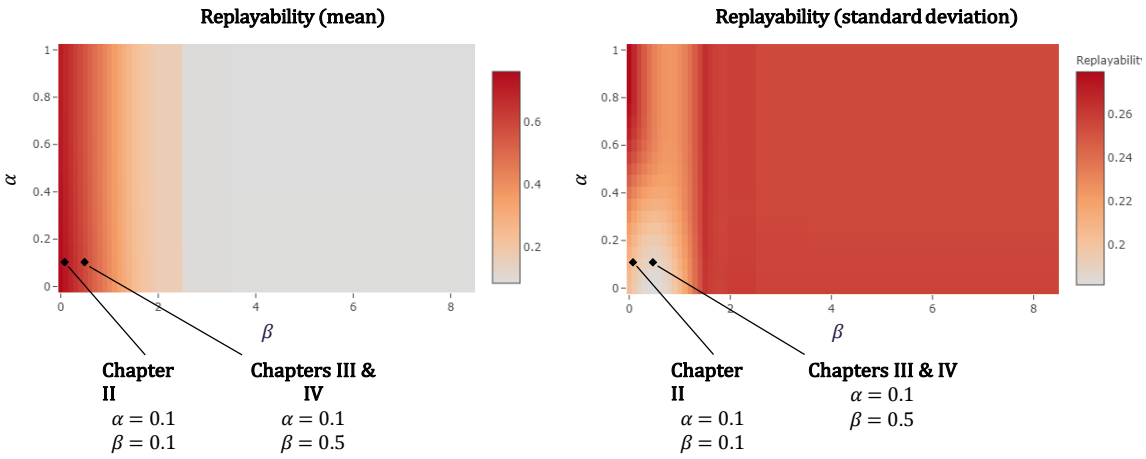


Figure B.1: Evolution of the replayability score obtained in function of α and β values: mean (left) and standard deviation (right) over 10 replications.

From these experiments, choosing $\alpha = 0.1$ and $\beta = 0.5$ is recommended, and was used for experiments conducted in Chapters 3 and 4. Qualitative experiments (for example by comparing obtained graphs for various values of α and β), are considered as future work on the impact of such parameters, both for descriptive and predictive process mining tasks.

Appendix C

Poster: Process Model-based Classification for Event Log Data

Hugo De Oliveira [1, 2], Vincent Augusto [1], Xiaolan Xie [1], Martin Prodel [2], Ludovic Lamarsalle [2]
[1] Mines Saint-Étienne, France; [2] HEVA, France

Data Science Summer School
June, 24th-28th 2019

model-based
cation for event log data

Introduction

Event logs are a widespread type of data structure carrying information of time and ordering of events. As the complexity increases when time-dependent processes are considered, human understanding of predictive models is a lever for acceptability and practical deployment. We present here a new binary classification algorithm, created for event log data. Moreover, the proposed algorithm provides transparency by producing a process model to explain training results and future predictions. Transparency of predictive models is a current challenge, particularly in healthcare where deep learning has become state of the art [1].

Computational experiment

Generation of traces with two graphs G_0 and G_1

After choosing a size configuration (pos_{max} for the length and div_e for the diversity), we create two graphs G_0 and G_1 with pos_{max} identical layers composed of div_e different nodes. Then, a proportion c_{pat} of shared patterns in G_1 is deleted. Traces are then created by randomly crossing the graphs, forming event logs L_0 and L_1 (10 per parameters combination).

Event logs

80% L_0^{train} L_1^{train} 20% L_0^{test} L_1^{test}

Training

DT*, RF**, MLP***
with event log flattening and hyperparameters tuning

Testing

$RepOpt$ and $DiffOpt$

Qualitative results

Process models

Quantitative results

AUC****

nethodology

ed here is the training of a binary classifier on event
es $L^{train}=(L_0^{train}, L_1^{train})$ of class 0 and 1.

$(PsM, \sigma)\in[0, 1]$ is a measure which quantifies the
odel PsM to represent a trace [2]. The idea behind the
k is the construction of a process model which well
om event log L_1^{train} while less representing traces
xtracts discriminative patterns from L_1^{train}.

unctions $RepOpt$ and $DiffOpt$ can be used to
with a metaheuristic to find optimal process model.

$$\max_{PsM} mean(R_1^{train})$$

$$\max_{PsM} mean(R_1^{train}) - mean(R_0^{train})$$

Event log and process model definitions

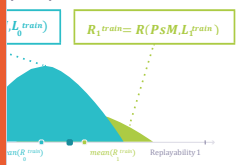
An **event log** $L=\{\sigma_1, \sigma_2, \ldots, \sigma_n\}$ regroups data in traces, each **trace** $\sigma_i = e_1 e_2 \ldots e_m$ being an ordered list of events, each **event** $e_k=(a, t)$ having a **label** $a\in A$ and a **time-stamp** t.

A **process model** $PsM=(N, E, L, P)$ is defined as a four-tuple with a set of **nodes** N and **edges** E, a **label function** L and a **position function** P. L and P map each node $n\in N$ respectively to a label $a\in A$ and a position $p\in\mathbb{N}^*$. Each edge links a node to another one of **strictly higher position.**

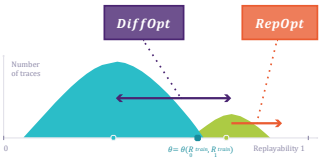
e of class prediction of a trace σ

epOpt or *DiffOpt*
L_1^{train}) to obtain replayability distributions $R_0^{train}= R(PsM, L_0^{train})$ and $R_1^{train}= R(PsM, L_1^{train})$

yability distributions



Distinct after optimization



separation
), e.g. using Gini impurity

3 **Compute the replayability** $r_\sigma = R(PsM, \sigma)$

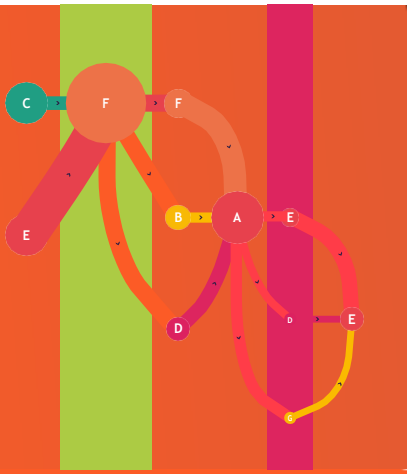
4 **Predict the class of the trace** σ by comparison between r_σ and θ

Qualitative results

A process model is used to discriminate the two classes using replayability. As a result, discriminative patterns extracted during the training procedure can be visualized.

The graph here highlights the specific patterns of G_1 which are absent of G_0 (for configuration c_{pat}=0.9, div_e=10, and pos_{max}=10).

Circles represent nodes of the model, and flux from circles represents edges. The size of nodes and edges are proportional to the number of traces represented.



Conclusion

Across this study, we proposed a new binary classification algorithm for event log data, based on process model optimization. Quantitative and qualitative results show the competitiveness and the transparency of the method. Future research will focus on the integration of more information into the model to increase prediction performance for other types of discriminative patterns.

* DT: Decision Tree; **RF: Random Forest; ***MLP: Multi Layer Perceptron; ****AUC: Area Under the Curve;

References
[1] C. Xiao, E. Choi, and J. Sun. Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. Journal of the American Medical Informatics Association, 06 2018.
[2] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie. Optimal process mining for large and complex event logs. IEEE Transactions on Automation Science and Engineering, 2018.

Appendix D

Poster: Process Mining for Predictive Analytics: a Case Study on NHS Data to improve Care for Sepsis Patients

Hugo De Oliveira [1,2], Martin Prodel [2], Ludovic Lamarsalle [2], Andi Orlowski [3]
1. Mines Saint-Etienne, France; 2. HEVA, France; 3. Imperial College Health Partners, England
Contact: hdeoliveira@hevaweb.com

CONFERENCE OF THE EUROPEAN WORKING GROUP
ON OPERATIONS RESEARCH APPLIED TO HEALTHCARE SERVICES
28 JULY - 02 AUGUST, 2019 - KARLSRUHE, GERMANY

Process mining for predictive analytics: a case study on NHS data to improve care for sepsis patients

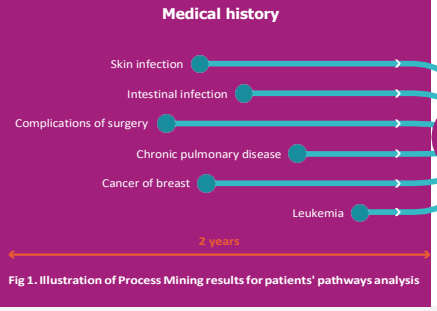
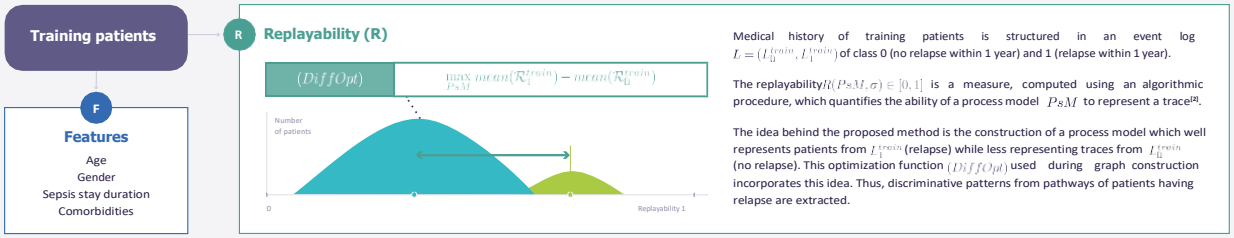


Fig 1. Illustration of Process Mining results for patients' pathways analysis

Context

Pattern extraction from medical history using process model optimization



Medical history of training patients is structured in an event log $L = (L_0^{train}, L_1^{train})$ of class 0 (no relapse within 1 year) and 1 (relapse within 1 year).

