Evaluation of risk factors for falls in the elderly based on a real data set using Bayesian networks

Gulshan Sihag

Thèse de doctorat
Pour obtenir le grade de Docteur de
l'UNIVERSITE POLYTECHNIQUE HAUTS-DE-FRANCE
et de l'INSA HAUTS-DE-FRANCE
et l'UNIVERSITE DE MONS

Discipline, :
Informatique, Mathématique et Recherche opérationnelle

Présentée et soutenue par Gulshan SIHAG
Le 3 Juillet 2023, à Valenciennes

Ecole doctorale :
Ecole Doctorale Polytechnique Hauts-de-France (ED PHF n°635)
Ecole Doctorale Université de Mons

Unité de recherche :
Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles et Humaines (LAMIH - UMR CNRS 8201)
Département Mathématique et Recherche opérationnelle, Université de Mons

Evaluation des facteurs de risque de chute chez les personnes âgées à l'aide de réseaux bayésiens basés sur un ensemble de données réelles

JURY

Président du jury
Jean-Baptiste BEUSCART, Professeur à CHU de Lille, France

Rapporteurs
Jean-Baptiste BEUSCART, Professeur à CHU de Lille, France
Rossana Maria de Castro ANDRADE, Professeur à Universidade Federal do Ceará, Brazil

Examinateurs
Marc PIRLOT, Professeur emeritus à Université de Mons, Belgique
Christina TIRNAUCA, Professeur à Universidad de Cantabria, Espagne
Pierre-Henri WUILLEMIN, Maître de conférences à LIP6, Sorbonne Université, France

Directeur de thèse
Sylvain PIECHOWIAK, Professeur à Université Polytechnique Hauts-de-France, France
Xavier SIEBERT, Professeur à Université de Mons, Belgique
Co-encadrant
Véronique DELCROIX, Maître de conférences à Université Polytechnique Hauts-de-France, France
Membres invités
Emmanuelle GRISLIN, Professeur à Université Polytechnique Hauts-de-France, France
François PUISIEUX, Professeur à CHU de Lille, France

PhD Thesis
Submitted for the degree of Doctor of Philosophy from
UNIVERSITE POLYTECHNIQUE HAUTS-DE-FRANCE
and INSA HAUTS-DE-FRANCE
and UNIVERSITY OF MONS

Subject :
Computer Science, Mathematics and Operations Research

Presented and defended by Gulshan SIHAG
On July 03, 2023, Valenciennes

Doctoral school :
Doctoral School Polytechnique Hauts-de-France (ED PHF n°635)
Doctoral School University of Mons

Research unit :
Laboratory of Industrial and Human Automation control Mechanical engineering and Computer science (LAMIH – UMR CNRS 8201)
Department of Mathematics and Operations Research, University of Mons

Evaluation of risk factors for falls in the elderly based on a real data set using Bayesian networks

JURY

President of jury
Jean-Baptiste BEUSCART, Professor at CHU Lille, France

Reviewers
Jean-Baptiste BEUSCART, Professor at CHU Lille, France
Rossana Maria de Castro ANDRADE, Professor at Universidade Federal do Ceará, Brazil

Examiners
Marc PIRLOT, Emeritus Professor at Université de Mons, Belgique
Christina TIRNAUCA, Professor at Universidad de Cantabria, Espagne
Pierre-Henri WUILLEMIN, Assistant Professor at LIP6, Sorbonne Université, France

Thesis directors
Sylvain PIECHOWIAK, Professeur à Université Polytechnique Hauts-de-France, France
Xavier SIEBERT, Professeur à Université de Mons, Belgique

Co-supervisor
Véronique DELCROIX, Assistant Professor at Université Polytechnique Hauts-de-France, France

Invited members
Emmanuelle GRISLIN, Professeur à Université Polytechnique Hauts-de-France, France
François PUISIEUX, Professeur à CHU de Lille, France

The Université Polytechnique Hauts-de-France and the Université de Mons neither endorse nor censure authors' opinions expressed in the theses: these opinions must be considered to be those of their authors.

The roots of education are bitter, but the fruit is sweet.

Aristotle

Abstract

Falls are a significant problem for the elderly, with identification and evaluation of risk factors being essential to reduce fall rates. However, fall prevention requires a pedagogical and repeated approach, time, and expertise to target actionable risk factors accurately. This thesis aims to evaluate the risk factors for falls using a real data set and Bayesian networks.

Using real data poses challenges, particularly in data preprocessing, which is time-consuming and requires expertise. Additionally, an application based on AI raises new challenges such as trustability, which depends on the interpretability and explainability of the results.

To address these challenges, this thesis proposes a knowledge model (Bayesian networks) that automatically evaluates the main actionable risk factors. The model is trained on a real data set combined with expert knowledge. Two iterations of the data preprocessing steps are presented and explained, including missing value imputation, variable selection, and the use of balancing techniques for imbalanced data. The first iteration included only the main variables to validate the process's feasibility, and the second iteration included as many variables as possible to improve the prediction and the process.

The model is compared with other well-known classifiers through different measures, including all or partial observation, and using or not balancing methods to manage the delicate question of imbalanced data. A Bayesian network is presented as a good solution, combining the quality of the results to evaluate the risk factors and the interpretability/explainability of the model from the expert's point of view.

The results show that predicting the presence or absence of risk factors for falls is a challenging task. While Bayesian Networks and other classifiers perform equivalently in terms of measures such as balanced accuracy and f1-score, the interest of Bayesian networks lies in their interpretability and the ability to use partial observations.

In summary, this thesis presents a contribution toward an application for fall prevention that facilitates automatic risk factor evaluation from partial observations of the patient, using a real data set and Bayesian networks. The proposed knowledge model (Bayesian networks) addresses the challenges of using real data and AI-based applications, respectively.

Keywords: Classification, Machine Learning, Problem of Falls, Fall Prevention, Bayesian Networks

Résumé

Les chutes constituent un problème important pour les personnes âgées, et l'identification et l'évaluation des facteurs de risque sont essentielles pour réduire les taux de chute. Cependant, la prévention des chutes nécessite une approche pédagogique et répétée, du temps et de l'expertise pour cibler avec précision les facteurs de risque exploitables. Cette thèse vise à évaluer les facteurs de risque de chute en utilisant un ensemble de données réelles et des réseaux bayésiens.

L'utilisation de données réelles pose des défis, en particulier en ce qui concerne le prétraitement des données, qui prend du temps et nécessite de l'expertise. De plus, une application basée sur l'IA soulève de nouveaux défis tels que la confiance, qui dépend de l'interprétabilité et de l'explicabilité des résultats.

Pour relever ces défis, cette thèse propose un modèle de connaissance (réseaux bayésiens) qui évalue automatiquement les principaux facteurs de risque pouvant faire l'objet d'une action. Le modèle est entraîné sur un ensemble de données réelles combinées à des connaissances d'experts. Deux itérations des étapes de prétraitement des données sont présentées et expliquées, y compris l'imputation des valeurs manquantes, la sélection des variables et l'utilisation de techniques d'équilibrage pour les données déséquilibrées. La première itération n'incluait que les principales variables pour valider la faisabilité du processus, et la seconde itération incluait autant de variables que possible pour améliorer la prédiction et le processus.

Le modèle est comparé à d'autres classificateurs bien connus par le biais de différentes mesures, y compris l'observation totale ou partielle, et l'utilisation ou non de méthodes d'équilibrage pour gérer la question délicate des données déséquilibrées. Un réseau bayésien est présenté comme une bonne solution, combinant la qualité des résultats pour évaluer les facteurs de risque et l'interprétabilité/explicabilité du modèle du point de vue de l'expert.

Les résultats montrent que prédire la présence ou l'absence de facteurs de risque de chute est une tâche difficile. Alors que les réseaux bayésiens et d'autres classificateurs ont des performances équivalentes en termes de mesures telles que la précision équilibrée et le score f1, l'intérêt des réseaux bayésiens réside dans leur interprétabilité et la possibilité d'utiliser des observations partielles.

En résumé, cette thèse présente une contribution à une application pour la prévention des chutes qui facilite l'évaluation automatique des facteurs de risque à partir d'observations partielles du patient, en utilisant un ensemble de données réelles et des réseaux bayésiens. Le modèle de connaissance proposé (réseaux bayésiens) répond aux défis de l'utilisation de données réelles et des applications basées sur l'IA, respectivement.

Mots clés : Classification, Apprentissage Automatique, Problème des chutes, Prévention des chutes, Réseaux Bayésiens

Acknowledgments

I would like to express my gratitude to all the people who have contributed to this thesis's realization. First, I would like to thank my co-supervisor, Veronique Delcroix, thank you for several hours of daily meetings, for many pieces of advice and discussions that helped me achieve this work. I also want to thank you for your moral support that helped me overcome the Ph.D. challenges. I truly appreciate all your efforts to make these three and half years an enjoyable experience for me. I have learned a lot from you. I also want to thank my advisor, Xavier Siebert, who played a big part in making this Ph.D. successful. Thanks for being so kind and for your constructive criticism. Your experience gave me confidence. I also want to thanks my co-advisers Sylvain Piechowiak and Emmanuelle Grislin, for their insightful suggestions through out this thesis. Besides my advisor and co-advisor, I would like to thank the members of my thesis committee: Jean-Baptiste Beuscart, Rossana Maria de Castro Andrade, Marc Pirlot, Christina Tirnauca, Pierre-Henri Wuillemin, and François Puisieux, for their time and feedback. I wish to show my gratitude to the current and former members of LAMIH team. Thanks for the amusing conversations at lunch, the generous pots, and for the journées du doctorant. I would like to thank the Région Hauts-de-France and University of Mons, Belgium immensely for the financial support that made this research possible. Last but by no means least, thanks to my family and friends who have provided me moral and emotional support in my life.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Context

Falls are a common and serious problem, particularly among older adults [23, 36, 93]. Falls can result in significant injuries, such as fractures and head trauma, and can lead to a loss of independence, reduced quality of life, and even death. Falls are a leading cause of injury-related hospitalizations and deaths among older adults. According to a report by the European Public Health Association, 3.8 million older people attend emergency departments each year with fall-related injuries, of which 1.4 million are admitted to hospitals for further treatment [118]. There are many risk factors associated with falls, including muscle weakness, gait, and balance problems, medication side effects, vision and hearing impairments, and environmental hazards such as slippery floors and poor lighting [53, 120]. Identifying and evaluating these risk factors is crucial to reduce fall rates and preventing injuries. However, accurately identifying and addressing fall risk factors can be a challenging task that requires a pedagogical and repeated approach, time, and expertise. Injuries from falls are also a financial burden. The study [118] shows that medical costs in Europe associated with fall-related injuries were approximately 25 billion euros per year.

The problem of falls is particularly relevant given the aging of the population in many countries. As the population in the world aged above 65 years is expected to increase from 9.3% in 2020 to 15.9% by 2050 [119], the incidence of falls is likely to increase, leading to a greater burden on healthcare systems and society as a whole. If effective prevention strategies are not established, the total cost of treating fall-related injuries in the European Union is also expected to increase up to 45 billion euros per year by 2050 [118]. As such, there is a need for effective and efficient methods to identify and evaluate fall risk factors, which can aid in fall prevention efforts and improve the health and well-being of older adults.

A large number of successful studies have addressed the problem of falls in elderly people [11, 23, 28, 36, 93, 120]. The authors in [19] state in their review that the problem of falls can be divided into two parts: Fall detection and fall prevention. The research in fall detection has been intensively investigated. The work related to fall detection is not part of this thesis.

The identification of risk factors for falls in an elderly person is essential to target an appropriate intervention to reduce that risk. Furthermore, a reduction of the risk factors could lead to a decrease in fall rate [96]. However, despite the volume of studies in the area, fall prevention remains an active subject of research [19, 28].

Fall prevention can be achieved by providing recommendations that help reduce the risk factors that are present for a given person (for instance: physiotherapy may improve balance and reduce the risk of muscular weakness). In the context of fall prevention, a large number of stakeholders can be considered, such as physicians, nurses, physiotherapists, close family members, caregivers, and the person himself. Various fall prevention systems have been proposed, often based on sensors [19, 28], but very few are based on a knowledge-based system [132, 18].

The evaluation of risk factors for falls requires time and expertise, and specific tests and devices may also be necessary. Therefore, a knowledge-based model could be useful to evaluate the risk factors to prevent falls. Moreover, the family physician who is one of the main actors in fall prevention generally does not have a lot of time, whereas fall prevention requires a pedagogical and repeated approach. As a consequence, the collection of information for a complete evaluation of risk factors for falls is not feasible regularly and the risk factors for falls of a person should be assessed from an incomplete set of observations. When there is no direct information for a specific risk factor, it is however possible to get a sense of it from general knowledge about its frequency in a specific context which is described by the other available information about the person. As an example, the experts know that a person who is afraid of falling and has neuropathy is much more likely to have balance problems than the average elderly. In this deduction, the expert combines general knowledge and reasoning with uncertainty. In order to support the identification of fall risks for a given person, we would like to be able to combine general knowledge and real data observed on this person.

The goal of this thesis is to automatically evaluate the main actionable risk factors for falls using a real data set combined with expert knowledge. To achieve that goal we propose the use of Bayesian networks (BN).

Bayesian networks are well-known models of knowledge. These probabilistic graphical models can manage uncertainty and allow updating the belief on a variable given information about other variables [58, 89]. In addition, BNs are understandable and modifiable by the expert thanks to the graph. These models are inherently explainable from their construction and thus allow transparency and visibility while decision-aiding. Another advantage of BNs is to update belief on any variable of the model from incomplete observations on any subset among the other variables. Also among the variables selected for this study, an arbitrary number of them can be observed, whether they are target or not. Moreover, risk factors are not independent of each other, meaning that when one of them is observed, it should be used to improve the evaluation of the others, in addition to other observed features. That situation makes very difficult the use of usual classifiers because a new model would have to be learned for each target variable, and each possible subset of observed variables. BN models allow overcoming that problem, since the same model can be used to evaluate any variable of the model, regarding any subset of observations. Another advantage of BN is that the model can be built both from data and expert knowledge which is very interesting in the context of health. It is also very important to make the model interpretable/ understandable by the final user (general practitioners) since it contributes to making the aiding system acceptable and augments the trust in results. So BN becomes the good choice to use because of the graphical representation that is easy to explain and understand.

Furthermore, in this thesis, we present two iterations of the data preprocessing steps, including missing value imputation, variable selection, and the use of balancing techniques for imbalanced data. The first iteration included only the main variables to validate the process's feasibility, and the second iteration included as many variables as possible to improve the prediction and the process. In addition, in order to evaluate the quality of prediction we compare the performance of Bayesian networks with other well-known classifiers through different measures, including all or partial observation, and using or not balancing methods to manage the delicate question of imbalanced data.

In summary, this thesis presents a contribution toward an application for fall prevention that facilitates the automatic evaluation of risk factors for falls from partial observations of the patient, using a real data set and Bayesian networks. Our results demonstrate the potential of Bayesian networks to manage uncertainty and provide a graphical view of the model of knowledge that contributes to the confidence of the expert and their validation of the model.

Plan of the thesis

The remainder of this dissertation is composed of 4 chapters as follows:

Chapter 2 - Basics and State of the art: This chapter describes the basic concepts of data preparation and machine learning algorithms followed by an overview of previous works that focus on the use of machine learning algorithms in healthcare.

Chapter 3 - Real data set: This chapter presents the details about the real data set from the service of fall prevention, hospital of Lille, France, used in our thesis and its analysis: manual selection of variables based on the ontology developed with the same service, steps of data preprocessing: data cleaning, reducing the size of the data, missing value imputation methods and the selected target risk factors to be evaluated.

Chapter 4 - Evaluate risk factors for falls using static data: This chapter describes the results obtained in order to evaluate the presence or absence of risk factors for falls using both iterations. Here we have presented the comparison of the predictive performance of Bayesian networks with other well-known classifiers, also the results when using a complete set or a specific subset, and the effect of using balancing techniques before training the classifier.

Chapter 5 - Conclusion: This chapter presents the conclusion of this thesis and the short and long-term perspectives for our research.

Chapter 2

Basics and State of the art

Outline of the current chapter

2.1 Introduction

Due to various advancements in data gathering technologies, such as physiological monitoring data and insurance claims data, hospitals, care centers, and other healthcare institutions are collecting an increasing amount of data [1].

Analysis and machine learning algorithms can aid in early illness identification, patient care, and community services as the amount of data in healthcare grow. Different approaches to healthcare prediction have been researched, and several methodologies have been evaluated [22].

In our work, we focus on the problem of falls. More precisely the identification of the presence or absence of risk factors for falls. With that aim, in this chapter, we first describe the problem of fall prevention.

Furthermore, the use of machine learning algorithms to detect health-related risks in patients is now common and a large number of studies in the literature have addressed this approach. With this aim, in the next part of this chapter, we describe the basics of machine learning followed by the literature review which shows an overview of different machine learning methods which have been tested on healthcare data. However, using machine learning in healthcare also poses some challenges such as dealing with imbalanced data, interpreting the results, ethical concerns, and data limitations. Imbalanced data can make it difficult to accurately predict the minority class. The results of machine learning models may be complex and hard to understand, and may not align with the clinical understanding of a disease or condition. Additionally, there are ethical concerns in using machine learning algorithms in healthcare such as privacy issues and ensuring fair and equitable treatment for all patients. Lastly, healthcare data is often siloed, scattered, incomplete and sensitive, making it difficult to access and use for machine learning purposes, and it's important to protect patient privacy and ensure compliance with regulations. In this context, in the last part of this chapter, we discuss the limitations of using machine learning with healthcare data.

2.2 Fall prevention problem

Falls present a striking danger to health and safety in older people [23, 36, 93]. According to a report by the European Public Health Association [118], 3.8 million older people in Europe attend emergency departments each year with fall-related injuries, of which 1.4 million are admitted to hospitals for further treatment. The most common consequences of these injuries are fractures, bruises, traumatic brain injury and reduced quality of life [53, 120]. Injuries from falls are also a financial burden. The study [118] shows that medical costs associated with fall-related injuries were approximately 25 billion euros per year.

The population aged above 65 years is expected to increase from 9.3% in 2020 to 15.9% by 2050 [119]. If effective prevention strategies are not established, the total cost of treating fall-related injuries in the European Union is also expected to increase up to 45 billion euros per year by 2050 [118].

A large number of successful studies have addressed the problem of falls in elderly people [11, 36, 93, 28, 23, 120]. In [19] the author stated in their review that the problem of falls can be divided into two parts: Fall Detection and Fall prevention.

The detection of risk factors for falls in an elderly person is essential to target an appropriate intervention. Reduction of the risk factors could lead to a decrease in fall rate [96]. Furthermore, research in fall detection has been intensively investigated; however, despite the volume of studies in the area, fall prevention remains an active subject of research [19, 28].

Fall prevention is a challenge to population aging, but it is one of the issues that have not been given sufficient attention. Since falls result from a complex interaction of risk factors, an important step in fall prevention is to detect the presence of risk factors for falls. Also, it has been shown that reducing the risk factors for falls reduces the risk of falls. Thus, fall prevention can be achieved by providing recommendations that help reduce the risk factors that are present for a given person (for instance: kinesiotherapy may improve balance and reduce the risk of muscular weakness). Among the risk factors for falls, some of them are reducible (or actionable) meaning that some actions can be carried out in order to reduce the risk factor. We focus on evaluating these risk factors. In the literature various fall prevention systems have been proposed, often based on sensors [19, 93, 36, 28], but very few are based on a knowledge-based system. Furthermore, the evaluation of risk factors for falls remains a challenge since it requires time and expertise, and specific tests and devices may also be necessary. Therefore, a knowledge-

based system could be useful to evaluate the risk factors to prevent falls. Moreover, the family physician who is one of the main actors in fall prevention generally does not have a lot of time, whereas fall prevention requires a pedagogical and repeated approach. As a consequence, the collection of information for a complete evaluation of risk factors is not feasible regularly and the risk factors for fall of a person should be assessed from an incomplete set of observations. In order to tackle that problem, this work is the first step toward a fall prevention aiding system.

Figure 2.1 shows a simplified view of the general architecture of the fall prevention aiding system. Lots of actors could contribute to fall prevention (doctors, family members, physiotherapists, nurses, specialists etc.). The objective of a fall prevention aiding system is to propose them a small number of adapted recommendations regarding the elderly they care for. These recommendations are selected on the basis of the most important risk factors for falls that are present in the elderly.

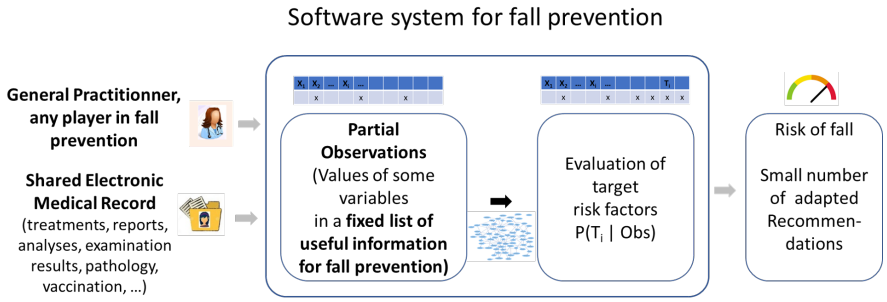


Figure 2.1: Proposition for a Fall Prevention system

The evaluation of risk factors for falls is a multi-factorial problem, and some risk factors can not be evaluated by a simple and rapid question. Moreover, the potential actors of fall prevention usually do not have much time for these questions, which makes it necessary to store useful information in a personal database. Fortunately, the value of some risk factors and useful variables to evaluate them could be automatically extracted from shared electronic medical records. However, some kinds of information are rarely present in the medical file, and the amount of information available for each elderly person is very different. Thus, the evaluation of the presence or absence of some risk factors for falls has to be done on the basis of information available about the person (partial or complete).

The use of machine learning algorithms to detect health-related risks in patients is now common [64, 78, 125] and a large number of successful studies have addressed the problem of falls in elderly [28, 36, 93, 120]. With that aim in our work, we use several machine-learning methods to evaluate the presence or absence of risk factors for falls. In the following section, we first introduce the basics of machine learning followed by a literature review that shows an overview of different machine learning algorithms which have been tested on healthcare data.

Summary of the Objectives

- ✓ focus on actionable risk factors for falls
- ✓ evaluate the level of risk for each (actionable) risk factor (allow ranking and a small number of recommendations)
- ✓ evaluation based on incomplete (partial) observations of the patient
- ✓ use a knowledge model (to make the system widely usable)
- ✓ provide possibility to view the graph of a causal model (augment trustability for expert users)

2.3 Basics of machine learning

In this section, we first describe the different types of machine learning methods, followed by the different classification algorithms. Then we present the different measures to evaluate the performance of a model.

2.3.1 Introduction to machine learning

Computer scientist and AI pioneer Tom Mitchell gave the definition of machine learning: *Machine Learning is the study of computer algorithms that improve automatically through experience.* More formally he states that: *A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E* [79].

Machine learning investigates the research and development of such algorithms and strategies that can learn from data and make predictions on data. These algorithms bypass the limitations of strictly static computer instructions by generating data-driven predictions or choices based on sample inputs [12]. Machine learning enables us to solve a wide range of computer tasks that would be impossible or impractical to handle using explicit algorithms or rule-based techniques.

Furthermore, in this section, we discuss the types of learning in machine learning. Figure 2.2 shows the classification of machine-learning algorithms by type of learning with a couple of examples.

2.3.1.1 Supervised learning

In this situation, a "teacher" presents the computer with sample inputs and desired outputs, and the objective is to learn a general rule that maps inputs to desired outputs. Desired output, also known as goal variable, is chosen to reflect the answer to a question that the organization would want to answer or a value unknown when the model is used to aid in decision making. Predictive modeling is another term for supervised learning [87]. Supervised learning includes:

- **Classification** - addresses the problem of determining which class (category) a new observation belongs to. In this situation, the target variables are categorical discrete variables. There are numerous categorization types based on the number of classes:

[1] Source: https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861

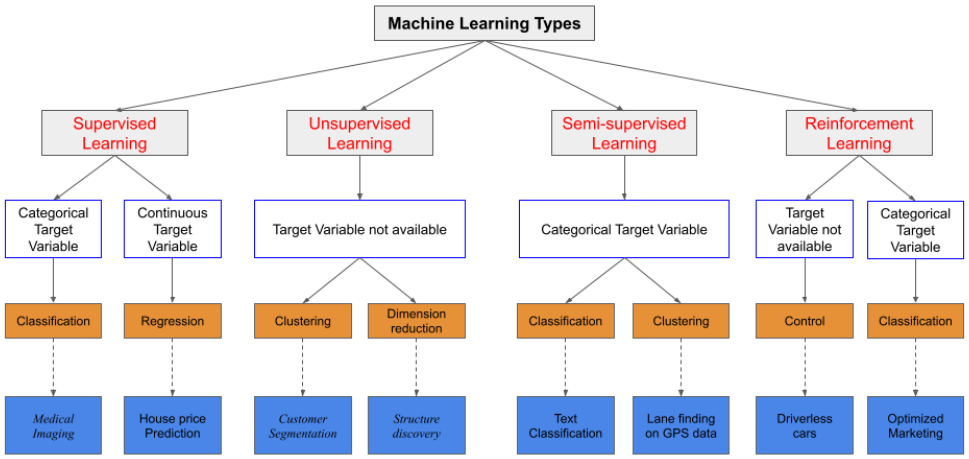


Figure 2.2: A description of the different types of machine learning[1]. On the left is supervised learning, which makes predictions using labeled data. Then we have unsupervised learning, which uses unlabeled data to try to uncover patterns and structures. Then we have semi-supervise learning, which makes predictions using a mixture of labeled and unlabeled data. At the very right is reinforcement learning, which attempts to learn from its own experience.

  - A problem having two classes is commonly referred to as a two-class or binary classification problem, for example, spam filtering.
  - A problem with more than 2 classes is called multi-class classification problems such as the categorization of photographs of fruits that may be oranges, apples, or pears.
  - A problem where input is assigned multiple classes is called a multi-label classification problem such as recognizing subjects in documents - the text might be about religion, politics, money, or education all at the same time, or none of these.

- **Regression** - These models operate with continuous target variables. Regression is a method for modeling the connection between one or more independent variables X and a dependent variable y (output/target variable) (explanatory variable). Predictor functions are used to characterize the connection between the dependent and independent variables. We distinguish the following forms of regression based on the number of independent variables:

  - *Simple linear regression* - 1 independent variable, such as predicting an employee's pay based on his years of experience. Salary is the dependent variable (y) in this example, while years of experience are the independent variable (X).
  - *Multiple linear regression* - is used when there is more than one independent variable. Predicting the price of an automobile, for example. The automobile attributes - brand, year, engine capacity, and mileage - are inputs. The output is the car's pricing. We have four independent variables in this case.
  - *Polynomial regression* - previous regression types used linear functions to describe the relationship between variables. Polynomial regression can be used when linear

models are too limiting. As an example, consider the prior case of predicting the price of an automobile. If there is no linear link between a car's qualities and its price, we may try polynomial regression.

2.3.1.2 Unsupervised learning

We simply have input data in this sort of learning; there is no goal variable. The goal of these approaches is to investigate the structure of data and discover patterns within it. The input space is structured in such a way that certain patterns appear more frequently than others, and we want to explore what happens and what does not. This is known as density estimation in statistics. Clustering is one way for estimating density. It is a problem of grouping a set of items so that those in the same group (cluster) are more similar to those in other groups (clusters) [87]. This approach is also used to detect outliers in the input data. Possible uses include customer segmentation, recommender systems, targeted marketing, structure discovery, etc.

2.3.1.3 Semi-supervised learning

It refers to a learning problem (and the algorithms developed to solve it) with a limited number of labeled instances and a large number of unlabeled examples from which a model must learn and predict new examples. As such, it is a learning problem that lies somewhere between supervised and unsupervised learning. A semi-supervised learning algorithm that is successful can outperform a supervised learning algorithm that is simply trained on labeled training instances [134]. Some examples include text classification, lane-finding on GPS data, etc.

2.3.1.4 Reinforcement learning

It is a branch of machine learning inspired by behaviorist psychology that is concerned with how software agents should behave in a given environment in order to maximize some concept of cumulative reward. It differs from supervised learning in that neither correct input/output pairings nor suboptimal behaviors are explicitly corrected.

The system's output is a series of actions. In this scenario, a single proper action is less crucial than a succession of correct actions leading to the objective. In any intermediate stage, there is no optimum action; an action is regarded as excellent if it is part of a good policy. A good application of reinforcement learning is game playing, where a single move is not as essential as a series of correct movements. A good move is one that is part of a solid game policy. Chess is an example of a game in which reinforcement learning was effectively employed. It is a game with few rules, but it is extremely difficult due to the enormous number of possible movements at each state and the large number of moves that a game may have. Once we have excellent algorithms that can learn to play games successfully, we can apply them to more obvious economic value applications [49].

As a reminder, our objective in this work is to evaluate the presence or absence of the risk factors for falls in the elderly which is an application of classification. With that objective, in the next section, we will discuss different types of classification algorithms.

2.3.2 Classification algorithms

In the literature, there exist numerous algorithms to do classification. In this section, we introduce several types of classification algorithms which we use in our work with their advantages

and disadvantages. First, a baseline or dummy classifier. Then we present logistic regression followed by the decision tree and random forest. Then we introduce the support vector machine and artificial neural networks and naive Bayes algorithm. In the end, we present the bayesian network classifiers.

2.3.2.1 Baseline or Dummy classifier

A baseline or dummy classifier makes predictions that ignore the input features. The role of the baseline classifier is to help to evaluate the quality of the results by comparing them against other more complex classifiers. The specific behavior of the baseline is selected with the strategy parameter. In our work, we use the most frequent strategy which means the baseline classifier always predicts the most frequent class in the target variable.

2.3.2.2 Logistic Regression

The technique of logistic regression has grown in significance in the field of machine learning. It enables the categorization of incoming information based on previous data by examining the correlation between one or more already present independent variables and forecasts a dependent data variable. Furthermore, it reduces complicated probability calculations to simple arithmetic problems. This helps to significantly reduce the impact of confounding factors and dramatically clarifies analyzing the impact of various variables. The algorithms become more accurate at making prediction classifications within data sets as new pertinent data is added [60]. This method has the benefit of directly providing the user with probabilities rather than just the class label information which can be helpful specifically when we have imbalanced datasets because the probability of each class allows it to predict the outcome of the minority class more accurately [99]. On the contrary, when the independent variables are strongly correlated, the model may not be able to correctly identify the outcomes because the estimates of the model coefficients may become unstable and inconsistent. Also, large quantities of noise or irrelevant characteristics in the data could be too much for the model to manage, since these might make the model less accurate. [97] Furthermore, a logistic regression classifier is applied in [4] to predict the chance of hospital readmission. The authors of [20] apply logistic regression to predict breast cancer patient survival.

2.3.2.3 Decision Tree

A decision tree consists of internal decision nodes and terminal leaves. Each node in the decision tree implements a test function with discrete output values that designate the branches. Each node accepts an input, runs the test, and then chooses one of the branches based on the outcome. This procedure begins at the root node and is continued recursively until a leaf node, which represents the class in which the input was classified, is reached. Figure 2.3 represents an example of a decision tree.

Decision trees are a type of recursive partitioning technique that is easy to define and execute. The following stages are shared by each variation of these algorithms [87]:

1. Assess the best approach to split the data into two or more divisions for each possible input variable, choose the optimal split, then divide the data into groups indicated by the split.
2. For all groups repeat step 1 (recursively).
3. Continue splitting until all records after a split belong to the same class or until another stop condition (statistical significance test or minimum record count) is met.

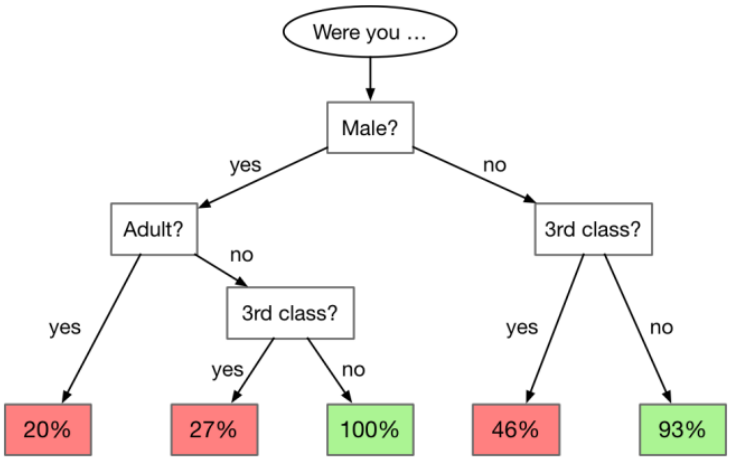


Figure 2.3: An example of a decision tree classifying/predicting a person's chances of survival on the Titanic. According to the results, a female from 1st/2nd class cabin or a male youngster from 1st/2nd class cabin has a good probability of being saved from the ship.

The main concept that distinguishes distinct algorithms in this category of classifiers is how they divide. They all offer a measure of the class distribution's purity. Some different types of decision tree algorithms are: ID3, C4.5, C5.0, etc [87]. According to [49], decision trees are one of the most often used classification algorithms. Because they are deemed simple to grasp, they are composed of a series of if-then-else principles that are more clear to ordinary people than mathematical formulas. In comparison to other types of classifiers, decision trees are simple to construct and scalable. They can deal with both numerical and categorical variables (some classification models only require numerical (neural networks) or categorical (naive bayes) variables). Sometimes, decision trees are likely to create over-complex trees that do not generalize the data well – this is called overfitting. In such circumstances, we may utilize strategies like pruning, limiting the number of samples necessary at a leaf node, or limiting the tree's maximum depth to minimize overfitting.

2.3.2.4 Ensemble methods

They combine numerous learning algorithms to achieve higher prediction performance than each of the constituent learning algorithms alone. A machine learning ensemble is made up of a limited number of different models, but it often enables far more flexible structures to exist within those models. Ensemble approaches aim to increase the capacity to generalize over a single estimate by combining the predictions of numerous base estimators created with a specific learning procedure. We identify two types of ensemble methods [87]:

- *Bagging methods* - the essential premise of averaging (bagging) approach is to create numerous estimators independently and then average their estimates. Due to lower variance, the combined estimator is generally superior to any of the single base estimators. Bagging, forests of randomized trees, and severely randomized trees are some examples.

- *Boosting methods* - unlike earlier methods, base estimators are generated successively and the bias of the aggregate estimator is attempted to be reduced. The key idea is to combine

multiple weak estimators to form a powerful ensemble. Examples include AdaBoost and Gradient Boosting.

Now, we present a type of bagging method called random forest.

Random Forest

Despite the fact that decision trees are relatively popular classifiers, they may overfit their training dataset despite producing generally decent results. Random forests solve the decision tree problem. Random forests are an ensemble learning approach for classification (as well as regression) and other tasks that work by creating several decision trees during training and outputting the class that is the mode of the classes. [87]

Random forests are a method of averaging numerous decision trees that have been trained on various regions of the same training dataset in order to reduce variation [55]. This comes at the expense of a slight increase in bias and some loss of interpretability, but it significantly improves the final model's performance. Bagging is applied to decision trees in random forests. Let $X = \{x_1, x_2, ..., x_n\}$ represent a collection of training samples with associated outputs. $Y = \{y_1, y_2, ..., y_n\}$, bagging repeatedly (M times), choose a random sample with X replacement and fit trees to these samples; for $m = (1, ..., M)$

1. Sample, with replacement, n training samples from X, Y; call these X_m, Y_m.
2. Train a decision tree f_m on X_m, Y_m.

After training, predictions for unseen samples $X^{'}$ are determined by taking the majority of votes of all trees. The output of a single tree is highly sensitive to training sample noise, whereas the average of several trees, assuming the trees are not connected, is not sensitive to training sample noise. Training several trees on a single training set may result in highly linked trees; bagging is a method of de-correlating the trees by exposing them to various training sets. This improves performance by lowering the variance of the model without raising the bias.

Moreover, the approach above describes the original bagging algorithm for trees. Random forests vary in just one way [14]: they employ a modified tree algorithm that picks a random subset of the features at each candidate split in the learning process. This is sometimes referred to as feature bagging. For a classification issue with N features, $\sqrt{N}$ features are typically employed in each split [55]. A graphic overview of a random forest classifier is given in figure 2.4.

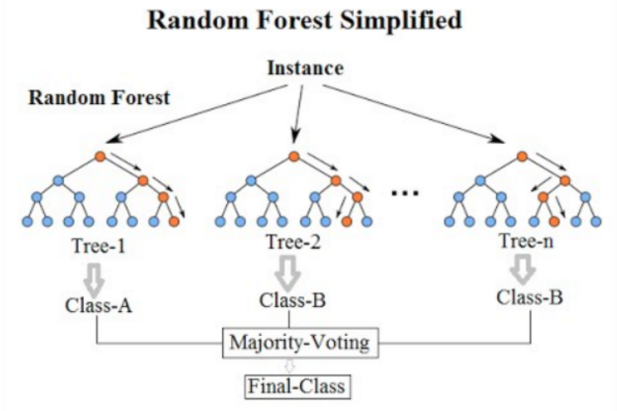


Figure 2.4: An example of a random forest model[2]

2.3.2.5 Support Vector Machine (SVMs)

SVMs are a type of supervised learning algorithm that can be used to classify and predict linear and nonlinear data. Assume we have an n-dimensional input vector training set. Each of these data items falls into one of two categories. The purpose is to separate them using an n-1 dimensional hyperplane, which is known as a linear classifier. Of course, there are several such classifiers that may meet this feature. However, we want to discover a hyperplane with the greatest margin/separation between two classes - that is, the greatest distance between the hyperplane and the nearest data points. The term support vectors refer to the vectors (data points) that are closest to this hyperplane. Figure 2.5 represents an example of an SVM classifier.