The replayability $R(PsM, \sigma) \in [0,1]$ is a measure, computed using an algorithmic procedure, which quantifies the ability of a process model PsM to represent a trace[2].

The idea behind the proposed method is the construction of a process model which well represents patients from L_1^{train} (relapse) while less representing traces from L_0^{train} (no relapse). This optimization function $(Diff Opt)$ used during graph construction incorporates this idea. Thus, discriminative patterns from pathways of patients having relapse are extracted.

Experiment

For all patients, features are used to train a decision tree for relapse prediction at sepsis episode release. For patients with exhaustive medical history (5 or more medical events within the 2 years before sepsis episode), the proposed methodology of pattern extraction is used to enrich feature data with the replayability score **(fig.2)**.

For each configuration, 80% of patients have been randomly selected as a 'train' set, the remaining 20% forming a 'test' set where area under the roc curve (AUC**) is computed as a performance measure. Decision tree (DT*) is used as a predictive algorithm, with maximum depth fixed as 4 for all models.

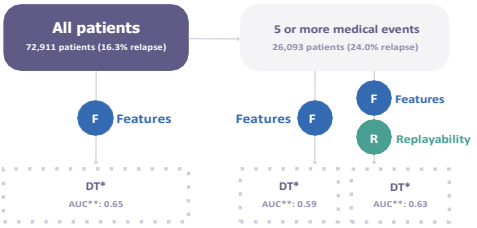


Fig 2. Schematic representation of the experiment

Results

Results show that for patients with 5 or more medical events, the addition of replayability within features increases performances in term of AUC** (from 0.59 to 0.63).

Moreover, replayability takes an important place in decision tree construction as the first split of the tree is performed using replayability (R > 0.003) **(fig 3.)**.

The result of optimization is a process model which shows extracted patterns of sepsis relapse within the medical history of patients. Thus, the visualization of such a process model provides insights to aid understanding of the risks of sepsis relapse. As a result, Leukemias, NHL*** or Neoplasms appear as particular events in medical history which are highly correlated with sepsis relapse **(fig 4.)**.

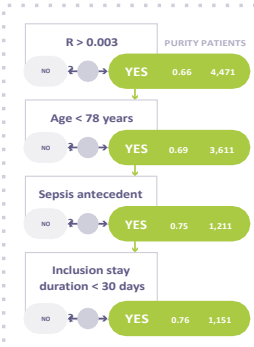


Fig 3. First 4 conditions of decision tree in favor of sepsis relapse, obtained for patients with 5 or more medical events (features+replayability)

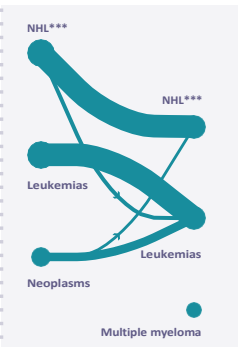


Fig 4. Resulting process model: pattern of sepsis relapse risk extracted from patients' medical history

Conclusion

Across this work, a methodology of pattern extraction from event log data (medical history) using process mining is presented. This method has been applied on a study case using NHS data to improve sepsis relapse prediction. Moreover, the obtained process model highlights some particular medical events within patients' history which will impact sepsis relapse risk.

The presented study is a proof of concept. Performances are not sufficient to be used in routine, but as the use of replayability increases performances, the methodology is encouraging. Future work will be focused on working with more precise data in order to improve performances and develop a tool to be used in practice, to identify at an early stage patients with a risk of sepsis relapse.

HEVA
IMPERIAL COLLEGE HEALTH PARTNERS

References
* DT: Decision Tree
**AUC: Area Under the Curve
*** NHL: Non-Hodgkin lymphoma

[1] Singer M, Deutschman CS, Seymour CW, et al. The third international consensus definitions for sepsis and septic shock (sepsis-3). JAMA 2016;315:801–10.
[2] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie. Optimal process mining for large and complex event logs. IEEE Transactions on Automation Science and Engineering, 2018.

Bibliography

[1] L. Hood, J. R. Heath, M. E. Phelps, and B. Lin, " Systems Biology and New Technologies Enable Predictive and Preventative Medicine," en, *Science*, vol. 306, no. 5696, pp. 640–643, Oct. 2004, Publisher: American Association for the Advancement of Science Section: Special Viewpoints, issn: 0036-8075, 1095-9203. doi: 10.1126/science.1104635 (cit. on p. 17).

[2] A. D. Weston and L. Hood, " Systems Biology, Proteomics, and the Future of Health Care: Toward Predictive, Preventative, and Personalized Medicine," en, *Journal of Proteome Research*, vol. 3, no. 2, pp. 179–196, Apr. 2004, issn: 1535-3893, 1535-3907. doi: 10.1021/pr0499693 (cit. on p. 17).

[3] L. Hood and S. H. Friend, " Predictive, personalized, preventive, participatory (P4) cancer medicine," en, *Nature Reviews Clinical Oncology*, vol. 8, no. 3, pp. 184–187, Mar. 2011, issn: 1759-4774, 1759-4782. doi: 10.1038/nrclinonc.2010.227 (cit. on pp. 17, 26).

[4] M. Flores, G. Glusman, K. Brogaard, N. D. Price, and L. Hood, " P4 medicine: How systems medicine will transform the healthcare sector and society," en, *Personalized Medicine*, vol. 10, no. 6, pp. 565–576, Aug. 2013, issn: 1741-0541, 1744-828X. doi: 10.2217/pme.13.57 (cit. on p. 17).

[5] J. McCarthy, " A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence," en, p. 3, (cit. on p. 17).

[6] M. Campbell, A. J. Hoane, and F.-h. Hsu, " Deep Blue," en, *Artificial Intelligence*, vol. 134, no. 1, pp. 57–83, Jan. 2002, issn: 0004-3702. doi: 10 . 1016 / S0004 – 3702(01)00129-1 (cit. on p. 17).

[7] D. A. Ferrucci, " Introduction to " This is Watson"," en, *IBM Journal of Research and Development*, vol. 56, no. 3.4, 1:1–1:15, May 2012, issn: 0018-8646, 0018-8646. doi: 10.1147/JRD.2012.2184356 (cit. on p. 17).

[8] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, " Mastering the game of Go with deep neural networks and tree search," en, *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016, issn: 0028-0836, 1476-4687. doi: 10.1038/nature16961 (cit. on p. 17).

[9] Q. V. Le, M. Ranzato, R. Monga, M. Devin, K. Chen, G. S. Corrado, J. Dean, and A. Y. Ng, "Building high-level features using large scale unsupervised learning," *arXiv:1112.6209 [cs]*, Jul. 2012, arXiv: 1112.6209 (cit. on p. 17).

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, " ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2012, pp. 1097–1105 (cit. on p. 17).

[11] I. Sutskever, O. Vinyals, and Q. V. Le, " Sequence to Sequence Learning with Neural Networks," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2014, pp. 3104–3112 (cit. on p. 17).

[12] D. Bahdanau, K. Cho, and Y. Bengio, " Neural Machine Translation by Jointly Learning to Align and Translate," *arXiv:1409.0473 [cs, stat]*, May 2016, arXiv: 1409.0473 (cit. on p. 17).

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, " BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423 (cit. on p. 17).

[14] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, Anglais. Cambridge, Massachusetts: MIT Press, Nov. 2016, isbn: 978-0-262-03561-3 (cit. on pp. 17, 18).

[15] Y. Bengio, A. Courville, and P. Vincent, " Representation Learning: A Review and New Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013, Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, issn: 1939-3539. doi: 10.1109/TPAMI.2013.50 (cit. on pp. 17, 30).

[16] T. D. Gunter and N. P. Terry, " The Emergence of National Electronic Health Record Architectures in the United States and Australia: Models, Costs, and Questions," EN, *Journal of Medical Internet Research*, vol. 7, no. 1, e3, 2005, Company: Journal of Medical Internet Research Distributor: Journal of Medical Internet Research Institution: Journal of Medical Internet Research Label: Journal of Medical Internet Research Publisher: JMIR Publications Inc., Toronto, Canada. doi: 10.2196/jmir.7.1.e3 (cit. on p. 18).

[17] P. Yadav, M. Steinbach, V. Kumar, and G. Simon, " Mining Electronic Health Records (EHRs): A Survey," *ACM Computing Surveys*, vol. 50, no. 6, 85:1–85:40, Jan. 2018, issn: 0360-0300. doi: 10.1145/3127881 (cit. on p. 18).

[18] D. L. Sackett, " Evidence-based medicine," en, *Seminars in Perinatology*, Fatal and Neonatal Hematology for the 21st Century, vol. 21, no. 1, pp. 3–5, Feb. 1997, issn: 0146-0005. doi: 10.1016/S0146-0005(97)80013-4 (cit. on p. 18).

[19] P. Tuppin, J. Rudant, P. Constantinou, C. Gastaldi-Ménager, A. Rachas, L. de Roquefeuil, G. Maura, H. Caillol, A. Tajahmady, J. Coste, C. Gissot, A. Weill, and A. Fagot-Campagna, " Value of a national administrative database to guide public decisions: From the système national d'information interrégimes de l'Assurance Maladie (SNIIRAM) to the système national des données de santé (SNDS) in France,"

en, *Revue d'Épidémiologie et de Santé Publique*, Réseau REDSIAM, vol. 65, S149–S167, Oct. 2017, issn: 0398-7620. doi: 10.1016/j.respe.2017.05.004 (cit. on p. 19).

[20] A. Kabeshova, Y. Yu, B. Lukacs, E. Bacry, and S. Gaïffas, " ZiMM: A deep learning model for long term and blurry relapses with non-clinical claims data," en, *Journal of Biomedical Informatics*, p. 103 531, Aug. 2020, issn: 1532-0464. doi: 10.1016/j.jbi.2020.103531 (cit. on pp. 19, 31).

[21] C. Villani, Y. Bonnet, C. Berthet, F. Levin, M. Schoenauer, A. C. Cornut, and B. Rondepierre, *Donner un sens à l'intelligence artificielle: pour une stratégie nationale et européenne*, fr. Conseil national du numérique, 2018, Google-Books-ID: Q7lUDwAAQBAJ, isbn: 978-2-11-145700-3 (cit. on p. 20).

[22] W. v. d. Aalst, *Process Mining: Data Science in Action*, en, 2nd ed. Berlin Heidelberg: Springer-Verlag, 2016, isbn: 978-3-662-49850-7 (cit. on pp. 20, 33–36, 40).

[23] E. Rojas, J. Munoz-Gama, M. Sepúlveda, and D. Capurro, " Process mining in healthcare: A literature review," en, *Journal of Biomedical Informatics*, vol. 61, pp. 224–236, Jun. 2016, issn: 1532-0464. doi: 10.1016/j.jbi.2016.04.007 (cit. on pp. 20, 36).

[24] T. G. Erdogan and A. Tarhan, " Systematic Mapping of Process Mining Studies in Healthcare," *IEEE Access*, vol. 6, pp. 24 543–24 567, 2018, issn: 2169-3536. doi: 10.1109/ACCESS.2018.2831244 (cit. on pp. 20, 36).

[25] W. El-Hage, C. Hingray, C. Lemogne, A. Yrondi, P. Brunault, T. Bienvenu, B. Etain, C. Paquet, B. Gohier, D. Bennabi, P. Birmes, A. Sauvaget, E. Fakra, N. Prieto, S. Bulteau, P. Vidailhet, V. Camus, M. Leboyer, M.-O. Krebs, and B. Aouizerate, " Les professionnels de santé face à la pandémie de la maladie à coronavirus (COVID-19) : Quels risques pour leur santé mentale ?" fr, *L'Encéphale*, vol. 46, no. 3, S73–S80, Jun. 2020, issn: 00137006. doi: 10.1016/j.encep.2020.04.008 (cit. on p. 24).

[26] B. Cardoen, E. Demeulemeester, and J. Beliën, " Operating room planning and scheduling: A literature review," en, *European Journal of Operational Research*, vol. 201, no. 3, pp. 921–932, Mar. 2010, issn: 03772217. doi: 10.1016/j.ejor. 2009.04.011 (cit. on p. 24).

[27] S. Saghafian, G. Austin, and S. J. Traub, " Operations research/management contributions to emergency department patient flow optimization: Review and research prospects," *IIE Transactions on Healthcare Systems Engineering*, vol. 5, no. 2, pp. 101–123, Apr. 2015, issn: 1948-8300. doi: 10.1080/19488300.2015.1017676 (cit. on p. 24).

[28] M. Fu, F. Glover, and J. April, " Simulation optimization: A review, new developments, and applications," in *Proceedings of the Winter Simulation Conference, 2005.*, ISSN: 1558-4305, Dec. 2005, 13 pp.–. doi: 10.1109/WSC.2005.1574242 (cit. on p. 24).

[29] K. Doi, " Computer-aided diagnosis in medical imaging: Historical review, current status and future potential," en, *Computerized Medical Imaging and Graphics*, Computer-aided Diagnosis (CAD) and Image-guided Decision Support, vol. 31, no. 4, pp. 198–211, Jun. 2007, issn: 0895-6111. doi: 10.1016/j.compmedimag.2007.02.002 (cit. on p. 24).

[30] Y. LeCun, Y. Bengio, and G. Hinton, " Deep learning," en, *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, issn: 0028-0836, 1476-4687. doi: 10.1038/nature14539 (cit. on p. 24).

[31] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, " Dermatologist-level classification of skin cancer with deep neural networks," en, *Nature*, vol. 542, no. 7639, pp. 115–118, Feb. 2017, Number: 7639 Publisher: Nature Publishing Group, issn: 1476-4687. doi: 10.1038/nature21056 (cit. on p. 24).

[32] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, R. Kim, R. Raman, P. C. Nelson, J. L. Mega, and D. R. Webster, " Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs," en, *JAMA*, vol. 316, no. 22, pp. 2402–2410, Dec. 2016, Publisher: American Medical Association, issn: 0098-7484. doi: 10.1001/jama.2016.17216 (cit. on p. 24).

[33] J.-Z. Cheng, D. Ni, Y.-H. Chou, J. Qin, C.-M. Tiu, Y.-C. Chang, C.-S. Huang, D. Shen, and C.-M. Chen, " Computer-Aided Diagnosis with Deep Learning Architecture: Applications to Breast Lesions in US Images and Pulmonary Nodules in CT Scans," *Scientific Reports*, vol. 6, Apr. 2016, issn: 2045-2322. doi: 10.1038/srep24454 (cit. on p. 24).

[34] A. Prasoon, K. Petersen, C. Igel, F. Lauze, E. Dam, and M. Nielsen, " Deep Feature Learning for Knee Cartilage Segmentation Using a Triplanar Convolutional Neural Network," en, in *Advanced Information Systems Engineering*, D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, C. Salinesi, M. C. Norrie, and Ó. Pastor, Eds., vol. 7908, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 246–253, isbn: 978-3-642-38708-1 978-3-642-38709-8. doi: 10.1007/978-3-642-40763-5_31 (cit. on p. 24).

[35] K. Fritscher, P. Raudaschl, P. Zaffino, M. F. Spadea, G. C. Sharp, and R. Schubert, " Deep Neural Networks for Fast Segmentation of 3D Medical Images," en, in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, and W. Wells, Eds., vol. 9901, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2016, pp. 158–165, isbn: 978-3-319-46722-1 978-3-319-46723-8. doi: 10.1007/978-3-319-46723-8_19 (cit. on p. 24).

[36] X. Su, H. Tong, and P. Ji, " Activity recognition with smartphone sensors," *Tsinghua Science and Technology*, vol. 19, no. 3, pp. 235–249, Jun. 2014, Conference Name: Tsinghua Science and Technology, issn: 1007-0214. doi: 10.1109/TST.2014.6838194 (cit. on p. 25).

[37] G. Bieber, M. Haescher, and M. Vahl, " Sensor requirements for activity recognition on smart watches," en, in *Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments - PETRA '13*, Rhodes, Greece: ACM Press, 2013, pp. 1–6, isbn: 978-1-4503-1973-7. doi: 10.1145/2504335.2504407 (cit. on p. 25).

[38] M. Memon, S. Wagner, C. Pedersen, F. Beevi, and F. Hansen, " Ambient Assisted Living Healthcare Frameworks, Platforms, Standards, and Quality Attributes," en, *Sensors*, vol. 14, no. 3, pp. 4312–4341, Mar. 2014, issn: 1424-8220. doi: 10.3390/s140304312 (cit. on p. 25).

[39] E. Zdravevski, P. Lameski, V. Trajkovik, A. Kulakov, I. Chorbev, R. Goleva, N. Pombo, and N. Garcia, " Improving Activity Recognition Accuracy in Ambient-Assisted Living Systems by Automated Feature Engineering," *IEEE Access*, vol. 5, pp. 5262–5280, 2017, Conference Name: IEEE Access, issn: 2169-3536. doi: 10.1109/ACCESS.2017.2684913 (cit. on p. 25).

[40] N. Varma, A. E. Epstein, A. Irimpen, R. Schweikert, and C. Love, " Efficacy and Safety of Automatic Remote Monitoring for Implantable Cardioverter-Defibrillator Follow-Up: The Lumos-T Safely Reduces Routine Office Device Follow-Up (TRUST) Trial," en, *Circulation*, vol. 122, no. 4, pp. 325–332, Jul. 2010, issn: 0009-7322, 1524-4539. doi: 10.1161/CIRCULATIONAHA.110.937409 (cit. on p. 25).

[41] E. Chiauzzi, C. Rodarte, and P. DasMahapatra, " Patient-centered activity monitoring in the self-management of chronic health conditions," en, *BMC Medicine*, vol. 13, no. 1, p. 77, Dec. 2015, issn: 1741-7015. doi: 10.1186/s12916-015-0319-2 (cit. on p. 25).

[42] M. H. McGillion, E. Duceppe, K. Allan, M. Marcucci, S. Yang, A. P. Johnson, S. Ross-Howe, E. Peter, T. Scott, C. Ouellette, S. Henry, Y. Le Manach, G. Paré, B. Downey, S. L. Carroll, J. Mills, A. Turner, W. Clyne, N. Dvirnik, S. Mierdel, L. Poole, M. Nelson, V. Harvey, A. Good, S. Pettit, K. Sanchez, P. Harsha, D. Mohajer, S. Ponnambalam, S. Bhavnani, A. Lamy, R. Whitlock, and P. Devereaux, " Postoperative Remote Automated Monitoring: Need for and State of the Science," en, *Canadian Journal of Cardiology*, vol. 34, no. 7, pp. 850–862, Jul. 2018, issn: 0828282X. doi: 10.1016/j.cjca.2018.04.021 (cit. on p. 25).

[43] C. P. Subbe, B. Duller, and R. Bellomo, " Effect of an automated notification system for deteriorating ward patients on clinical outcomes," en, *Critical Care*, vol. 21, no. 1, p. 52, Dec. 2017, issn: 1364-8535. doi: 10.1186/s13054-017-1635-z (cit. on p. 25).

[44] D. O. Nahmias, E. F. Civillico, and K. L. Kontson, " Deep learning and feature based medication classifications from EEG in a large clinical data set," en, *Scientific Reports*, vol. 10, no. 1, p. 14 206, Aug. 2020, Number: 1 Publisher: Nature Publishing Group, issn: 2045-2322. doi: 10.1038/s41598-020-70569-y (cit. on p. 25).