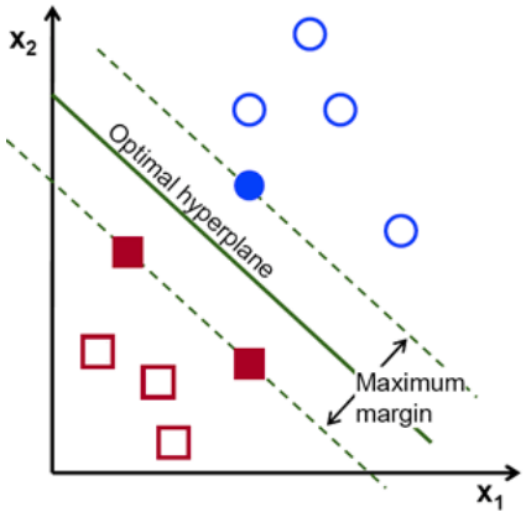


Figure 2.5: An example of an SVM classifer[3]

Later, a soft margin was created as a modification to the maximum margin [15]. Misclassifications are permitted in this situation, but they penalize the function to minimize by a factor proportional to a parameter C and the distance of the errors from the margin. In other words, SVM optimizes the margin between classes while reducing the penalization term, which is weighted by parameter C, which serves as a limit for the number of misclassifications. Also using kernel functions, which transform the data into a higher-dimensional space, SVM model can represent non-linear interactions between dependent and independent variables. On the contrary, the SVM model can be computationally expensive to train, particularly when the dataset is big or complex kernel functions are used. [24]

Furthermore, text classification tasks such as spam detection and sentiment analysis are performed using SVMs. They are also widely utilized in picture recognition tasks and in various fields of handwritten digit recognition, such as postal automation services. They are memory efficient because they employ a subset of training samples in the decision function (support vectors). [111]

[2]Source: https://williamkoehrsen.medium.com/random-forest-simple-explanation-377895a60d2d

[3]Source: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

2.3.2.6 Naive Bayes

The supervised machine learning algorithms that are primarily employed for classification also include Naive Bayes. It is referred to as "naive" because of the presumption that the input features used to build the model are independent. Therefore, altering one input feature will have no impact on the others. It is therefore naive in the sense that it is highly unlikely that this assumption is accurate. Furthermore, it is based on the Bayes theorem and is often suitable for very high-dimensional data sets. Bayes theorem can be described as follows:

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

where, $P(A|B)$ is called posterior which represents the probability of hypothesis A on the observed event B, and $P(B|A)$ is called likelihood which means the probability of the evidence given that the probability of a hypothesis is true, and $P(A)$ is called prior which represents the probability of hypothesis before observing the evidence, and $P(B)$ is marginal that represents the probability of evidence [95].

Furthermore, naive Bayes models are simple to set up. When there is a large correlation between feature variables, we may have low classification accuracy. Naive Bayes models are named after their assumption of feature independence. Despite their oversimplified assumptions, they produce extremely promising results in a variety of real-world problems, including document categorization and spam filtering [97]. They can be highly quick compared to other machine learning models. The main distinction between naive Bayes models and other machine learning models is that Naive Bayes models only consider one class at a time. They compute a likelihood for each class, and the class with the highest probability is allocated to the sample.

2.3.2.7 Artificial Neural Networks

In some ways, Artificial Neural Networks (ANN) resembles how the human brain learns. Neurons are the building blocks of an artificial neural network, and they in turn create layers. Each layer has a unique nonlinear activation function that aids in the learning process and the layer's output. The output of each layer is transferred to the following layer. Each epoch updates the weights connected to the neurons and is in charge of the overall predictions. Several optimizers are used to optimize the learning rate. Every ANN has a cost function, which is minimized as learning progresses. Then, the weights that produce the best results according to the cost function are used. The relationship between the neuron's input and output can be described as follows:

$$y = f(\sum_{i=1}^{n} w_i x_i + b),$$

where x_i denotes the input signal, w_i denotes the weight, y denotes the output, b denotes the threshold, and f denotes the activation function. These neurons are linked together to form ANN. Figure 2.6 represents an example of an ANN algorithm.

The ANN prediction algorithm has the advantage of not requiring an exact mathematical relationship between input and output parameters. The incorporation of spatial information, for example, does not necessitate its explicit parameterization. Another advantage of the ANN prediction algorithm is that as more data sets become available, the training sample sizes can easily be increased. On the contrary, the computational time of the ANN prediction algorithm rapidly increases as the number of parameters (layers) increases. Another disadvantage of ANN is to obtain optimal performance, as it requires careful adjustment of the hyperparameters (e.g.,

the number and size of the hidden layers), which can be time-consuming. Because it is not always evident how the network makes choices, and it can be challenging to interpret. [97, 130]

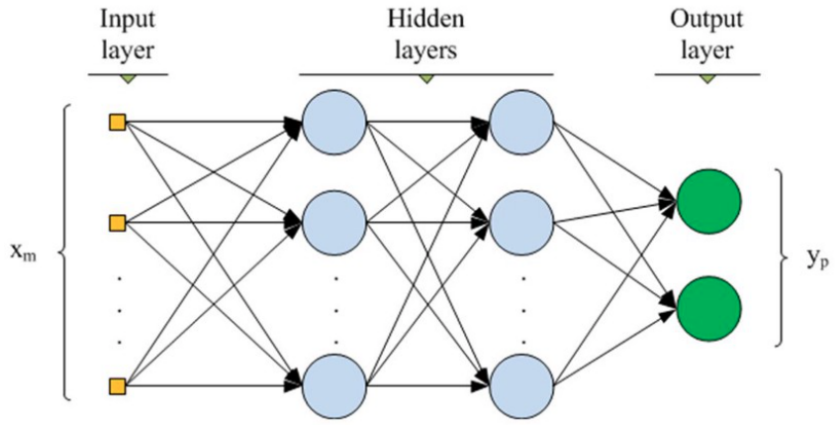


Figure 2.6: An example of a neural network model[4]

2.3.2.8 Bayesian Networks

A particular kind of probabilistic graphical model called Bayesian networks (BNs) is an effective tool for capturing uncertainty and evaluating risk [89]. The Bayes theorem and conditional probability theory are used to structure BNs. By computing the posterior probability distribution of any unobserved variable given new data input in a specific state, Bayes' theorem allows us to reason logically, rationally, and consistently. More formally, a Bayesian network is a graphical representation of a set of variables U = $\{X_1, X_2, \ldots, X_n\}$ with a joint probability that can be factorized as follows:

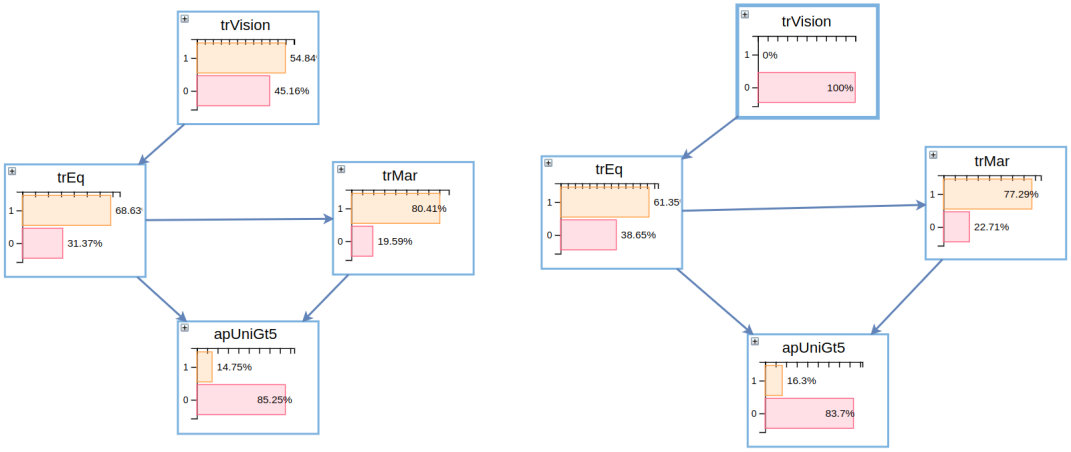
$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i | Parent(X_i))$$

where $Parent(X_i)$ is the set of variables that correspond to direct predecessors of X_i in the graph. It consists of a directed acyclic graph where each node represents a distinct random variable and each edge represents a conditional dependency. It also contains a set of the local probability distributions for each node/variable.

Figure 2.7 shows the graphical representation of a BN where each box represents a variable and the colored partitions inside each box represent the distributions of probability for each state of the variable. It represents a part of the graph learned during our experiment. Here left side (figure 2.7a) represents the BN with general knowledge about the population and the right side (figure 2.7b) shows that if we know that a given person has no problem in vision ($trVision = 0$), it increases our belief that this person has less chance of having balance disorder *(trEq)*.

Modeling of a BN includes (1) selecting the variables to be included in the model, (2) establishing network structure, and (3) obtaining a conditional probability table (parameters) for each variable. We now present different structure learning and parameter learning approaches for BN.

[4]Source: `https://williamkoehrsen.medium.com/deep-neural-network-classifier-32c12ff46b6c`



(a) BN with general knowledge

(b) BN after information $trVision = 0$

Figure 2.7: A graphical representation of a Bayesian Network

Structure learning in BN: There are two types of approaches in the literature for learning the structure of a BN from data. The first category of approaches is based on learning the conditional independence relations of the BN, from which the network is learned. These approaches are sometimes referred to as constraint-based approaches. The second category of methods, known as score-based approaches, views structure learning as an optimization issue, with scores targeted at maximizing the probability of the data given to the model. However, both approaches are known to provide NP-hard formulations, necessitating the employment of heuristic methods to discover near-optimum solutions with high probability in a reasonable amount of repetitions. We now briefly describe the basic concept behind this class of techniques. We refer to [89] for a more in-depth examination of this subject.

Constraint-Based Approaches: This family of approaches aims to construct a graph structure that reflects the dependency and independence relationships in the data that correspond to the empirical distribution. Nonetheless, the number of conditional independence tests that such algorithms would have to conduct among every pair of nodes to test all conceivable relations is exponential, necessitating the usage of some approximations. Some examples of this type of algorithm are the PC algorithm, the Incremental Association Markov Blanket (IAMB) algorithm, etc.

Score-Based Approaches: These techniques attempt to maximize the likelihood L of a collection of observed data D, which may be calculated as the product of each observation's probability. Because we aim to infer the optimal model G from the observed data, we define the likelihood of observing the data given a certain model G as:

$$LL(G:D) = \prod_{d \in D} P(d|G)$$

In practice, however, the most likely graph for any random collection of data is always the fully connected one, because adding an edge can only raise the likelihood of the data, i.e., this strategy overfits the data. To compensate for this constraint, the likelihood score is nearly usually paired with a regularization term that penalizes model complexity in favor of sparser solutions. As previously stated, such an optimization problem is intractable due to the vast search space

for viable solutions. As a result, heuristic approaches are frequently used to tackle optimization tasks. Some examples of such heuristic approaches include Hill Climbing, Tabu Search, Genetic algorithms, etc.

There is another type of method to learn the structure of BN called *Hybrid approach* which is the combination of the above two mentioned groups.

Parameter learning in BN: Parameter learning entails estimating CPT values from a data set for a known structure. There are two main approaches to dealing with the parameter-estimation task: one based on maximum likelihood estimation, and the other using Bayesian approaches [89].

So far we have described some of the numerous existing algorithms to do classification. Now, we present different methods to evaluate the performance of a model.

2.3.3 Measuring model's performance

What constitutes a good model is determined by the organization's interests and is described as the business success criterion. These criteria must be translated into predictive modeling criteria before they can be used to choose candidate models. If we require very precise forecasts, we utilize accuracy measures. However, in order to better comprehend the predictions, a more transparent model may be used. In such circumstances, we employ subjective measurements to gain more understanding. Some projects may utilize a combination of both to avoid selecting the most accurate model when a less accurate but more transparent model with roughly the same accuracy is available [55]. In that aim, we now present the different measures used to evaluate the performance of a classifier.

A number of techniques can be used to assess machine learning models. Analytical research is anticipated to expand with the use of a variety of evaluation tools. A classification model's (or "classifier's") performance on a set of test data for which the true values are known is described by a confusion matrix. Shown in Table 2.1, where TN (TP) is number of negative (positive) samples correctly classified, and FP (FN) is number of negative (positive) samples incorrectly classified as positive (negative). Also, Actual positive (negative) means that the sample is positive (negative) [110].

Table 2.1: A confusion matrix

	Predict Positive (PP)	**Predict Negative (PN)**
Actual Positive (P)	TP	FN
Actual Negative (N)	FP	TN

Accuracy is calculated as the ratio of the total number of correct predictions to the total number of predictions made by the model. It provides a general sense of how well the model is performing in terms of making correct predictions overall. It is a useful metric when the classes in the dataset are roughly balanced, meaning that there are roughly equal numbers of samples in each class. However, when the classes are imbalanced, accuracy may not be an appropriate metric to use because it can be misleading. For example, if 95% of the samples belong to one class, a classifier that always predicts that class would achieve an accuracy of 95%, but would not be very useful in practice. Accuracy can be defined as follows:

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

Misclassification Rate (Error Rate) tells overall, how often is the classifier wrong. It can be defined as follows:

$$Error\ Rate = \frac{FP + FN}{TP + FN + TN + FP} = 1 - Accuracy$$

True Positive Rate (Sensitivity or Recall) measures the proportion of actual positive samples that are correctly classified as positive by the model, out of the total number of actual positive samples. It should be used when the goal is to identify as many positive samples as possible while minimizing the number of false negative predictions. For example, in medical diagnosis, the true positive rate is a critical metric because we want to minimize the number of false negative predictions, that is, cases where the model incorrectly predicts that a patient does not have a disease when in fact they do. In this case, we would want to maximize the true positive rate to ensure that we capture as many positive cases as possible. Mathematically, it is defined as:

$$Recall = \frac{TP}{TP + FN}$$

False Positive Rate tells how often the model predicts True if the actual value is False. It should be used when the goal is to minimize the number of false positive predictions while capturing as many true negative cases as possible. For example, in airport security screening, the fast positive rate is a critical metric because we want to minimize the number of false positive predictions, i.e., cases where the model incorrectly predicts that a passenger has a dangerous item when in fact they do not. In this case, we would want to minimize the fast positive rate to ensure that we do not unnecessarily delay or inconvenience passengers, while still maintaining a high level of security. Mathematically, it is defined as:

$$False\ Positive\ Rate = \frac{FP}{TN + FP}$$

Specificity tells how often the model predicts False if the actual value is False. In the context of imbalanced data, specificity is particularly useful when the negative class is the minority class, and the model's performance is dominated by its ability to correctly identify the positive class. Specificity provides additional insight into the model's ability to correctly identify negative cases, which is also important in many applications. For example, suppose a medical test is designed to identify patients who have a rare disease. If the disease is indeed rare, the dataset will be imbalanced, with the negative class (healthy patients) being the majority. In this case, a high specificity will indicate that the model is able to correctly identify healthy patients, which is crucial for preventing unnecessary treatment or alarm. Mathematically it is defined as:

$$Specificity = \frac{TN}{TN + FP} = 1 - \ False\ Positive\ Rate$$

Precision tells how often is the model correct if it predicts True. Precision is useful in situations where the cost of a false positive is high, such as in medical diagnosis, fraud detection, or spam filtering. In these cases, it is important to minimize false positives, even at the cost of increased false negatives. In the context of imbalanced data, precision can be a more informative metric than accuracy, as it is less affected by the imbalance. For example, in a dataset with 90% negative examples and 10% positive examples, a classifier that always predicts negative will have an accuracy of 90%, but a precision of 0% for the positive class. Mathematically is defined

as:

$$Precision = \frac{TP}{TP + FP}$$

Prevalence tells how often the True condition actually occurs in data and is defined as:

$$Prevalence = \frac{P}{P + N}$$

Balanced accuracy Both binary and multi-class classification use balanced accuracy. It is widely used when working with imbalanced data, or when one of the target classes shows up much more frequently than the other. It is the arithmetic mean of sensitivity and specificity.

$$Balanced\ Accuracy = \frac{1}{2}(\frac{TP}{TP + FN} + \frac{TN}{TN + FP})$$

Area under ROC curve (AUC-ROC) Area Under the Receiver Operator Characteristic Curve provides a summary of the trade-offs between true positive rates and false-positive rates for the given predictive model. When the observations are evenly distributed among the classes, AUC-ROC produces good results. Also, it is a good metric to use with imbalanced data because it is not affected by the imbalance in the dataset. It is also a better metric than accuracy for imbalanced datasets because accuracy can be misleading when the dataset is imbalanced. AUC ROC considers both the True Positive Rate and the False Positive Rate, making it more suitable for evaluating the performance of models on imbalanced datasets.

Area under PR curve (AUC-PR) A simple graph with precision values on the y-axis and recall values on the x-axis makes up a PR curve. AUC-PR measures the quality of a classifier by computing the area under the PR curve. It is particularly useful when the dataset is imbalanced because it provides a more informative evaluation of the performance of a classifier when the positive class is rare. In such cases, accuracy may be misleading since a model that predicts the majority class most of the time can achieve high accuracy, despite being ineffective at identifying the minority class. By contrast, AUC-PR considers precision and recall, which are more relevant metrics in imbalanced datasets.

F1 score F1-score maintains a balance between recall and precision. F1 score (also known as F-measure, or balanced F-score) is an error metric that measures model performance by calculating the harmonic mean of precision and recall. It is useful when dealing with imbalanced data because it takes into account both true positives and false negatives. It is often used in classification tasks such as spam filtering, fraud detection, and medical diagnosis, where correctly identifying both positive and negative cases is important. Mathematically, it is defined as:

$$F1 - score = \frac{2 * precision * recall}{recall + precision}$$

F2 score The F2 score is based on the premise that recall should be given more weight than precision. It is useful in imbalanced datasets where recall is more important than precision. For example, in medical diagnosis tasks, it may be more important to correctly identify all positive cases, even at the cost of some false positives. Mathematically it is defined as:

$$F2 - score = \frac{5 * precision * recall}{4 * recall + precision}$$

So far we have presented the basics of machine learning algorithms and how to evaluate the quality of the prediction for a given machine learning algorithm. But, building a good machine learning model requires a good data set. Furthermore, working with data is more difficult than it may appear since it necessitates, first of all, careful handling of the data. Moreover, according to The State of Data Science 2020 report[5], data preparation and understanding is one of the most important and time-consuming tasks of the machine learning project lifecycle. With that aim in the next section, we present the basics of data preparation.

2.4 Basics of data preparation

Datasets typically need considerable preparation before they can produce significant insights since the majority of machine learning algorithms require data to be structured in a very specific way. In this section, we discuss some of the different steps of data preparation such as missing value imputation, the problem of imbalance in data, and feature selection to evaluate a target for a given machine learning algorithm.

2.4.1 Missing Value Imputation

The enhancement of technology in the modern world is relying on data. There is a lot of chaos incurred in the lively datasets. Missing data is a common problem faced with real-world datasets. Missing data can be anything from missing sequence, incomplete features, files missing, incomplete information, data entry error, etc. The origin of missing values can be caused by different reasons and depending on these origins missing values should be considered differently and dealt with in different ways [9].

Missing data are commonly classified as 'missing completely at random' (MCAR), 'missing at random' (MAR), and 'missing not at random' (MNAR) [5, 9, 34, 37, 41, 61, 92, 108, 123]. When the probability of missing data on a given variable is independent of the values of that variable, and of the values of other variables in the data set, the data are assumed to be MCAR [41]. For example, an observation is missing when a questionnaire of a study subject is accidentally lost when it is not related to any other patient characteristics. The missing data here would be MCAR. When the probability of non-response is independent of the missing value but is related to the values of another variable in the data set, the data are considered to be MAR [41]. For example, suppose we want to evaluate the predictive value of a diagnostic test for depression, and the test results are known for all diseased patients but unknown for a random sample of the non-diseased patient. In this case, the missing data would be MAR. When a missing data point is not dependent on other variables in the data set but is dependent on the unobserved missing value itself, missingness is assumed to be MNAR [41]. For example, suppose we want to conduct a depression survey, but some patients failed to fill in a depression survey because of their level of depression.

Furthermore, there are several options for handling missing values each with its own PROS and CONS. The choice of method for dealing with missing data is crucial for the validity of conclusions and should be based on careful consideration of the reasons for the missing data, missing data patterns and the availability of auxiliary information [61]. Researchers and Scientists discussed various techniques in their published work. Some of the basic techniques are discussed as under:

[5] https://www.anaconda.com/state-of-data-science-2020?utm_medium=blog&utm_source=anaconda&utm_campaign=sods-2020&utm_content=data-prep-blog

- **Listwise deletion:** Listwise deletion removes all data for a case with one or more missing values. It is the simplest technique to handle missing data but comes with an assumption that data are MCAR which is not always the case and hence produces biased results [92].
- **Pairwise Deletion:** Unlike to listwise deletion, pairwise deletion attempts to minimize the loss and maximizes all data available on an analysis-by-analysis basis. It means that pairwise deletion will not delete a case completely from the analyses but delete cases based on the variables included in the analysis. As a result, analyses may be completed on subsets of the data depending on where values are missing. But, the disadvantage of pairwise deletion is that it also requires the MCAR assumption to produce unbiased parameter estimates [92].
- **Mean Imputation:** Mean imputation is a method in which the mean of the available cases replaces the missing value on a certain variable. It is easy to use and can be used regardless of whether the data are MCAR, MAR, or MNAR. But the variability in the data is reduced, so the standard deviations and the variance estimates tend to be underestimated. Also, the magnitude of the covariance and correlation decreases by restricting the variability. Hence this method often causes biased estimates [92].
- **Last observation carried forward:** As the name states, it replaces the missing value with the last observation collected. This method makes the assumption that the observation of the individual has not changed at all since the last measured observation, which is mostly unrealistic [37].
- **K-Nearest Neighbour Imputation:** KNN approximates the missing value in the data with the help of k nearest values of that missing value. It measures the distance statistic between the case of missing data and all other cases in the data set and imputes missing values considering k number of values that are mostly similar to the values of interest. The disadvantage of using KNN is that whenever it looks for the most similar instances, the algorithm searches through all the data set.
- **Regression Imputation:** In regression imputation, the imputed value is predicted from a regression equation. It uses the information provided in the complete observations to predict the values of the missing observations [37].

Researchers have also developed some complex techniques to handle missing values such as missForest [123], EM algorithm, Maximum Likelihood Estimation, and Multiple Imputation [92].

2.4.2 Imbalance in data

Most of the machine learning classifiers trained on data with an uneven distribution of classes are prone to over-predicting the majority class. As a result, the minority class has a higher rate of misclassification. In addition, classification algorithms penalize false positives and false negatives equally, which is not adapted for imbalanced data.

An imbalanced data set occurs when there is an unequal representation of classes. A severe imbalance in a data set may raise a problem in the machine learning algorithm. These algorithms are more likely to classify a new observation in the majority class since the probability of belonging to the majority class is higher and the algorithm tries to minimize errors [57]. We encounter the imbalanced classification problem when our training data's class distribution has a significant skew. Even though the skew may not always be extreme (it can vary), we still consider

imbalanced classification to be a problem because it can affect how well our machine learning algorithms performs. One way of handling imbalanced data is oversampling. Oversampling is the duplicating of samples using the minority class. Another way is undersampling which includes deleting samples from the majority class. We can also handle the imbalanced data using hybrid methods which include both oversampling and undersampling. However, undersampling is interesting when the total number of cases is large enough, and deleting some cases does not lead to a loss of information.

One of the oversampling techniques to address the imbalance issue is SMOTE (synthetic minority oversampling technique) [21]. By increasing minority class examples at random and duplicating them, it seeks to balance the distribution of classes. SMOTE creates new minority instances by combining minority instances that already exist. For the minority class, it creates virtual training records using linear interpolation. For each example in the minority class, one or more of the k-nearest neighbors are randomly chosen to serve as these synthetic training records. SVM-SMOTE [133] is an additional oversampling technique. After training SVMs on the initial learning set, SVM-SMOTE uses support vectors to roughly estimate the borderline area. Each minority class support vector will be connected at random to a few of its closest neighbors by lines of synthetic data. Another SMOTE variant that doesn't concentrate on neighbors or borders is adaptive synthetic sampling (ADASYN) [56]. Instead, it emphasizes data density and generates fictitious data in line with that.

In the medical field, imbalanced data is frequent, and balancing techniques can be used to improve classification performance. In a recent study [42], Recurrent Neural Network (RNN) is utilized for classification, and Synthetic Minority Over-sampling Technique (SMOTE) is used to solve the problem of data imbalance. The SMOTE approach uses over- and under-sampling of the attributes based on the k Nearest Neighbor (kNN) algorithm. For categorization, the RNN processes the instance without reference to the prior instance [42].

In the real world, classifying imbalanced data is a difficult task for many data sets. In another study [17], SMOTE, Borderline-SMOTE, and ADASYN are put to the test to see how well they handle data set imbalance and what effect it has on classification accuracy. In this study, a classifier based on gradient boosting is deployed across seven datasets, and F1-Score, AUC, accuracy, recall, and precision are used to gauge classifier performance. Studies for the data sets Mammography, Liver Disorders, Diabetes (Pima Indian), Indian Liver, Habberman, and Immunotherapy indicated that oversampling technique increased accuracy from 2% to 11%. When compared to other oversampling techniques, borderline-SMOTE boosts accuracy more significantly. Surprisingly, Breast Cancer Wisconsin consistently achieves accuracy, whether oversampling is used or not [17].

2.4.3 Feature Selection

Feature selection is primarily focused on removing non-informative or redundant predictors from the model. Some predictive modeling problems have many features that can slow the development and training of models and require a large amount of system memory. Additionally, the performance of some models can degrade when including input features that are not relevant to the target feature. In literature, many studies focus on the evaluation of the features which are most affecting a given risk factor. For example, the risk factors for orthostatic hypotension, which is an important risk factor for falls, are studied in [44], and the factors associated with the fear of falling are examined in [45]. Generally, feature selection methods are classified into 3 categories:

- **Wrapper methods:** Wrappers require some method to search the space of all possible subsets of features, assessing their quality by learning and evaluating a classifier with

that feature subset. The feature selection process is based on a specific machine learning algorithm that we are trying to fit on a given dataset. It follows a greedy search approach by evaluating all the possible combinations of features against the evaluation criterion. e.g.- Forward features selection, backward feature selection, recursive feature elimination, etc. [16].

- **Filter methods:** Filter methods pick up the intrinsic properties of the features measured via univariate statistics instead of cross-validation performance. These methods are faster and less computationally expensive than wrapper methods. When dealing with high-dimensional data, it is computationally cheaper to use filter methods. e.g. - chi-square, mutual information gain, pearson correlation, fisher method, etc. [16].

- **Embedded methods:** These methods encompass the benefits of both the wrapper and filter methods, by including interactions of features but also maintaining reasonable computational cost. Embedded methods are iterative in the sense that they take care of each iteration of the model training process and carefully extract those features which contribute the most to the training for a particular iteration. e.g.- lasso regularization (L1), random forest importance, etc. [16].

So far, we have discussed in this chapter the basics of machine learning and how to prepare the data in order to gain some insights. In the following, we present a brief literature review of the use of machine learning algorithms in healthcare.

2.5 State of the art of using machine learning in healthcare

In the recent 20 years, as more data becomes available, the field of machine learning in healthcare is increasing. In order to construct methods for recognizing patterns in data, machine learning incorporates several diverse topics, including computer science, statistics, and optimization. Using those patterns, one can gain a deeper knowledge of a present situation or make predictions about a future one [128].

This section provides an overview of machine learning approaches to solve healthcare challenges. Table 2.2 provides an overview of several machine-learning approaches used in the healthcare domain as well as related references. We searched on *Scopus* database with the following string {KEY (machine-learning AND classification AND healthcare) }. We selected articles that are open source, in English, and published between 2013 and 2023 in a journal or conference proceedings. This process resulted in a total of 166 articles. We then excluded the articles which deal with image analysis, text data, big data, sensor data, and time series data. These steps lead us to 22 articles from this search strategy. We also added 4 more articles previously known to the author. Finally, we included a total of 26 articles for our analysis.

Furthermore, the selection of these 26 articles specifically focuses on the use of machine learning in healthcare for several reasons. Firstly, the articles showcase the various applications of machine learning in healthcare, such as predictive diagnostics, personalized treatment plans, and real-time monitoring of patients. Secondly, they provide insights into the current state of the field, including its challenges and limitations. These articles aim to provide a comprehensive overview of the use of machine learning in healthcare, including its benefits and limitations, helping to drive further research and development in this area. It can be seen from table 2.2 that all the classifiers listed are used, with a slightly smaller number of uses for ANN and BN. Furthermore, 10 out of 26 articles use only 1 or 2 classifiers whereas 9 articles use 4 classifiers or more. In this thesis, we use all 7 classifiers mentioned above. Furthermore, a brief explanation

of each of these references is provided as follows in order to provide insights into the recent literature on machine learning in healthcare.

Table 2.2: Overview of references using different machine learning methods

Reference	**LR**	**DT**	**RF**	**SVM**	**NB**	**ANN**	**BN**	Total
[2]	✗	✗	✗	✗	✗	✗		6
[3]	✗	✗	✗	✗		✗		5
[7]			✗		✗			2
[8]				✗				1
[10]		✗		✗	✗			3
[32]		✗	✗	✗	✗			4
[38]	✗		✗	✗				3
[46]	✗	✗	✗	✗	✗	✗		6
[48]	✗	✗	✗	✗				4
[51]	✗	✗	✗	✗	✗			5
[52]				✗	✗		✗	3
[54]		✗						1
[62]	✗	✗						2
[66]	✗		✗	✗				3
[67]	✗	✗	✗	✗		✗		5
[74]			✗		✗			2
[81]		✗		✗		✗		3
[84]							✗	1
[91]	✗		✗					2
[82]	✗		✗	✗	✗			4
[107]	✗	✗	✗	✗	✗			5
[112]							✗	1
[115]	✗			✗	✗			3
[124]	✗		✗					2
[126]				✗	✗			2
[127]	✗		✗			✗		3
26 studies	15	12	16	17	12	6	3	

The study [2] proposes non-invasive machine learning models for continuous glucose monitoring in diabetes patients. Random Forest and Decision Tree algorithms performed best, with 84% accuracy for the PIDD dataset[6] and 70% accuracy for the iGLU dataset[7]. The proposed device provides an excellent solution for continuous glucose monitoring compared to similar methods.

In [3] the author proposes an enhanced approach for identifying potential risk factors and predicting the incidence of stroke using ten classification models, including advanced boosting classifiers. The method achieved a high accuracy rate of 97% on all feature classifications, with gradient and ensemble boosting-tree-based models being the most suitable for predicting strokes in real-world situations. The study identifies age, heart disease, glucose level, hypertension, and marital status as the most significant risk factors, with other attributes also playing essential roles in obtaining the best performance.

[6]Source: `https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database`
[7]Source: `https://paperswithcode.com/dataset/iglu`

The author in [7] discusses the use of data mining techniques in the healthcare sector, specifically for the diagnosis of heart disease. The author explains that extreme values in data sets can reduce the accuracy of classification, and data conversion is an important step to pre-configure the process of converting data into suitable mining models. The authors applied classification methods such as Naive Bayes and Random Forest to original datasets and datasets with feature selection methods. The research was conducted on three different sets of heart disease data to analyze the pre-treatment effect in terms of accuracy.

The author in [8] explores the use of machine learning methods for predicting intensive care unit (ICU) admissions of COVID-19 patients and guiding hospital decision-makers in resource allocation. The authors analyzed the clinical and laboratory data of 100 patients with laboratory-confirmed COVID-19 tests using a weighted radial kernel support vector machine coupled with Recursive Feature Elimination. The proposed method outperformed other classification methods in discriminating between ICU and non-ICU admissions and identified a set of significant features that can assist in resource allocation and mobilization between intensive care and isolation units. The authors note that the study was retrospective and will require training to forecast prospectively.

Several data mining and classification methods are used in [10] to forecast breast cancer risk and diagnosis. Support Vector Machine, decision tree, Naive Bayes, and k-Nearest Neighbors are all compared by the authors. These techniques are used on the well-known Wisconsin Breast Cancer dataset. When comparing accuracy, it is demonstrated that SVM produces the best results.

The study [32] discusses the development of a diagnosis system to detect chronic kidney disease (CKD) using machine learning algorithms with the support of a hybrid feature selection approach. The study used clinical data from 400 CKD patients, prepared the dataset for the prediction model, and proposed a feature selection approach to remove redundant features. The Extra trees classifier was found to have the highest accuracy at 98%, while the Bagging classifier performed worst with only 60% accuracy. Early detection of CKD is important for saving lives, and this study offers a promising method for detecting the disease.

The author in [38] compares the performance of logistic regression with several other machine learning methods for estimating the risk of death in patients following emergency hospital admission using first blood test results and physiological measurements. The logistic model performed well compared to other methods, with a calibration slope of 0.90 and an area under the receiver operating characteristic curve of 0.847. The authors suggest that, given the complexity of tuning parameters for other methods, logistic regression with transformations is a competitive option for predicting in-hospital mortality with no evidence of overfitting.

In [46] the author presents a new hybrid predictive model for early mortality prediction in the Intensive Care Unit (ICU) using a combination of Genetic Algorithm, Stacking, and Boosting ensemble methods. The new model is designed to solve the highly imbalanced data problem using the SVM-SMOTE method. The study compared the new model with various machine learning models and achieved better performance than other classifiers. The proposed model was also benchmarked against state-of-the-art predictive models applied to the MIMIC dataset and outperformed them. The new model has the potential to provide valuable information about patients' lives and reduce costs at the earliest possible stage.

The author in [48] discusses the use of data mining and machine learning techniques to diagnose CKD at an early stage. The study evaluates the performance of five ML classification models, including support vector machine, random forest, logistic regression (LR), K-nearest neighbor, and decision tree, on a benchmark CKD dataset from UCI repository. The proposed model involves data pre-processing in two stages and the results show that the random forest model has accomplished the best results with a maximum precision of 0.99, recall of 0.99, and

F-score of 0.99 with a minimal error rate of 0.012.

The study [51] describes a case study conducted in a small city in Pakistan where healthcare facilities are limited to handle the COVID-19 pandemic. The study focuses on developing a machine learning classification model to predict the severity of COVID-19 patients to manage resources effectively. Among seven tested algorithms, the SVM was chosen to predict the severity of patients, which classified patients into mild, moderate, and severe levels with an accuracy of 60%.

In [52] the author discusses decision assistance for identifying patients at high risk of developing hyperlactatemia. The objective is to anticipate hyperlactatemia early on so that healthcare personnel can intervene and patient health can be improved. The scientists used 741 patients' electronic health records in their investigation. Various classification approaches, such as naive Bayes, support vector machines, Gaussian models, and Markov models, were applied. It was demonstrated that only three characteristics, the median lactate levels, the mean arterial pressure, and the median absolute deviation of the respiratory rate, may be used to make reasonable predictions.

The authors of [54] created a technique that practitioners may use to anticipate the diagnosis of diabetes patients. For prediction, this system employs a decision tree and k-nearest neighbors. The decision tree produced the best results, with an accuracy of more than 90%.

The author in [62] discusses how CKD is a growing global health crisis and that early prediction of CKD is important to improve patient outcomes. The article presents a methodology that uses machine learning techniques, such as logistic regression, decision tree classification, and K-nearest neighbor, to predict CKD status using clinical data. The study found that logistic regression had the highest accuracy rate of approximately 97%. The dataset used in the study was the CKD dataset, and the results show that the models employed in this study are more trustworthy than those used in previous studies.

In [66] the author discusses the issue of imbalanced medical datasets, where negative cases outnumber positive cases. Using two lung cancer datasets, the study compares the performance of 23 class imbalance methods with three classifiers to determine the best technique for medical datasets. The results show that class imbalance learning can improve the classification ability of the model. Over-sampling techniques have the lowest standard deviation and are generally more stable, with the random forest classifier performing the best with the random over-sampling method.

The author in [67] discusses the use of machine learning techniques to develop a predictive model for diagnosing and predicting the severity of cardiovascular disease (CVD). Various machine learning algorithms such as artificial neural networks, support vector machine, logistic regression, decision tree, random forest, and AdaBoost were applied to a heart disease dataset to predict the disease. The study constructed fusion models by combining the decisions of two algorithms using a weighted sum rule and applying a weighted score fusion approach to improve classification performance. The proposed approach was experimented with different test training ratios for binary and multiclass classification problems, with the highest accuracy being 95% for binary and 75% for multiclass classification.

In [74] the author discusses the development of an e-diagnosis tool for coronary artery disease (CAD) based on machine learning algorithms. ML methods, such as RandomForest, XGboost, MultilayerPerceptron, J48, AdaBoost, NaiveBayes, LogitBoost, and KNN were applied to medical datasets to predict and detect CAD. To improve accuracy, an ensemble model using majority voting was designed to combine the forecasts of individual classifiers. The results showed that the ensemble majority voting approach based on the top 3 classifiers, MultilayerPerceptron, RandomForest, and AdaBoost, achieved the highest accuracy of 88.12%. The study demonstrates that the majority voting ensemble approach proposed is the most accurate ML classification

approach for the prediction and detection of CAD.