[45] N. Sridhar, A. Shoeb, P. Stephens, A. Kharbouch, D. B. Shimol, J. Burkart, A. Ghoreyshi, and L. Myers, " Deep learning for automated sleep staging using instantaneous heart rate," en, *npj Digital Medicine*, vol. 3, no. 1, p. 106, Dec. 2020, issn: 2398-6352. doi: 10.1038/s41746-020-0291-x (cit. on p. 25).

[46] U. R. Acharya, H. Fujita, O. S. Lih, Y. Hagiwara, J. H. Tan, and M. Adam, " Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network," en, *Information Sciences*, vol. 405, pp. 81–90, Sep. 2017, issn: 0020-0255. doi: 10.1016/j.ins.2017.04.012 (cit. on p. 25).

[47] S. L. Oh, E. Y. K. Ng, R. S. Tan, and U. R. Acharya, " Automated diagnosis of arrhythmia using combination of CNN and LSTM techniques with variable length heart beats," en, *Computers in Biology and Medicine*, vol. 102, pp. 278–287, Nov. 2018, issn: 0010-4825. doi: 10.1016/j.compbiomed.2018.06.002 (cit. on p. 25).

[48] S. Sheikhalishahi, R. Miotto, J. T. Dudley, A. Lavelli, F. Rinaldi, and V. Osmani, " Natural Language Processing of Clinical Notes on Chronic Diseases: Systematic Review," en, *JMIR Medical Informatics*, vol. 7, no. 2, e12239, 2019, Company: JMIR Medical Informatics Distributor: JMIR Medical Informatics Institution: JMIR Medical Informatics Label: JMIR Medical Informatics Publisher: JMIR Publications Inc., Toronto, Canada. doi: 10.2196/12239 (cit. on p. 25).

[49] Y. Xiong, B. Tang, Q. Chen, X. Wang, and J. Yan, " A Study on Automatic Generation of Chinese Discharge Summary," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Nov. 2019, pp. 1681–1687. doi: 10.1109/BIBM47256.2019.8983293 (cit. on p. 25).

[50] S. H. Lee, " Natural language generation for electronic health records," en, *npj Digital Medicine*, vol. 1, no. 1, pp. 1–7, Nov. 2018, Number: 1 Publisher: Nature Publishing Group, issn: 2398-6352. doi: 10.1038/s41746-018-0070-0 (cit. on p. 25).

[51] J. C. Venter, M. D. Adams, E. W. Myers, *et al.*, " The Sequence of the Human Genome," en, *Science*, vol. 291, no. 5507, pp. 1304–1351, Feb. 2001, Publisher: American Association for the Advancement of Science Section: Special Reviews, issn: 0036-8075, 1095-9203. doi: 10.1126/science.1058040 (cit. on p. 25).

[52] E. S. Lander, L. M. Linton, B. Birren, *et al.*, " Initial sequencing and analysis of the human genome," en, *Nature*, vol. 409, no. 6822, pp. 860–921, Feb. 2001, Number: 6822 Publisher: Nature Publishing Group, issn: 1476-4687. doi: 10.1038/35057062 (cit. on p. 25).

[53] R. A. Gibbs, " The Human Genome Project changed everything," en, *Nature Reviews Genetics*, Aug. 2020, issn: 1471-0056, 1471-0064. doi: 10.1038/s41576-020-0275-3 (cit. on p. 26).

[54] E. D. Esplin, L. Oei, and M. P. Snyder, " Personalized sequencing and the future of medicine: Discovery, diagnosis and defeat of disease," *Pharmacogenomics*, vol. 15, no. 14, pp. 1771–1790, Nov. 2014, issn: 1462-2416. doi: 10.2217/pgs.14.117 (cit. on p. 26).

[55] D. Ravì, C. Wong, F. Deligianni, M. Berthelot, J. Andreu-Perez, B. Lo, and G.-Z. Yang, " Deep Learning for Health Informatics," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 1, pp. 4–21, Jan. 2017, issn: 2168-2208. doi: 10.1109/JBHI.2016.2636665 (cit. on pp. 26, 30, 32).

[56] S. Ekins, A. C. Puhl, K. M. Zorn, T. R. Lane, D. P. Russo, J. J. Klein, A. J. Hickey, and A. M. Clark, " Exploiting machine learning for end-to-end drug discovery and development," en, *Nature Materials*, vol. 18, no. 5, pp. 435–441, May 2019, issn: 1476-1122, 1476-4660. doi: 10.1038/s41563-019-0338-z (cit. on p. 26).

[57] E. Smalley, " AI-powered drug discovery captures pharma interest," en, *Nature Biotechnology*, vol. 35, no. 7, pp. 604–605, Jul. 2017, issn: 1087-0156, 1546-1696. doi: 10.1038/nbt0717-604 (cit. on p. 26).

[58] Y. Cao, J. Romero, and A. Aspuru-Guzik, " Potential of quantum computing for drug discovery," en, *IBM Journal of Research and Development*, vol. 62, no. 6, 6:1–6:20, Nov. 2018, issn: 0018-8646, 0018-8646. doi: 10.1147/JRD.2018.2888987 (cit. on p. 26).

[59] S. J. Lewis, Z. Gandomkar, and P. C. Brennan, " Artificial Intelligence in medical imaging practice: Looking to the future," *Journal of Medical Radiation Sciences*, vol. 66, no. 4, pp. 292–295, 2019, issn: 2051-3909. doi: 10.1002/jmrs.369 (cit. on p. 26).

[60] " Data protection as a pillar of citizens' empowerment and the EU's approach to the digital transition - two years of application of the General Data Protection Regulation," en, Communication from the comission to the European parliament and the concil, Jun. 2020 (cit. on p. 27).

[61] T. Ryan-Mosley, *There is a crisis of face recognition and policing in the US*, en, MIT Technology Review, Aug. 2020 (cit. on p. 27).

[62] L. Nordling, " A fairer way forward for AI in health care," en, *Nature*, vol. 573, no. 7775, S103–S105, Sep. 2019, Number: 7775 Publisher: Nature Publishing Group. doi: 10.1038/d41586-019-02872-2 (cit. on p. 27).

[63] M. E. Charlson, P. Pompei, K. L. Ales, and C. R. MacKenzie, " A new method of classifying prognostic comorbidity in longitudinal studies: Development and validation," en, *Journal of Chronic Diseases*, vol. 40, no. 5, pp. 373–383, Jan. 1987, issn: 0021-9681. doi: 10.1016/0021-9681(87)90171-8 (cit. on p. 27).

[64] A. Morton, E. Marzban, G. Giannoulis, A. Patel, R. Aparasu, and I. A. Kakadiaris, " A Comparison of Supervised Machine Learning Techniques for Predicting Short-Term In-Hospital Length of Stay among Diabetic Patients," in *2014 13th International Conference on Machine Learning and Applications*, Dec. 2014, pp. 428–431. doi: 10.1109/ICMLA.2014.76 (cit. on p. 27).

[65] T. A. Daghistani, R. Elshawi, S. Sakr, A. M. Ahmed, A. Al-Thwayee, and M. H. Al-Mallah, " Predictors of in-hospital length of stay among cardiac patients: A machine learning approach," en, *International Journal of Cardiology*, vol. 288, pp. 140–147, Aug. 2019, issn: 01675273. doi: 10.1016/j.ijcard.2019.01.046 (cit. on p. 28).

[66] L. Turgeman, J. H. May, and R. Sciulli, " Insights from a machine learning model for predicting the hospital Length of Stay (LOS) at the time of admission," en, *Expert Systems with Applications*, vol. 78, pp. 376–385, Jul. 2017, issn: 0957-4174. doi: 10.1016/j.eswa.2017.02.023 (cit. on p. 28).

[67] J. Billings, J. Dixon, T. Mijanovich, and D. Wennberg, " Case finding for patients at risk of readmission to hospital: Development of algorithm to identify high risk patients," en, *BMJ*, vol. 333, no. 7563, p. 327, Aug. 2006, issn: 0959-8138, 1468-5833. doi: 10.1136/bmj.38870.657917.AE (cit. on p. 28).

[68] M. Di, M. Bojarnejad, S. King, W. Duan, M. Di, D. Zheng, A. Murray, and P. Langley, " Robust prediction of patient mortality from 48 hour intensive care unit data," en, *Computing in Cardiology*, 2012 (cit. on p. 28).

[69] A. Salcedo-Bernal, M. P. Villamil-Giraldo, and A. D. Moreno-Barbosa, " Clinical Data Analysis: An Opportunity to Compare Machine Learning Methods," en, *Procedia Computer Science*, CENTERIS/ProjMAN / HCist 2016, vol. 100, pp. 731–738, Jan. 2016, issn: 1877-0509. doi: 10 . 1016 / j . procs . 2016 . 09 . 218 (cit. on p. 28).

[70] F. Rahimian, G. Salimi-Khorshidi, A. H. Payberah, J. Tran, R. Ayala Solares, F. Raimondi, M. Nazarzadeh, D. Canoy, and K. Rahimi, " Predicting the risk of emergency admission with machine learning: Development and validation using linked electronic health records," en, *PLOS Medicine*, vol. 15, no. 11, A. Sheikh, Ed., e1002695, Nov. 2018, issn: 1549-1676. doi: 10.1371/journal.pmed.1002695 (cit. on p. 28).

[71] O. Ben-Assuli, R. Padman, M. Leshno, and I. Shabtai, " Analyzing Hospital Readmissions Using Creatinine Results for Patients with Many Visits," en, *Procedia Computer Science*, The 7th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2016)/The 6th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2016)/Affiliated Workshops, vol. 98, pp. 357–361, Jan. 2016, issn: 1877-0509. doi: 10.1016/j.procs.2016.09.054 (cit. on p. 28).

[72] J. Futoma, J. Morris, and J. Lucas, " A comparison of models for predicting early hospital readmissions," en, *Journal of Biomedical Informatics*, vol. 56, pp. 229–238, Aug. 2015, issn: 1532-0464. doi: 10.1016/j.jbi.2015.05.016 (cit. on p. 28).

[73] T. Desautels, R. Das, J. Calvert, M. Trivedi, C. Summers, D. J. Wales, and A. Ercole, " Prediction of early unplanned intensive care unit readmission in a UK tertiary care hospital: A cross-sectional machine learning approach," en, *BMJ Open*, vol. 7, no. 9, e017199, Sep. 2017, issn: 2044-6055, 2044-6055. doi: 10.1136/bmjopen-2017-017199 (cit. on p. 28).

[74] E. K. Lee, F. Yuan, D. A. Hirsh, M. D. Mallory, and H. K. Simon, " A clinical decision tool for predicting patient care characteristics: Patients returning within 72 hours in the emergency department," eng, *AMIA Annual Symposium proceedings*, vol. 2012, pp. 495–504, 2012, issn: 1942-597X (cit. on p. 28).

[75] D. Hooijenga, R. Phan, V. Augusto, X. Xie, and A. Redjaline, " Discriminant analysis and feature selection for emergency department readmission prediction," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, Nov. 2018, pp. 836–842. doi: 10.1109/SSCI.2018.8628938 (cit. on p. 28).

[76] Z. Xing, J. Pei, and E. Keogh, " A brief survey on sequence classification," *ACM SIGKDD Explorations Newsletter*, vol. 12, no. 1, pp. 40–48, Nov. 2010, issn: 1931-0145. doi: 10.1145/1882471.1882478 (cit. on p. 29).

[77] S. B. Needleman and C. D. Wunsch, " A general method applicable to the search for similarities in the amino acid sequence of two proteins," en, *Journal of Molecular Biology*, vol. 48, no. 3, pp. 443–453, Mar. 1970, issn: 0022-2836. doi: 10.1016/0022-2836(70)90057-4 (cit. on p. 29).

[78] T. F. Smith and M. S. Waterman, " Identification of common molecular subsequences," en, *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195–197, Mar. 1981, issn: 0022-2836. doi: 10.1016/0022-2836(81)90087-5 (cit. on p. 29).

[79] B.-J. Yoon, " Hidden Markov Models and their Applications in Biological Sequence Analysis," *Current Genomics*, vol. 10, no. 6, pp. 402–415, Sep. 2009, issn: 1389-2029. doi: 10.2174/138920209789177575 (cit. on p. 29).

[80] S. Blasiak and H. Rangwala, " A Hidden Markov Model Variant for Sequence Classification," en, in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, 2011, p. 6. doi: 10.5591/978-1-57735-516-8/IJCAI11-203 (cit. on p. 29).

[81] J. Klema, L. Novakova, F. Karel, O. Stepankova, and F. Zelezny, " Sequential Data Mining: A Comparative Case Study in Development of Atherosclerosis Risk Factors," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 1, pp. 3–15, Jan. 2008, issn: 1558-2442. doi: 10.1109/TSMCC.2007.906055 (cit. on p. 29).

[82] C. Zhou, B. Cule, and B. Goethals, " Pattern Based Sequence Classification," en, *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 5, pp. 1285–1298, May 2016, issn: 1041-4347. doi: 10.1109/TKDE.2015.2510010 (cit. on p. 29).

[83] R. J. C. Bose and W. M. van der Aalst, " Discovering signature patterns from event logs," en, in *2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, Singapore, Singapore: IEEE, Apr. 2013, pp. 111–118, isbn: 978-1-4673-5895-8. doi: 10.1109/CIDM.2013.6597225 (cit. on p. 29).

[84] F. Wang, N. Lee, J. Hu, J. Sun, S. Ebadollahi, and A. F. Laine, " A Framework for Mining Signatures from Event Sequences and Its Applications in Healthcare Data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 2, pp. 272–285, Feb. 2013, issn: 1939-3539. doi: 10.1109/TPAMI.2012.111 (cit. on p. 29).

[85] M. Vandromme, J. Jacques, J. Taillard, A. Hansske, L. Jourdan, and C. Dhaenens, " Extraction and optimization of classification rules for temporal sequences: Application to hospital data," en, *Knowledge-Based Systems*, vol. 122, pp. 148–158, Apr. 2017, issn: 0950-7051. doi: 10.1016/j.knosys.2017.02.001 (cit. on p. 29).

[86] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, " Deep learning for healthcare: Review, opportunities and challenges," en, *Briefings in Bioinformatics*, vol. 19, no. 6, pp. 1236–1246, Nov. 2018, issn: 1467-5463. doi: 10.1093/bib/bbx044 (cit. on p. 30).

[87] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi, " Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 5, pp. 1589–1604, Sep. 2018, issn: 2168-2208. doi: 10.1109/JBHI.2017.2767063 (cit. on pp. 30–32).

[88] C. Xiao, E. Choi, and J. Sun, " Opportunities and challenges in developing deep learning models using electronic health records data: A systematic review," en, *Journal of the American Medical Informatics Association*, vol. 25, no. 10, pp. 1419–1428, Oct. 2018. doi: 10.1093/jamia/ocy068 (cit. on pp. 30–32).

[89] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, " Doctor AI: Predicting Clinical Events via Recurrent Neural Networks," en, in *Machine Learning for Healthcare Conference*, Dec. 2016, pp. 301–318 (cit. on p. 31).

[90] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley, " Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records," en, *Scientific Reports*, vol. 6, no. 1, p. 26 094, May 2016, issn: 2045-2322. doi: 10.1038/srep26094 (cit. on pp. 31, 32).

[91] A. Rajkomar, E. Oren, K. Chen, A. M. Dai, N. Hajaj, M. Hardt, P. J. Liu, X. Liu, J. Marcus, M. Sun, P. Sundberg, H. Yee, K. Zhang, Y. Zhang, G. Flores, G. E. Duggan, J. Irvine, Q. Le, K. Litsch, A. Mossin, J. Tansuwan, D. Wang, J. Wexler, J. Wilson, D. Ludwig, S. L. Volchenboum, K. Chou, M. Pearson, S. Madabushi, N. H.

Shah, A. J. Butte, M. D. Howell, C. Cui, G. S. Corrado, and J. Dean, "Scalable and accurate deep learning with electronic health records," en, *npj Digital Medicine*, vol. 1, no. 1, pp. 1–10, May 2018, issn: 2398-6352. doi: 10.1038/s41746-018-0029-1 (cit. on p. 31).

[92] X. S. Zhang, F. Tang, H. H. Dodge, J. Zhou, and F. Wang, " MetaPred: Meta-Learning for Clinical Risk Prediction with Limited Patient Electronic Health Records," en, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowl- edge Discovery & Data Mining*, Anchorage AK USA: ACM, Jul. 2019, pp. 2487– 2495, isbn: 978-1-4503-6201-6. doi: 10.1145/3292500.3330779 (cit. on p. 31).

[93] A. Ashfaq, A. Sant'Anna, M. Lingman, and S. Nowaczyk, " Readmission prediction using deep learning on electronic health records," en, *Journal of Biomedical Informatics*, vol. 97, p. 103 256, Sep. 2019, issn: 1532-0464. doi: 10.1016/j.jbi.2019.103256 (cit. on p. 31).

[94] E. Choi, Z. Xu, Y. Li, M. Dusenberry, G. Flores, E. Xue, and A. Dai, " Learning the Graphical Structure of Electronic Health Records with Graph Convolutional Transformer," en, *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, pp. 606–613, Apr. 2020, issn: 2374-3468, 2159-5399. doi: 10.1609/aaai.v34i01.5400 (cit. on p. 31).

[95] T. Janssoone, C. Bic, D. Kanoun, P. Hornus, and P. Rinder, " Machine Learning on Electronic Health Records: Models and Features Usages to predict Medication Non-Adherence," *arXiv:1811.12234 [cs, stat]*, Nov. 2018 (cit. on p. 31).

[96] X. Min, B. Yu, and F. Wang, " Predictive Modeling of the Hospital Readmission Risk from Patients' Claims Data Using Machine Learning: A Case Study on COPD," en, *Scientific Reports*, vol. 9, no. 1, p. 2362, Feb. 2019, issn: 2045-2322. doi: 10.1038/ s41598-019-39071-y (cit. on p. 31).

[97] A. Henriksson, J. Zhao, H. Boström, and H. Dalianis, " Modeling heterogeneous clinical sequence data in semantic space for adverse drug event detection," in *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, Oct. 2015, pp. 1–8. doi: 10.1109/DSAA.2015.7344867 (cit. on p. 31).

[98] Z. Zhu, C. Yin, B. Qian, Y. Cheng, J. Wei, and F. Wang, " Measuring Patient Similarities via a Deep Architecture with Medical Concept Embedding," en, in *2016 IEEE 16th International Conference on Data Mining (ICDM)*, Barcelona, Spain: IEEE, Dec. 2016, pp. 749–758, isbn: 978-1-5090-5473-2. doi: 10.1109/ICDM.2016.0086 (cit. on p. 31).

[99] I. Landi, B. S. Glicksberg, H.-C. Lee, S. Cherng, G. Landi, M. Danieletto, J. T. Dudley, C. Furlanello, and R. Miotto, " Deep representation learning of electronic health records to unlock patient stratification at scale," en, *npj Digital Medicine*, vol. 3, no. 1, p. 96, Dec. 2020, issn: 2398-6352. doi: 10.1038/s41746-020-0301-z (cit. on p. 31).

[100] E. Choi, M. T. Bahadori, E. Searles, C. Coffey, M. Thompson, J. Bost, J. Tejedor-Sojo, and J. Sun, " Multi-layer Representation Learning for Medical Concepts," en, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco California USA: ACM, Aug. 2016, pp. 1495–1504, isbn: 978-1-4503-4232-2. doi: 10.1145/2939672.2939823 (cit. on p. 31).