In study [81] the author focuses on predicting readmission of patients with Chronic Obstructive Pulmonary Disease (COPD) using machine learning algorithms, with the goal of reducing healthcare costs and improving the quality of care. The study evaluates the performance of different machine learning models using Area Under Curve (AUC) and Accuracy (ACC) as criteria for prediction power. The study also identifies important variables for predicting readmission and achieves the highest accuracy of 91%.

In [84], the author suggests building a system to facilitate intelligent decision-making in the diagnosis of preeclampsia using Bayesian networks to assist specialists in the pregnant's care. They used information on the symptoms of 20 pregnant women with varying degrees of hypertension severity to evaluate and validate the suggested technique.

The study [91] investigated 26525 adult cancer patients. The author's objective is to forecast six-month mortality. The authors' approaches include random forest, gradient boosting, and logistic regression. The gradient boosting technique has also been used in real-time to categorize patients as being at high risk. The outcomes of this real-world application met the practitioners' expectations.

The study [82] aimed to evaluate the performance of the six best classifiers for analyzing the Autism Spectrum Disorder (ASD) screening training dataset. The study found that J48 produced promising results compared to other classifiers when tested in both circumstances, with and without missing values, and could assist health practitioners in making accurate diagnoses of ASD occurrences in patients. The study also addressed the issue of missing values in the dataset through an imputation method, where missing values were replaced with the mean of the available records in the dataset. The study's outcome may help health practitioners predict the occurrence of ASD more accurately using machine learning algorithms.

The author in [107] discusses using machine learning algorithms for binary classification in decision support systems to improve operations and reduce costs. The study compares the performance of different algorithms using the Scikit-learn machine learning library for Python and evaluates them on public diabetes and human resource dataset. The best-performing algorithm for supervised learning was Random Forest, while Balanced Iterative Reducing and Clustering Using Hierarchies and Spectral Clustering algorithms performed best for unsupervised clustering. The study suggests that applying unsupervised clustering as a preprocessing step can boost performance in supervised techniques.

The author of [112] proposes an architecture for tracking the patient's hand motions. Fog and cloud gateways for real-time response generation are employed for frequent monitoring of arthritis sufferers. The suggested architecture includes a thread protocol and a Bayesian network classifier to achieve reliable communication and anomaly detection, respectively. A dataset of 431 arthritis patients is collected in real-time and simulated using the OMNet++ simulator. In comparison to not employing the fog and thread protocol, observations reveal that the packet delivery ratio is increased by 15-20%, the response time is lowered by 20-30%, and the packet delivery rate is enhanced by 25-35%.

The authors of [115] attempt to predict postoperative sepsis and acute kidney injury. This topic is approached using a variety of models, including logistic regression, generalized additive models, naive Bayesian, closest shrunken centroid, and support vector machines. The area under the receiver operating characteristic curve, accuracy, and positive projected value is used to compare the various models. It is discovered that logistic regression, generalized additive models, and support vector machines outperform the naïve Bayesian model, with AUC scores as high as 0.858 for acute kidney injury and as high as 0.909 for severe sepsis.

The authors of [124] employ logistic regression and random forest to predict the progression of inflammatory bowel disease. Their technique was validated on a dataset of over 20000 patients.

The random forest produces the best performance, with an area under the receiver operating characteristic curve of 0.85. The authors state that this model can help practitioners differentiate between patients who are at high and low risk of a disease flare, which can help them tailor their therapy.

In study [126] the author discusses the use of machine learning algorithms to classify and discriminate between malignant and benign breast cancer cases. Three algorithms, k-NN, Naive Bayes, and Support Vector Machine, were used to analyze and classify data from the Wisconsin breast cancer database. The accuracy of the classifiers was evaluated using train/test split and cross-validation techniques, with k-NN providing the highest accuracy at 97.07%. The results show the potential for machine learning in improving predictive performance in breast cancer classification.

The authors of [127] conducted research on 378256 patients utilizing clinical data collection. The study's purpose is to evaluate machine learning algorithms for detecting cardiac failure. The authors contrast four machine learning approaches: random forest, logistic regression, gradient boosting, and neural networks. Furthermore, they compare the outcomes of those approaches to the results of a commonly used algorithm in the United States. A neural network produced the best results. When compared to the accepted method, 355 more people experienced cardiac failures in this situation.

So far we have presented an overview of the literature review which represents the use of machine-learning algorithms to solve healthcare problems. Now we present some insights we gained from this literature review. Table 2.3 represents the main topic, number of cases, variables, targets, and classes in a given reference article. From this table, we can say that machine learning is used in a wide range of applications in healthcare such as predicting diabetes, cancer, cardiovascular diseases, and so on. We also noted that all studies mentioned here use the real data set for their analysis except 3 studies. In addition, 3 studies use data sets from 2 different sources. Furthermore, the size of the data sets used ranges from 99 cases to 378256 cases, with 18 studies having a data set including less than 1000 cases. and 5 studies use a data set with more than 10000 cases and 3 variables up to 559 variables for different studies. Also, the focus of most of these studies is to predict a single target variable (mostly binary target except 3 multi-target studies) except for 1 study which predicts 2 target variables and 3 studies did not mention the number of target variables.

As a reminder in this thesis, we work with a real data set of about 1810 patients obtained from the hospital of Lille, France. In our work, we have 12 target risk factors for falls we want to evaluate. Our data set initially includes 445 columns including redundancy and various details. we finally selected 45 variables in the first iteration and 90 variables in the second iteration. It is also important to keep in mind that when evaluating a given target risk factor the other 11 targets are considered features of the patient as they represent important information about the patient. Another point has to be noted that the prediction is usually achieved with the assumption that all other variables are observed. We relax this strong assumption in this work, (this point is explained in the next chapter). In the next chapter, we will discuss in detail the process and reason for the selection of these target variables.

In this section, we provided an overview of the machine-learning algorithms used to solve healthcare challenges. In the next section, we discuss some of the limitations and unresolved issues when using machine learning models in healthcare.

Table 2.3: Analysis of references about classification in healthcare: represents the main topic, number of cases, variables, targets, and classes in a given reference article

Reference	main topic	data (real?)	#cases	#variables	#targets	#classes
[2]	predict diabetes	yes (2 data)	768 (99)	8 (3)		
[3]	predict risk factor for stroke		5110	12	1	2
[7]	predict heart disease		394	14	1	
[8]	predict covid19 admission in ICU	yes	100		1	2
[10]	Brest cancer risk prediction	yes	699	11	1	2
[32]	predict kidney disease	yes	400	26	1	2
[38]	predict mortality in hospital	yes	38173	10	1	2
[46]	predict mortality in ICU	yes	1999	21	1	2
[48]	predict kidney disease	yes	400	24	1	2
[51]	covid prediction	yes	992	11	1	3
[52]	hyperlactatemia prediction	yes	741	7	1	2
[54]	diabetes prediction		768	8	1	2
[62]	predict kidney disease	yes	400	14	1	2
[66]	predict lung cancer	yes (2 data)	80672 (53452)	13	1	2
[67]	predict cardiovascular disease	yes	303	13	1	5
[74]	predict artery disease	yes	303	13	1	2
[81]	predict readmission in COPD patients	yes	195	32	1	2
[84]	preeclampsia prediction	yes	164	11	1	4
[91]	predict mortality among cancer patients	yes	26525	559	1	
[82]	predict autistic spectrum disorder	yes	292	21	1	2
[107]	Evaluation of decision support system	yes (2 data)	768 (14999)	8 (9)	1	2
[112]	arthritis analysis	yes	431			
[115]	predict sepsis and kidney injury		50318	60	2	
[124]	predict hospitalization in IBD patient	yes	20368			
[126]	predict breast cancer	yes	699		1	2
[127]	cardiovascular risk prediction	yes	378256	30		

2.6 Limitations and open challenges

In this chapter, we have seen that increased data availability in healthcare has opened up a variety of new opportunities. Several machine-learning methods have been used to solve a wide range of issues. These algorithms can provide valuable insights into healthcare by extracting patterns and predicting outcomes from real data sets. However, the quality of the output from these models depends heavily on the quality of the input data, and healthcare data sets often suffer from issues such as imbalanced data, missing values, and too many variables. Imbalanced data occurs when the distribution of the classes of interest is not equal, which can lead to models that are biased toward the majority class. Missing values are another common issue in healthcare data, which can occur due to incomplete data collection, data entry errors, or patients not reporting certain information. Finally, selecting the appropriate variables or features to use in a machine learning model can be challenging due to a large number of potential predictors, some of which may be irrelevant or even harmful to the model's performance.

In the previous section, we presented an overview of the literature review representing the use of different machine learning algorithms in healthcare problems. Now we present the insights from these studies regarding the challenges mentioned above. Table 2.4 represents the list of reference articles dealing with missing values, imbalanced data, and variables selection. We see from this table that out of 26 articles, 14 of them mentioned the problem of missing value in the data, in which missing values were present in 10 of them. Furthermore, only 5 out of 26 articles mentioned the problem of imbalanced data, and only 9 out of 26 mentioned the variable selection procedures.

Furthermore, there are several other challenges that need to be addressed in order to ensure the accuracy and reliability of these algorithms. One such challenge is data limitations, where the quality or quantity of data available for analysis is limited. This can lead to reduced accuracy and limited generalizability of the algorithm. Multiple studies have highlighted this problem in their research, for example, the author in [33] emphasizes the importance of technological advancement as well as a dedication to open science in order to fully achieve the promise of machine learning in healthcare. The author in [40] mentioned the quality of data as a critical concern in healthcare. It also states that, in order for data to be valuable, it must be of high quality, and so it must be properly kept and retrieved. They also emphasize data preprocessing as a critical issue, claiming that machine learning algorithms in healthcare lag behind those in other fields due to a lack of consistent and trustworthy data management in hospitals.

Also, interpretability is a major challenge, as many machine learning algorithms are considered "black boxes" and it is difficult to understand how the algorithm arrived at its predictions. This lack of interpretability makes it difficult to trust and apply the algorithm's output in clinical practice. Multiple studies have highlighted it in their research, for example, the authors of [114] recognize the benefits of machine learning in healthcare, particularly in identifying sickness patterns but underline the significance of taking into account variables such as patient trust, transparency of the methods utilized, and potential bias by algorithms. Also, the author in [122] emphasizes one key problem with machine learning models: they are frequently complicated and nonlinear, making them difficult to examine and explain. This is a critical challenge, particularly in healthcare, and it may limit the use of machine learning models in practice. The authors advocate for the use of data and model visualization, as well as the incorporation of healthcare practitioners' expertise into the creation of data-driven procedures.

As a reminder, in this thesis, we deal with all the challenges mentioned above: missing values, imbalanced data, variable selection, data quality, and interoperability, in the context of the evaluation of risk factors for falls in elderly patients. Also, using the right measure is a really important point in the evaluation of the quality of a classifier since all measures are not

Table 2.4: Analysis of references about classification in healthcare: represents if a given reference article deals with missing values, imbalanced data, and variables selection

Reference	**missing data**	**imbalance**	**variable selection**
[2]	yes		yes
[3]	yes	yes	yes
[7]			yes
[8]			yes
[10]			
[32]	yes		yes
[38]	no		
[46]	no	yes	
[48]	yes		
[51]			
[52]	no		
[54]			
[62]	no		yes
[66]	yes	yes	
[67]	yes	yes	
[74]		yes	yes
[81]			
[84]			
[91]			
[82]	yes		
[107]			yes
[112]			
[115]	yes		yes
[124]	yes		
[126]			
[127]	yes		
#yes	10	5	9
#no	4		

equivalent and some measures can be completely inadequate such as accuracy when dealing with imbalanced data. However, regarding the data limitation problem, we are aware that a larger data set would be interesting but in this work, we use a limited data set with 1810 cases. Moreover, a larger data collection is currently ongoing (thanks to the PREMOB project, lead by Lille's hospital) Furthermore, to address the problem of interpretability we also work with experts in order to help in our analysis.

2.7 Conclusion

The use of machine learning algorithms in healthcare is an area of growing interest and has the potential to improve the quality and efficiency of healthcare services. However, the success of these algorithms heavily relies on the quality of data used for training, as well as the ability to handle and process the data effectively. This chapter has identified several challenges (missing values, imbalanced data, variable selection, data quality, and interoperability) that need to be addressed to ensure the accuracy and effectiveness of machine learning algorithms in healthcare.

Furthermore, machine learning algorithms can be used as a way to aid in early illness identification, patient care, and community services as the amount of data in healthcare grow. In our work, we focus on the problem of the prevention of falls. In this context, machine learning algorithms can be used to detect health-related risks in patients, which can aid in the evaluation of risk factors for falls. With that aim, in this chapter, we first briefly described the problem of fall prevention followed by the basics of machine learning. But, building a good machine learning model requires a good data set. Furthermore, working with data is more difficult than it may appear since it necessitates, first of all, careful handling of the data. In that aim, we presented the basics of data preparation more specifically about handling missing data, the problem of imbalance in data, and the selection of relevant variables to build a good machine learning model. We also presented an overview of a literature review that shows the use of different machine learning algorithms in the healthcare domain and the challenges faced by researchers.

Despite these challenges, there is a growing need for the development of accurate and reliable predictive models for fall risk assessment. With the rapid growth in healthcare data, machine learning has the potential to revolutionize the way falls are prevented through a facilitated evaluation of modifiable risk factors, even when available data on the patient is very partial. As a result, it is important for future research to continue exploring methods for addressing these challenges, improving the interpretability of results, and ensuring the clinical applicability of the models.

In conclusion, while machine learning has the potential to provide valuable insights for fall risk assessment, it is important to approach the development and implementation of these models with caution. By acknowledging the challenges and limitations of machine learning algorithms and taking steps to address them, we can ensure that the predictive models are reliable and clinically applicable, ultimately leading to improved patient outcomes.

Chapter 3

Real data set

Outline of the current chapter

3.1 Introduction

Oxford Dictionary defines a data-set[1] as "a collection of data that is treated as a single unit by a computer". This means that a dataset contains a lot of separate pieces of data but can be used to train an algorithm with the goal of finding predictable patterns inside the whole dataset.

Data is an essential component of an AI or ML model and, one of the main reasons for the spike in the popularity of machine learning that we witness today. It can come in many forms. Machine learning models rely on certain data types such as numerical data, categorical data, time series data, text data, and so on[2].

[1] https://www.oxfordlearnersdictionaries.com/definition/english/data-set
[2] https://www.datarobot.com/blog/the-importance-of-machine-learning-data/

- **Numerical data**, also known as quantitative data, is a type of information that can be measured and expressed in numbers, such as height, weight, or cost. To identify numerical data, one can apply mathematical operations like averaging or sorting. Numerical data can be classified as discrete, which are whole or exact numbers like the number of students in a class, or continuous, which are numbers falling within a certain range, such as interest rates. It is important to note that numerical data is not time-specific but represents raw figures.

- **Categorical data** is organized based on distinct features, such as gender, social class, ethnicity, hometown, or other similar labels. It is important to note that categorical data is non-numerical, which means that it cannot be summed up, averaged or arranged in chronological order. Categorical data is useful for grouping individuals or ideas that share common characteristics, enabling machine learning models to analyze data in a more streamlined manner.

- **Time series data** is comprised of data points that are indexed at different time intervals, usually collected regularly. Utilizing and understanding time series data enables us to compare data points from different time periods, such as weeks, months, or years. Unlike numerical data, time series data is characterized by having specific starting and ending points, which allows for a more meaningful analysis of the data over time.

- **Text data** refers to written language, consisting of words, sentences, or paragraphs, which can provide valuable insights for machine learning models. However, because these words are often challenging for models to understand on their own, they are typically analyzed through techniques such as text classification, sentiment analysis, or word frequency analysis. These methods enable the text data to be processed and grouped, allowing for a more accurate interpretation and utilization of the information.

Moreover, the sources for collecting a dataset vary and strongly depend on the domain we are working on. Broadly speaking, there are four main sources of data: real-world usage data, survey data, public data sets, and simulated data [15, 49].

- **Real-World Usage Data:** When AI products are already in the market, real-world data from actual consumers might be a valuable resource. We may look at searches, total results, which results users click on, and what they look at and purchase, for example, using a search engine or search function. Social media platforms can collect information about what users publish, like, share, and comment on. Smartphones, in-car systems, and home assistants with speech recognition capabilities can capture spoken requests and computer answers. There is additional data broadcast from music providers and websites such as YouTube that may track what users look at. We know that utilizing actual data correctly represents how people use the system, and we don't have to spend to produce it. However, there are legal questions associated with collecting it, as well as privacy concerns.

- **Survey Data:** The second source for machine learning data is surveys. We go directly to the users or prospective users, and ask what they like or don't like, and what we can improve about the product. This approach gives data from actual users and gets around privacy concerns and legal issues as, by taking the survey, people are opting to participate. Surveys provide context and the opportunity to follow up on anything that's unclear. We also have some control over what people say and do in that we can direct them to the specific topics we want to address. On the other hand, survey data is somewhat unreliable, because what people say they do and want on the survey might be quite different than what they actually

do and want. Additionally, survey data is often skewed toward dissatisfied users, as people who get what they want are less motivated to provide feedback.

- **Public Data Sets:** There are a number of different types of public data sets available from search engines, social media, Amazon Web Services, Wikipedia, universities, data science communities, and other data repositories. There's also an enormous amount of public data from academic efforts in speech and language processing from the last 40 years, licensable from various organizations. For most commercial purposes, affordability is the real advantage of these data sets. This kind of data is often used for applications like basic language recognition or machine translation.

- **Engineered or Collected Data:** The fourth main way to collect quality data is to make it ourselves. This is often the only way to proceed with a new solution when there aren't any users or usage data yet. We can simulate the user experience by hiring speakers and professionals, and gathering and annotating the data for project-specific needs. We can mimic the conditions where people will use the product like driving in a car on a city street, etc. On one hand, we can get exactly what we need faster this way because we are in control and always know the context. We can follow up with the professionals and speakers if there's a question. And, since we are not using real data, there are no legal or privacy concerns. Most importantly, the model will produce a better end result. On the other hand, this type of data collection will require a larger investment.

In addition, high quality is an essential thing to take into consideration when collecting a dataset for a machine learning project. It is as important as quantity even if we have implemented great algorithms for machine learning models.

Acquiring a high-quality dataset is an essential prerequisite for constructing real-world AI/ML applications. However, working with real-world datasets can be challenging as they are often complex, unstructured, and difficult to work with. The performance of any Machine Learning or Deep Learning model is directly linked to the quantity, quality, and relevance of the dataset, and striking the right balance can be difficult. Fortunately, there is a wealth of open-source datasets available that has encouraged researchers and the AI community to undertake state-of-the-art research and develop AI-enabled products. Nonetheless, even with the abundance of datasets, there are still significant challenges in addressing new problem statements. These include[3]:

- Insufficient Data: A lack of large samples of data points required by Machine Learning algorithms can limit the accuracy and effectiveness of the model.

- Bias and Human Error: The tools used for data collection can often result in human error or bias towards specific aspects, potentially skewing the model's results.

- Quality: Real-world datasets are frequently disorganized and complex, making it difficult to ensure high data quality.

- Privacy and Compliance: Some sources do not share their data due to privacy and compliance regulations, such as medical or national security data.

- Data Annotation Process: Manually labeling datasets for quality can be a time-consuming and expensive process that is prone to errors due to human intervention.

[3] https://www.datatobiz.com/blog/datasets-in-machine-learning/

Addressing these challenges is crucial for data scientists to produce robust and effective AI applications. It is important to note that these challenges stem from the inherent complexities of working with real-world datasets and the tools used for data collection. As such, it is necessary to develop solutions that can mitigate these challenges and produce high-quality datasets that can be used to develop powerful AI/ML models.

Furthermore, sufficient volumes of data allow us to analyze the trends and hidden patterns and make decisions based on the dataset [69]. However, while it may look rather simple, working with data is more complicated since it requires, first of all, proper treatment of the data you have, from the purposes of using a dataset to the preparation of the raw data for it to be actually usable [26]. As per The State of Data Science 2020 report[4], data preparation and analysis is a crucial and time-consuming task in the Machine Learning project lifecycle. It reveals that a significant portion of a Data Scientist's or an AI developer's time is spent on data analysis, accounting for around 70% of their work time. The remaining 30% of their time is allocated to other essential activities such as model selection, training, testing, and deployment.

In this work, we focus on using the real-world data collected at the service of fall prevention, hospital of Lille, France. This data contains numerical variables as well as some other variables, including text data. The size of the initial data set provided by the hospital is very small which includes information about 440 variables for 1810 patients who visited the service between January 2005 and December 2016. Also, there are many missing values present in the data maybe some of them due to human error. Furthermore, the privacy of the person who took part in this study is preserved as the information used in this study is anonymous.

In that aim, in this chapter, we provide the description of the data in section 3.2. Section 3.3 is focused on the ontology of risk factors for fall prevention, which was developed with the help of experts from the same service of fall prevention. Finally, in section 3.4 we describe the different steps of data preprocessing.

3.2 Subjects and data

In this section, we will first describe the data source from where we get the data. Also, the procedure for data collection. Furthermore, we will discuss the characteristics of the initial dataset.

3.2.1 Data source and collection

The data for this study were collected at a specialized service for fall prevention. This service is part of the Lille University Hospital Center[5] which is one of the largest public health establishments in Northern Europe. It provides geriatric care in three hospitalization sectors (acute medicine, follow-up care, and rehabilitation and long-term care) and two complementary units in rehabilitation and radiology [6]. It is lead by Prof. François PUISIEUX.

On the day of consultation in the service for fall prevention, the patients are admitted for a full day, during which they interact with various medical professionals who each look into a variety of factors, including past falls, diet, physical activity, and medical tests like a balance test. The patient's information is recorded at each step. The patient's case file is then discussed among a group of experts on the subject of elderly falls, and they discuss to summarize the

[4] https://www.anaconda.com/state-of-data-science-2020?utm_medium=blog&utm_source=anaconda&utm_campaign=sods-2020&utm_content=data-prep-blog

[5] https://www.chu-lille.fr/

[6] https://fr.wikipedia.org/wiki/Centre_hospitalier_universitaire_de_Lille

patient's risk factors for falls identified during the day. They also partially sort the list of risk factors regarding their importance. Based on that, they discuss the best course of action for the patient. At the end of the day, one of the experts receives the patient and explains to them a few suitable recommendations. The patient is invited to return to the hospital six months later for a brief consultation so that the recommendations and the number of falls over the previous six months can be evaluated. This data is first collected on the paper file of the patient and later copied onto the excel file that was given to us for analysis.

3.2.2 Data description

The data for this study includes 1810 patients who visited the service between January 2005 and December 2016, of which 28% of them are male and 72% are female. The age of patients ranges from 51 years old to 100 years old, with an average age of 81 years old. Table 3.1 represents the description of variables present in the initial data set based on their category. In total, we have a total number of 440 variables present in the initial file.

Category	Number of columns	Description / explanation
Empty column	2	
Grayed column	59	During the 12 years of use of the file, some columns were copied, modified, and grayed
Error value	3	Error value due to a formula with a problem
Recommendation	17	
Collected after 6 months	36	Collected during the second appointment 6 months after the hospital day
Value is a free text	41	Often provide text detail related with a previous column
Administrative information	4	
Personal characteristics	9	Examples are age, sex, size, weight, body mass index, ...
Disease and health problem	78	Disease, symptom, disorder, history of health problem (except fractures), incapacity
Fracture	11	Different kind of fractures
Behavior	19	Activity of daily living
Medicine	100	Medicine, drugs associations
Medical test	41	Gait and balance test, cognitive test, blood pressure, heart frequency, etc.
Autonomy	10	Autonomy in the activity of daily living
Environmental variables	5	
Number of falls	5	Number of falls during the last six months
Total number of columns	**440**	

Table 3.1: Description of 440 variables present in the initial data set based on their category.

Among those 440 variables, some variables are duplicated, or partially duplicated. In some

cases, this is due to the evolution of the file over more than 10 years (for instance, the textual domain of a variable is replaced by numerical values in a new column). In other cases, the same piece of information is deliberately collected several times (for instance regarding orthostatic hypotension, since it is not always visible, achieving several texts is the right way to detect it). The 440 variables of the initial file are not fully listed in this manuscript, but the two subsets of variables that we used in this study are completely described in section 3.4.2.

It should be noted that no variable reports the risk of falls, even if that information is sometimes mentioned in the paper report written at the end of the first appointment. When information is available about the evaluated risk of fall of the patient, it is usually qualified as "faible", "modéré" or "élevé", even if many other words can be used.

Out of a total of 1810 patients, the paper reports of 530 patients were manually analyzed, and information about the risk of falls was found in 330 cases. For the 200 remaining patients, the report was not found or in progress for 19 cases and no information about the risk of falls was found in the report for 181 cases. Among those 330 cases with information about the risk of falls, we have:

- 2% with a low risk of fall ("faible" or equivalent),
- 46% with a moderate risk of fall ("modéré" or equivalent),
- 52% with a high risk of fall ("élevé" or equivalent).

It is important to remark that this distribution does not represent the (french) population of elderly people (over 65). Indeed, the population represented by this data set includes only people who had an appointment in the service of fall prevention at Lille's hospital, naturally having an important proportion of persons with a high risk of falls. Thus, the use of a data set representing a very specific population has to be kept in mind when analyzing the results, and the evaluation of risk factors for falls in the general population of elderly people should be adapted to take into account the true distributions. However, the benefit of such a data set is to concentrate on targeted people, meaning those with a high risk of falls, and thus offers a very detailed analysis of that population.

We now present the ontology of risk factors for falls that was intensively used to understand the variable and build subsets of variables used in this work.

3.3 Ontology about risk factors for falls

With the aim of developing an application for fall prevention, it is necessary for both medical experts and the computer science team to understand each other. In that aim, an ontology of fall prevention fall has been developed[7], allowing a better understanding of the multiple factors that can cause falls in the elderly [19]. Those risk factors have their origins in the aging process, but they are also impacted by a person's behavior, habits, and surroundings [96]. By addressing all of these risk factors, falls can be prevented. The goal of such a system is to enable proper and continuous follow-up of the individual through a pedagogical and instructive approach, which includes evaluation of risk factors for falls and giving adopted recommendations. Following that context, in this section, we describe the methodology used to design an ontology for fall prevention followed by the resulting ontology for the risk of falling.

[7] The work of this section was previously done and published in [30], and it is not part of this thesis.

3.3.1 The Ontology Design Methodology

There are several approaches for creating an ontology, for example, [6, 121]. The creation of the ontology was based on methods in [90, 121], which included four steps: defining the purpose, conceptualization, formalization, and validation. The schematic picture for the definition of ontology is shown in figure 3.1.

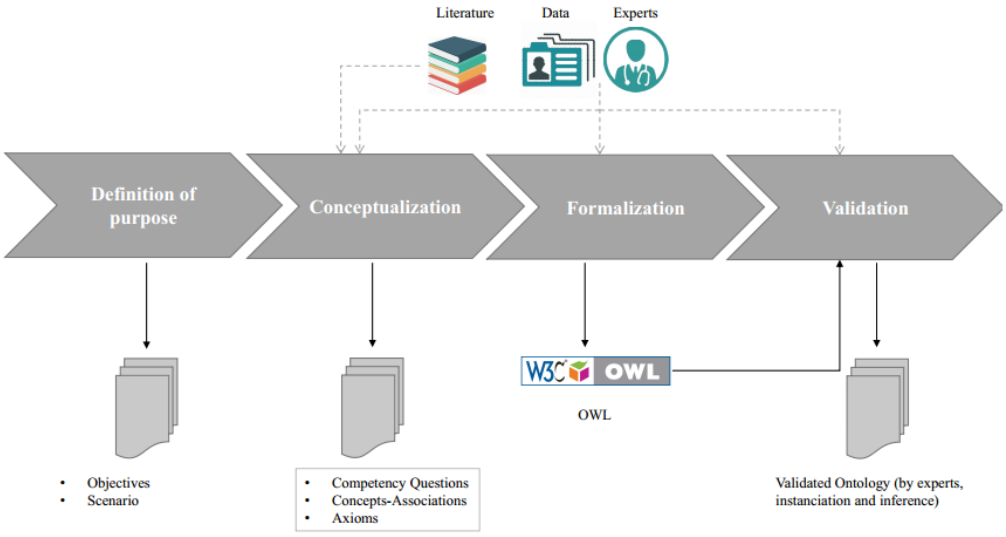


Figure 3.1: The ontology definition process

The purpose of this ontology is to support the evaluation of risk factors of falls in the elderly in order to prevent falls. This ontology is the basis for the development of the fall prevention software system. The conceptualization requires the definition of the ontology's scope, concepts, relations, and constraints, and a description of a glossary for all concepts and attributes specified. This ontology serves as the foundation for the creation of the fall prevention software system. It symbolizes the modeling of knowledge itself. To begin, we followed the advice of ontology methodology [73, 90], and set competence questions (i.e., requirements in the form of questions that the ontology must answer) as follows:

1. What are the important characteristics to be observed for a person at risk of falling?
2. What are the falling risk factors?
3. What are the appropriate recommendations for fall prevention?

Once we identify the traits of a person that are suggested in risk factors recognized by physicians, the first two competency questions become intimately interrelated. Based on the characteristics of the elderly and the risk factors he/she has, physicians define specific recommendations. Since the last part (providing recommendations) is out of scope for this thesis, we focus on the first 2 parts. We organized the ontology which focuses on the elderly person at risk of falling (to address the first two competency questions, presented above). We present the resulting ontology in section 3.3.2. Furthermore, formalization entailed creating the ontology

using the W3C Web Ontology Language (OWL)[8]. Finally, the ontology was validated using physicians' analysis and by instantiating it with real cases from the hospital unit's historical record.

3.3.2 Ontology about Elderly Person at Risk of Falling

The fall of the elderly is a multifaceted condition, according to physicians and literature [13, 19, 98]. When attempting to comprehend the factors that contribute to a person's risk of falling, it is vital to understand several aspects of their disease, the medicine they take, their daily activities, and the environment in which they lives. All of these variables that characterize a person show which risk factors exist for that individual and may be refined based on the possible risk it may cause. To that end, the ontology depicted in Figure 3.2 has been developed, which focuses on the older person at risk of falling.

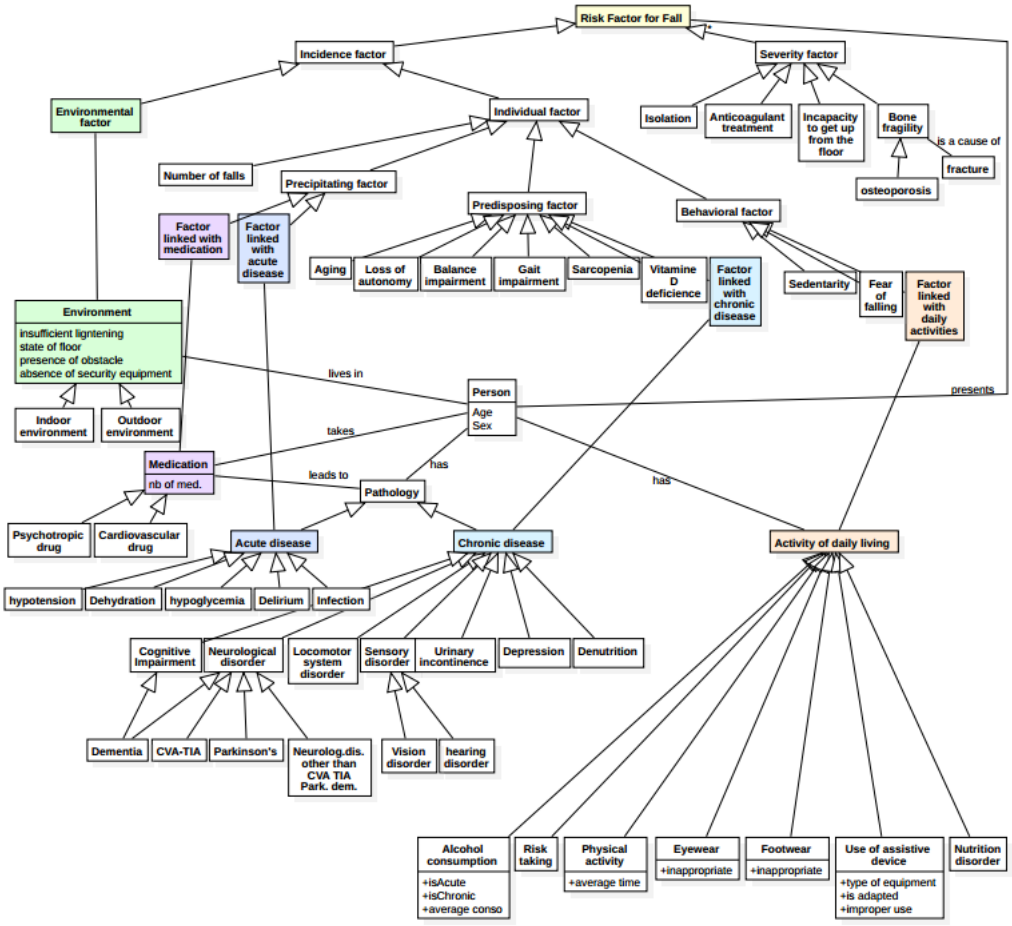


Figure 3.2: The ontology about the elderly person at risk of falling

The fundamental features of a person (age and sex) are relevant for analyzing risk factors in all of the aspects shown in figure 3.2. Even if these characteristics are not assessed in an

[8] https://www.w3.org/OWL/

isolated way, they might have an impact on the evaluation. Women, for example, are more prone than males to fall [117]. Furthermore, the combined effects of aging and age-related diseases increase the risk of falls [13, 116]. Some chronic or acute diseases are known to increase the risk of falling in the elderly [109]. In addition, according to [13], different types of chronic disease (cognitive impairment and dementia, locomotor system disorder, sensory disorder, neurological disorder, musculoskeletal affections, neurosensory disorder, neurological condition, dementia, urinary incontinence, depression, denutrition) and acute disease (hypotension, dehydration, hypoglycemia, delirium, infection confusional state) are important factors that influence the risk of falls. Furthermore, the elderly frequently have many diseases, necessitating the usage of multiple medications. Psychotropic medications (such as antidepressants, sedative-hypnotics, tranquilizers, and neuroleptics) and cardiovascular medications (for instance, antiarrhythmic drugs, nitrates, and diuretics) are the two types of drugs that are more specifically connected with an increased risk of falls [13, 71, 72, 94].

In addition to the medicine taken, the activities of daily life and environmental features must be considered when assessing the risk of falling. The literature (INEPS[9], 2017) and the hospital's historical database allow for the identification of the following factors connected to activities of daily living:

- Alcohol consumption - This includes both excessive alcohol consumption over a short period of time and chronic alcohol consumption with the associated mean level of consumption.
- Wearing inappropriate shoes - The question of the type of shoes is quite well documented (for instance the use of rigid-soled shoes, closed).
- Physical activity - Regular physical activity delays the onset of major chronic diseases. In particular, it promotes mobility, which can help to reduce falls. Insufficient physical activities lead progressively to sarcopenia. Daily thirty minutes of physical activity is recommended.
- Nutrition disorders - The lack of food is a cause of muscular weakness, which may cause gait and balance disorders.
- Risk taking - Some behaviors such as hurrying, or climbing on a chair contribute to increasing the risk of fall. Wearing inappropriate glasses may impact balance;
- Use of auxiliary equipment - The use of a walking stick or a walking frame may become a source of risk either because of bad use of the equipment or because the equipment is not well adapted to the person.

In addition, various research provided in INEPS[10](2005) has demonstrated that the great majority of older people's houses constitute environmental hazards. The involvement of the following elements is noted in the literature: inadequate illumination; soil condition; the presence of barriers; and lack of safety equipment (for example, handrail, grab bar).

Furthermore, severity factors (or gravity factors) influence the consequences of falls and the severity level of related injuries and complications. According to our geriatric experts, the most important ones are:

- Bone fragility - augment the risk of fracture when falling

[9] https://www.ineps.fr/
[10] https://www.ineps.fr/

- Incapacity to get up from the floor - augment the risk to keep a long time on the floor, especially in case of isolation.

- Anticoagulant treatment - augment the risk of bleeding following a fall, especially in case of prolonged stance on the ground.

- Living alone (isolation) - augment the risk of keeping a long time on the floor when the person cannot get up from the floor.

Incidence factors are more important in lowering the risk of falling. In reality, most preventative measures rely on external factors. These variables do not cause falls systematically, but their presence in an aged person with limited physical capacity might generate a risk. Identifying these characteristics helps us to make required environmental changes and acquire new behaviors that can make life simpler and safer for the elderly. Advanced age adds to an increase in the risk of falling. It invariably causes physical, cognitive, and behavioral changes (including sensory, musculoskeletal, neurological, and metabolic changes). These changes are characterized as precipitating causes, which occur on a regular basis, such as acute pathologies, pharmaceutical effects, and so on, and predisposing factors, which are mostly connected to the impacts of aging and chronic diseases. In addition to the aforementioned considerations, the history of prior falls is regarded as one of the strongest predictors of a future fall: elderly people who have one or more falls may have decreased physical ability and experience fear. As a result, the senior's quality of life is reduced which led him to restrict his mobility, which favors the loss of muscular strength, balance, and reflexes. Following a fall, a person having one or more of these factors is likely to have more serious fall-related consequences, which may affect his balance and mobility, hence an increased risk of future falls.

This ontology of risk factors for falls constitutes a strong basis that for our understanding of the variables of the data set. In the following, we present the steps of data preprocessing that lead to the definition of the two subsets of our data set that we used in our analysis.

3.4 Data Preprocessing

Data preprocessing has a significant impact on the performance of machine learning models because unreliable samples may lead to wrong outputs [26, 69]. To perform a meaningful data preprocessing, either the domain expert should be integrated in the data analysis or the domain should be extensively studied before the data is preprocessed [39]. In this study we have used expert knowledge to provide a better understanding of data. Also, the understanding of the data was made easier by the help of an ontology about fall prevention [30] developed previously with the same service of fall prevention of Lille's Hospital as discussed in section 3.3. Moreover, we used an iterative approach for the pre-processing of data. It is divided into two iterations. The objective for the first (second) iteration is to select the minimum (maximum) number of variables from the initial data set in order to evaluate the risk factors for falls respectively. Here, in the first iteration, the goal was to provide a model with a reasonable size and in the second iteration to improve the results obtained by the 1st iteration and try to keep as many variables as possible. In this section, we first describe the steps taken for cleaning the data followed by the list of variables and the target selected. Furthermore, we describe the steps for the imputation of missing values in our data.

3.4.1 Data cleaning and variable definition

Detecting and repairing dirty data is one of the perennial challenges in research, and failure to do so can result in unreliable decisions [43]. The data can have many irrelevant and incomplete variables with missing information. Cleaning is required to get understandable information from this kind of data [39]. In our study, we also need some cleaning of the data prior to building any network.

Furthermore, data forms the foundation of any machine learning algorithm. The size of a data set consists of: number of cases, number of variables and size of the domain of each variable. The data can contain a huge number of variables, some of which are not even required. Such redundant information makes modeling complicated. Thus reduction in the number of variables when using machine learning algorithms is very important. We will discuss about how we reduced the number of variables in the following section. Furthermore, the size of domain of a variable is also an important factor when using machine learning algorithms. In study [70], the author shows that the model trained on sparse data performed poorly in the test dataset. In other words, the model during the training learns noise and they are not able to generalize well. Hence they overfit. Thus the size of the domain of a variable is also an important factor when using machine learning algorithms. We discuss how we discretize the continuous variables and variables with large domain sizes in the following section.

Furthermore, in our case, we have a data set that consists of information about 1810 patients. We used the following 2 criteria to include a person in this study:

- the person should be 65 years or older.
- the person should be able to walk.

After this cleaning, in total, we have information for 1745 patients.

Now in the following sections, we present the data cleaning steps that are common to iterations 1 and 2, followed by the specific steps for each iteration of data cleaning as presented in Figure 3.3.

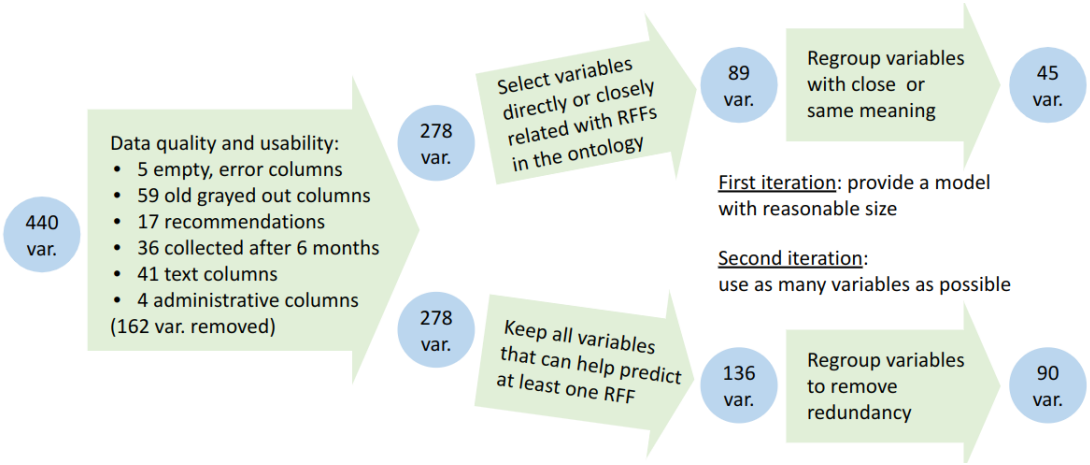


Figure 3.3: Data cleaning main steps: common steps (left); first iteration (upper right); second iteration (bottom right); here RFFs means risk factors for falls

3.4.1.1 Common steps of data cleaning

In order to identify the variables to be kept for our study, we first use the criterion of data quality to remove unusable variables, such as old ones that were replaced by new ones but still present in the data file, or columns with an error value due to a problem with a formula in the excel file. These variables correspond to the first categories of variables shown in Table 3.1. We also removed variables whose value is free text, since we consider they are both not usable and not interesting, as agreed with our experts. We also use a second criterion to remove variables that can not be used to evaluate some risk factors for falls. In that aim, we remove variables related to information that comes after the present time where we aim to evaluate the risk factors. It concerns the recommendations provided at the end of the day at the hospital, and all variables collected during the second appointment, six months later. We also removed administrative information such as the date of the meeting, since they are of no interest in the evaluation of any risk factors. Now we provide the detail of the number of variables removed following these two criteria.

Data quality - In order to have relevant and understandable observations, we removed:

- 2 empty columns
- 3 columns with error due to the excel formula
- 59 old grayed-out variables,
- 41 text columns

Focus on the characteristics related to risk factors of fall - We remove variables that are not useful for the evaluation of risk factors for fall:

- 17 variables associated with recommendations
- 36 variables associated with the second appointment after 6 months
- 4 administrative columns

(total 162 variables removed)

3.4.1.2 Data cleaning and variable definition for Iteration 1

After the removal of the variables guided by the above two criteria, we use a third criterion that is specific for the data cleaning of the first iteration and consist in providing a model with reasonable size. Indeed, the aim of the first iteration is to have a first view of the evaluation of the risk factors for falls, by using only the most important variables, instead of including all interesting parameters. In order to keep a small number of variables, we used ontology and we had discussions with our experts to identify the most important variables. In that aim, we first identified the variables of the data file directly or closely related to the main concepts of the ontology. We found one or several variables for most of the concepts present in the ontology. In the second step, for each concept of the ontology, we either selected one representative variable or regrouped several variables with the same or close one using the OR operator. This step was achieved thanks to an interview with our main expert (Pr. Puisieux).

Furthermore, the first column of Table 3.2 represents the concepts of the ontology that is represented in our data means concepts for which at least one variable was found in the initial data file provided by Lille's hospital. We rearrange these concepts into different categories based on their meaning. The second column provides the number of variables in the initial data file that directly or closely correspond to a concept. In that step, we did not consider variables

with more than 30% of missing values. Indeed, using such variables would require specific treatments that are not the objective of this first iteration. Moreover, this threshold was decided as a compromise between not adding too much noise and the maximum possible information for a given patient. The third column states the method used to define the final variable: When considering a subgroup of variables corresponding to a risk factor for fall in the ontology, we used one of the next two methods: either we select one variable from the subgroup that is a good representative of the subgroup (designated by « selection » method), or we define a new variable as a disjunction of a set of variables (designated by «logical OR operator » method). when only one variable was found for a given risk factor for falls, we kept that variable (the column method is left empty). The fourth column is the short name of the resulting variable, corresponding to the abbreviation of the french name of the variable.

The next paragraphs provide some explanations about Table 3.2.

The variable *demence* was defined with the following formula to manage a ternary variable ($dem3$) and a continuous variable (MMS): $demence = 1$ if $dem = 1$ or $dem3 = 1$ or $dem3 = 2$ or $demAP = 1$ or $MMS < 24$.

The variable "auTrNeur" regroups the following eight neurological disorders: neuropathy, ataxia, cerebellar ataxia, myopathy, proprioception disorder, lack of deep sensibility, cerebellar syndrom, and vestibular syndrome.

The loss of autonomy is represented by the index of the activity of Daily Living (ADL). In addition, we kept two other variables that are closely related to the loss of autonomy: the fact to drive his car ($conduit$) and the difficulty to use the toilet ($difWC$).

Regarding the concept *Incapacity to get up from the floor*, we kept two related variables in the initial data file: the person stayed on the ground for more than one hour after a fall ($gt1hSol$), and the person was able to get up from the floor after a fall ($aSuSeRel$).

Among the selected variables, three of them were non-binary variables, namely the age of the person, the body mass index, and the index of activity of Daily Living (ADL). We discretized them as follows: the variable $agegt80$ is 1 when the age is greater than 80 and 0 else; the variable $ADLinf5$ is 1 when the ADL index is less than 5, corresponding to a loss of autonomy, and 0 else; the variable $BMI4$ is discretized in four values: $BMI4 = 0$ if $BMI < 18.5$, $BMI4 = 1$ if $BMI \in [18.5, 25[$, $BMI4 = 2$ if $BMI \in [25, 30[$.

We also added the following variables after an interview with our expert in Lille's hospital: use of a diuretic, heart disease cardiopathy and cardiac arrhythmia ($cardiop$ and $arithm$), other important diseases such as pneumopathy ($pneumo$), diabetes ($diabete$) and High blood pressure (HTA), and additional behavioral factors such as smoking ($tabac$), the ability to drive his/her car ($conduit$) and the difficulty to use the toilet ($difWC$) which is closely related with the loss of autonomy. Also, cardiovascular diseases are associated with greater fall risk [63], we have two variables (cardiopathy $cardiop$ and cardiac arrhythmia $arithm$) associated with these diseases in our data.

Some of the concepts in the ontology of risk factors for falls are not represented in the data file. This is the case for the properties of the indoor and outdoor environment such as insufficient lightening, state of the floor, presence of obstacles, and the absence of security equipment. This is because the data collection is made during a day hospital that does not allow the collection of detailed data on the person's living space. One variable is however present in the initial data file stating the presence or not of any environmental risk factors. This is the same for some risk factors linked with the daily activity that is not directly represented in the data file (risk-taking and inappropriate footwear). Another concept present in the ontology is the use of the cardiovascular drug or anticoagulant treatment. Since these variables can not be directly identified in the initial data file, they do not belong to the set of selected variables. Most of the precipitating factors are not present in the initial data file (dehydration, hypoglycemia, delirium,

Concept of the ontology that is represented in the data	**NbV**	**Method**	**Variable name**
Personal characteristics and number of falls			
Sex	1		*sexe*
Age	1		*agegt*80
Body mass index	1		*BMI*4
Number of falls	3	Selection	*nbChu*2
Precipitating factors			
Psychotropic drug	1		*gt*1*psych*
Number of medicine	2	Selection	*nbMed*3
Orthostatic hypotension	3	OR	*newHypoT*
Psychotropic drug	4	Selection	*gt*1*psych*
Predisposing factors			
Loss of Autonomy	1		*ADLinf*5
Loss of Autonomy (closely related)	1		*difWC*
Loss of Autonomy (closely related)	1		*conduit*
Balance impairment	1		*trEq*
Balance impairment (closely related)	1		*apUniGt*5
Gait impairment	9	Selection	*trMar*
Gait impairment (closely related)	3	Selection	*GUGOgt*20
Sarcopenia	2	OR	*dfOuFaiM*
Dementia	4	OR specific	*demence*
CVA - TIA	1		*AVC* – *AIT*
Parkinson's disease	5	Selection	*parkOuSP*
Neurological disorder other than CVA, TIA, Parkinson's disease and dementia	8	OR	*auTrNeur*
Locomotor system disorder (instance of)	2	OR	*arthPoly*
Vision disorder	1		*trVision*
Hearing disorder	1		*trAudit*
Urinary Incontinence	1		*pathUro*
Depression	1		*dep*
Behavioral factors			
Fear of falling	1		*peurTom*
Fear of falling (closely related)	1		*evitSort*
Alcohol consumption	1		*alc*
Physical activity (closely related)	3	Selection	*sort*
Use of assistive device	2	Selection	*utiATM*
Severity factors			
Isolation	1		*vitSeul*
Isolation (closely related)	1		*maisRet*
Incapacity to get up from the floor	2	Selection	*aSuSeRel*
Incapacity to get up from the floor	1		*gt*1*hSol*
Fracture	2	OR	*fracturA*
Osteoporosis	2	Selection	*osteoConf*
Osteoporosis (closely related)	2	OR	*newTrOst*
Environmental factors			
Environmental factors	2	Selection	*factEnv*
Secondary risk factors			
Diuretic drug	2	Selection	*diuretiq*
Smoking	2	Selection	*tabac*
Heart disease	1		*cardiop*
Heart disease	1		*arithm*
Pneumopathy	1		*pneumo*
Diabete	1		*diabete*
High blood pressure	1		*HTA*

Table 3.2: 45 Variables selected for the first iteration based on the ontology. The second column (NbV) shows the number of variables in the initial data file that corresponds to a concept of the ontology

and infection). This can be explained by the short-term (or non-regular) nature of these factors, making them difficult to be anticipated. The other risk factors in the ontology that are not present in the initial are aging, vitamin D deficiency, and sedentarity.

Finally, as visible in Figure 3.3, we found 89 variables directly or closely related to the main risk factors for falls defined in the ontology. Among them, we regrouped 64 variables into 20 variables, by either selecting a variable or merging by using a disjunction of a subgroup. The final result of this cleaning leads to 45 variables.

3.4.1.3 Data cleaning and variable definition for Iteration 2

After completing the first iteration (data preprocessing, prediction of the targets, and evaluation), we started a second iteration with the aim to improve each step of the first iteration. This second iteration also consists of data cleaning and variables definition, which were achieved in three steps as for the first iteration (see Figure 3.3). The first step is common with the first iteration and led to 278 variables. We present as follows the other two steps: the manual selection of variables from the initial data set based on ontology about risk factors, and the grouping of variables to avoid redundancy. At the end of this cleaning, we have in total of 90 variables.

The manual selection of variables from the initial data set in the second iteration

In this second iteration, our goal is to improve the evaluation of the risk factors for falls, and thus we use as many variables as possible. For that aim, we conducted manual variable selection with the aid of our expert. Our guideline was to keep any variable that could help in the evaluation of at least one of the target variables. We started with 278 variables and removed 142 variables.

Most of the variables of the first iteration were kept for the second iteration except four of them: the variables related to environmental factors, smoking, cardiac arrhythmia, and pneumopathy. Regarding environmental factors, they are clearly listed among the main risk factors for falls, meaning that their presence increases the risk of falling. However, they are completely independent of all other risk factors, meaning that knowing the presence or the absence of environmental factors for a given person does not help to evaluate any other risk factors for falls for her. Regarding smoking, cardiac arrhythmia, and pneumopathy, which we considered secondary risk factors, they were finally dismissed by the expert.

Regrouping variables to remove redundancy in the second iteration

After this manual cleaning in the second iteration, in total, we have information about 136 variables, among which:

- 97 binary variables
- 6 tertiary variables
- 21 discrete variables with larger domain
- 12 continuous variables

To understand better we further divide these variables into four groups presented in Table 3.3.

Now, we describe the steps done in each group to remove redundancy and reduce the size of the domain of the variables.

Group A: subgroups of binary or ternary variables with same or close meaning

Group A includes 36 variables corresponding to 10 subgroups of binary or ternary variables with the same or close meaning. We describe the way how we combined them as follows. Table 3.4 presents the 10 resulting variables.

Group	Description	Number before grouping	Number after grouping
A	Subgroups of binary or ternary variables with same or close meaning	36	10
B	Subgroups of variables with same or close meaning including variables with a larger domain or continuous variables	29	12
C	Variables with large domain and continuous variables for which no other variable exists with a very close meaning	16	13
D	Binary variables for which no other var exists with a very close meaning	55	55
	Total	**136**	**90**

Table 3.3: Different groups of variables and their numbers before and after regrouping variables with the same or close meaning

- *sarcopen* - This subgroup consists of 2 binary variables: Lower limb strength deficit (defmmi) and lower limb muscle weakness (faibmmi). We combine variables in this group using the OR function as follows:

$$sarcopen = \begin{cases} 1, & \text{if defmmi = 1 OR faibmmi = 1} \\ 0, & \text{if defmmi = 0 AND faibmmi = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *hypotenO* - This subgroup consists of 3 binary variables: orthostatic hypotension in consultation (hypot) and symptomatic orthostatic hypotension during the consultation test (hypotenc) and known orthostatic hypotension before consultation (hypotavt). We combine variables in this group using the OR function as follows:

$$hypotenO = \begin{cases} 1, & \text{if hypot = 1 OR hypotenc = 1 OR hypotavt = 1} \\ 0, & \text{if hypot = 0 AND hypotenc = 0 AND hypotavt = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *antiArit* - This subgroup consists of 8 binary variables related to medicine taken by the patient: digoxin, Ia, Ib, Ic, II, III, cordarone, IV. We combine variables in this group using the OR function as follows:

$$antiArit = \begin{cases} 1, & \text{if digoxin = 1 OR Ia = 1 OR Ib = 1 OR Ic = 1 OR II = 1 OR III = 1} \\ & \text{OR cordarone = 1 OR IV = 1} \\ 0, & \text{if digoxin = 0 AND Ia = 0 AND Ib = 0 AND Ic = 0 AND II = 0} \\ & \text{AND III = 0 AND cordarone = 0 AND IV = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *osteopor* - This subgroup consists of 2 binary variables: osteoporosis reported history or BMD (osteo) and confirmed osteoporosis with BMD score < 2.5 (osteodmo). We combine

variables in this group using the OR function as follows:

$$osteopor = \begin{cases} 1, & \text{if osteo = 1 OR osteodmo = 1} \\ 0, & \text{if osteo = 0 AND osteodmo = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *aSuSeRel* - This subgroup consists of 2 binary variables (asuserel0 and asuserel1) with similar descriptions regarding whether a person could stand up on his own. We combine variables in this group using the OR function as follows:

$$aSuSeRel = \begin{cases} 1, & \text{if asuserel0 = 1 OR asuserel1 = 1} \\ 0, & \text{if asuserel0 = 0 AND asuserel1 = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *aidTecMa* - This subgroup consists of 3 binary variables (uniaid0, uniaid1, and uniaid2) with similar descriptions regarding if a person uses an assistive device while walking. We combine variables in this group using the OR function as follows:

$$aidTecMa = \begin{cases} 1, & \text{if uniaid0 = 1 OR uniaid1 = 1 OR uniaid2 = 1} \\ 0, & \text{if uniaid0 = 0 AND uniaid1 = 0 AND uniaid2 = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *demence* - This subgroup consists of 5 variables: dementia (dem)= yes or no; dementia (dem3)= yes likely or yes confirmed or no; dementia (demAP) = yes or no, with history or diagnosis made at the end of the consultation; 2 similar variable representing MMS less than 24 (mmslt24) = yes or no, discretized to a single binary variable. We combine variables in this group using the OR function as follows:

$$demence = \begin{cases} 1, & \text{if dem = 1 OR dem3 = 1 } or \text{ 2 OR demAP = 1 OR mmslt24 = 1} \\ 0, & \text{if dem = 0 AND dem3 = 0 AND demAP = 0 AND mmslt24 = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *fracture* - This subgroup consists of 2 binary variables: fracture when falling from height excluding spontaneous fractures (fracExS) and fracture when falling from height or spontaneous fractures (fracS). We combine variables in this group using the OR function as follows:

$$fracture = \begin{cases} 1, & \text{if fracExS = 1 OR fracS = 1} \\ 0, & \text{if fracExS = 0 AND fracS = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *parkOuSP* - This subgroup consists of 6 variables: parkinson disease = yes or no (parkmal2); parkinson or parkinsonian syndrome (parksydA)= yes or no; parkinson's syndrome (parksyd2) = yes or no; parkinson disease (parkmal3) = yes probably or yes confirmed or no; parkinson's syndrome (parksyd3) = yes probably or yes confirmed or no; parkinson or parkinsonian syndrome diagonosis made at the end of the consultation (parkousp)= yes or no. We combine variables in this group using the OR function as follows:

$$ParkOuSP = \begin{cases} 1, & \text{if parkmal2 = 1 OR parksydA = 1 OR parksyd2 = 1 OR} \\ & \text{parkmal3 = 1 } or \text{ 2 OR parksyd3 = 1 } or \text{ 2 OR parkousp = 1} \\ 0, & \text{if parkmal2 = 0 AND parksydA = 0 AND parksyd2 = 0 AND} \\ & \text{parkmal3 = 0 AND parksyd3 = 0 AND parkousp = 0} \\ NA, & \text{otherwise} \end{cases}$$

- *traAnOst* - This subgroup consists of 3 binary variables: biphosphonates (biphosp); osteoporosis treatment before consultation (tranost); and calcium / vitamine D (calvitD). We excluded biphosp variable because it is included in the definiiton of tranost. Also, we excluded calvitD following the recommendation of the domain expert.

Short name	Description	Details of variable involved
sarcopen	Sarcopenia	2 variables: defmmi, faibmmi
hypotenO	Orthostatic hypotension	3 variables: hypot, hypotenc, hypotavt
antiArit	Anti arrhythmic	8 variables: digoxin, Ia, Ib, Ic, II, III, cordarone, IV
osteopor	Osteoporosis	2 variables: osteo, osteodmo
aSuSeRel	was able to get up from floor on his own	2 variables: asuserel0, asuserel1
aidTecMa	use an assistive device while walking	3 variables: uniaid0, uniaid1, uniaid2
demence	Dementia	4 variables: dem, dem3, demAP, mmslt24
fracture	fracture when falling from height	2 variables: fracExS, fracS
parkOuSP	Parkinson disease	6 variables: parkmal2, parksydA, parksyd2, parkmal3, parksyd3, parkousp
traAnOst	anti-osteoporosis treatment	3 variables: biphosp, tranost, calvitD

Table 3.4: Summary of variables in group A

Group B: subgroups of variables with same or close meaning including variables with a larger domain or continuous variables

Group B includes 29 variables corresponding to 12 subgroups of variables with the same or close meaning including variables with a large domain or continuous variables. We describe the way how we combined these subgroups as follows. Table 3.5 presents the 12 variables obtained from group B.

- *nbmed3* - This subgroup consists of 2 variables: number of medications and class variable where
$$class = \begin{cases} 0 & if \text{ no. of medication } \in [0,4] \\ 1 & if \text{ no. of medication } \in [5,8] \\ 2 & if \text{ no. of medication } \geq 9 \end{cases}$$
we selected the class variable for our analysis because the domain size if this variable is smaller than the other variable and named it *nbmed3*.

- *nbchu2* - This subgroup consists of 5 variables related to the number of falls in the past 6 months with different discretization. we decided to use the variable with a domain size

equal to 2 and is given as follows:

$$nbchu3 = \begin{cases} 0 & \text{if no. of falls } \in [0,1] \\ 1 & \text{if no. of falls } \geq 2 \end{cases}$$

- *bmi_lt19* - This subgroup consists of 2 variables related to the Body mass index (BMI) of a given person with different discretization. We decided to use the variable with a domain size equal to 2 and is given as follows:

$$bmi\_lt19 = \begin{cases} 0 & \text{if BMI } \geq 19 \\ 1 & \text{if BMI } < 19 \end{cases}$$

- *apUniGt5* - This subgroup consists of 3 variables: time standing on one leg (left and right each) and binary variable if a person can stand on one leg (either left or right) more than 5 sec. We use the binary variable from this group and named it apUniGt5.

- *TUGgt20* - This subgroup consists of 3 variables: time taken in performing timed up and go tests with simple and double tasks respectively and binary variable if a person takes time more than 20 seconds to perform the test with the simple task. We use the binary variable from this group and named it TUGgt20.

- *vitMar* - This subgroup consists of 2 variables regarding the time taken in walking 10 meters and speed in meters per second. We selected the walking speed variable and discretized it as follows [47, 83]:

$$vitMar = \begin{cases} 0 & \text{if walking speed } < 0.7 \text{ m/s} \\ 1 & \text{if walking speed } \in [0.7 - 1.1) \text{ m/s} \\ 2 & \text{if walking speed } \geq 1.1 \text{ m/s} \end{cases}$$

- *X, no. of X type variables* - This category consists of 6 small groups regarding the number of diuretics, nitro derivatives, neuroleptics, sedative drugs, antidepressant drugs, and psychotropic drugs. We keep the binary variable for groups of variables about diuretics (*diuretiq*), nitro derivatives (*derivNit*), neuroleptic (*neurolep*), sedative drugs taken (*a1medSad*), antidepressant drugs (*a1AntiDep*) respectively, because we have a very small number of data points other than 0 and 1, in variables number of X in these groups. The number of psychotropic drugs is discretized with variable with domain size 3 because the number of persons with at least 2 psychotropic drugs is important and also because psychotropic drugs regroup several subcategories such as "antidepresseurs", "sédatifs", etc. it is defined as follows:

$$nbPsych3 = \begin{cases} 0 & \text{if no. of psychotropic drugs } = 0 \\ 1 & \text{if no. of psychotropic drugs } = 1 \\ 2 & \text{if no. of psychotropic drugs } \geq 2 \end{cases}$$

Group C: variables with large domain and continuous variables for which no other variable exists with a very close meaning

Group C consists in 16 variables for which no other variable with the same or close meaning exists. Group C includes 3 continuous variables and 13 variables with larger domain sizes. We present as follows how we discretized 13 of these variables and why we finally removed 3 of them.

- *age4* - In the data set from the hospital of Lille, we have
  - 83 patients whose age is between 65 and 70 years old;

Short name	Variable description	Details (if any)
nbmed3	number of medications	variable with large domain
nbchu2	number of falls in last 6 months	variable with large domain
bmi_lt19	BMI	continuous variable
apUniGt5	standing more than 5 seconds on one leg	2 continuous variables combined
TUGgt20	Get up and Go in more than 20 seconds	2 continuous variables combined
vitMar	walking speed	2 continuous variables combined
diuretiq	numbers of diuretics	variable with large domain
derivNit	numbers of nitro derivatives	variable with large domain
neurolep	numbers of neuroleptic drugs	variable with large domain
a1medSed	numbers sedative drugs taken	variable with large domain
a1AntiDep	numbers of antidepressant drugs taken	variable with large domain
nbPsych3	numbers of psychotropic drugs taken	variable with large domain

Table 3.5: Summary of variables in group B

- 111 patients from 70 to 80 years old,
- 933 patients from 80 to 90 years old and
- 155 patients older than 90 years of age.

We discretized this variable based on the equal length principle and is given as follows:

$$age4 = \begin{cases} 0 & \text{if age } \in [65,70) \\ 1 & \text{if age } \in [70,80) \\ 2 & \text{if age } \in [80,90) \\ 3 & \text{if age } \geq 90 \end{cases}$$

- *ADLlt5* - This variable represents the activity of daily life score. A score of 6 indicates the full function, 4 indicates moderate impairment, and 2 or less indicates severe functional impairment[11]. Based on the interview with the expert from the hospital of Lille we propose the following discretization of this variable:

$$\text{ADLlt5} = \begin{cases} 0 & \text{if ADL score } \in [5,6] \\ 1 & \text{if ADL score } \in [0,5) \end{cases}$$

- *LSAi4* - This variable represents the life space assessment score of a given person. The maximum LSA score of a given person can be 120 [113]. To discretize this variable we used the equal-frequency algorithm. The discretization is given as follows:

$$LSAi4 = \begin{cases} 0 & \text{if LSA score } < 17 \\ 1 & \text{if LSA score } \in [17,32) \\ 2 & \text{if LSA score } \in [32,57] \\ 3 & \text{if LSA score } \in [57,120] \end{cases}$$

- *nFrac4* - This variable represents the total number of fractures a given person has. Based

[11] https://www.alz.org/careplanning/downloads/katz-adl.pdf

on interviews with the experts in fall prevention, we discretized this variable as follows:

$$nFrac4 = \begin{cases} 0 & \text{if no. of fractures } = 0 \\ 1 & \text{if no. of fractures } = 1 \\ 2 & \text{if no. of fractures } = 2 \\ 3 & \text{if no. of fractures } \geq 3 \end{cases}$$

- *variables related to habits of a person* - In this group, we have 7 variables representing the habits of a person namely: (1)Dressing and undressing; (2) taking a shower or bath; (3) getting up from a chair or shitting down; (4) going down or up;(5) reaching over their head or to the ground; (6) going down or up a slope; (7) going out example a church, etc. All these variables are a score between 0 and 4.

- *antiHT3* - This variable represents the number of anti-hypertensive drugs taken and is discretized as follows:

$$antiHT3 = \begin{cases} 0 & \text{if no. of anti-hypertensive drugs } = 0 \\ 1 & \text{if no. of anti-hypertensive drugs } = 1 \\ 2 & \text{if no. of anti-hypertensive drugs } \geq 2 \end{cases}$$

- *respHypo3* - This variable represents the number of drugs possibly responsible for the orthostatic hypotension and is discretized as follows:

$$respHypo3 = \begin{cases} 0 & \text{if no. of drugs } = 0 \textit{ or } 1 \\ 1 & \text{if no. of drugs } = 2 \textit{ or } 3 \\ 2 & \text{if no. of drugs } \geq 4 \end{cases}$$

- *other variables* - We have 3 variables in this categories:
"*alpha blocker*" is removed because it is counted as the number of anti-hypertensive drugs;
"*alpha blocker for urinary use*" and "*Ldopa*" are removed because they are counted as the number of drugs responsible for hypotension.

Short name	**Variable description**	**Details (if any)**
age4	Age	variable with large domain
ADLlt5	Activities of daily living	continuous variable
LSAi4	Life space assessment	continuous variable
nFrac4	Number of fractures	variable with large domain
habiDhab	Dress and undress	variable with large domain
doucBain	Take a shower or a bath	variable with large domain
levChais	Get up from a chair or sit down	variable with large domain
montdesc	Going up or down stairs	variable with large domain
atQqchHB	Reaching for something above your head or on the ground	variable with large domain
marPente	Descending or ascending a slope	variable with large domain
sortir	Going out (e.g. church service)	variable with large domain
antiHT3	Number of anti-hypertensive drugs	variable with large domain
respHypo3	Number of drugs possibly responsible for orthostatic hypotension	variable with large domain

Table 3.6: Summary of variables in group C

Group D: binary variables for which no other variable exists with a very close meaning
We kept these 55 variables with no change in the second iteration.

Finally, after grouping variables as described above for groups A, B, and C, this last step of data cleaning for the second iteration leads us to 90 variables. In the next section, we present these 90 variables following their category in the ontology.

3.4.2 Description of the variables selected for iteration 1 and 2

In this section, we describe the variables selected after the cleaning of the data during both the first and second iterations. We organize the list of variables based on their categories in the ontology described in section 3.3 :

- variables related to a person's characteristics (Table 3.7)
- variables related to severity factors (Table 3.8)
- variables related to predisposing factors associated with chronic disease (Table 3.9)
- variables related to other predisposing factors (Table 3.10)
- variables related to precipitating factors (Table 3.11)
- variables related to behavioral factors (Table 3.12)

Variables related to characteristics of a person
In this category, we have 5 variables namely, sex, age, body mass index of the person as well as the number of falls in the last 6 months, and if the person went to higher studies. In the first iteration, the age variable was used as a binary variable (if a person is older than 80 years or not) whereas in the second iteration, is used with a domain size equal to 4. This difference in discretization was done after the recommendation by the experts. Also, the high level of study variable was not present during the first iteration. Table 3.7 represents the list of variables related to the characteristics of a person.

Variable description (french)	**Variable description (english)**	**Short name**	
		selection 1	**selection 2**
sexe de la personne	sex of the person	sex	sexe
âge de la personne	age of the person	agegt80	age4
Indice de masse corporelle de la personne	Body mass index of the person	BMI4	bmi_lt19
haut niveau d'étude	high level of study		htNivEtu
nombre de chutes au cours des 6 derniers mois	number of falls in last 6 months	nbChu2	nbchu2

Table 3.7: List of variables related to the characteristics of a person

Variables related to severity factors
In this category, we have in total 6 variables of which 2 variables are related to isolation, 2 variables are related to the incapacity of a person to get up from the floor, and 2 variables are related to bone fragility. In the first iteration, the variable regarding fracture is a combination

of fracture and vertebral collapse whereas the latter was removed in the second iteration by following the recommendation of the experts. Also, the variable number of fractures is not present in iteration 1. Table 3.8 represents the list of variables related to severity factors.

Variable description (french)	**Variable description (english)**	**Short name**	
		selection 1	**selection 2**
vit seul	lives alone	vitSeul	vitSeul
vit en maison de retraite	lives in retirement home	maisRet	maisRet
repos au sol > 1 heure	rest on the floor > 1 hour	gt1hSol	gt1hSol
a su se relever tout seul	was able to get up on his own	aSuSeRel	aSuSeRel
fracture lors d'une chute de sa hauteur	fracture when falling from height	fracturA	fracture
nombre de fractures	number of fractures		nbFrac4

Table 3.8: List of variables related to severity factors

Variables related to predisposing factors associated with chronic disease
In this category, we have in total of 32 variables related to predisposing factors associated with chronic disease. In the first iteration, we have one variable *arthPoly* which is a combination of two different variables *arth* and *polyArth*, we use these variables separately in the second iteration. Also, in the first iteration, we have one variable about other neurological diseases (*auTrNeur*)which is combined with 8 different variables, we use these variables separately in the second iteration. Furthermore, we have 10 variables in this category which are used only in the second iteration. This selection and combination of variables are done with the help of the experts. Table 3.9 represents the list of variables related to predisposing factors associated with some type of chronic disease.

Variables related to other predisposing factors
In this category, we have in total of 15 variables of which 2 variables are associated with functional tests related to gait, 1 variable is associated with a functional test related to balance, 8 variables are related to loss of autonomy, and 4 variables are related to functional difficulties of the person. From this category, we have 8 variables of which 7 related to the daily activities of a person and 1 related to walking speed are not present in the first iteration. Table 3.10 represents the list of variables related to other predisposing factors.

Variables related to precipitating factors
In this category, we have in total of 22 variables of which 2 variables are associated with acute disease, and 20 variables are related to medication taken by the person. From this category, we have 17 variables that are not present in the first iteration. Furthermore, the number of psychotropic variables is used with domain size equal to 2 and 3 in the first and second iterations respectively. Table 3.11 represents the list of variables related to other predisposing factors.

Variables related to behavioral factors
In this category, we have in total 11 variables of which 2 variables are associated with fear of falling, and 9 variables are related to the daily activities of the person. From this category, we have 5 variables that are not present in the first iteration. Table 3.12 represents the list of variables related to behavioral factors.

Other variables
In this category, we have in total 8 variables that do not belong to the categories defined above but are important factors to evaluate the risk of falls. From this category, we have 3 variables that are not present in the first iteration as well as 4 variables that are not present in the second iteration. Table 3.13 represents the list of other selected variables.