[101] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, " GRAM: Graph-based Attention Model for Healthcare Representation Learning," en, in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Halifax NS Canada: ACM, Aug. 2017, pp. 787–795, isbn: 978-1-4503-4887-4. doi: 10.1145/3097983.3098126 (cit. on p. 31).

[102] A. L. Beam, B. Kompa, A. Schmaltz, I. Fried, G. Weber, N. Palmer, X. Shi, T. Cai, and I. S. Kohane, " Clinical Concept Embeddings Learned from Massive Sources of Multimodal Medical Data," eng, *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, vol. 25, pp. 295–306, 2020, issn: 2335-6936 (cit. on p. 31).

[103] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, " Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," en, *Information Fusion*, vol. 58, pp. 82–115, Jun. 2020, issn: 1566-2535. doi: 10.1016/j.inffus.2019.12.012 (cit. on p. 32).

[104] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, San Francisco, California, USA: Association for Computing Machinery, Aug. 2016, pp. 1135–1144, isbn: 978-1-4503-4232-2. doi: 10.1145/2939672.2939778 (cit. on p. 32).

[105] S. M. Lundberg and S.-I. Lee, " A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017, pp. 4765–4774 (cit. on p. 32).

[106] C. Rudin, " Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," en, *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, May 2019, issn: 2522-5839. doi: 10.1038/s42256-019-0048-x (cit. on p. 32).

[107] A. Vellido, " The importance of interpretability and visualization in machine learning for applications in medicine and health care," en, *Neural Computing and Applications*, Feb. 2019, issn: 0941-0643, 1433-3058. doi: 10.1007/s00521-019-04051-w (cit. on p. 32).

[108] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, " RETAIN: An Interpretable Predictive Model for Healthcare using Reverse Time Attention Mechanism," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds., Curran Associates, Inc., 2016, pp. 3504–3512 (cit. on p. 32).

[109] H. Suresh, N. Hunt, A. Johnson, L. A. Celi, P. Szolovits, and M. Ghassemi, " Clinical Intervention Prediction and Understanding with Deep Neural Networks," en, in *Machine Learning for Healthcare Conference*, Nov. 2017, pp. 322–337 (cit. on p. 32).

[110] X. Xu, Y. Wang, T. Jin, and J. Wang, " Learning the Representation of Medical Features for Clinical Pathway Analysis," en, in *Database Systems for Advanced Applications*, J. Pei, Y. Manolopoulos, S. Sadiq, and J. Li, Eds., ser. Lecture Notes in Computer Science, Cham: Springer International Publishing, 2018, pp. 37–52, isbn: 978-3-319-91458-9. doi: 10.1007/978-3-319-91458-9_3 (cit. on p. 32).

[111] J. Zhang, K. Kowsari, J. H. Harrison, J. M. Lobo, and L. E. Barnes, " Patient2Vec: A Personalized Interpretable Deep Representation of the Longitudinal Electronic Health Record," *IEEE Access*, vol. 6, pp. 65 333–65 346, 2018, issn: 2169-3536. doi: 10.1109/ACCESS.2018.2875677 (cit. on p. 32).

[112] D. A. Kaji, J. R. Zech, J. S. Kim, S. K. Cho, N. S. Dangayach, A. B. Costa, and E. K. Oermann, " An attention based deep learning model of clinical events in the intensive care unit," en, *PLOS ONE*, vol. 14, no. 2, I. Safro, Ed., e0211057, Feb. 2019, issn: 1932-6203. doi: 10.1371/journal.pone.0211057 (cit. on p. 32).

[113] C. Yin, R. Zhao, B. Qian, X. Lv, and P. Zhang, " Domain Knowledge Guided Deep Learning with Electronic Health Records," en, in *2019 IEEE International Conference on Data Mining (ICDM)*, Beijing, China: IEEE, Nov. 2019, pp. 738–747, isbn: 978-1-72814-604-1. doi: 10.1109/ICDM.2019.00084 (cit. on p. 32).

[114] A. R. C. Maita, L. C. Martins, C. R. L. Paz, L. Rafferty, P. C. K. Hung, S. M. Peres, and M. Fantinato, " A systematic mapping study of process mining," *Enterprise Information Systems*, vol. 12, no. 5, pp. 505–549, May 2018, issn: 1751-7575. doi: 10.1080/17517575.2017.1402371 (cit. on p. 33).

[115] C. d. S. Garcia, A. Meincheim, E. R. Faria Junior, M. R. Dallagassa, D. M. V. Sato, D. R. Carvalho, E. A. P. Santos, and E. E. Scalabrin, " Process mining techniques and applications – A systematic mapping study," en, *Expert Systems with Applications*, vol. 133, pp. 260–295, Nov. 2019, issn: 0957-4174. doi: 10.1016/j.eswa.2019.05.003 (cit. on pp. 33, 34, 37).

[116] W. van der Aalst, A. Adriansyah, A. K. A. de Medeiros, F. Arcieri, T. Baier, T. Blickle, J. C. Bose, P. van den Brand, R. Brandtjen, J. Buijs, A. Burattin, J. Carmona, M. Castellanos, J. Claes, J. Cook, N. Costantini, F. Curbera, E. Damiani, M. de Leoni, P. Delias, B. F. van Dongen, M. Dumas, S. Dustdar, D. Fahland, D. R. Ferreira, W. Gaaloul, F. van Geffen, S. Goel, C. Günther, A. Guzzo, P. Harmon, A. ter Hofstede, J. Hoogland, J. E. Ingvaldsen, K. Kato, R. Kuhn, A. Kumar, M. La Rosa, F. Maggi, D. Malerba, R. S. Mans, A. Manuel, M. McCreesh, P. Mello, J. Mendling, M. Montali, H. R. Motahari-Nezhad, M. zur Muehlen, J. Munoz-Gama, L. Pontieri, J. Ribeiro, A. Rozinat, H. Seguel Pérez, R. Seguel Pérez, M. Sepúlveda, J. Sinur, P. Soffer, M. Song, A. Sperduti, G. Stilo, C. Stoel, K. Swenson, M. Talamo, W. Tan, C. Turner, J. Vanthienen, G. Varvaressos, E. Verbeek, M. Verdonk, R. Vigo, J. Wang, B. Weber, M. Weidlich, T. Weijters, L. Wen, M. Westergaard, and M. Wynn, " Process Mining Manifesto," en, in *Business Process Management Workshops*, F. Daniel, K. Barkaoui, and S. Dustdar, Eds., ser. Lecture Notes in Business Information Processing, Berlin, Heidelberg: Springer, 2012, pp. 169–194, isbn: 978-3-642-28108-2. doi: 10.1007/978-3-642-28108-2_19 (cit. on p. 33).

[117] J. E. Cook and A. L. Wolf, " Automating process discovery through event-data analysis," in *Proceedings of the 17th international conference on Software engineering*, ser. ICSE '95, Seattle, Washington, USA: Association for Computing Machinery, Apr. 1995, pp. 73–82, isbn: 978-0-89791-708-7. doi: 10.1145/225014.225021 (cit. on p. 34).

[118] W. van der Aalst, T. Weijters, and L. Maruster, " Workflow mining: Discovering process models from event logs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 9, pp. 1128–1142, Sep. 2004, issn: 1558-2191. doi: 10.1109/TKDE.2004.47 (cit. on p. 34).

[119] A. K. A. D. Medeiros, V. B. F. Dongen, V. D. W. M.P. Aalst, and A. J. M. M. Weijters, " Process mining for ubiquitous mobile systems : An overview and a concrete algorithm," English, in *Ubiquitous Mobile Information and Collaboration Systems: Second CAiSE Workshop, UMICS 2004, Riga, Latvia, June 7-8, 2004, Revised Se- lected Papers*, Springer, 2004, pp. 151–165. doi: 10.1007/978-3-540-30188-2_12 (cit. on p. 34).

[120] L. Wen, W. M. P. van der Aalst, J. Wang, and J. Sun, " Mining process models with non-free-choice constructs," en, *Data Mining and Knowledge Discovery*, vol. 15, no. 2, pp. 145–180, Oct. 2007, issn: 1384-5810, 1573-756X. doi: 10.1007/s10618- 007-0065-y (cit. on p. 34).

[121] L. Wen, J. Wang, W. M. van der Aalst, B. Huang, and J. Sun, " Mining process models with prime invisible tasks," en, *Data & Knowledge Engineering*, vol. 69, no. 10, pp. 999–1021, Oct. 2010, issn: 0169023X. doi: 10.1016/j.datak.2010.06.001 (cit. on p. 34).

[122] H. R. Motahari-Nezhad, J. Recker, and M. Weidlich, Eds., *Business Process Management: 13th International Conference, BPM 2015, Innsbruck, Austria, August 31 – September 3, 2015, Proceedings*, en, ser. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, vol. 9253, isbn: 978-3-319-23062-7 978-3-319-23063-4. doi: 10.1007/978-3-319-23063-4 (cit. on p. 34).

[123] A. J. M. M. Weijters, V. D. W. M.P. Aalst, and A. K. A. D. Medeiros, *Process mining with the HeuristicsMiner algorithm*, English. Technische Universiteit Eindhoven, 2006, isbn: 978-90-386-0813-6 (cit. on p. 34).

[124] A. Weijters and J. Ribeiro, " Flexible Heuristics Miner (FHM)," in *2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, Apr. 2011, pp. 310–317. doi: 10.1109/CIDM.2011.5949453 (cit. on p. 34).

[125] C. W. Günther and W. M. P. van der Aalst, " Fuzzy Mining – Adaptive Process Simplification Based on Multi-perspective Metrics," en, in *Business Process Management*, D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, G. Alonso, P. Dadam, and M. Rosemann, Eds., vol. 4714, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 328–343, isbn: 978-3-540-75182-3 978-3-540-75183-0. doi: 10.1007/978-3-540-75183-0_24 (cit. on p. 34).