Variable description (french)	Variable description (english)	Short name	
		selection 1	selection 2
épilepsie	epilepsis		epilep
dépression	depression	dep	dep
arthrite	arthritis	arthPoly	arth
polyarthrite rhumatoïde	rheumatoid arthritis		polyArth
malnutrition	malnutrition		malnut
pathologie urologique	urological pathology	pathUro	pathUro
trouble de la vision	vision impairment	trVision	trVision
trouble de l'audition	hearing disorder	trAudit	trAudit
problème podologique	podiatric problem	pbPodo	pbPodo
neuropathie	neuropathy		neurPath
ataxie périphérique ou sensitive	peripheral or sensory ataxia		ataxPeri
myopathie ou atteinte neurologique proximale	myopathy or proximal neurological damage		myopat
ataxie cérébelleuse	cerebellar ataxia	auTrNeur	ataxCer
trouble proprioceptif	proprioceptive disorder		trProp
troubles de la sensibilité profonde	deep sensory disturbances		trSensPer
syndrome cérébelleux	cerebellar syndrome		syndCer
syndrome vestibulaire	vestibular syndrome		syndVes
akinésie	akinesia		akines
tremblement	tremor		trembl
démence	dementia	demence	demence
parkinson ou syndrome parkinsonien	parkinson or parkinsonian syndrome	parkOuSP	parkOuSP
tumeur cérébrale	brain tumor		tumCer
pathologie médullaire	spinal pathology		pathMed
pathologie radiculaire	root pathology		pathRad
cardiopathie	heart disease	cardiop	cardiop
hypertension	hypertension	HTA	HTA
diabète	diabetes	diabete	diabete
hyperthyroïdie	hyperthyroidism		hyperThy
ostéoporose	osteoporosis	osteoConf	osteopor

Table 3.9: List of variables related to predisposing factors associated with chronic disease

Variable description (french)	Variable description (english)	Short name	
		selection 1	selection 2
Timed get up and go > 20 s.	Timed get up and go > 20 s.	GUGOgt20	TUGgt20
Vitesse de marche	walking speed		vitMar
Appui unipodal > 5 sec	unipodal support > 5 s.	apUniGt5	apUniGt5
Vous habiller et vous déshabiller	Dress and undress		habiDhab
Prendre une douche ou un bain	Take a shower or a bath		doucBain
Vous lever d'une chaise ou vous asseoir	Get up from a chair or sit down		levChais
Monter ou descendre les escaliers	Going up or down stairs		montDesc
Atteindre quelque chose au dessus de votre tête ou par terre	Reaching for something above your head or on the ground		atQqchHB
Descendre ou monter une pente	Descending or ascending a slope		marPente
Sortir (par ex. service religieux)	Going out (e.g. church service)		sortir
Activités de la vie quotidienne	Activities of Daily Living	ADLinf5	ADLlt5
Difficulté à utiliser les toilettes	difficulty using the toilet	difWC	difWC
Trouble de la marche	walking disorder	trMar	trMar
Trouble de l'équilibre	balance disorder	trEq	trEq
Manque de force musculaire ou faiblesse musculaire des membres inférieurs	lack of muscle strength OR muscle weakness of the lower limbs	dfOuFaiM	sarcopen

Table 3.10: List of variables related to other predisposing factors

Variable description (french)	Variable description (english)	Short name	
		selection 1	selection 2
confusion	confusion		confusion
hypotension orthostatique	orthostatic hypotension	newHypoT	hypotenO
traitement anti HTA	anti-hypertension treatment		trAnHTA
au moins 1 antalgique classe II ou III	at least 1 class II or III analgesic		gt1antal
anticholinergiques	anticholinergics		antiChol
AGONISTES dopaminergiques	dopaminergic AGONISTS		agonDopa
au moins 2 psychotropes	at least 2 psychotropic drugs		gt2psych
corticothérapie > 3 mois	corticosteroid therapy > 3 months		corticTh
antiandrogénique (décapeptyl, ...)	antiandrogens (decapeptyl, etc.)		antiAndr
anti arithmique	anti arithmic		antiArit
nombre de médicaments	number of drugs	nbMed3	nbmed3
traitement anti-ostéoporose avant consultation	anti-osteoporosis treatment before consultation	newTrOst	traAnOst
au moins un médicament diurétique	at least one diuretic drug	dieretiq	diuretiq
au moins un dérivé nitré	at least 1 nitrate derivative		derivNit
au moins un neuroleptique	at least 1 neuroleptic		neurolep
au moins un sédatif	at least 1 sedative drug		sedatif
au moins un antidépresseur	at least 1 antidepressant		antidepr
nombre de médicaments psychotropes	number of psychotropic drugs	gt1psych	nbPsych3
au moins un antihypertenseur	at least 1 antihypertensive drug		antiHT3
nombre de médicaments éventuellement responsables d'hypotension orthostatique	number of drugs possibly responsible for orthostatic hypotension		respHypo3

Table 3.11: List of variables related to precipitating factors

Variable description (french)	Variable description (english)	Short name	
		selection 1	selection 2
Evite de sortir de peur de tomber	avoid going out for fear of falling	evitSort	evitSort
Peur de tomber	fear of falling	peurTom	peurTom
Aide humaine	human aid		aideHum
Alcool	alcohol	alc	alcool
Conduit sa voiture	drive his car	conduit	conduit
Sortir (quitter son domicile)	go.out (leaving his home)	sort	sort
Porter des bas de contention	wearing compression stockings		basCount
Sortir avec quelqu'un	Going out with someone company		sortAcc
Sortir seul à pied	Going out alone on foot		sortSeul
Utilisation d'une aide technique de marche	use of a walking aid	utiATM	aidTecMa
Évaluation de l'espace de vie (initial)	Life Space Assessment (initial)		LSAi4

Table 3.12: List of variables related to behavioral factors

Variable description (french)	Variable description (english)	Short name	
		selection 1	selection 2
AVC-AIT	AVC-AIT	AVC_AIT	avc_ait
Hématome ED/SD	ED/SD hematoma		hematome
lipothymie / syncope concomitante à la chute	lipothymia / syncope concomitant with the fall		lipoth
ovariectomie < 45 ans ou ménopause précoce	oophorectomy < 45 years or early menopause		ovariect
facteurs environnementaux	environmental factors	factEnv	
tabac	tobacco	tabac	
arythmie cardiaque	cardiac arrhythmia	arythm	
pneumo (maladie pulmonaire obstructive chronique (MPOC), asthme)	pneumo (chronic obstructive pulmonary disease (COPD), asthma)	pneumo	

Table 3.13: List of other selected variables

3.4.3 Target Variables selected

Several target variables have been chosen for prediction from the list of variables selected because it is important to assess their value. Indeed, outside of specialized fall prevention services, information about these risk factors is frequently unavailable. It's interesting for a number of reasons to assess how likely it is that these factors will exist in the present or the future:

1. All of these factors go into determining fall risk, and since they are all modifiable, it is possible to take certain steps to lower that risk.

2. Since depression, dementia, orthostatic hypotension, Parkinson disease, and other neurological disorders are not always diagnosed, assessing the likelihood of their occurrence enables one to alert a doctor to the need for additional testing.

3. In order to prevent osteoporosis and loss of autonomy, it's interesting to evaluate their likelihood of developing positively in the future even if they don't already exist.

Table 3.14a provides the list of target variables and their prevalence selected during the first iteration. Similarly, table 3.14b represents the list of target variables and their prevalence selected during the second iteration. We distinguish two groups among these target variables:

- Group *M0* - the risk factors with majority class 0
- Group *M1* - the risk factors with majority class 1.

The target variables are listed in decreasing order of their prevalence.

Group	Target variable	prevalence of the RFF
M1	trMar	83.3 %
M1	peurTom	77.2 %
M1	trEq	74.5 %
M1	auTrNeur	70.1 %
M1	dFouFaiM	66 %
M1	nbChu2	58.4 %
M0	demence	42.2 %
M0	newHypoT	32.5 %
M0	dep	28.4 %
M0	ADLinf5	25.5 %
M0	osteoConf	19.2 %
M0	parkOuSP	16.5 %

(a) using iteration 1

Group	Target variable	prevalence of the RFF
M1	trMar	82.5 %
M1	peurTom	75.6 %
M1	trEq	73.4 %
M1	sarcopen	62 %
M1	nbchu2	57.9 %
M0	demence	42.1 %
M0	osteopor	33.2 %
M0	hypotenO	32.1 %
M0	dep	27.8 %
M0	ADLlt5	22.9 %
M0	parkOuSP	17.1 %

(b) using iteration 2

Table 3.14: Target Risk Factors for Falls and their group

3.4.4 Missing value imputation

Missing data is a common problem faced with real-world datasets. Missing data can be anything from missing sequence, incomplete features, files missing, incomplete information, data entry error, etc [9].

In our data, we also have missing values meaning for some variables we do not have any information about a given person. Moreover, there are many unusual values in the data set. These values are mostly present because of human error which is very common during the data collection [26]. For example, let us assume, a variable "X" from the data set is binary. For some patients, the value of this variable "X" is 11 or 8 or any other text. We treated this type of value as missing values. Now, we present the different steps taken to do the imputation of missing values in our data set during both iterations.

3.4.4.1 Iteration 1

As a reminder for this iteration, the data contains values for 1810 patients with 45 features of the mixed type that is numerical and categorical. All the methods we have discussed earlier (see section 2.4.1) have their advantages and disadvantages, but we selected KNN Imputation over other methods. Since our data is of mixed type using KNN is a suitable choice [5]. The other reasons behind selecting KNN are as follows
- It is very simple and easy to use as compared to others.
- It can be applied irrespective of the data that is whether data are MCAR, MAR or MNAR [5] (which is the same situation we have with our data)

The number of neighbors is set to five after evaluating different choices.

3.4.4.2 Iteration 2

In this iteration, our data set consists of information on 90 variables for 1745 patients which also includes some missing information. Figure 3.4 represents the distribution of missing information in our data selected after the second iteration. The X-axis represents the percentage of missing values and the y-axis represents the number of variables with less than or equal to that percentage of missing values. Furthermore, in this section, we will discuss the interest in doing missing value imputation and the algorithm to use when doing missing value imputation after the second selection of variables.

Should we do missing value imputation or not?
Here, the objective is to evaluate the interest in the imputation of missing values. In that aim, we first extract a subset of complete data, we call it "no_mv". It includes data from 912 patients (out of 1745) for 67 variables (out of 90 variables). We have 9 targets (out of 11) present in this selection. This subset is the result of a compromise between the number of variables and the number of cases. Keeping all the 95 variables leads to a subset of complete data including only 112 cases, which is not usable. Then we perform the prediction of each of these 9 RFFs on the basis of the 66 remaining variables. Second, we impute missing values on the remaining cases by using Naive Bayes, resulting in a second data set including all available cases (1745 cases). In order to compare the results, we keep the same set of 67 variables. We call it "after_mvi". Table 3.15 represents the accuracy of the prediction of 9 target RFF by using 4 well-known classifiers namely Logistic regression (LR), Support vector machine (SVM), Random forest (RF) and Bayesian Networks (BN) when using subset "no_mv" versus subset "after_mvi". The first column represents the target risk factor, second column represents the categories of the subset used, third column represents the prevalence (proportion of most frequent class) for a given target, and column 4 to 7 represents the improvement or decrement in accuracy form the prevalence using different classifiers.

In order to see the difference in accuracy when using "no _mv" subset and "after_mvi" subset, we first subtract the results when using "no _mv" subset from results using the subset "after_mvi". Furthermore, we took the average increment or decrement of the difference in accuracy using

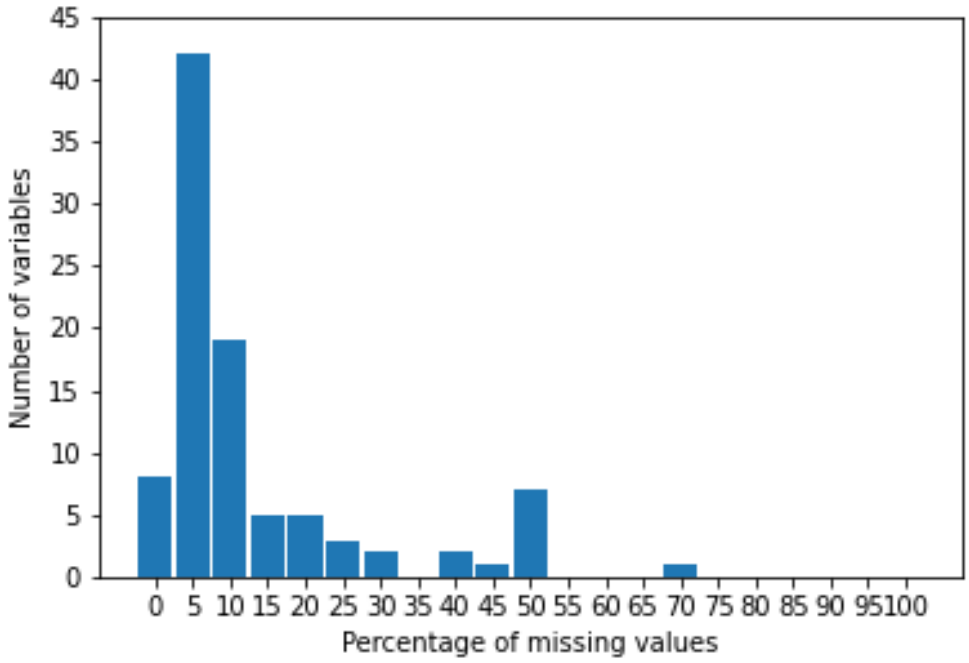


Figure 3.4: Distribution of missing values in our data set selected after the second iteration

Table 3.15: Accuracy of prediction of 9 target RFFs when using subset "no_mv" versus subset "after_mvi"

		prevalence	SVM	LR	RF	BN
trMar	no_mv	81.58	0.89	2.64	3.52	-1.86
	after_mvi	82.53	2.8	3.26	3.31	-3.67
peurTom	no_mv	78.73	0.43	0.21	1.86	-7.03
	after_mvi	75.65	0.68	1.6	1.54	-3.79
trEq	no_mv	74.02	8.33	5.8	6.58	-0.55
	after_mvi	73.46	8.44	7.69	7.23	2.65
defORfaib	no_mv	64.15	3.4	2.3	3.73	-3.96
	after_mvi	62.81	3.38	3.55	4.24	-2.41
nbchu3	no_mv	44.95	-1.09	-1.31	-3.18	-0.33
	after_mvi	42.18	3.32	3.78	3.95	2.41
hypot_OR	no_mv	65.79	-1.87	-4.06	0.43	-31.58
	after_mvi	66.71	-0.81	-1.61	0.16	-3.28
dep	no_mv	72.15	12.4	11.74	12.5	12.51
	after_mvi	72.2	11.87	11.75	11.7	10.03
osteo_OR	no_mv	66.22	10.3	10.95	11.83	8.43
	after_mvi	66.31	11.75	12.26	10.48	5.9
park_OR	no_mv	82.35	0.11	-0.11	0	-9.54
	after_mvi	82	0.01	0.86	0.52	-14.95

a given classifier for a given target. Table 3.16 represents the average difference (increment or decrement) in accuracy when using subset "after_mvi" than subset "no_mv".

Table 3.16: Average difference (increment or decrement) in accuracy when using subset "after_mvi" than subset "no_mv"

var	**SVM**	**LR**	**RF**	**BN**	**avg**
trMar	1.91	0.62	-0.21	-1.81	**0.13**
peurTom	0.25	1.39	-0.32	3.24	**1.14**
trEq	0.11	1.89	0.65	3.2	**1.46**
defORfaib	-0.02	1.25	0.51	1.55	**0.82**
nbchu3	4.41	5.09	7.13	2.74	**4.84**
hypot_OR	1.06	2.45	-0.27	28.3	**7.89**
dep	-0.53	0.01	-0.8	-2.48	**-0.95**
osteo_OR	1.45	1.31	-1.35	-2.53	**-0.28**
park_OR	-0.1	0.97	0.52	-5.41	**-1.01**
avg	**0.95**	**1.66**	**0.65**	**2.98**	

From table 3.16 we can see that on average using SVM, we got 0.95% better results with data after missing value imputation and on average, using LR we got 1.66% better results with data after missing value imputation. In addition, on average using RF, we got 0.65% better results with data after missing value imputation, and on average using BN we got 2.98% better results with data after missing value imputation. Moreover, when we take an average of all classifiers for trMar we got 0.13% more accurate results when using data after missing value imputation. Also, we can see similar results for other variables. We can conclude from these results that doing missing value imputation is the right step in our case. Furthermore, in the next part, we will address the question of choosing an algorithm to do missing value imputation.

What algorithm for missing value imputation?
Here, the objective is to evaluate the interest of an algorithm for the imputation of missing values. As we discussed at the beginning of this section, there are plenty of algorithms to perform missing value imputation. In our work, we try to use simpler and easier-to-perform algorithms for the imputation of missing values since this is not the main objective of our thesis. In order to accomplish that, we compare the KNN algorithm with the naive Bayes algorithm for values of k ranging from 1 to 19 when imputing missing values from our data set. In figure 3.5 we present the performance of KNN versus naive Bayes for doing imputation of missing values. For a given target variable we present in the figure the improvement in accuracy (y-axis) from the baseline given the frequency of the most frequent class (x-axis). Each point in the figure represents one target variable. From these results, we can conclude that naive bayes is the right choice for the imputation of missing values in our case.

3.5 Conclusion

Data is an essential component of an AI or ML model. As data collection becomes simpler as a result of increased digitization, it is also becoming more widely accessible in the healthcare sector. The information is collected at hospitals, care centers, and other healthcare institutions. These data could include, among other things, information about biological processes, administrative processes, and insurance claims. The amount of data being used to solve healthcare-related issues is increasing, as is the use of data-driven methods.

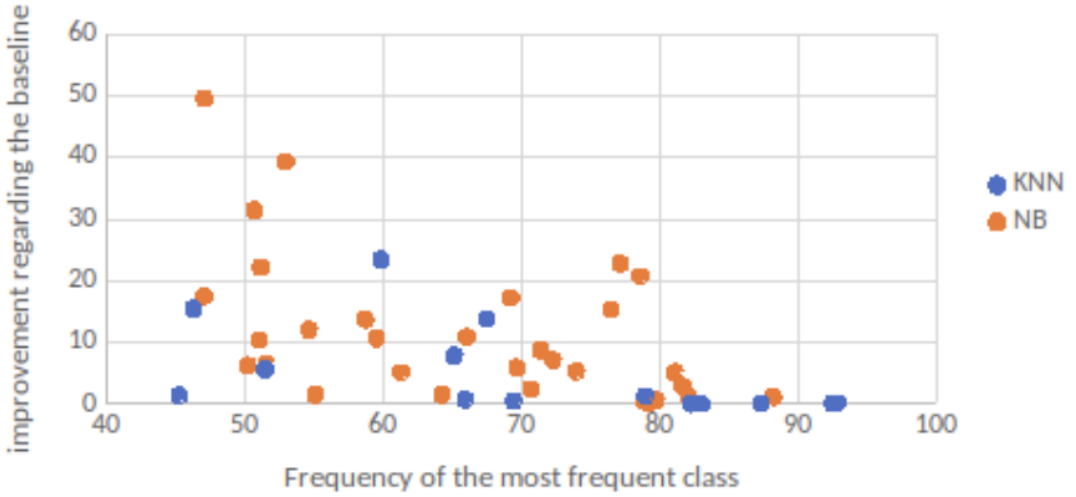


Figure 3.5: Missing value imputation using Naive Bayes versus KNN

In this chapter, firstly, we described the source (Service of Fall Prevention, Hospital of Lille, France) from where we got our data and the description of the data used. The initial data includes 1810 patients who visited the service between January 2005 and December 2016, of whom 28% are male and 72% are female, with ages ranging from 51 years old to 100 years old, with an average age of 81 years old. In our study, we included a person who can walk and is 65 years of age or older.

Secondly, we described the methodology used to design an ontology for fall prevention, followed by the resulting ontology for the risk of falling. The goal of this ontology, which served as the foundation for the creation of the fall prevention software system, is to support the assessment of elderly individuals' risk factors for falls. It also provides a solid foundation for our knowledge of the variables in our data set.

Finally, we presented the different steps of data preprocessing that led to the definition of the two data sets that we used in our analysis. In order to identify the variables to be kept for our study, we first used the criterion of data quality to remove unusable variables (2 empty, 3 with errors in formula, 59 grayed out, and 41 text columns), and a second criterion is to remove variables which can not be used to evaluate any risk factors for falls. In pursuit of that aim, we removed 17 variables related to recommendations, 36 variables associated with the second appointment after 6 months, and 4 administrative variables. After the variables guided by the first two criteria are removed, we also use a third criterion that is specific to the first (second) iteration of data cleaning and consists of providing a model with a reasonable size (aim to improve each step of the first iteration). This cleaning led to a data set containing 45 variables in the first and 90 variables second iterations. Additionally, we have identified 12 target risk factors (11 in the second iteration because the other 1 is not present in the data after cleaning) to evaluate from our data.

In the subsequent chapter, we present the various findings for evaluating a specific target risk factor using the data sets chosen during both iterations 1 and 2.

Chapter 4

Evaluate risk factors for fall using static data

Outline of the current chapter

4.1 Introduction

In the previous chapter, we have explained the description and preprocessing of our data. As a reminder, the main objective of our work is to predict the presence or absence of risk factors for falls based on the information for a given person. To that aim, we have also described the various methods we propose to use according to the characteristics of the data. In this chapter, we present the results obtained. As mentioned earlier, we used an iterative approach for analyzing our data. It is divided into two iterations. The objective for the first (resp. second) iteration is to select the minimum (resp. maximum) number of variables from the initial data set in order to evaluate the risk factors for falls respectively. Here, in the first iteration, the goal was to provide

a model with a reasonable size and in the second iteration to improve the results obtained by the 1st iteration and try to keep as many variables as possible. With that in mind, in this chapter, we focus on the following questions:

Question to focus on using iteration 1

→ Should we use a complete set of variables or a specific subset of variables for the evaluation of a given target risk factor?

→ How to evaluate a target risk factor based on the available partial observations for a given person?

→ Should we balance the data before using a classifier or not?

Question to focus on using iteration 2

→ Should we use a complete set of variables or a specific subset of variables for the evaluation of a given target risk factor?

→ Which classifier to use to evaluate a target risk factor for a given person?

→ How to assess the quality of prediction when evaluating a given target risk factor using a given classifier?

We organize the chapter as follows: first, we present the following results when using iteration 1: prediction using a complete set of variables versus a specific subset; prediction based on the percentage of observations available; prediction based on imbalanced data versus balanced data. Furthermore, we present the following results when using iteration 2: prediction using all variables versus a specific subset of variables; prediction of different targets using a single BN model versus a specific BN for each target; procedure to learn the structure of BN model; comparison of the predictive performance of BN with other usual classifiers. To compute all these results we used the following python libraries: scikit-learn[1], pandas[2], numpy[3], matplotlib[4], pyAgrum[5], tensorflow[6], keras[7], and scikeras[8].

4.2 Results using iteration 1

In this section, we describe the results obtained when using data obtained after the first iteration of data preprocessing. We first describe whether we should be using a specific subset or a complete set of variables to predict each of the target risk factors for falls. After that, we examine how the performance of different classifiers evolves when the evaluation is based on partial observations. To that aim we consider different percentages of observations to make the

[1] https://scikit-learn.org/sTable/
[2] https://pandas.pydata.org/
[3] https://numpy.org/
[4] https://matplotlib.org/
[5] https://pyagrum.readthedocs.io/en/latest/
[6] https://www.tensorflow.org/
[7] https://keras.io/
[8] https://www.adriangb.com/scikeras/sTable/

prediction. The next question concerns the problem of imbalance in the data, more precisely, should we use the imbalanced data as it is (that is in the original form) or should we balance the data using some balancing techniques before using the prediction models? Furthermore, the results presented in this section are published in [100, 104, 105, 106].

4.2.1 Should we use a specific subset or a complete set of variables?

As we have seen in the previous chapter, after preprocessing of data for this iteration we have information on 45 variables for 1810 patients. Since our goal is to evaluate each of the 12 target risk factors for falls thanks to these 45 variables, we first try to answer the question of whether should be using a specific subset or a complete set of variables to predict the target risk factors for falls. With that aim, in this first iteration, we compare the results of the prediction using the complete set of variables and a specific subset obtained by variable selection using different classifiers, namely Bayesian Networks (BN), Logistic Regression (LR), Decision Tree (DT), Random Forest (RF) and Support Vector Machine (SVM). In that context, we use the chi-square method in order to identify a subset of variables associated with each of the target risk factors for falls. We consider a significance level of 0.05, which is the usual (most commonly used) value, meaning that all the variables with a significance level of less than 0.05 are selected for a given target risk factor. We also compare the results with a significance level of 0.02 and it makes no change. Figure 4.1 represents the schematic diagram of the methodology used.

Figure 4.1: Methodology used when evaluating the interest of using all variables versus a specific subset of variables in iteration 1

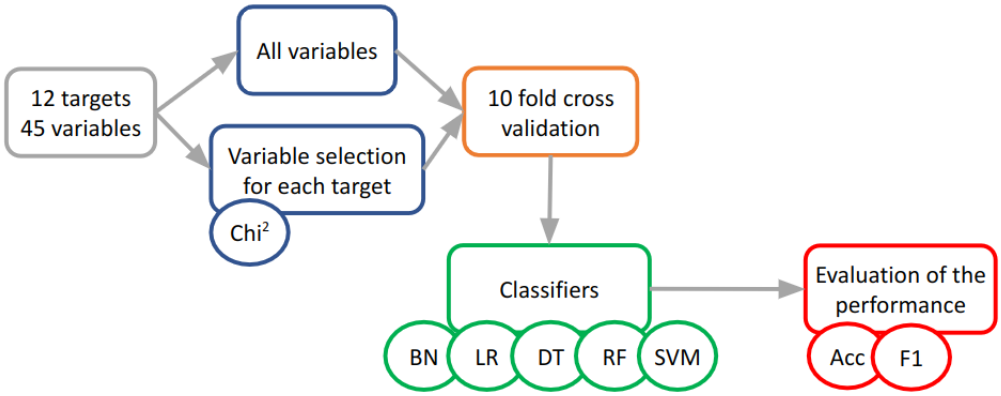


Table 4.1 presents the difference between the results when using a complete set of variables (45 variables) and a specific subset of variables selected using hypothesis testing for each risk factor. We compared the accuracy and F1 score for each risk factor shown in the first and second rows respectively. The maximum value of accuracy for a given risk factor is shown in blue (red for the F1 score).

Table 4.1 shows that for the prediction of any given risk factor, the results using a complete set of variables are very similar when using the specific subset of variables. However, the difference is always very small and hence can be neglected and we can use any of the discussed scenarios for prediction. This can seem counter-intuitive since it is known that unuseful variables may alter the performance of the classification but we choose to use the complete set of variables. Indeed the advantage of using a complete set of variables over the specific subset comes from

Table 4.1: Comparison of accuracy (acc) and F1 score (F1) using specific subset (sss) for each risk factor vs complete data (45var).

RFFs		**BN**		**LR**		**DT**		**RF**		**SVM**	
		sss	45var	sss	45var	sss	45var	sss	45var	sss	45var
trMar	acc	86.24	86.8	86.88	87.06	80.65	80.36	86.65	86.57	**87.28**	87.24
	F1	0.92	0.92	0.92	0.92	0.88	0.88	0.92	0.92	***0.93***	***0.93***
peurTom	acc	78.7	**79.8**	78.98	79.04	73.94	72.93	77.16	79.02	78.82	78.99
	F1	0.87	***0.88***	0.87	0.87	0.83	0.82	0.86	***0.88***	0.87	***0.88***
dfOufaim	acc	68.69	69.49	70.06	69.5	60.04	60.19	69.32	**70.99**	70.15	70.59
	F1	***0.8***	***0.8***	0.79	0.79	0.69	0.69	0.79	***0.8***	***0.8***	***0.8***
trEq	acc	82.2	**82.51**	81.47	81.03	71.06	70.19	81.3	81.46	82.09	82.06
	F1	***0.89***	***0.89***	0.88	0.88	0.8	0.8	0.88	0.88	***0.89***	***0.89***
auTrNeur	acc	71.01	71.17	71.57	71.17	58.14	60.19	68.83	**71.71**	71.12	71.69
	F1	0.82	0.82	0.82	0.82	0.69	0.71	0.8	0.82	0.82	***0.83***
nbchu2	acc	57.15	59.19	61.57	61.34	55.26	55.55	58.52	61.61	**62.14**	61.98
	F1	0.69	0.71	0.7	0.7	0.61	0.62	0.67	0.71	***0.72***	0.71
ADLinf5	acc	78.5	79.16	**80.4**	80.22	70.9	79.71	79.68	79.91	80	79.9
	F1	0.53	***0.55***	0.54	0.54	0.45	0.44	0.49	0.47	0.48	0.49
demence	acc	66.48	67.22	68.15	**69.27**	58.18	58.64	65.88	68.6	67.86	68.8
	F1	0.55	0.54	0.59	***0.61***	0.51	0.51	0.56	0.58	0.58	0.58
newHypoT	acc	67.39	67.47	**67.72**	66.87	60.2	56.61	62.48	67.55	67.37	67.35
	F1	0.01	0.02	0.18	0.22	0.33	***0.35***	0.31	0.14	0.04	0.03
dep	acc	73.73	73.9	75.07	73.7	72.82	67.62	73.7	73.78	**75.11**	74.56
	F1	0.45	0.42	***0.47***	0.46	0.46	0.45	0.5	0.35	***0.47***	0.4
osteoconf	acc	81.65	82.26	83.24	**83.46**	76.41	76.2	80.97	82.3	82.72	82.52
	F1	0.49	***0.52***	0.46	0.48	0.38	0.4	0.39	0.3	0.4	0.32
parkOuSP	acc	83.4	83.48	**83.88**	83.69	74.94	72.91	81.7	83.3	83.44	83.47
	F1	0.06	0	0.12	0.19	***0.27***	0.25	0.16	0.02	0.01	0

the fact that if we use a specific subset to evaluate a given risk factor we are restricting our model to use a smaller set of information. Also, this small set is not always entirely available in our context where only partial information can be obtained, and not always the same part of information. Moreover, the final objective of our work is to provide an aiding system that will be used by the general practitioner in real-life situations. Hence, if we use the complete set of variables we have more chances to get the information about a given patient, since a piece of the available information may partially compensate for the absence of another element. We will discuss more about partial information as follows.

In summary, it is better to use the complete set of variables to build the model instead of using the specific subset to predict the risk factors for falls because as shown in this study the results using a complete set of variables are as good as the other. Also, in real-life situations, the number of observations can be different for each patient. So for each new patient, we have to learn a new model using the available information which can be very time-consuming. With that in mind, in the next section, we present the prediction of the presence or absence of the target risk factors based on the available information.

4.2.2 Comparison of classifiers based on percentages of observations

So far we evaluated whether should we use a specific subset or a complete set of variables to predict the target risk factor and it comes out based on results presented in the previous section that using a complete set of variables is a better choice in our situation. With that in mind, in this section, we present the results when using the complete set of variables for the prediction of target risk factors for falls when using a partial observation. Here, by the partial observations set, we mean for example if we have a total of 44 variables and we say we only have 25% of the information of a given person, that means for that person we only have information available for 11 variables and we have to predict the presence or absence of our target risk factor based on these 11 variables instead of the complete set with 44 variables.

In order to estimate the risk factors based on available information, we build a Bayesian Networks (BN) model and compare the results with other classifiers, namely Logistic Regression (LR), Decision Tree (DT), Random Forest (RF) and Support Vector Machine (SVM). Figures 4.2 and 4.3 show a schematic diagram of the methodology to predict the presence or absence of the risk factors using BN and other classifiers (LR, DT, RF, SVM) respectively. The approach is roughly the same on both algorithms, except that the BN model (graph and parameters) is learned only once whereas the other classifiers have to be learned again for each target variable and each subset of variables. To evaluate the prediction model performance we used 10-fold cross-validation. In each fold, 10% cases were used as testing sets and 90% cases as training sets. Then the average over these 10-fold evaluations is compared to the results with the baseline classifier. The above procedure is repeated for different sets of observations with different sizes.

Furthermore, to evaluate the quality of prediction by a classifier we compare the results with a baseline classifier that always predicts the most frequent class when comparing accuracy scores and the positive class when comparing accuracy F1 scores. Because, when we have a negative class as the majority and we chose the baseline to predict always majority class, the recall will always be 0 hence F1 score for the baseline classifier will always be zero. Tables 4.2, 4.3, and 4.4 represent the accuracy (left) and F1 score (right) for our targets calculated using a complete set of variables for different percentages of available observations. The horizontal axis represents the percentage of randomly selected observations used to predict the target risk factor starting from 10% up to 100%.

Based on the results presented in Tables 4.2, 4.3, and 4.4 we can see that the accuracy of all target variables except *ParkOuSP* and *newHypoT* increases with the percentage of observations for

Figure 4.2: Methodology used when evaluating the target risk factors using BN based on partial observations in iteration 1

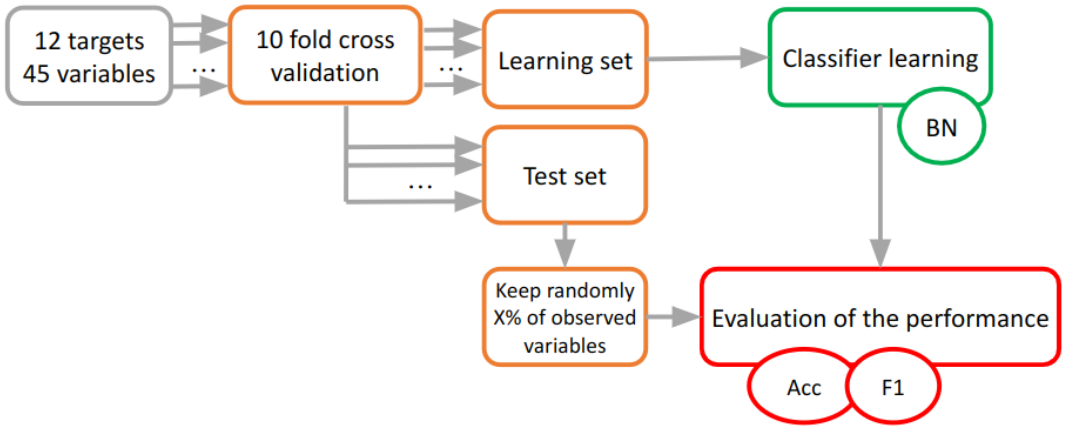


Figure 4.3: Methodology used when evaluating the target risk factors using the usual classifier (LR, DT, RF, SVM) based on partial observations in iteration 1

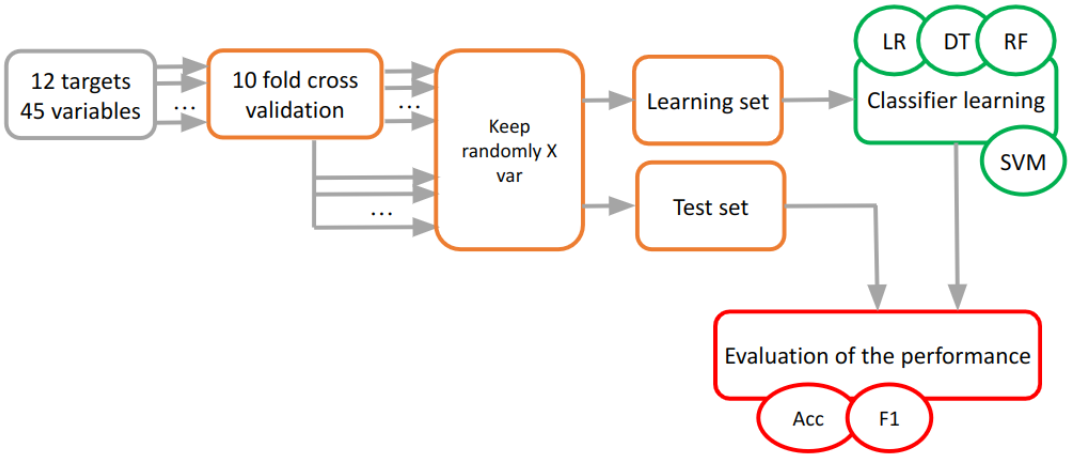


Table 4.2: Accuracy and F1 score for each risk factor respectively. Horizontal axis represents the % of available observations among the 44 remaining variables

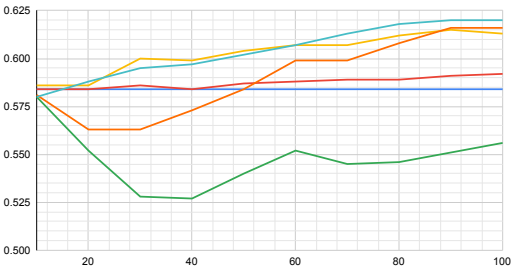
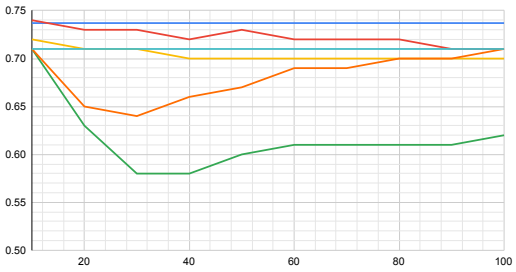
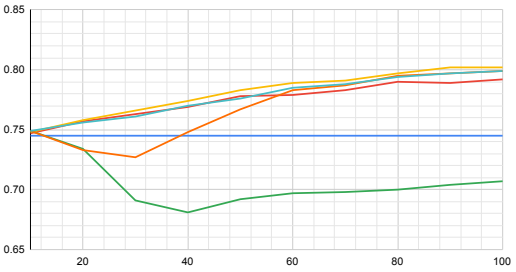
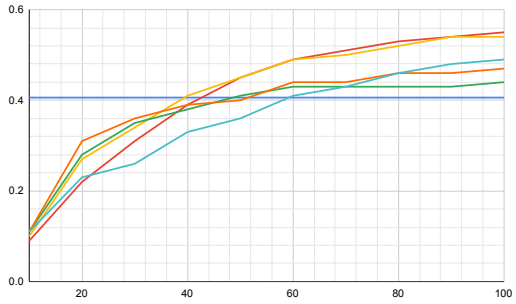
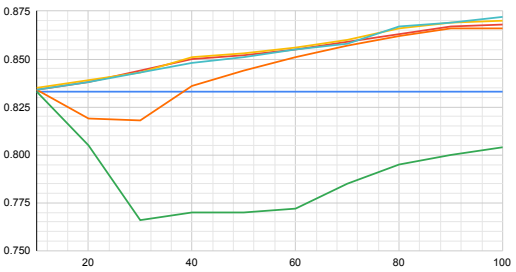
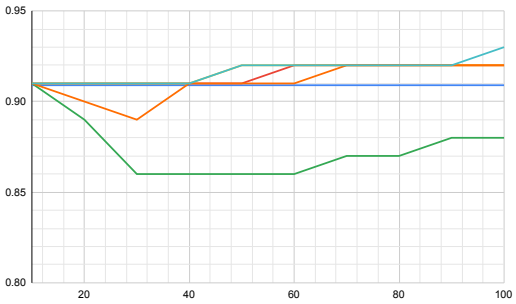
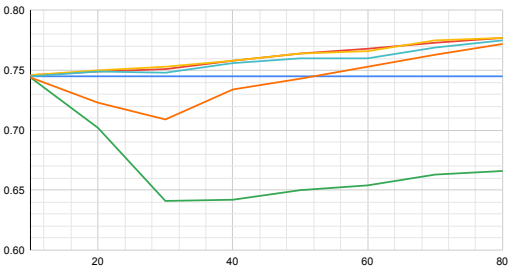
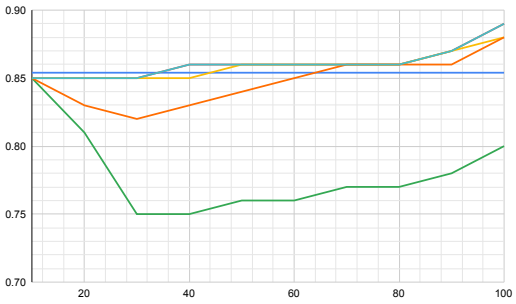
Accuracy	F1 score
nbChu2	baseline bn lr dt rf svm
	
ADLinf5	
	
trMar	
	
trEq	
	

Table 4.3: Accuracy and F1 score for each risk factor respectively. Horizontal axis represents the % of available observations among the 44 remaining variables

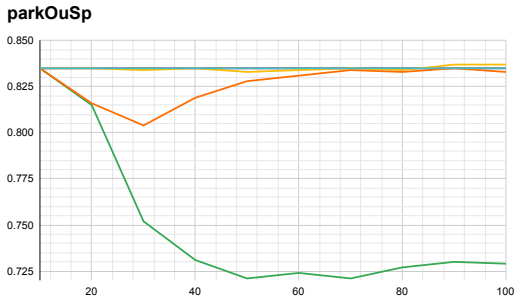
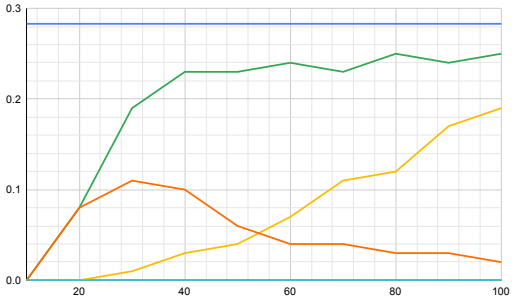
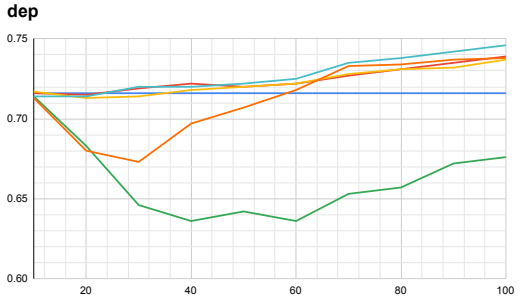
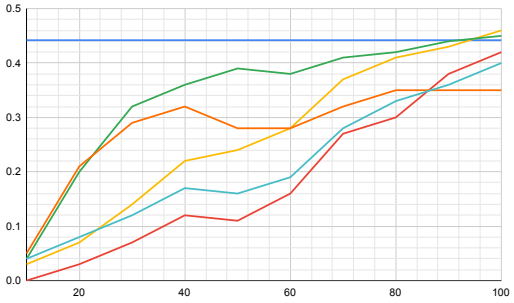
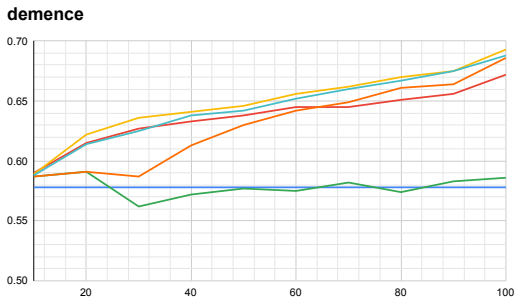
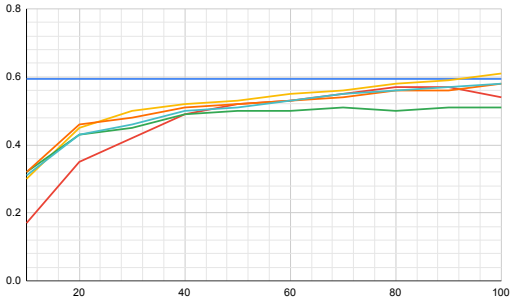
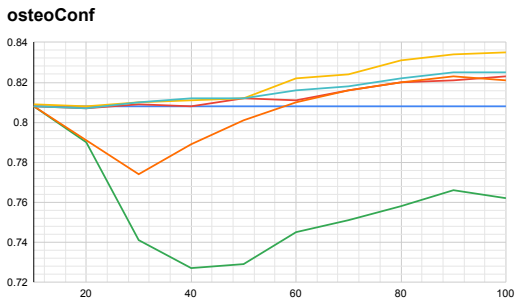
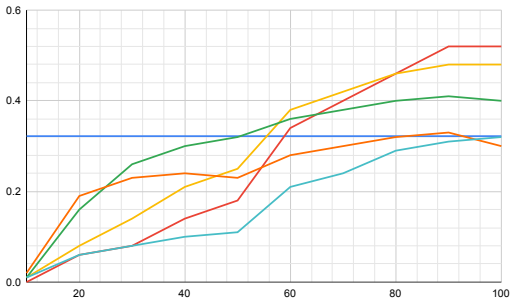
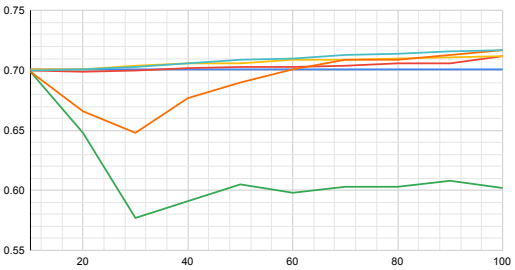
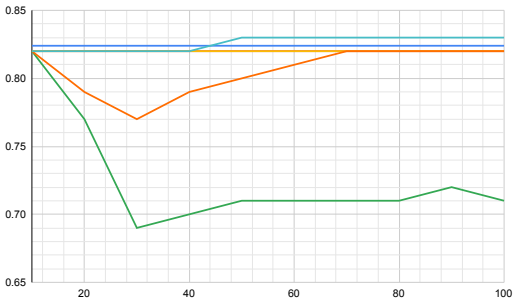
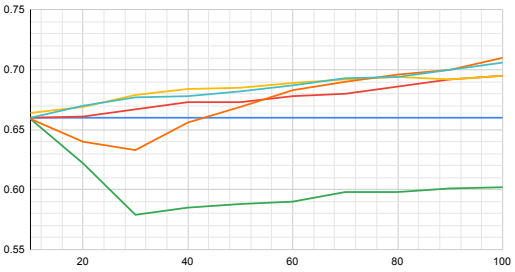
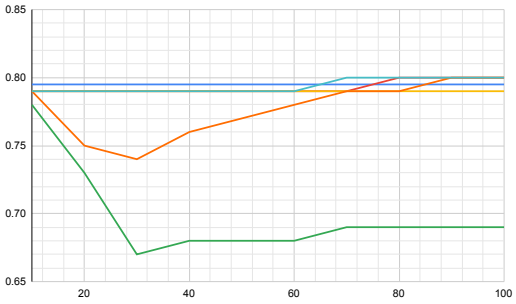
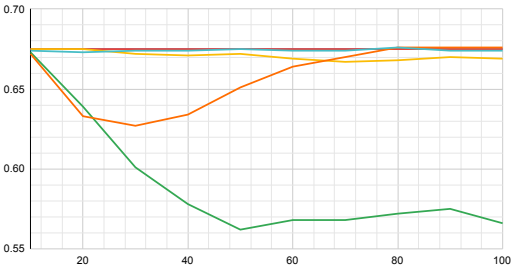
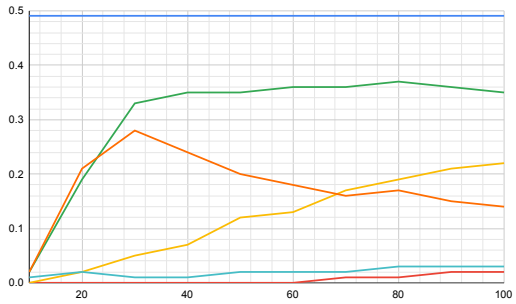
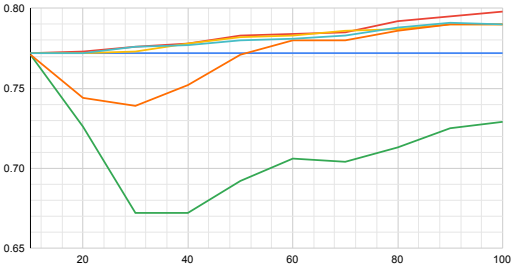
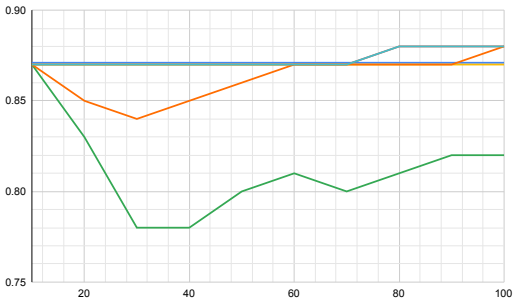
Accuracy	F1 score
	
	
	
	

Table 4.4: Accuracy and F1 score for each risk factor respectively. Horizontal axis represents the % of available observations among the 44 remaining variables

Accuracy	F1 score
auTrNeur 	
dfOufaim 	
newHypoT 	
peurTom 	

all classifiers except DT. Regarding the F1 score, it clearly increases along with the percentage of observation only for the targets *ADLinf5*, *osteoconf*, *trEq*, *dep*, and *demence*, and to a small extent for *trMar*. However, regarding the variables *dep* and *demence*, no classifier clearly outperforms the baseline in terms of the F1 score. For the targets *ParkOuSP* and *newHypoT*, the performance of the classifiers seems to be independent of the percentage of available observation, meaning that the classifiers fail to provide a correct evaluation of the RFF.

After analyzing the behavior of the classifiers regarding the percentage of available information, we consider some other points in the analysis of these results: A first point is that it can be seen that the predictive performance of the BN model is comparable to the other classifiers used. We note that BNs provide roughly the same quality of results for the different target variables as the other classifiers with some variations according to the target risk factors for falls. None of the classifiers that we tested is clearly better than the others, even if LR provides sometimes slightly better results and DT sometimes slightly lower performance. For example, for the number of falls ($nbChu2$) (Table 4.2 first row), there is from 2 to 3% difference between the baseline accuracy and the best three classifiers here (LR, SVM, RF) when at least 60% of observations are available, whereas BN's accuracy is slightly lower, and DT's accuracy is lower than the baseline classifier. With regards to the F1 score for the prediction of the number of falls, the value for any of the classifiers is less than the baseline value. About dementia variable ($demence$), the accuracy of the prediction is rather good whatever the classifier, except DT, but the F1 score is lower than the baseline value. This reveals the inability to detect dementia from this data set, with the only positive result for this variable being to state the absence of dementia. However, the results are better for other variables such as for activities of daily living ($ADLinf5$) and osteoporosis confirmed ($osteoConf$). The model's accuracy and F1 score increase from baseline as the percentage of available observations increases respectively (as shown in Table 4.2).

These results show that we are able to evaluate most of the risk factors (even if with a low improvement compared with the baseline classifier) except orthostatic hypotension ($newHypoT$) and Parkinson ($parkOuSP$). One of the reasons may be the data we used for this study is from a very specific population that is at high risk of falling, most of them presenting several risk factors. This bias probably making more difficult to get a clearer separation. Another reason, as stated by the expert regarding orthostatic hypotension, is that our set of variables does not include enough details about the class of drugs that are known to be predictive of hypotension. As for the variable $parkOuSP$, the expert states that it is not predictable from our set of variables.

In summary, based on the results presented above, we can conclude that all classifiers except DT allow to improve the accuracy with partial information, but only some targets are correctly predicted in terms of F1 score. In particular, the prediction of the targets osteoConf, ADLinf5, and trEq keeps interesting even when based on only 50% of the observations. In addition, BN's predictive performance is equivalent to that of the other classifiers implemented since it yields predictions that are approximate of a similar standard for various target variables, varying somewhat depending on the target risk factor for falls. Furthermore, among the 45 variables selected for this study, an arbitrary number of them can be observed, whether they are targets or not. Moreover, risk factors are not independent of each other, meaning that when one of them is observed, it should be used to improve the evaluation of the others, in addition to other observed features. That situation makes more difficult the use of usual classifiers because a new model would have to be learned for each target variable, and for each possible subset of observed variables. BN models allow overcoming that problem, since the same model can be used to evaluate any variable of the model, regarding any subset of observations. In addition, BNs allow the combination of general statistical knowledge and specific individual information, and to update belief on any node from incomplete observations. These features, the unicity of the model for any target variable and the possibility to combine knowledge from different sources, exactly

answer the problem of predicting risk factors in real-life situations. Another advantage of BN is that the model can be built both from data and expert knowledge which is very interesting in the context of health. It is also very important to make the model interpretable/ understandable by the final user (general practitioners) since it contributes to making the aiding system acceptable and augments the trust in results. So BN becomes the good choice to use because of the graphical representation that is easy to explain and understand.

However, the above-mentioned classifiers do not account for the data imbalance while making predictions. When classes are not equally represented, the data collection is imbalanced. A classifier may have trouble if the data set is severely imbalanced. Due to the increased chance of belonging to the majority class and the algorithm's attempt to reduce mistakes, these algorithms are more likely to classify a new observation in the majority class. With that in mind, in the next section, we deal with this problem of imbalance and present the use of some balancing techniques and their effects on the prediction of risk factors for falls.

4.2.3 Prediction using imbalanced versus balanced data

So far we have discussed the use of a complete set of variables to predict the presence or absence of a risk factor for falls and presented the results of prediction based on the available information about a patient (partial or complete). In this section, we describe the effect of balancing techniques when doing the prediction versus when we have imbalanced data. With that aim, we have compared the outcomes for six different classifiers namely Logistic Regression (LR), Random Forest (RF), Artificial Neural Networks (ANN), Support Vector Machine (SVM), Naive Bayes (NB), and Bayesian Networks (BN), in order to see the differences between utilizing imbalanced data for classifications and using the data after balancing with various balancing approaches (three oversampling methods: SMOTE, SMOTE-SVM, and ADASYN), with the aim of predicting separately 12 target variables. Figure 4.4 represents the schematic diagram of the methodology used.

Figure 4.4: Schematic diagram of the methodology used when evaluating the interest of using oversampling techniques in iteration 1

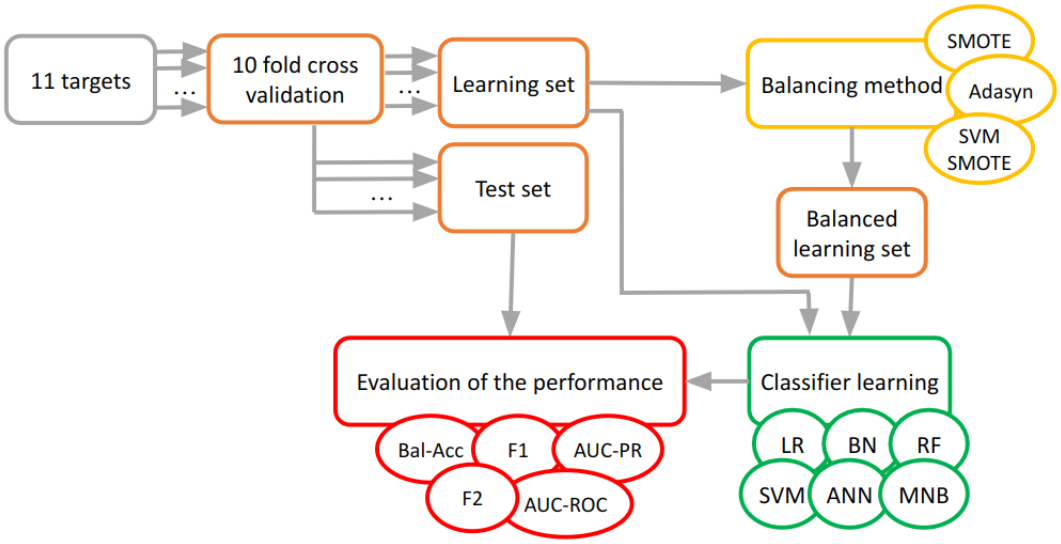
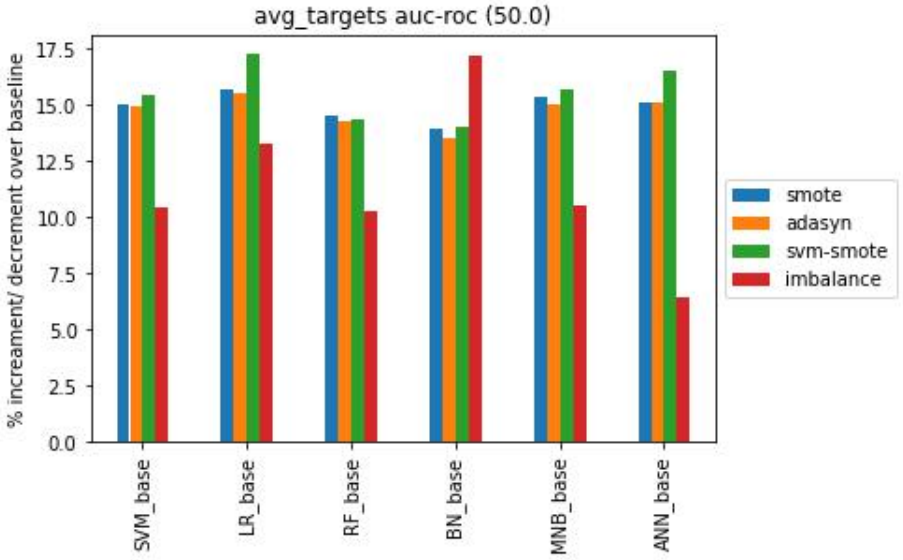


Figure 4.5: Percentage of increment from baseline regarding AUC-ROC, when averaging over all targets for each classifier, using imbalanced and balanced data with SMOTE, ADASYN and SVM-SMOTE respectively.



In this section, we first show the results for each classifier when averaging over all targets. Second, we present the results for each target when averaging over each classifier. In each case, we show the statistical t-test to summarize our findings.

4.2.3.1 Results by classifiers, averaging over targets

Figures 4.5 to 4.9 present the results when averaging over all targets for each classifier regarding (1) AUC-ROC, (2) AUC-PR, (3) Balanced accuracy, (4) F1 score, (5) F2-score, using imbalanced and balanced data (with the above balancing methods). The horizontal axis represents the different classifiers used. The vertical axis represents the percentage of increment or decrement from the baseline results when comparing using AUC-ROC, AUC-PR, and balanced accuracy and percentage of the score when comparing F1 and f2 scores. Here for F1 and F2 scores, we did not show the increment or decrement from baseline results because when the majority class is negative the baseline F1 (or f2) score is not defined because precision is not defined. The baseline results are computed using a dummy classifier which always predicts the majority class.

Regarding the results of all measures (Figures 4.5 to 4.9), it can be seen that when using imbalanced data, the average improvement regarding the baseline classifier is variable depending on the classifier. For example, regarding AUC-ROC (Figure 4.5), ANN improves the results from the baseline by about 6 points on average while the Bayesian network improves it by about 16 points when using the imbalanced data. On the other hand, the use of an oversampling method makes the improvement of all classifiers very similar with differences of only a few points. A second remark is that using an oversampling technique provides improvement for all classifiers except the Bayesian network which performs generally better when using imbalanced data, and for all measures. A third point to be noted is that SVM-SMOTE always leads to a slightly better improvement than SMOTE and ADASYN.

From the result presented here, the first conclusion is that the results of different classifiers are

Figure 4.6: Percentage of increment from baseline regarding AUC-PR, when averaging over all targets for each classifier, using imbalanced and balanced data with SMOTE, ADASYN and SVM-SMOTE respectively.

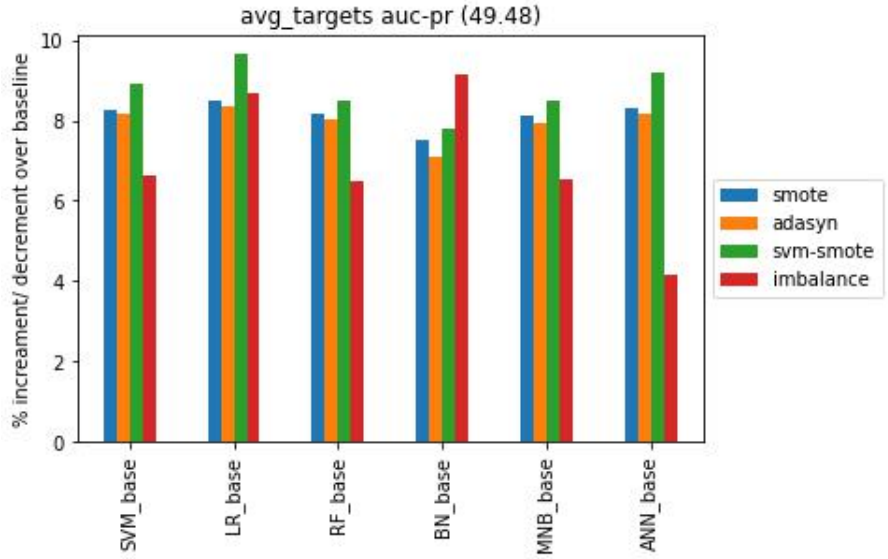


Figure 4.7: Percentage of increment from baseline regarding balanced accuracy, when averaging over all targets for each classifier, using imbalanced and balanced data with SMOTE, ADASYN and SVM-SMOTE respectively.

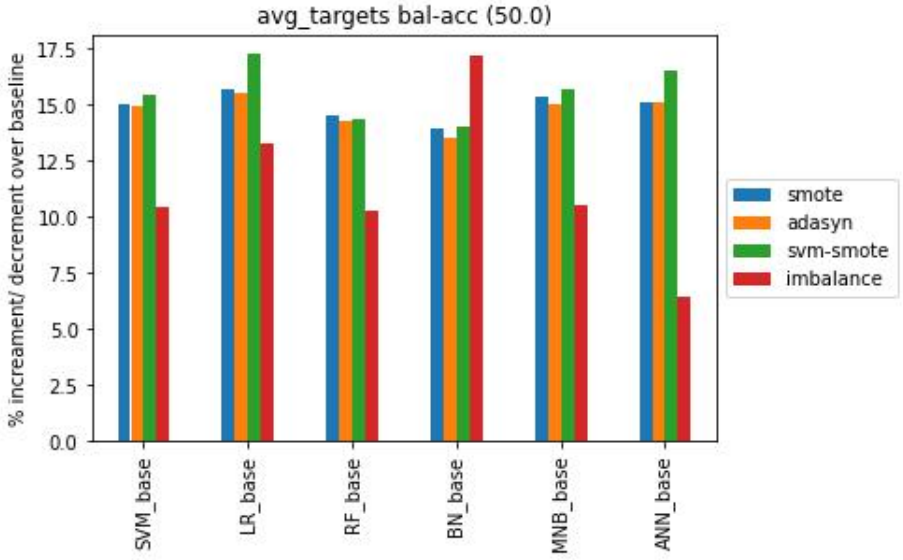


Figure 4.8: Percentage of F1 score, when averaging over all targets for each classifier, using imbalanced and balanced data with SMOTE, ADASYN and SVM-SMOTE respectively.

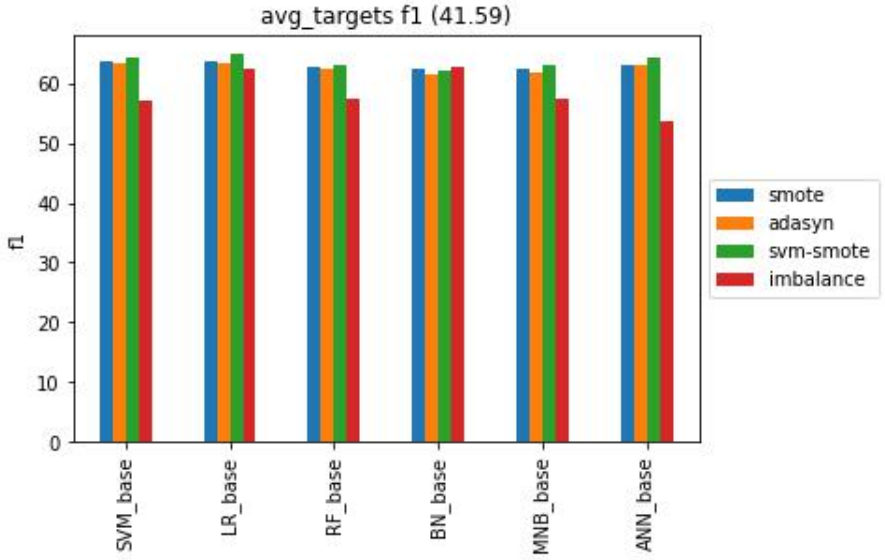
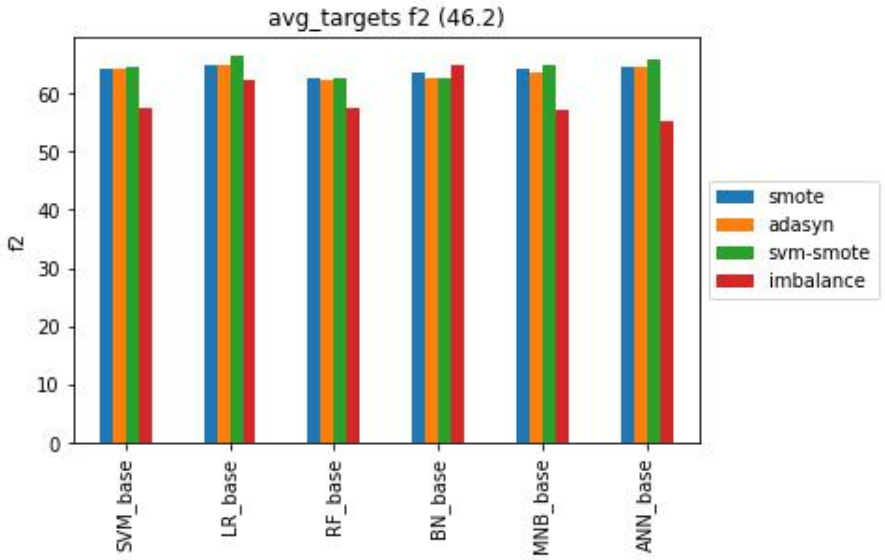


Figure 4.9: Percentage of F2-score, when averaging over all targets for each classifier, using imbalanced and balanced data with SMOTE, ADASYN and SVM-SMOTE respectively.



rather similar for all the considered measures, except the Bayesian network whose results using imbalanced data are comparable with the results of other classifiers when using balanced data. In order to evaluate the significance of the improvement after using oversampling techniques, we present the p-values as follows.

4.2.3.2 Statistical tests when averaging over targets

We now use a t-test to check the significance of improvement in the prediction results after using the oversampling techniques. We use classification techniques, SVM, LR, RF, BN, MNB, and ANN, to classify the averaged target variable for the original (imbalance) dataset and obtain the corresponding values of accuracy measures; AUC-ROC, AUC-PR, Bal-acc, F1, and F2 scores. Similarly, we obtain the values of each of these accuracy measures after using the oversampling techniques; SMOTE, ADASYN, and SVM-SMOTE, corresponding to each classifier. We then use a one-tailed t-test to test the null hypothesis which states that there is no improvement in the results for a given measure by using oversampling techniques. If the p-value is smaller than 0.05 then we reject the null hypothesis and conclude that the improvement is significant. Table 4.5 presents the p-values of a one-tailed t-test for improvement of prediction results under different measures after using the oversampling techniques.

Table 4.5: One-tailed t-test when averaging over targets

	p-values		
	SMOTE	ADASYN	SVM-SMOTE
AUC-ROC	**0.037**	**0.047**	**0.029**
AUC-PR	0.095	0.135	**0.042**
Bal-acc	**0.037**	**0.047**	**0.029**
F1	**0.015**	**0.023**	**0.012**
F2	**0.013**	**0.02**	**0.015**

Results provided in Table 4.5 show that there is a significant improvement in prediction for the three oversampling methods for AUC-ROC, Bal-acc, F1 score, and F2- score, and in some cases for AUC-PR. Regarding AUC-PR, only SVM-SMOTE provides a significant improvement compared with unbalanced data, after averaging on all targets.

In the next part, we analyze the results for each target separately but averaging over the classifiers. This makes sense since we saw that the results of different classifiers are rather similar.

4.2.3.3 Results by target, averaging over classifiers

Figures 4.10 to 4.14 represent the results when we average each quality measure over the five classifiers for each target, using imbalanced and balanced data (with different balancing methods) respectively. The horizontal axis represents the different targets predicted. The vertical axis represents the percentage of scores when showing results for F1 and F2 measures and the percentage of increment or decrement from the baseline results when showing results for other measures.

Regarding AUC-ROC, AUC-PR, and balanced accuracy (Figures 4.10 to 4.12), using balanced data provides better results than imbalanced data for 10 targets out of 12 whatever the balancing method (the prediction of the variables *demence* and *ADLinf5* is not, or very slightly, improved for those measures). Let's also remark that the prediction of the variable *newHypoT* is hardly better than the baseline, whatever the data set. Looking more in detail, this result also not depends on

Figure 4.10: Percentage of increment or decrement from the baseline results regarding AUC-ROC when averaging over classifiers for each target.

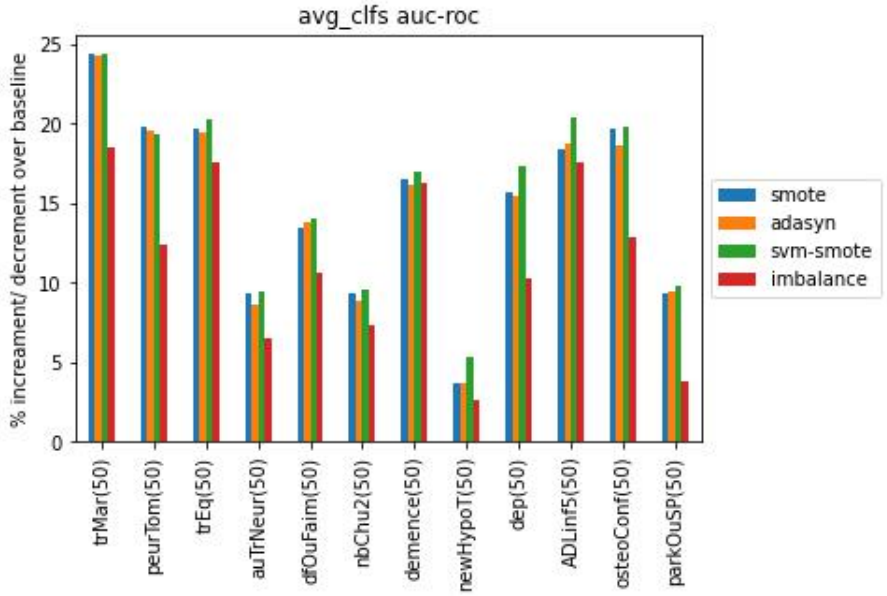


Figure 4.11: Percentage of increment or decrement from the baseline results regarding AUC-PR when averaging over classifiers for each target.

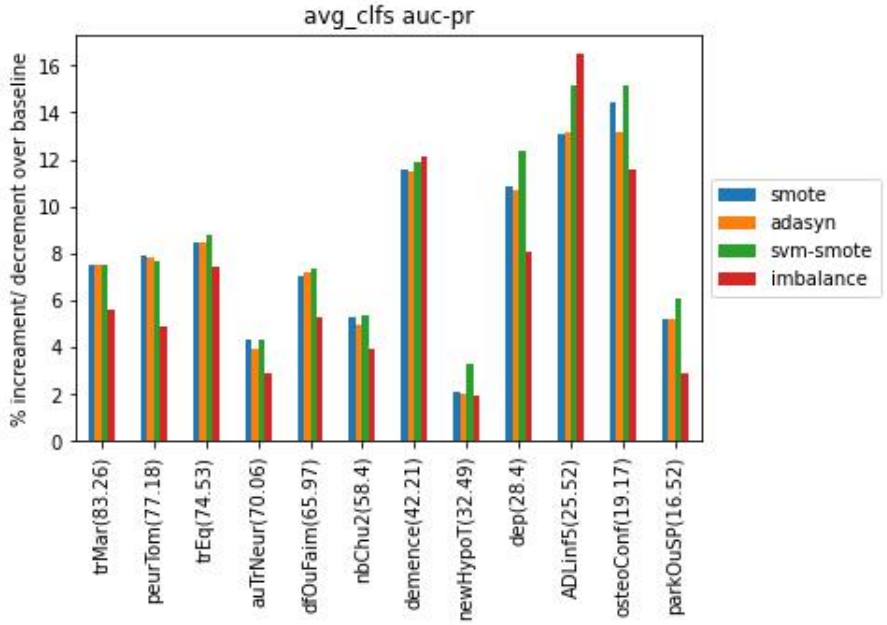


Figure 4.12: Percentage of increment or decrement from the baseline results regarding balanced accuracy when averaging over classifiers for each target.

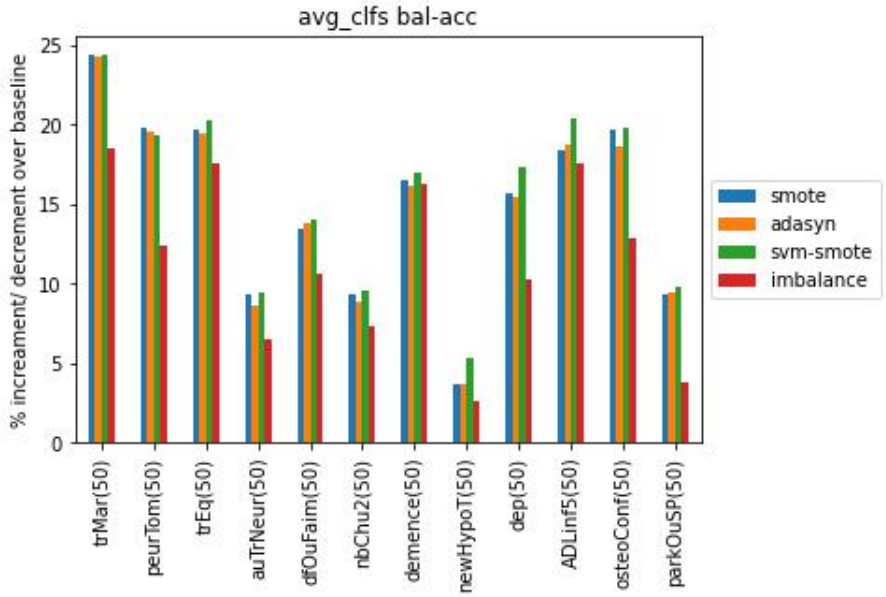


Figure 4.13: Percentage of F1 score when averaging over classifiers for each target.

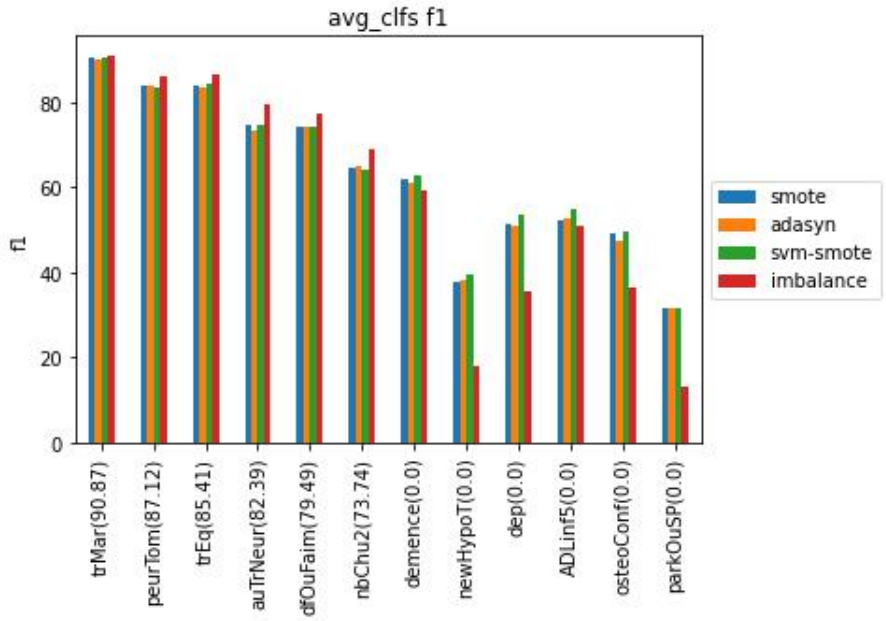
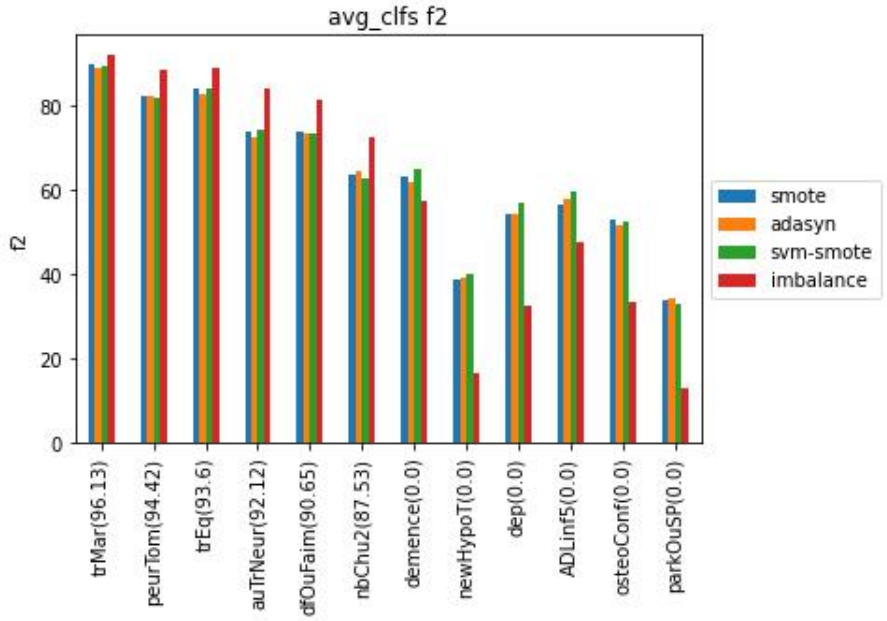


Figure 4.14: Percentage of F2-score when averaging over classifiers for each target.



the classifier; thanks to a discussion with an expert, it appears that some important variables that could help to predict hypotension are not part of the 45 selected variables. For these three measures (AUC-ROC, AUC-PR, and balanced accuracy), the balancing using SVM-SMOTE most often provides slightly better improvement than SMOTE and ADASYN.