[126] A. K. A. de Medeiros, A. J. M. M. Weijters, and W. M. P. van der Aalst, " Genetic process mining: An experimental evaluation," en, *Data Mining and Knowledge Discovery*, vol. 14, no. 2, pp. 245–304, Apr. 2007, issn: 1384-5810, 1573-756X. doi: 10.1007/s10618-006-0061-7 (cit. on p. 34).

[127] J. C. A. M. Buijs, B. F. van Dongen, and W. M. P. van der Aalst, " On the Role of Fitness, Precision, Generalization and Simplicity in Process Discovery," en, in *On the Move to Meaningful Internet Systems: OTM 2012*, R. Meersman, H. Panetto, T. Dillon, S. Rinderle-Ma, P. Dadam, X. Zhou, S. Pearson, A. Ferscha, S. Bergamaschi, and I. F. Cruz, Eds., ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer, 2012, pp. 305–322, isbn: 978-3-642-33606-5. doi: 10.1007/978-3-642-33606-5_19 (cit. on p. 34).

[128] S. J. J. Leemans, D. Fahland, and W. M. P. van der Aalst, " Discovering Block-Structured Process Models from Event Logs - A Constructive Approach," en, in *Application and Theory of Petri Nets and Concurrency*, ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer, 2013, pp. 311–329, isbn: 978-3-642-38697-8. doi: 10.1007/978-3-642-38697-8_17 (cit. on p. 34).

[129] S. J. J. Leemans, D. Fahland, and W. M. P. van der Aalst, " Discovering Block-Structured Process Models from Event Logs Containing Infrequent Behaviour," en, in *Business Process Management Workshops*, ser. Lecture Notes in Business Information Processing, Cham: Springer International Publishing, 2014, pp. 66–78, isbn: 978-3-319-06257-0. doi: 10.1007/978-3-319-06257-0_6 (cit. on p. 34).

[130] S. J. J. Leemans, D. Fahland, and W. M. P. van der Aalst, " Discovering Block-Structured Process Models from Incomplete Event Logs," en, in *Application and Theory of Petri Nets and Concurrency*, ser. Lecture Notes in Computer Science, Cham: Springer International Publishing, 2014, pp. 91–110, isbn: 978-3-319-07734-5. doi: 10.1007/978-3-319-07734-5_6 (cit. on p. 34).

[131] S. J. J. Leemans, D. Fahland, and W. M. P. van der Aalst, " Scalable Process Discovery with Guarantees," en, in *Enterprise, Business-Process and Information Systems Modeling*, ser. Lecture Notes in Business Information Processing, Cham: Springer International Publishing, 2015, pp. 85–101, isbn: 978-3-319-19237-6. doi: 10.1007/978-3-319-19237-6_6 (cit. on p. 34).

[132] S. Ferilli, " WoMan: Logic-Based Workflow Learning and Management," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 6, pp. 744–756, Jun. 2014, issn: 2168-2232. doi: 10.1109/TSMC.2013.2273310 (cit. on p. 34).

[133] S. Ferilli, F. Esposito, D. Redavid, and S. Angelastro, " Predicting Process Behavior in WoMan," en, in *AI*IA 2016 Advances in Artificial Intelligence*, G. Adorni, S. Cagnoni, M. Gori, and M. Maratea, Eds., ser. Lecture Notes in Computer Science, Cham: Springer International Publishing, 2016, pp. 308–320, isbn: 978-3-319-49130-1. doi: 10.1007/978-3-319-49130-1_23 (cit. on p. 34).

[134] D. Chapela-Campa, M. Mucientes, and M. Lama, " Discovering Infrequent Behavioral Patterns in Process Models," en, in *Business Process Management*, J. Carmona, G. Engels, and A. Kumar, Eds., ser. Lecture Notes in Computer Science, Cham: Springer International Publishing, 2017, pp. 324–340, isbn: 978-3-319-65000-5. doi: 10.1007/978-3-319-65000-5_19 (cit. on p. 34).

[135] A. Augusto, R. Conforti, M. Dumas, M. La Rosa, and A. Polyvyanyy, " Split miner: Automated discovery of accurate and simple business process models from event logs," en, *Knowledge and Information Systems*, vol. 59, no. 2, pp. 251–284, May 2019, issn: 0219-3116. doi: 10.1007/s10115-018-1214-x (cit. on p. 34).

[136] J. De Weerdt, M. De Backer, J. Vanthienen, and B. Baesens, " A multi-dimensional quality assessment of state-of-the-art process discovery algorithms using real-life event logs," en, *Information Systems*, vol. 37, no. 7, pp. 654–676, Nov. 2012, issn: 03064379. doi: 10.1016/j.is.2012.02.004 (cit. on p. 34).

[137] A. Augusto, R. Conforti, M. Dumas, M. L. Rosa, F. M. Maggi, A. Marrella, M. Mecella, and A. Soo, " Automated Discovery of Process Models from Event Logs: Review and Benchmark," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 4, pp. 686–705, Apr. 2019, issn: 1558-2191. doi: 10.1109/TKDE.2018.2841877 (cit. on p. 34).

[138] H. A. Reijers and J. Mendling, " A Study Into the Factors That Influence the Understandability of Business Process Models," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 41, no. 3, pp. 449–462, May 2011, issn: 1558-2426. doi: 10.1109/TSMCA.2010.2087017 (cit. on p. 35).

[139] W. M. P. van der Aalst, " A practitioner's guide to process mining: Limitations of the directly-follows graph," en, *Procedia Computer Science*, CENTERIS 2019 - International Conference on ENTERprise Information Systems / ProjMAN 2019 - International Conference on Project MANagement / HCist 2019 - International Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN/HCist 2019, vol. 164, pp. 321–328, Jan. 2019, issn: 1877-0509. doi: 10.1016/j.procs.2019.12.189 (cit. on p. 35).

[140] W. M. P. van der Aalst, " Process mining and simulation: A match made in heaven!" In *Proceedings of the 50th Computer Simulation Conference*, ser. SummerSim '18, Bordeaux, France: Society for Computer Simulation International, Jul. 2018, pp. 1– 12 (cit. on p. 36).

[141] V. Augusto, X. Xie, M. Prodel, B. Jouaneton, and L. Lamarsalle, " Evaluation of discovered clinical pathways using process mining and joint agent-based discrete-event simulation," in *2016 Winter Simulation Conference (WSC)*, Dec. 2016, pp. 2135–2146. doi: 10.1109/WSC.2016.7822256 (cit. on pp. 36, 40).

[142] R. Phan, V. Augusto, D. Martin, and M. Sarazin, " Clinical Pathway Analysis Using Process Mining and Discrete-Event Simulation: An Application to Incisional Hernia," en, in *2019 Winter Simulation Conference (WSC)*, National Harbor, MD, USA: IEEE, Dec. 2019, pp. 1172–1183, isbn: 978-1-72813-283-9. doi: 10 . 1109 / WSC40007.2019.9004944 (cit. on p. 36).

[143] M. Prodel, V. Augusto, X. Xie, B. Jouaneton, and L. Lamarsalle, " Stochastic simulation of clinical pathways from raw health databases," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, Aug. 2017, pp. 580–585. doi: 10.1109/COASE.2017.8256167 (cit. on pp. 36, 40).

[144] A. E. Márquez-Chamorro, M. Resinas, and A. Ruiz-Cortés, " Predictive Monitoring of Business Processes: A Survey," *IEEE Transactions on Services Computing*, vol. 11, no. 6, pp. 962–977, Nov. 2018, Conference Name: IEEE Transactions on Services Computing, issn: 1939-1374. doi: 10.1109/TSC.2017.2772256 (cit. on p. 36).

[145] A. P. Kurniati, O. Johnson, D. Hogg, and G. Hall, " Process mining in oncology: A literature review," in *2016 6th International Conference on Information Communication and Management (ICICM)*, Oct. 2016, pp. 291–297. doi: 10 . 1109 / INFOCOMAN.2016.7784260 (cit. on p. 37).

[146] G. P. Kusuma, M. Hall, C. Gale, and O. Johnson, " Process Mining in Cardiology: A Literature Review," en, *International Journal of Bioscience, Biochemistry and Bioinformatics*, vol. 8, no. 4, pp. 226–236, Oct. 2018, issn: 2010-3638 (cit. on p. 37).

[147] W. Yang and Q. Su, " Process mining for clinical pathway: Literature review and future directions," in *2014 11th International Conference on Service Systems and Service Management (ICSSSM)*, Jun. 2014, pp. 1–5. doi: 10.1109/ICSSSM.2014.6943412 (cit. on p. 37).

[148] C. L. Ireson, " Critical pathways: Effectiveness in achieving patient outcomes," eng, *The Journal of Nursing Administration*, vol. 27, no. 6, pp. 16–23, Jun. 1997, issn: 0002-0443. doi: 10.1097/00005110-199706000-00008 (cit. on p. 37).

[149] K. Baker, E. Dunwoodie, R. G. Jones, A. Newsham, O. Johnson, C. P. Price, J. Wolstenholme, J. Leal, P. McGinley, C. Twelves, and G. Hall, " Process mining routinely collected electronic health records to define real-life clinical pathways during chemotherapy," en, *International Journal of Medical Informatics*, vol. 103, pp. 32–41, Jul. 2017, issn: 1386-5056. doi: 10.1016/j.ijmedinf.2017.03.011 (cit. on p. 37).

[160] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie, " Optimal Process Mining for Large and Complex Event Logs," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1309–1325, Jul. 2018, issn: 1558-3783. doi: 10.1109/TASE.2017.2784436 (cit. on pp. 37, 38, 40, 41, 81).