Regarding F1 and F2 scores, (Figures 4.13 and 4.14) clearly shows the difference between targets for which the majority class is 1 (on the left), and for which majority class is 1, on the right. As mentioned above, F1 score is usually used for targets with a majority class 0. For the targets in groups with majority class 0, the use of oversampling method clearly improves the F1 score as well as F2-score. Furthermore, in order to evaluate the significance of the improvement after using oversampling techniques, we present the p-values as follows.

4.2.3.4 Statistical tests when averaging over classifiers

We use a t-test to check the significance of improvement in the results for each of the 12 target variables after using the oversampling techniques, averaging over all the classifiers. We use classification techniques, SVM, LR, RF, BN, MNB, and ANN, to classify target variables for the original (imbalance) dataset and obtain the values of evaluation measures; AUC-ROC, AUC-PR, Bal-acc, F1, and F2-scores for each target variable. Similarly, we obtain the values of each of these evaluation measures after using the oversampling techniques, SMOTE, ADASYN, and SVM-SMOTE, for each of the target variables. After averaging the values of the evaluation measures of different classifiers, we use a one-tailed t-test to test the null hypothesis which states that there is no improvement in the results by using oversampling techniques. If the p-value is smaller than 0.05 then we reject the null hypothesis and conclude that the improvement is significant. Table 4.6 presents p-values for each of the oversampling techniques using all evaluation measures.

It is clear from Table 4.6 that there is a significant improvement when comparing AUC-

Table 4.6: One-tailed t-test when averaging over classifiers

	p-values		
	SMOTE	ADASYN	SVM-SMOTE
AUC-ROC	**2.13E-4**	**2.63E-4**	**2.12E-5**
AUC-PR	**0.021**	**0.031**	**0.001**
Bal-acc	**2.13E-4**	**2.63E-4**	**2.12E-5**
F1	0.064	0.079	0.051
F2	**0.014**	0.134	0.103

ROC, AUC-PR, and balanced accuracy for all the target variables when averaging over all classifiers used. Regarding F1 score, when using smote and adasyn the difference from using the imbalanced data is significant with 93% and 92% significance levels respectively. However, when using svm-smote the difference is significant with a 94% significance level. When considering F2-score for comparison, we can see that using smote gave a significant difference and when using adasyn and svm-smote, the difference is significant with 86% and 89% significance levels respectively. The fluctuation of significant difference when comparing F1 and F2-scores may be due to the dominance of results for the targets for which the majority class is 1 as we can see in Figures 4.13 and 4.14.

4.2.4 Summary of results using iteration 1

We have first presented the results in order to answer the question: should we use a specific subset of variables or the complete set of variables when evaluating a given target? From the results presented, we can see that there is no big difference in using the two to predict the risk factors for falls because as shown in this study the results using a complete set of variables are as good as the other. Also, in real-life situations, the number of observations can be different for each patient. So for each new patient, we have to learn a new model using the available information which can be time and memory consuming.

Furthermore, to evaluate the quality of prediction results based on partial observations, we have compared the results using BN with other classifiers namely, LR, DT, RF, and SVM. Based on those results we can conclude that BN's predictive performance is equivalent to that of the other classifiers implemented since it yields predictions that are approximate of a similar standard for various target variables, varying somewhat depending on the target risk factor for falls. The results have also shown that when using partial observations, the quality of the prediction in terms of F1 score is clearly better than the baseline for only 4 target variables.

In addition, we have discussed the problem of classification with imbalanced data and analyzed the impact of three oversampling methods SMOTE, SMOTE-SVM, and ADASYN. In order to see the difference when using original imbalanced data versus the data after balancing with given oversampling methods, we have compared the results using several classifiers namely Logistic Regression, Random Forest, Artificial Neural Networks, Naive Bayes, and Bayesian Networks. To evaluate the performance of different classifiers, we use several measures: Balanced Accuracy, F1 score, F2-score, the area under the Precision-Recall curve, and the area under the Receiver Operating Characteristic curve. We have presented the results summarised by the classifier (averaging over targets) and by target (averaging over classifiers).

As observed, the results of different classifiers used on Lille's data set when averaging over all targets are rather similar for all the considered measures, except the Bayesian network whose results using imbalanced data are comparable with results of other classifiers when using

balanced data. Similarly, the results of different targets when averaging over all classifiers shows the improvement in each type of measure used when using the balanced data with oversampling methods versus using imbalanced data. In addition, we also see that SVM-SMOTE gives slightly better results as compared to other oversampling techniques.

Furthermore, the one-tailed t-test confirms our findings that when averaging over targets, there are significant improvements in AUC-ROC, AUC-PR, F1 score, and balanced accuracy for all classifiers when using oversampling methods. Also from the one-tailed t-test when averaging over classifiers, we can conclude that there are significant improvements in AUC-ROC, AUC-PR, and balanced accuracy when using oversampling methods. For F1 score the results are dominated by target variables with a majority class 0.

Also, recall that fall prevention requires to provide a small number of recommendations depending on the risk factors present for a person. Thus the evaluation of risk factors is the basis of fall prevention. Also, in real life, imbalanced data sets are very common. So based on the results and discussion presented, we propose using balancing techniques, specifically SVM-SMOTE, as a possible solution to the data imbalance problem.

Points to remember after analyzing the results using iteration 1

- ✓ We decided to use a complete set of variables for the evaluation of a given target risk factor
- ✓ BN's predictive performance is as good as the other classifiers used
- ✓ Using SVM-SMOTE is a better choice when the data is imbalanced except for the BN classifier

4.3 Results using iteration 2

In this section, we describe the results when using data obtained after the second iteration of data preprocessing. In that aim, we first describe whether should be using a specific subset or a complete set of variables to predict the target risk factor for falls. After that, we show the results of different structure learning algorithms for the BN model. Furthermore, as we learned from the results of the previous iteration, using SVM-SMOTE before training the classifier improves the performance of a given classifier. Now, we further investigate if this claim is still true when using BN with our data set as we have a different set of input variables in this iteration. Additionally, in order to evaluate the quality of prediction results we present the comparison of the results of BN with other classifiers.

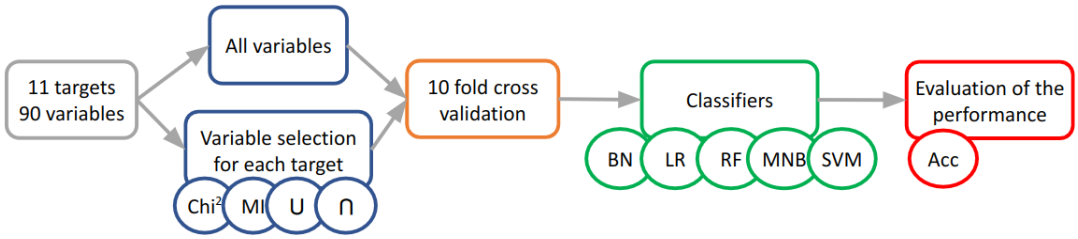
4.3.1 Should we use all variables or a specific subset

In this iteration, our data set consists of information on 90 variables listed in Tables 3.7 to 3.12 (see Chapter 3, section 3.4.2). As the literature suggests, not all the features are important to evaluate specific risk factors. With that aim, in this section, we discuss the interest in doing feature selection from this subset of variables extracted from our data during the second iteration.

Should we do feature selection or not?

Here the objective is to evaluate the interest in doing feature selection. To that aim, we use chi-square (chi2) score (scores with associated p-values) and mutual information (mi) score to evaluate the importance of a feature for a given target risk factor. Figure 4.15 represents the schematic diagram of the methodology used.

Figure 4.15: Methodology used when evaluating the interest of using all variables versus a specific subset of variables in iteration 2



Methodology:

- use chi2 and mi score for each target to obtain 2 sorted lists of variables selected for each risk factor
- threshold used to select a variable using chi2: we select a variable for a given target if its p-value is less than 0.05 that is we select a variable with 95% confidence. We called this set of selected variables "chi2".
- threshold used to select a variable using mi: we select a variable for a given target if its mi score is greater than 0. We called this set of selected variables "mi".

Now, **for a given target "y" we have the following set of selected variables**:

- chi2: variables selected using chi2 score
- mi: variables selected using mi score
- intersection: set of variables which are commonly selected using chi2 and mi
- union: set of variables which are selected using either chi2 or mi

Classifiers used to evaluate the effect of feature selection:
support vector machine (svm); logistic regression (lr); random forest (rf); bayesian networks (bn); multinomial naive bayes (mnb)

Results:
Table 4.7 represents the average increment or decrement in the accuracy from baseline for each classifier over all the targets. Here, baselineAcc represents the accuracy of the baseline classifier meaning the prevalence in the M1 group and (1- prevalence) in the M0 group, and the number presented in all other columns is the increment or decrement in accuracy using a given classifier from baseline accuracy. It shows that when we use the subset of variables selected using chi-square, SVM and LR give the best prediction: 6.91% and 6.83% more accurate predictions than the baseline classifier respectively. whereas when we use all variables, RF and BN give the best prediction (6.4% and 1.33% more accurate predictions than the baseline classifier respectively). Moreover, MNB gives the best prediction (3.36% more accurate predictions than the baseline classifier respectively) when using the intersection.

Similarly, Table 4.8 represents the average increment or decrement in the accuracy from baseline when predicting a given target after averaging over all the classifiers used. It shows that for 4 out of 11 targets we have the best prediction when using the subset of variables

Table 4.7: Average percentage increment or decrement in the accuracy from baseline for each classifier over all the targets

	all	**chi2**	**mi**	**intersection**	**union**
baselineAcc	68.43	68.43	68.43	68.43	68.43
SVM - baselineAcc	6.5	**6.91**	6.46	6.41	6.59
LR - baselineAcc	5.94	**6.83**	6.06	6.31	6.38
RF - baselineAcc	**6.4**	5.33	5.95	4.72	6.12
BN - baselineAcc	**1.33**	1.3	1.31	1.2	1.07
MNB - baselineAcc	2.62	3.24	3.06	**3.36**	2.89

selected using chi-square and for 3 out of 11 we have the best prediction when using the subset of variables selected using mutual information; for 2 out of 11 using all variables, for 1 out of 11 using the intersection, and 1 out of 11 using the union.

Table 4.8: Average percentage increment or decrement in the accuracy from baseline when predicting a given target over all the classifiers used

	all	**chi2**	**mi**	**intersection**	**union**
trMar	1.38	1.53	1.4	**1.65**	1.32
peurTom	3.16	**3.42**	2.53	2.74	3.06
sarcopen	2.91	**3.04**	2.93	2.81	3.03
trEq	4.78	**5.03**	4.46	4.37	4.85
nbchu2	**4.81**	4.37	4.12	3.49	4.34
ADLlt5	**0.21**	0.09	-0.19	-0.09	-0.19
demence	8.82	9.27	8.82	8.49	**9.45**
hypotenO	-1.77	-1.66	**-1.05**	-1.17	-1.77
dep	10.57	**11.16**	11.02	10.97	10.98
osteopor	10.77	11.09	**11.16**	10.56	10.74
parkOuSP	4.5	4.57	**5.05**	4.6	4.86

From Tables 4.7 and 4.8, we can see that no method for the variable selection clearly outperforms the other. We summarize these results in Table 4.9. It represents, the average percentage difference in accuracy when using a classifier with all variables versus a specific subset selected using a given feature selection method. Here, we see that for using the subset of variables selected by chi-square score we have 0.16% more accurate results than using all variables, whereas for using the subset of variables selected by mi score we have 0.1% more accurate prediction as compared to using all variables. From these results, we can conclude that on average there is no difference when using a specific subset of variables or all variables when predicting a given target.

4.3.1.1 A specific BN for each target versus a single BN for all targets

So far we have seen that using a specific subset is not very different than using a complete set of variables. Since our focus is on using BN, we investigate this question further with a different set of structure learning algorithms for BN. Furthermore, in the previous section, we used chi-square, mutual information, their intersection, and their union to select a specific subset of variables, but now we use chi-square from this list for the selection of variables because the number of variables selected is always less and if we repeat this process the number of variables

Table 4.9: Average percentage difference in accuracy when using a classifier with all variables vs specific subset selected using a given feature selection method

	Avg difference
all	0
chi2	**0.16**
mi	0.01
intersection	-0.16
union	0.05

selected is always the same. Table 4.10 represents the number of variables selected for a given target using a given feature selection method.

Table 4.10: Number of variables selected for a given target using a given feature selection method

Targets	chi2	mi	intersection	union
trMar	45	63	36	72
peurTom	30	46	26	50
sarcopen	35	50	23	62
trEq	43	53	28	68
nbchu2	28	48	13	63
ADLlt5	47	67	39	75
demence	46	58	32	72
hypotenO	19	48	13	54
dep	22	44	14	52
osteopor	33	52	26	59
parkOuSP	32	59	23	68

In addition, we have selected 3 target risk factors to evaluate the difference between using a single BN with all variables to predict all targets or a specific BN using a specific subset of variables to predict a given target. To select these targets we used the results presented in the previous section. We have the following criteria to select a given target:

1. **good prediction results:** we have selected the target for which we have the overall best results given any classifier with any measure.

2. **different percentages or prevalence:** we selected 3 targets whose prevalence is not in the same range to have more generality in our selection.

3. **interesting to predict:** we have selected those variables which are interesting and important to predict because they are very important risk factors, often difficult to diagnose (or often undiagnosed).

4. **the number of variables selected using chi-square:** we have selected those variables for which the number of variables selected by the chi-square test is less. The objective here is to have a small number of variables with maximum information.

Now we present in detail the procedure to select these 3 targets for our analysis. We have the following 6 variables with the negative class as the majority class: *ADLlt5*, *dep*, *demence*, *osteopor*, *hypotenO*, *parkOuSP*. Based on criteria 1, we have selected *dep*, *demence*, and *osteopor*

because they are the ones with good results for any given measure and given classifiers, and finally, we have selected *dep* and *demence* from this group because they are very important to predict, generally not diagnosed and they have a small number of variables in the subset selected using chi-square. Furthermore, we have 5 variables in this group: *trMar*, *trEq*, *peurtom*, *sarcopen*, *nbchu2*. Based on criteria 1, we have selected *trEq* from this group because it has the best results than others for a given classifier using a given measure.

In order to see the difference when using a single BN model using all variables for all targets versus a specific BN model using a specific subset for a given target we compared the balanced accuracy score for a given model and for a given target. With that aim we used 13 different structure learning algorithms for BN, namely: greedy hill climbing (GHC) and Tabu list (Tabu), each with scores: Akaike information criterion (AIC), Bayesian Information Criterion (BIC), Bayesian-Dirichlet scoring (BD), Bayesian-Dirichlet equivalent uniform (BDeu), K2 score, and log2 likelihood ratio test (Log2); and Multivariate Information based Inductive Causation (MIIC).

Table 4.11 represents the percentage difference in balanced accuracy score using all variables minus using a specific subset selected using chi-square when predicting a given target using a given algorithm. For example, the number -1.32 (see first row, the second column in the Table) means that the balanced accuracy score is 1.32% less when using all variables as compared to using a specific subset selected by the chi-square method when predicting trEq using the BN model learned using GHC with AIC score.

Table 4.11: Difference in balanced accuracy between using all variables and using variables selected with chi2 when predicting *trEq*, *demence*, and *dep* using BNs learned with different structure learning algorithms

Method / target	trEq	demence	dep
GHC_AIC	-1.32	-1.87	2.01
GHC_BIC	0.43	0.72	0.12
GHC_BD	0.31	-0.01	0.1
GHC_BDeu	0.32	0.21	-0.06
GHC_K2	-1.84	-0.08	-0.83
GHC_Log2	0.32	0.21	-0.06
Tabu_AIC	1.17	-1.81	-1.49
Tabu_BIC	0.57	0.41	-0.16
Tabu_BD	-0.95	-0.25	-0.24
Tabu_BDeu	0.18	-2.21	-0.37
Tabu_K2	-1.37	-0.6	-0.6
Tabu_Log2	0.18	-2.21	-0.37
MIIC	-1.33	-1.4	-1.7
mean	**-0.26**	**-0.68**	**-0.28**
std	**0.96**	**1.07**	**0.89**

We can see from Table 4.11 that when predicting *trEq*, we have a difference of less than 1% for 8 out of 13 algorithms of which for 7 of them using a single BN using all variables is better. For the remaining 5 out of 13 algorithms, the difference is always between 1% to 2%. Also, the mean difference in balanced accuracy using all algorithms is 0.26% which is very close to no difference with a standard deviation of 0.96%. Similarly, when predicting *demence*, we have a difference of less than 1% for 8 out of 13 algorithms of which for 4 of them using a single BN is better and for 4 of them using specific BN is better. For the remaining 5 out of 13 algorithms, we have a difference for 3 algorithms between 1% and 2%, and for the remaining 2 algorithms

we have a difference equal to 2.21%. Also, the mean difference using all algorithms is 0.68% with a standard deviation of 1.07%. Likewise, when predicting *dep*, we have a difference of less than 1% for 10 out of 13 algorithms. for the remaining 3 algorithms, we have differences of 2.01%, -1.49%, -1.7% respectively. Also, the mean difference using all algorithms is -0.28% with a standard deviation of 0.89%. Furthermore, in order to see if the difference when using BN with all variables versus a specific subset of variables selected using "chi2" is significant or not we perform a one-tailed t-test. We have the null hypothesis as there is no significant difference and with the alternative hypothesis as there is a significant difference when using a BN with all variables versus a specific subset of variables selected using the chi-square method. Table 4.12 represents the p-values when comparing balanced accuracy for a given target. Here, if the p-value is less than 0.05 we reject the null hypothesis. From this Table, we can say that there is no significant difference when predicting *trEq* and *dep*. However, the difference is significant when predicting *demence* but as we can see from Table 4.11 the difference is not too much. After seeing these results we propose the use of a single BN model using all variables to predict all target risk factors for falls.

Table 4.12: one-tailed t-test for difference in using all variables versus specific subset for BN

	p-values		
	trEq	**demence**	**dep**
Bal-acc	0.178	0.02	0.139

Points to remember from this section

- ✓ We decided to use a single BN model using a complete set of variables for the evaluation of the target risk factors

4.3.2 BN Structure learning

So far we have seen that it is better to use a single BN model to predict all the target risk factors from our data set. But it is not very clear which algorithms should we use to learn the structure of the BN model. As discussed in the earlier chapter, there are many possible algorithms to learn the structure of the BN model. To decide which learning algorithms to use, we focus on the following two criteria: (1) the predictive performance; (2) the understandability of the graph. With that aim, in this section, we present the comparison of prediction results using different structure learning algorithms. We use the following 13 different structure learning algorithms for BN, namely: greedy hill climbing (GHC) and Tabu list (Tabu), each with scores: Akaike information criterion (AIC), Bayesian Information Criterion (BIC), Bayesian-Dirichlet scoring (BD), Bayesian-Dirichlet equivalent uniform (BDeu), K2 score, and log2 likelihood ratio test (Log2); and Multivariate Information based Inductive Causation (MIIC). Furthermore, we present our approach to get a more understandable structure with the help of domain experts.

4.3.2.1 Comparison of performance for different algorithms

Tables 4.13 and 4.14 represent the percentage increment from baseline balanced accuracy when predicting a given target using a BN model with a given structure learning algorithm. For example, the number 30.58 (first row, first column in Table 4.13) means when predicting trMar using a BN model constructed using GHC structure learning algorithms with AIC score the

improvement in balanced accuracy from baseline score is 30.58% that is the actual balanced accuracy score is 80.58%. At the bottom of both Tables, we show the mean and standard deviation over all algorithms for each given target risk factor. The results suggest that there is only a small variation in balanced accuracy for a given target using the different BN structure learning algorithms. The standard deviation from the mean is less than 1% except for the variables *demence* (1.1%) and *osteopor* (1.12%). Since the predictive performance of all the BN models with different structure learning used is rather similar, we focus on our second criterion about the understandability of the BN graph.

Table 4.13: Percentage increment from baseline balanced accuracy score (baseline bal-acc = 50%) when predicting a given target with the positive class as majority class using a BN model with different structure learning algorithms

algorithm	trMar	peurTom	sarcopen	trEq	nbchu2
ghc_aic_base	30.58	28.87	11.13	23	6.75
ghc_bic_base	30.12	29.22	10.27	22.4	6.04
ghc_bd_base	30.75	27.89	9.91	23.22	7.16
ghc_bdeu_base	29.39	27.5	9.86	22.97	8.04
ghc_log2_base	29.39	27.5	9.86	22.97	8.04
ghc_k2_base	27.82	29.15	9.55	24.06	7.81
tabu_aic_base	31.21	27.94	10.64	23.09	6.54
tabu_bic_base	29.53	29.18	10.27	21.42	6.04
tabu_bd_base	29.4	28.16	10.02	22.83	6.98
tabu_bdeu_base	29.99	27.15	9.61	23.1	7.97
tabu_log2_base	29.99	27.15	9.61	23.1	7.97
tabu_k2_base	30.67	26.67	9.72	23.96	7.81
miic_base	28.74	26.75	11.78	22.8	8.06
mean	**29.81**	**27.93**	**10.17**	**22.99**	**7.32**
std	**0.88**	**0.89**	**0.64**	**0.62**	**0.75**

4.3.2.2 Towards an understandable BN graph: using mandatory arcs on local BN around each target

So far we have seen that using different structure learning algorithms for the BN model does not make any difference in terms of predictive performance. As a reminder, one of the main objectives of our work is to support the general practitioner to evaluate the risk of falls. With that aim, we want to provide a model which can be understandable by the final user such that the results can be explained. So, we want to focus on the relationship between the variables in our BN model to make the graph more understandable.

Furthermore, several configurations are possible between two variables A and B:

1. A is a cause of B (or B is a cause of A)
2. A and B are dependent but A is not a cause of B and B is not a cause of A (this can occur when another variable is a common cause of A and B)
3. A and B are "independent"

In a causal graph, the first case is represented by an arc A–> B in the BN network graph. However, usual BN structure learning algorithms can not learn the causal direction of the arcs. It

Table 4.14: Percentage increment from baseline balanced accuracy score (baseline bal-acc = 50%) when predicting a given target with the negative class as majority class using a BN model with different structure learning algorithms (continued)

algorithm	ADLlt5	demence	hypotenO	dep	osteopor	parkOuSP
ghc_aic_base	20.04	16.34	8.66	30.03	24.8	30.9
ghc_bic_base	18.92	12.7	8.96	29.98	25.44	30.85
ghc_bd_base	19.27	16.02	7.43	29.55	26.74	30.93
ghc_bdeu_base	19.12	15.3	8.15	30.1	27.72	30.78
ghc_log2_base	19.12	15.3	8.15	30.1	27.72	30.78
ghc_k2_base	19.25	15.59	8.77	30.11	26.47	31.01
tabu_aic_base	19.1	16.13	8.08	30.36	26.25	30.41
tabu_bic_base	18.9	12.95	8.96	30.34	24.75	30.85
tabu_bd_base	18.22	15.03	7.93	30.28	27.37	30.99
tabu_bdeu_base	19.06	14.42	7.89	29.9	27.24	30.16
tabu_log2_base	19.06	14.42	7.89	29.9	27.24	30.16
tabu_k2_base	20.8	15.88	7.41	29.82	26.12	30.81
miic_base	19.05	14.56	7.9	27.83	24.37	30.47
mean	**19.22**	**14.97**	**8.17**	**29.87**	**26.33**	**30.7**
std	**0.59**	**1.1**	**0.5**	**0.63**	**1.12**	**0.29**

is thus possible to get an arc A–>B when the causality is in the opposite direction. In the second case, when the common parent is not part of the set of considered variables, the dependence between the variable is also represented by an arc between the two variables, whereas this arc does not represent causality. Finally, the third case should result in no arc between A and B in the BN graph.

The property of completeness means that any common cause of several variables does belong to the set of considered variables. In our case, the hypothesis of completeness is probably not verified, meaning that the second case may occur. Moreover, given some hypotheses, it is sometimes assumed that the arcs involved in a V-structure can be causal. However, in our case, we clearly found several examples of V-structure whose arcs are clearly not causal (for example, several arcs toward the variable *sexe* are certainly not causal ones since no variable can be considered as a cause of the *sexe*). With that context, in this study, we aim to get a causal graph since it contributes to the understandability of the graph by the experts. In that aim, we worked with domain experts and defined some mandatory arcs to be included in the structure of the BN model learned. Since we had limited time with the domain expert and the number of arcs learned by any given algorithm is very high. We focus on the possible arcs around each target risk factor. In order to define causal arcs we focus on the Markov blanket for a given target. Markov blanket for a given node X is the set of nodes MB(X) such that X is independent of the rest of the nodes given MB(X).

Also, we first define the mandatory arcs and use the 13 structure learning algorithms previously mentioned in order to see whether using this list of mandatory arcs makes any difference in predictive performance for any given model. The methodology we use to define mandatory arcs is as follows:

- select the nodes that are present in the Markov blanket of a given target node for a given model (produced by one of the BN structure learning algorithms)
- count the presence of a given node in the MB obtained by the different structures for each

target (list given in Tables 4.15 and 4.16)

- select the nodes that are present in most of the MBs for a given target
- from the selected nodes, define the relationship with the target node using the general knowledge with the help of the domain expert. We defined the mandatory arcs only for the direct cause or direct consequences of each target variable. Meaning that some variables of the MB of a risk factor for falls have not been included in the list of mandatory arcs. Figures 4.16 and 4.17 show the local BN graph around each target variable, composed of the nodes of the Markov blanket with mandatory arcs for each target. (see B.1 for the full BN graph learned)

Table 4.15: list of nodes in the Markov blanket of each target in group M1 and their number of occurrences regarding the 13 BN structure learning algorithms

trMar		**peurTom**		**trEq**		**sarcopen**		**nbchu2**	
var	**count**	**var**	**count**	**var**	**count**	**var**	**count**	**var**	**count**
sarcopen	13	sortir	13	trProp	13	trMar	13	a1AntiDep	13
trEq	13	evitsort	13	pbPodo	13	aidTecMa	13	TUGgt20	13
aidTecMa	13	sexe	7	trMar	13	myopat	13	agonDopa	1
aideHum	13	sortSeul	7	basCont	13	vitSeul	1	trMar	1
apUniGt5	13	LSAi4	5	trAudit	11	agonDopa	1	parkOuSP	1
ataxPeri	11	montDesc	4	age4	11	antiHT	1	ADLlt5	1
ataxCer	11	a1medSed	1	apUniGt5	8	osteopor	1	gt1hSol	1
vitMar	10	dep	1	sexe	6	fracture	1	lipoth	1
neurPath	6	pbPodo	1	trVision	2	sort	1		
difWC	6	htNivEtu	1	neurPath	1	BMI_lt19	1		
trProp	5	osteopor	1	htNivEtu	1	evitsort	1		
TUGgt20	3	alcool	1	hypotenO	1	trAnHTA	1		
sexe	1	akines	1	osteopor	1				
pathUro	1	traAnOst	1	aideHum	1				
evitsort	1			nFrac4	1				
agonDopa	1								
trAudit	1								
syndCer	1								
trembl	1								
parkOuSP	1								
ADLlt5	1								
nbchu2	1								

Now in order to see the difference when using a BN model with structure learned with or without mandatory defined arcs we compare the balanced accuracy when predicting a given target. Tables 4.17 and 4.18 represent the percentage difference in balanced accuracy when using mandatory arcs and when not using mandatory arcs. For example, the number 1.27 (first row, first column in Table 4.17) means for predicting trMar when we use BN model with structure learned using GHC with AIC score and with predefined mandatory arcs, we have 1.27% better balanced accuracy than using BN without predefined mandatory arcs.

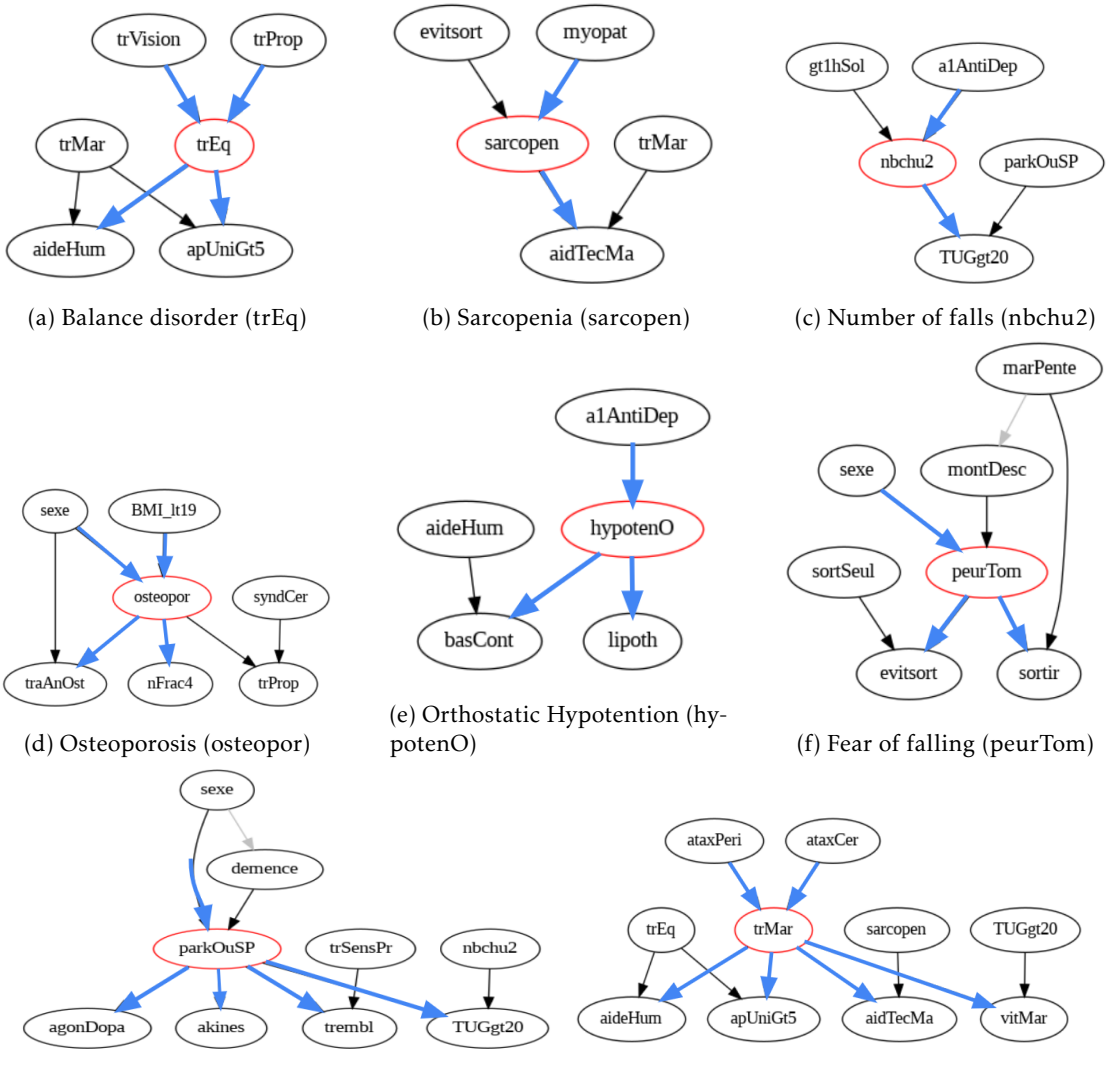
In order to summarize these results, we also present the mean and standard deviation over all algorithms used as well as all targets predicted. As we can see from Table 4.17, when averaging over all algorithms, for 2 out of 5 targets, the difference is positive, meaning a better balanced accuracy when using mandatory arcs, and for 3 out of 5 targets the difference is negative means using BN model without mandatory arcs is better. But the difference is always between 0%

Table 4.16: list of nodes in the Markov blanket of each target in group M0 and their number of occurrences regarding the 13 BN structure learning algorithms (continued)

demence		osteopor		hypotenO		dep		ADLlt5		parkOuSP	
var	**count**	**var**	**count**	**var**	**count**	**var**	**count**	**var**	**count**	**var**	**count**
sexe	13	sexe	13	a1AntiDep	13	a1AntiDep	13	sexe	13	agonDopa	13
htNivEtu	13	nFrac4	13	basCont	13	gt2psych	13	difWC	13	akines	13
ADLlt5	13	traAnOst	13	lipoth	5	sexe	4	demence	13	trembl	13
akines	13	pbPodo	11	sortir	2	aideHum	3	TUGgt20	13	alcool	11
conduit	13	conduit	6	pbPodo	1	hydrocep	3	pathUro	13	sexe	10
epilep	9	diabete	6	trEq	1	syndVes	3	sortSeul	13	neurolep	9
confusion	9	nbmed3	6	aideHum	1	ADLlt5	2	conduit	12	a1medSed	9
trembl	5	BMI_lt19	5	trembl	1	antiEmet	2	vitMar	10	trSensPr	7
aideHum	4	gt1antal	4			a1medSed	1	maisRet	10	difWC	6
malnut	4	malnut	4			pbPodo	1	LSAi4	10	antiEmet	2
tumCer	4	sortir	4			htNivEtu	1	akines	8	vitSeul	1
vitMar	1	dep	1			osteopor	1	a1AntiDep	3	trProp	1
medPsych	1	htNivEtu	1			alcool	1	evitsort	2	trMar	1
pbPodo	1	neurPath	1			peurTom	1	dep	2	ataxPeri	1
osteopor	1	alcool	1			akines	1	gt2psych	1	ADLlt5	1
basCont	1	peurTom	1			traAnOst	1	gt1hSol	1	nbchu2	1
evitsort	1	sort	1					agonDopa	1		
sortSeul	1	trAnHTA	1					trAudit	1		
traAnOst	1	trProp	1					trMar	1		
derivNit	1	trEq	1					aideHum	1		
		akines	1					trembl	1		
		basCont	1					a1medSed	1		
		derivNit	1					medPsych	1		
		a1medSed	1					aidTecMa	1		
		antiHT	1					parkOuSP	1		
		sarcopen	1					nbchu2	1		
		demence	1					apUniGt5	1		

Table 4.17: Percentage difference in balanced accuracy when predicting a given target using BN with mandatory arcs minus without mandatory arcs for a given structure learning algorithm

algorithm	trMar	peurTom	sarcopen	trEq	nbchu2	mean	std
ghc_aic	1.27	-0.23	-0.32	-0.18	1.41	**0.39**	**0.87**
ghc_bic	1.94	0.1	-0.65	0.2	1.42	**0.6**	**1.05**
ghc_bd	-0.67	-1.16	-0.61	-0.2	1.51	**-0.23**	**1.03**
ghc_bdeu	0.72	0.16	1.2	-0.95	0.66	**0.36**	**0.82**
ghc_log2	0.72	0.16	1.2	-0.95	0.66	**0.36**	**0.82**
ghc_k2	2.6	-1.7	0.57	-1.32	-0.4	**-0.05**	**1.72**
tabu_aic	0.43	-0.42	0.17	-0.22	0.4	**0.07**	**0.38**
tabu_bic	1.2	-2.26	-0.85	0.7	-1.32	**-0.51**	**1.43**
tabu_bd	1.28	-2.1	-0.57	-1.68	-0.95	**-0.8**	**1.31**
tabu_bdeu	0.27	-0.3	0.87	-1.65	-1.53	**-0.47**	**1.11**
tabu_log2	0.27	-0.3	0.87	-1.65	-1.53	**-0.47**	**1.11**
tabu_k2	-1.27	-1.27	0.54	-1.38	-1.95	**-1.07**	**0.94**
miic	1.71	-0.51	-0.46	-1.01	0	**-0.05**	**1.05**
mean	**0.81**	**-0.76**	**0.15**	**-0.79**	**-0.12**	**-0.14**	**0.67**
std	**1.05**	**0.85**	**0.76**	**0.78**	**1.24**	**0.94**	**0.21**



(a) Balance disorder (trEq)

(b) Sarcopenia (sarcopen)

(c) Number of falls (nbchu2)

(d) Osteoporosis (osteopor)

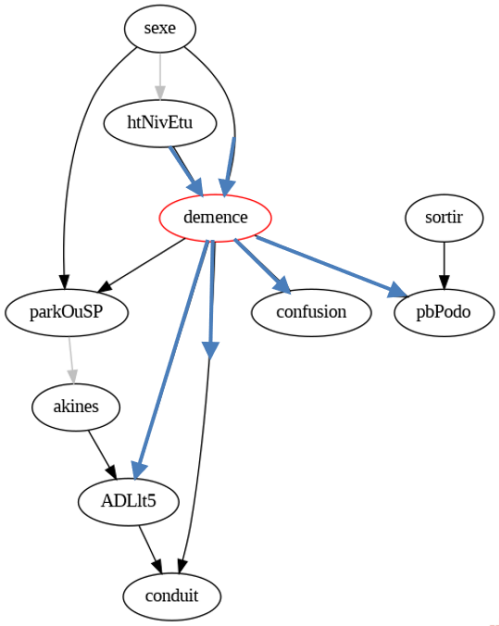
(e) Orthostatic Hypotention (hypotenO)

(f) Fear of falling (peurTom)

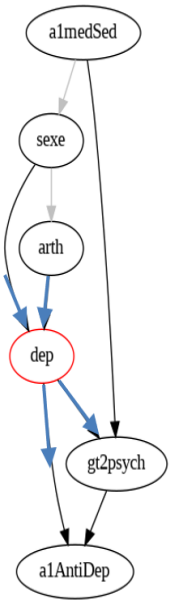
(g) Parkinson Syndrome (parkOuSP)

(h) Gate disorder (trMar)

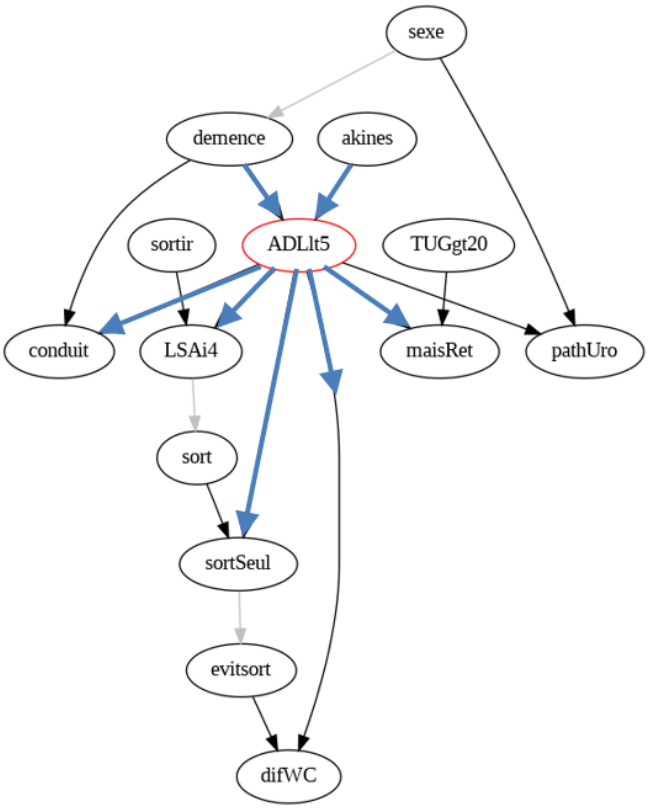
Figure 4.16: Markov Blanket with Mandatory arcs (blue line) for each target



(a) Dementia (demence)



(b) Depression (dep)



(c) Activities of daily living (ADLlt5)

Figure 4.17: Markov Blanket with Mandatory arcs (blue line) for each target (continued)

Table 4.18: Percentage difference in balanced accuracy when predicting a given target using BN with mandatory arcs minus without mandatory arcs for a given structure learning algorithm (continued)

algorithm	ADLlt5	demence	hypotenO	dep	osteopor	parkOuSP	mean	std
ghc_aic	1.98	0.17	-0.18	1.27	0.82	-0.33	**0.62**	**0.9**
ghc_bic	2.97	2.66	1.44	1.36	-0.98	-1.32	**1.02**	**1.8**
ghc_bd	0.08	-0.11	1.05	1.39	-0.82	-1.76	**-0.03**	**1.17**
ghc_bdeu	0.68	0.8	-0.11	0.92	-1.59	-1.88	**-0.2**	**1.25**
ghc_log2	0.68	0.8	-0.11	0.92	-1.59	-1.88	**-0.2**	**1.25**
ghc_k2	1.71	0.27	0.27	-0.15	-0.51	-0.86	**0.12**	**0.89**
tabu_aic	4.07	-1.75	0.25	0.89	-0.46	-1.45	**0.26**	**2.12**
tabu_bic	2.58	1.51	1.33	0.95	0.83	-1.39	**0.97**	**1.31**
tabu_bd	2.52	-0.48	0.69	-0.01	-1.71	-1.58	**-0.1**	**1.58**
tabu_bdeu	1.98	-0.4	0.45	0.25	-1.15	-0.93	**0.03**	**1.14**
tabu_log2	1.98	-0.4	0.45	0.25	-1.15	-0.93	**0.03**	**1.14**
tabu_k2	1.91	-1.17	2.05	0.2	-0.46	-1.46	**0.18**	**1.51**
miic	0.29	0	0	0	0.47	0.63	**0.23**	**0.28**
mean	**1.8**	**0.15**	**0.58**	**0.63**	**-0.64**	**-1.16**	**0.23**	**1.04**
std	**1.14**	**1.13**	**0.7**	**0.56**	**0.88**	**0.7**	**0.85**	**0.24**

and 1%. In addition, the standard deviation is also close to 1% which means that the results using different algorithms do not vary too much. Also, when averaging over these 5 targets, the difference is always less than or very close to 1%. The average difference is -0.14% with a standard deviation of 0.21% which means there is no big difference in predicting these five targets.