[151] R. Gatta, M. Vallati, C. Fernandez-Llatas, A. Martinez-Millana, S. Orini, L. Sacchi, J. Lenkowicz, M. Marcos, J. Munoz-Gama, M. Cuendet, B. de Bari, L. Marco-Ruiz, A. Stefanini, and M. Castellano, " Clinical Guidelines: A Crossroad of Many Research Areas. Challenges and Opportunities in Process Mining for Healthcare," en, in *Business Process Management Workshops*, C. Di Francescomarino, R. Dijkman, and U. Zdun, Eds., ser. Lecture Notes in Business Information Processing, Cham: Springer International Publishing, 2019, pp. 545–556, isbn: 978-3-030-37453-2. doi: 10.1007/978-3-030-37453-2_44 (cit. on pp. 37, 38).

[152] M. Rovani, F. M. Maggi, M. de Leoni, and W. M. van der Aalst, " Declarative process mining in healthcare," en, *Expert Systems with Applications*, vol. 42, no. 23, pp. 9236–9251, Dec. 2015, issn: 09574174. doi: 10.1016/j.eswa.2015.07.040 (cit. on p. 38).

[153] F. Mannhardt and D. Blinde, " Analyzing the trajectories of patients with sepsis using process mining," English, in *RADAR+EMISA 2017, Essen, Germany, June 12-13, 2017*, CEUR-WS.org, 2017, pp. 72–80 (cit. on pp. 38, 40).

[154] A. B. Jensen, P. L. Moseley, T. I. Oprea, S. G. Ellesøe, R. Eriksson, H. Schmock, P. B. Jensen, L. J. Jensen, and S. Brunak, " Temporal disease trajectories condensed from population-wide registry data covering 6.2 million patients," en, *Nature Communications*, vol. 5, no. 1, p. 4022, Jun. 2014, issn: 2041-1723. doi: 10.1038/ncomms5022 (cit. on p. 38).

[155] G. Kusuma, S. Sykes, C. McInerney, and O. Johnson, " Process Mining of Disease Trajectories: A Feasibility Study:" en, in *Proceedings of the 13th International Joint Conference on Biomedical Engineering Systems and Technologies*, Valletta, Malta: SCITEPRESS - Science and Technology Publications, 2020, pp. 705–712, isbn: 978-989-758-398-8. doi: 10.5220/0009166607050712 (cit. on p. 38).

[156] A. Alharbi, A. Bulpitt, and O. Johnson, " Improving Pattern Detection in Healthcare Process Mining Using an Interval-Based Event Selection Method," en, in *Business Process Management Forum*, Springer, Cham, Sep. 2017, pp. 88–105. doi: 10.1007/978-3-319-65015-9_6 (cit. on p. 38).

[157] U. Kaymak, R. Mans, T. v. d. Steeg, and M. Dierks, " On process mining in health care," in *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, ISSN: 1062-922X, Oct. 2012, pp. 1859–1864. doi: 10.1109/ICSMC.2012.6378009 (cit. on p. 38).

[158] B. F. A. Hompes, H. M. W. Verbeek, and W. M. P. van der Aalst, " Finding Suitable Activity Clusters for Decomposed Process Discovery," en, in *Data-Driven Process Discovery and Analysis*, P. Ceravolo, B. Russo, and R. Accorsi, Eds., vol. 237, Series Title: Lecture Notes in Business Information Processing, Cham: Springer International Publishing, 2015, pp. 32–57, isbn: 978-3-319-27242-9 978-3-319-27243-6. doi: 10.1007/978-3-319-27243-6_2 (cit. on p. 38).

[159] A. Alharbi, A. Bulpitt, and O. A. Johnson, " Towards Unsupervised Detection of Process Models in Healthcare," eng, *Studies in Health Technology and Informatics*, vol. 247, pp. 381–385, 2018, issn: 1879-8365 (cit. on pp. 38, 40).

[161] N. Martin, A. Martinez-Millana, B. Valdivieso, and C. Fernández-Llatas, " Interactive Data Cleaning for Process Mining: A Case Study of an Outpatient Clinic's Appointment System," en, in *Business Process Management Workshops*, C. Di Francescomarino, R. Dijkman, and U. Zdun, Eds., vol. 362, Cham: Springer International Publishing, 2019, pp. 532–544, isbn: 978-3-030-37452-5 978-3-030-37453-2 (cit. on p. 38).

[150] P. Weber, R. Backman, I. Litchfield, and M. Lee, " A Process Mining and Text Analysis Approach to Analyse the Extent of Polypharmacy in Medical Prescribing," in *2018 IEEE International Conference on Healthcare Informatics (ICHI)*, Jun. 2018, pp. 1–11. doi: 10.1109/ICHI.2018.00008 (cit. on p. 38).

[162] D. R. Jones, M. Schonlau, and W. J. Welch, " Efficient Global Optimization of Expensive Black-Box Functions," en, *Journal of Global Optimization*, vol. 13, no. 4, pp. 455–492, Dec. 1998, issn: 1573-2916. doi: 10.1023/A:1008306431147 (cit. on p. 39).

[163] H. De Oliveira, M. Prodel, and V. Augusto, " Binary Classification on French Hospital Data: Benchmark of 7 Machine Learning Algorithms," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2018, pp. 1743–1748. doi: 10.1109/SMC.2018.00301 (cit. on p. 39).

[164] H. T. C. Nguyen, S. Lee, J. Kim, J. Ko, and M. Comuzzi, " Autoencoders for improving quality of process event logs," en, *Expert Systems with Applications*, vol. 131, pp. 132–147, Oct. 2019, issn: 09574174. doi: 10.1016/j.eswa.2019.04.052 (cit. on p. 40).

[165] H. De Oliveira, M. Prodel, L. Lamarsalle, M. Inada-Kim, K. Ajayi, J. Wilkins, S. Sekelj, S. Beecroft, S. Snow, R. Slater, and A. Orlowski, " "Bow-tie" optimal pathway discovery analysis of sepsis hospital admissions using the Hospital Episode Statistics database in England," *JAMIA Open*, vol. 3, no. 3, pp. 439–448, Sep. 2020, issn: 2574-2531. doi: 10.1093/jamiaopen/ooaa039. eprint: https://academic.oup.com/jamiaopen/article-pdf/3/3/439/34283031/ooaa039.pdf (cit. on p. 68).

École Nationale Supérieure des Mines
de Saint-Étienne

NNT: 2020LYSEM021

Author: Hugo De Oliveira

Title: Predictive Modeling of Patient Pathways using Process Mining and Deep Learning

Speciality: Industrial Engineering

Keywords: predictive modeling, deep learning, process mining, patient pathway, non-clinical claims data, explainability.

Abstract

Initially created for a reimbursement purpose, non-clinical claim databases are exhaustive Electronic Health Records (EHRs) which are particularly valuable for evidence-based studies. The objective of this work is to develop predictive methods for patient pathways data, which leverage the complexity of non-clinical claims data and produce explainable results. Our first contribution focuses on the modeling of event logs extracted from such databases. New process models and an adapted process discovery algorithm are introduced, with the objective of accurately model characteristic transitions and time hidden in non-clinical claims data. The second contribution is a preprocessing solution to handle one complexity of such data, which is the representation of medical events by multiple codes belonging to different standard coding systems, organized in hierarchical structures. The proposed method uses auto-encoders and clustering in an adequate latent space to automatically produce relevant and explainable labels. From these contributions, an optimization-based predictive method is introduced, which uses a process model to perform binary classifica- tion from event logs and highlight distinctive patterns as a global explanation. A second predictive method is also proposed, which uses images to represent patient pathways and a modified Variational Auto-Encoders (VAE) to predict. This method globally explains predictions by showing an image of identified predictive factors which can be both frequent and infrequent.

ÉCOLE DES MINES DE SAINT-ÉTIENNE
158, cours Fauriel – CS 62362
42023 Saint-Étienne Cedex 2 – France
Tél. +33(0)4 77 42 01 23
www.mines-stetienne.fr

École Nationale Supérieure des Mines
de Saint-Étienne

NNT: 2020LYSEM021

Auteur: Hugo De Oliveira

Titre: Modélisation prédictive des parcours de soins à l'aide de techniques de Process Mining et de Deep Learning

Spécialité: Génie Industriel

Mots-Clefs: modélisation prédictive, deep learning, process mining, parcours patient, données médico-administratives, explicabilité.

Abstract

Les bases de données médico-administratives sont des bases de données de santé particulièrement exhaustives. L'objectif de ce travail réside dans le développement d'algorithmes prédictifs à partir des données de parcours patients, considérant la complexité des données médico-administratives et produisant des résultats explicables. De nouveaux modèles de processus et un algorithme de process mining adapté sont présentés, modélisant les transitions et leurs temporalités. Une solution de prétraitement des journaux d'événements est également proposée, permettant une représentation des évènements complexes caractérisés par de multiples codes appartenant à différents systèmes de codage, organisés en structures hiérarchiques. Cette méthode de clustering par auto-encodage permet de regrouper dans l'espace latent les événements similaires et produit automatiquement des labels pertinents pour le process mining, explicables médicalement. Un premier algorithme de prédiction adapté aux parcours est alors proposé, produisant via une procédure d'optimisation un modèle de processus utilisé pour classifier les parcours directement à partir des données de journaux d'événements. Ce modèle de processus sert également de support pour expliquer les patterns de parcours distinctifs entre deux populations. Une seconde méthode de prédiction est présentée, avec un focus particulier sur les événements médicaux récurrents. En utilisant des images pour modéliser les parcours, et une architecture d'auto-encodage variationnel modifiée pour l'apprentissage prédictif, cette méthode permet de classifier tout en expliquant de manière globale, en visualisant une image des facteurs prédictifs identifiés.

ÉCOLE DES MINES DE SAINT-ÉTIENNE
158, cours Fauriel – CS 62362
42023 Saint-Étienne Cedex 2 – France
Tél. +33(0)4 77 42 01 23
www.mines-stetienne.fr