Similarly, we can see from Table 4.18 that when averaging over all algorithms, the difference is always less than 1% except for ADLlt5 (1.8%) and parkOuSP (-1.16%). Additionally, when averaging over all targets, the difference is always close to or less than 1%. Overall, the mean difference when using all algorithms and predicting all targets is 0.23% with a standard deviation of 0.24%.

From the results presented here, we can conclude that in terms of predictive performance when using the mandatory arcs there is no difference or very little difference as compared to not using the mandatory arcs. Hence we will use these mandatory arcs to learn the structure of the BN model.

4.3.2.3 Total number of arcs

So far we have seen that there is no difference in terms of predictive performance for all the 13 algorithms used to learn the structure of the BN model. However, the structure learned using a different algorithm is different as we can see from Tables 4.19 and 4.20 which represents the structural hamming distance (SHD) between all the algorithms used. Two perfectly identical structures have SHD equal to 0 and the more the SHD means the more different the structures are.

So in order to select an algorithm to learn the structure in our case we refer to using the algorithm which produces the less number of arcs in total. Table 4.21 represents the total number of arcs learned using each algorithm. We can see that when using GHC with the BIC score provides us with 121 arcs, hence we decided to use this algorithm to learn the structure of the BN model from our data.

Table 4.19: Structural hamming distance (SHD)

	ghc_aic	ghc_bic	ghc_bd	ghc_bdeu	ghc_k2	ghc_log2
ghc_aic	0					
ghc_bic	50	0				
ghc_bd	33	76	0			
ghc_bdeu	31	75	5	0		
ghc_k2	35	66	29	28	0	
ghc_log2	11	58	30	31	38	0
tabu_aic	65	103	86	87	89	74
tabu_bic	94	60	112	114	105	100
tabu_bd	77	113	70	71	89	81
tabu_bdeu	76	112	69	70	88	80
tabu_k2	77	105	86	87	75	83
tabu_log2	70	107	86	87	91	67
miic	129	141	141	141	141	133

Table 4.20: Structural hamming distance (SHD) (continued..)

	tabu_aic	tabu_bic	tabu_bd	tabu_bdeu	tabu_k2	tabu_log2	miic
tabu_aic	0						
tabu_bic	79	0					
tabu_bd	43	107	0				
tabu_bdeu	43	105	4	0			
tabu_k2	41	98	22	26	0		
tabu_log2	9	83	42	42	42	0	
miic	143	155	154	155	153	145	0

Table 4.21: Total number of arcs for a given algorithm for structure learning

	ghc_aic	ghc_bic	ghc_bd	ghc_bdeu	ghc_k2	ghc_log2	tabu_aic
#arcs	144	121	160	160	150	152	149

	tabu_bic	tabu_bd	tabu_bdeu	tabu_k2	tabu_log2	miic
#arcs	122	161	161	152	156	195

Points to remember from this section

- ✓ All the structure learning that we used provided very similar performance in terms of balanced accuracy
- ✓ We decided to use the greedy hill climbing (GHC) algorithm with a BIC score to learn the structure of our BN model
- ✓ We introduce some mandatory arcs involving the target variables based on general knowledge and with help from domain experts

4.3.3 Using BN with oversampling versus without oversampling

We have seen in previous sections that using a classifier after balancing the data gives better results. Since we focus on using BN and also the average results when using BN were good, we further investigate whether this claim is still true for BN. With that aim, in this section, we present the difference in balanced accuracy when predicting a given target when using oversampling versus without oversampling. As we have also seen that using SVM-SMOTE gives the best results in our case. We will focus on the difference when using SVM-SMOTE as the balancing technique. Figure 4.18 represents the schematic diagram of the methodology used: for each target, the 10-fold cross-validation method is applied; a new learning set is obtained after balancing the original one with SVM-SMOTE according to the imbalance ratio of the target; a first BN is learned based on the non-modified learning set and a second one based on the balanced learning set. The performance of these two BNs is compared regarding balanced accuracy.

Figure 4.18: Methodology used when evaluating the interest of using BN after balancing the data versus imbalanced data in iteration 2

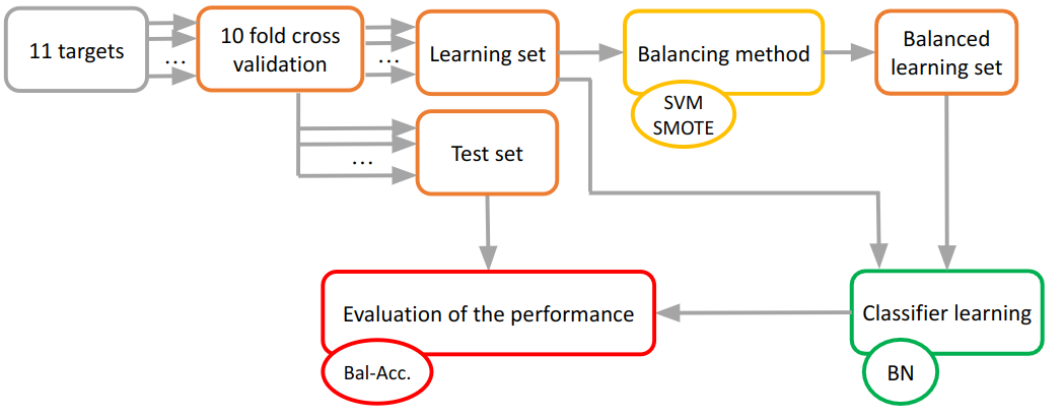
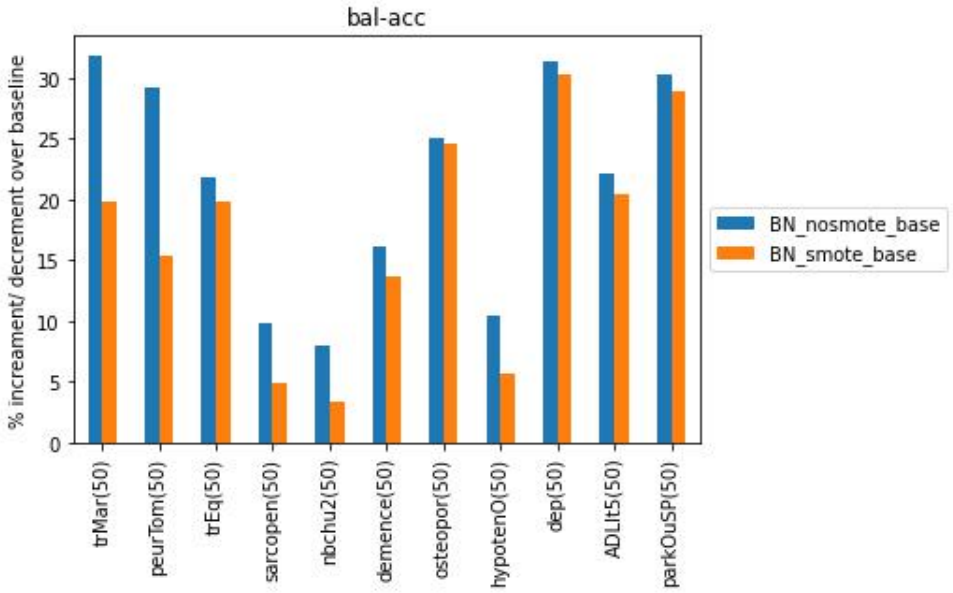


Figure 4.19 represents the balanced accuracy when predicting all targets using BN with SVM-SMOTE versus with imbalanced data.

We can clearly see from the Figure that using BN with imbalanced data is always better than using BN after balancing the data using SVM-SMOTE. The difference ranges from approximately 1% up to 14%. Balanced accuracy is a symmetric measure regarding positive and negative classes. The highest gap is obtained for two highly imbalanced targets (*trMar* and *peurTom*)

Figure 4.19: Percentage of increment or decrement from the baseline results regarding Balanced accuracy when using BN with imbalanced data versus using BN with balanced data (using SVM-SMOTE).



with a difference of more than 12% in favor of using the original imbalanced data set. For three targets, the difference is between 4% and 5% (*sarcopen*, *nbChu2*, *hypotenO*) and six targets show a difference in balanced accuracy between 0.5 and 3%, always in favor of using the imbalanced data set. Hence for our further analysis, we decide to use BN with imbalanced data. Now in the next section, we present the comparison of results when using BN versus all other classifiers mentioned earlier.

Points to remember from this section

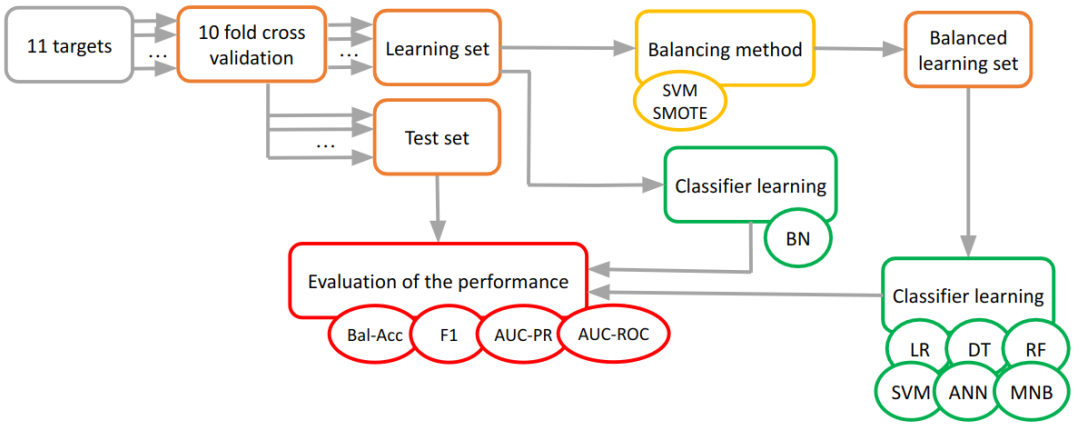
✓ We decided to use the BN model with original imbalanced data for the evaluation of a given target risk factor

4.3.4 Comparison of BN with other classifiers

So far we have seen that BN provides good prediction results. But in order to see the quality of prediction, in this section, we compare the results of prediction using BN with other usual classifiers namely: LR, DT, RF, SVM, ANN, and MNB. Here the BN is built using original imbalanced data and other classifiers are built after balancing the data using SVM-SMOTE (as discussed earlier). In addition, BN is learned using GHC with BIC score and mandatory arcs (as defined in previous section). Furthermore, we use balanced accuracy, F1 score, the area under ROC, and PR curves for evaluation. We use 10-fold cross-validation with a stratified strategy and an exhaustive search over specified parameter values for an estimator (gridsearchcv) for hyperparameter tuning. Figure 4.20 represents the schematic diagram of the methodology used.

Table 4.22 represents the list of parameters and their different values, and Tables 4.23 and 4.24 represent the values selected for each parameter for a given classifier after tuning of

Figure 4.20: Methodology used when comparing the predictive performance of BN with all other classifiers used in iteration 2



hyperparameter using gridsearchcv. We can see that the best value of a parameter for a given classifier differs for each target. For example: for the classifier logistic regression (lr), the best value of the parameter **C** is either 0.01 or 0.1 (and C=1 for the target *trEq*); and the best value for parameter **solver** is *liblinear* for all variables in group M0 and *lblgs* for variables in group M1, except the variable **trMar**.

Table 4.22: Possible combination of parameter for tuning

	Parameters	**Values**					
lr	**C**	0.01	0.1	1	10	100	
	Solver	lbfgs	liblinear	newton-cg	sag	saga	
dt	**criterion**	gini	entropy				
	max_depth	10	30	50	70	90	None
	max_features	log2	sqrt	None			
rf	**criterion**	gini	entropy				
	max_depth	10	30	50	70	90	None
	max_features	log2	sqrt	None			
	n_estimators	10	50	100	150	200	250
svm	**C**	0.1	1	10	100		
	kernel	linear	poly	rbf	sigmoid		
ANN	**batch_size**	10	20	40	60	80	100
	epochs	10	50	100			

Now we provide the comparison of results using different classifiers when predicting a given target. Figures 4.21 to 4.24 show the comparison of the performance of the BN with other classifiers regarding balances accuracy, F1 score, the area under the precision-recall curve and the area under the ROC curve. In those Figures, the target risk factors for falls are sorted according to their prevalence: variables of group M1 (prevalence more than 50%) are on the left and those in group M0 (prevalence less than 50%) are on the right. The variables in the middle have a better balance in their classes. Figure 4.21 represents the percentage increment or decrement when comparing balanced accuracy for a given classifier when predicting a given

Table 4.23: Selected parameter after tuning for each target in group M1

	Parameters	trMar	peurTom	trEq	sarcopen	nbchu2
lr	**C**	0.1	0.1	1	0.1	0.01
	solver	liblinear	lbfgs	lbfgs	lbfgs	lbfgs
dt	**criterion**	gini	entropy	gini	entropy	entropy
	max_depth	10	30	10	30	10
	max_features	None	None	None	sqrt	None
rf	**criterion**	gini	gini	entropy	entropy	gini
	max_depth	30	30	30	50	10
	max_features	log2	log2	sqrt	log2	log2
	n_estimators	100	250	250	250	100
svm	**C**	1	1	1	1	1
	kernel	rbf	rbf	rbf	rbf	rbf
ANN	**batch_size**	100	10	10	20	100
	epochs	50	10	10	10	10

Table 4.24: Selected parameter after tuning for each target in group M0

	Parameters	demence	osteopor	hypotenO	dep	ADLlt5	parkOuSP
lr	**C**	0.1	0.01	0.01	0.1	0.1	0.1
	solver	liblinear	liblinear	liblinear	liblinear	liblinear	liblinear
dt	**criterion**	gini	entropy	entropy	gini	gini	gini
	max_depth	10	10	10	10	30	30
	max_features	None	None	log2	None	sqrt	None
rf	**criterion**	gini	gini	entropy	entropy	entropy	entropy
	max_depth	10	30	30	10	30	30
	max_features	log2	sqrt	log2	sqrt	log2	None
	n_estimators	200	250	150	200	100	100
svm	**C**	0.1	1	1	1	100	100
	kernel	linear	rbf	rbf	rbf	rbf	rbf
ANN	**batch_size**	20	40	80	20	80	20
	epochs	10	10	10	10	100	10

target. We can see from this Figure that overall LR, SVM, and BN have the best balanced accuracy. Furthermore, we have 10-15% increments from the baseline for targets *nbchu2* and *hypotenO*; 15-20% for targets *sarcopen*, and *trEq*; 20-25% for targets *trEq* and *ADLlt5*; 25-30% for targets *trMar*, *peurTom*, *osteopor*, *dep*, and *parkOuSP*; where the baseline balanced accuracy is always 50%.

Figure 4.21: Comparison of prediction results of BN with other classifiers when for a given target, percentage of increment or decrement from the baseline results regarding Balanced accuracy.

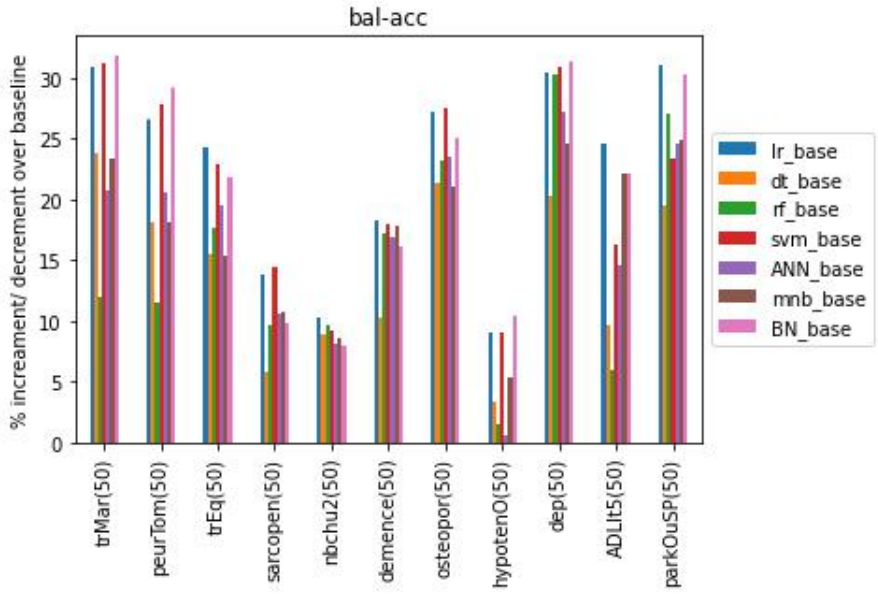


Figure 4.22 represents the percentage increment or decrement when comparing the area under the ROC curve (AUC-ROC) for a given classifier when predicting a given target. We can see from this Table that overall LR, RF, SVM, and BN have the best evaluation. Furthermore, we have 10-15% increments from the baseline for targets *nbchu2* and *hypotenO*; 15-20% for targets *sarcopen*; 20-30% for targets *trEq* and *demence*; 30-40% for targets *trMar*, *peurTom*, *osteopor*, *dep*, *ADLlt5*, and *parkOuSP*; where the baseline AUC-ROC is always 50%.

Figure 4.23 represents the percentage increment or decrement when comparing the area under the PR curve (AUC-PR) for a given classifier when predicting a given target. We can see from this Table that overall LR, RF, SVM, ANN, and BN have the best evaluation. Furthermore, we have 10-15% increments from the baseline for targets *trMar*, *sarcopen*, *nbchu2* and *hypotenO*; 15-20% for targets *peurTom*, *trEq* and *demence*; 30-40% for targets *ADLlt5*; 40-50% for targets *osteopor*, *dep*, and *parkOuSP*; where the baseline AUC-PR for each target is mentioned in the horizontal axis with their respective names.

Figure 4.24 represents the F1 score (F1) for a given classifier when predicting a given target. We can see from this Table that overall LR, RF, SVM, ANN, and BN have the best evaluation. For targets *hypotenO* we have an F1 score of 40-50%; 50-60% for ADLlt5; 60-70% for *demence*, *osteopor*, *dep*, *nbchu2* and *parkOuSp*; 70-80% for sarcopen; 80-90% for *trMar*, *peurTom*, and *trEq*.

Figure 4.22: Comparison of prediction results of BN with other classifiers when for a given target, percentage of increment or decrement from the baseline results regarding area under ROC curve.

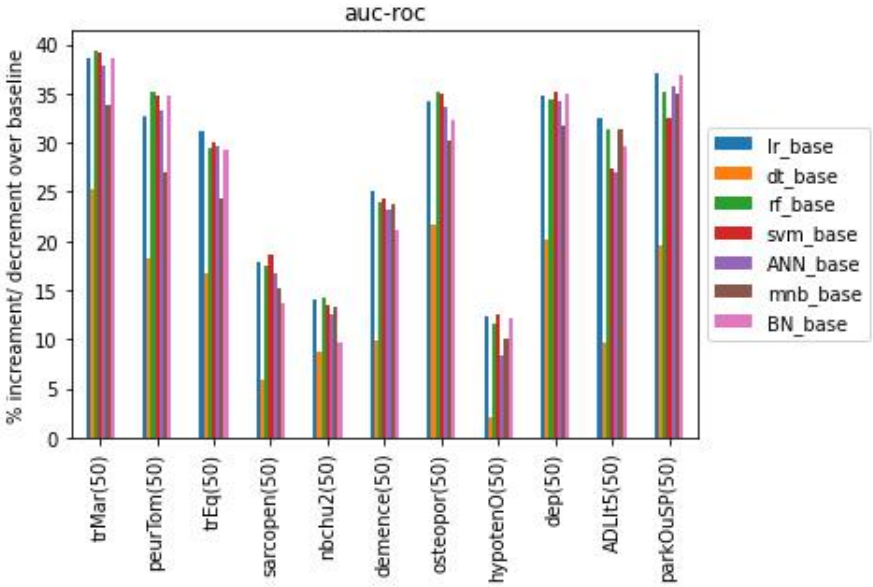


Figure 4.23: Comparison of prediction results of BN with other classifiers when for a given target, percentage of increment or decrement from the baseline results regarding area under PR curve.

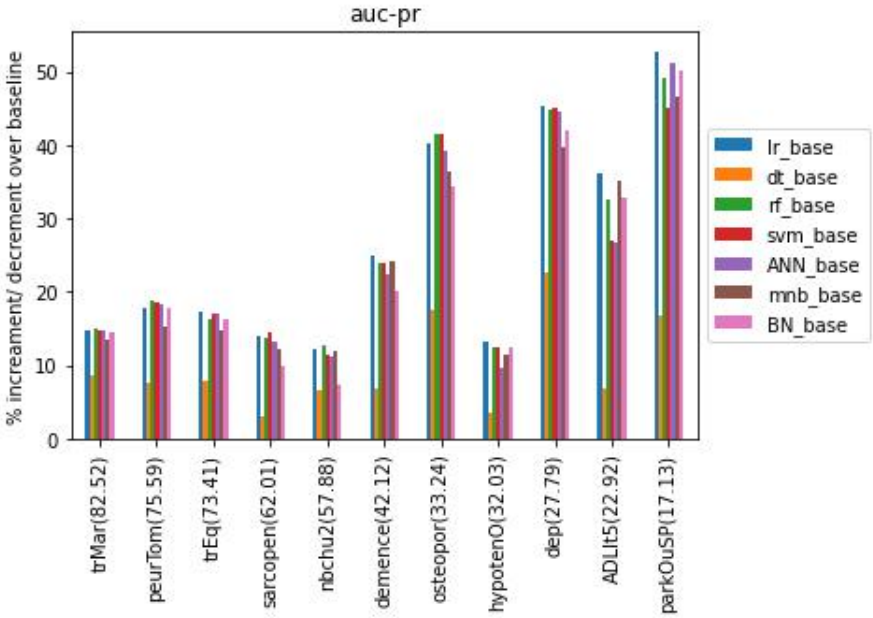
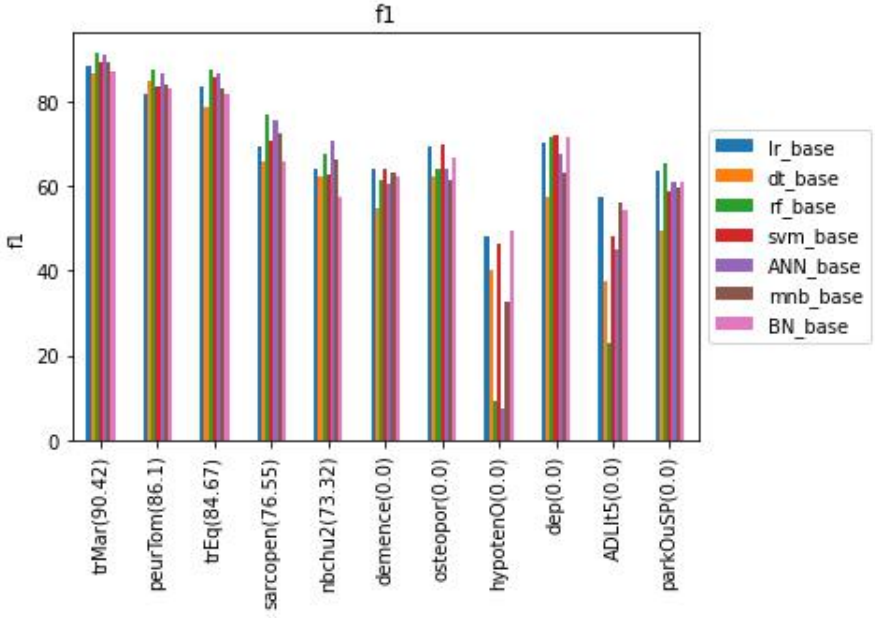


Figure 4.24: Comparison of prediction results of BN with other classifiers when for a given target, percentage of F1 score.



Points to remember from this section

- ✓ BN's predictive performance is as good as some other classifiers used

4.3.5 Summary of results using iteration 2

In this section, we have first presented the results in order to answer the question: should we use a specific subset of variables or the complete set of variables to evaluate a given target risk factor? We have used chi-square and mutual information methods to select a specific subset of variables. From the results presented, we concluded that on average there is no difference when using a specific subset of variables or all variables when predicting a given target. In addition, since our focus is more on using BN, we further investigated if we should use a single BN to predict all targets or use a specific subset of variables to learn a specific BN for a given target risk factors. Here we selected 3 target risk factors for comparison based on several criteria and also used 13 different BN models. From the results presented, we conclude that it is better to use a single BN model using all variables to predict a given target risk factor.

Furthermore, we also compared 13 different structure learning algorithms in order to have a structure that has a good representation of the data, an understandable relationship between variables, and good prediction results. From the results presented, we decided to use the greedy hill climbing (GHC) algorithm with a BIC score for structure learning for the BN model. We also introduce some mandatory arcs with the help of domain experts and based on general knowledge in order to make the graph more understandable. In addition, we also investigated whether to use the oversampling method before building the BN model or use the imbalanced

data itself. The results suggest that the BN model performs better when build using imbalanced data.

Furthermore, to evaluate the quality of the prediction results of BN we compared the results with several well-used classifiers namely: LR, DT, RF, SVM, ANN, and MNB. To evaluate the performance of different classifiers we presented balanced accuracy, F1 score, the area under ROC, and PR curves. As can be seen from the results presented, BN is among the best-performing classifiers when predicting a given target risk factor and we are able to predict most of the targets with some variability in the results.

4.4 Benefits of Iteration 2 compared to Iteration 1

So far we have presented the results obtained using both the first and second iterations. Now in this section, we present the difference in those results. Table 4.25 presents the difference between iteration 1 and iteration 2. The first column represents the different steps with the second column focusing on the criteria used for each step. The third and fourth columns represent the values for each criterion when using iterations 1 and 2 respectively.

Table 4.25: Differences in iteration 1 and iteration 2

	Criteria	Iteration 1	Iteration 2
manual var selection	**# var**	45	90
	# targets	12	11
missing values	**imputation**	KNN (k=5)	NB, KNN (k= [1, 2, ... , 19])
all VS sss	**var selection**	chi2	chi2, mi, union, intersection
	classifier	BN, LR, DT, RF, SVM	BN, LR, RF, SVM, MNB
	measures	Acc, F1	Acc
imbalance problem	**oversampling**	SMOTE, ADASYN, SVM-SMOTE	SVM-SMOTE
	classifier	BN, LR, RF, SVM, ANN, MNB	BN
	measures	Bal-acc, F1, F2, AUC-ROC, AUC-PR	Bal-acc
risk factor prediction	**classifier**	BN, LR, DT, RF, SVM	BN, LR, DT, RF, SVM, MNB, ANN
	measures	Acc, F1, Bal-acc	Bal-acc, F1, AUC-ROC, AUC-PR
	use % observation	Yes	No

We can see from the above Table that in the first iteration we selected 45 variables after preprocessing of data whereas, in the second iteration, we selected 90 variables. The goal of the first iteration was to get a first view of the feasibility, the methodology, and the quality of results while keeping a small model in order to make easier this first round of the study. In that aim, the 45 selected variables correspond to the most important variables involved in fall prevention. On the opposite, the aim in the second iteration was to improve the quality of the prediction, and in that aim, we considered any variable likely to contribute to the prediction of at least one

of the targets. The difference at this step is not only just in terms of the number of variables selected but also in the quality of the way the variables have been either selected or defined by combining several variables with the same or very close meaning. In addition, the number of target variables was reduced from 12 to 11 in iteration 2. This is due to the removal of the variable *auTrNeur* that had been defined in the first iteration. This variable regrouped a set of 8 neurological disorders except for *dementia*, CVA-TIA, or Parkinson's disease. These last three neurological disorders are among the main risk factors for falls whereas the eight others play a secondary role. It finally appeared that it was not relevant to predict the presence of one of these eight neurological disorders. We have thus added each of them as distinct variables in the second iteration, but not considered as target variables. Regarding the missing value imputation, in iteration 1, we only used KNN (k=5) in order to get a completed data set; in the second iteration, we tried to optimize this step, first by answering the question of whether to use MVI or not, and then with what algorithm. In that aim, in iteration 2, we compared the performance of NB and KNN with the value of k ranging from 1 to 19 (see section 3.4.4).

In order to evaluate the interest in using a complete set of variables versus a specific subset of variables, in iteration 1 we only used the chi2 method for variables selection and BN, LR, DT, RF, and SVM for prediction with accuracy and F1 score as measures to evaluate the quality of prediction. But in iteration 2 we achieved a more complete test to answer the question, and we used chi2, mi, their intersection, and union for variable selection with BN, LR, RF, MNB, and SVM for prediction with accuracy as measures.

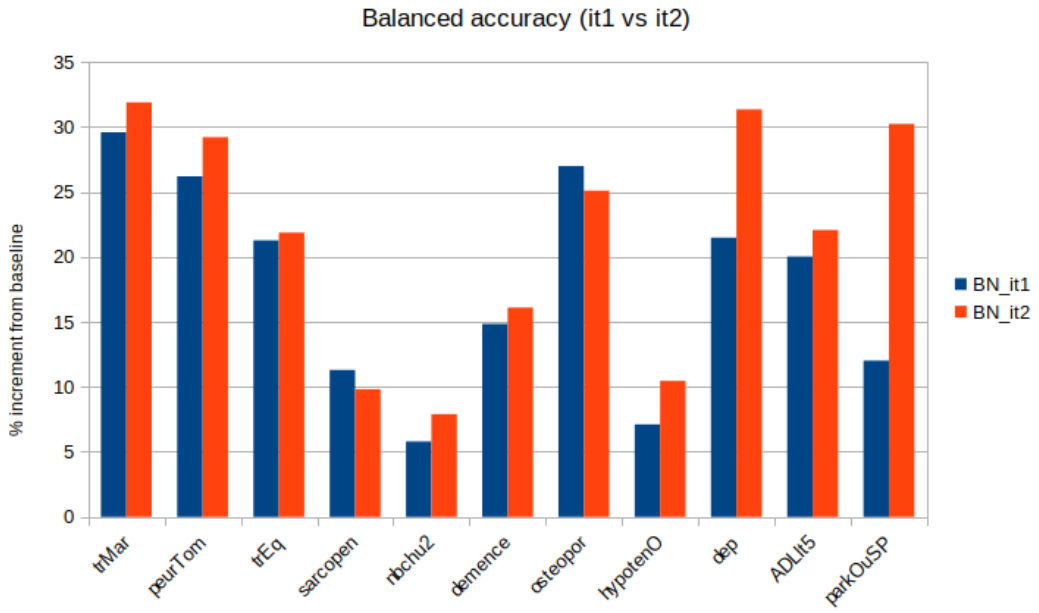
In order to evaluate the interest in using imbalanced data versus balanced data for the prediction of a given target risk factor, in the first iteration we compare the results using the following oversampling methods: SMOTE, ADASYN, SVM-SMOTE with the following classifiers: BN, LR, RF, SVM, ANN, MNB and evaluated the quality of results using the following measures: Bal-acc, F1, F2, AUC-ROC, AUC-PR. This first comparison showed (1) the interest to use the balancing method for usual classifiers, (2) the difference regarding the Bayesian network, and (3) the advantage of SVM-SMOTE over SMOTE and ADASYN. Based on those results, in the second iteration 2, we only investigated the use of SVM-SMOTE with BN using bal-acc as a measure to evaluate the quality of prediction. For other classifiers, we use them with SVM-SMOTE (based on the results from iteration 1).

Since our main focus is on using BN to evaluate the risk factor, in the first iteration we only used GHC algorithms with AIC score to learn the structure of BN whereas, in the second iteration, we compared 13 different structure learning algorithms namely GHC and Tabu list each with scores: AIC, BIC, BD, Bdeu, K2, and Log2; and MIIC. Furthermore, in order to evaluate the quality of the predictive performance of BN for a given target, in the first iteration we compare the results of BN with the following classifiers: BN, LR, DT, RF, and SVM with the following measures: accuracy and F1 score. But for the second iteration, we used the following classifiers: LR, DT, RF, SVM, MNB, and ANN with the following measures: Bal-acc, F1, AUC-ROC, and AUC-PR. Furthermore, in the first iteration, we evaluated the interest or prediction of a given target risk factor using the partial observation available but in the second iteration, we did not perform this evaluation based on partial observation because of time constraints. In future perspectives, we plan to evaluate the target risk factors based on partial observation using the data selected using iteration 2.

Now, we present the difference in results of the prediction of BN using iteration 1 versus iteration 2. Figure 4.25 represents the percentage increment in balanced accuracy from baseline when using BN with iteration 1 versus iteration 2. We can see from the figure that using BN after iteration 2 clearly gives a higher balanced accuracy when predicting all the targets except *sarcopen* and *osteopor*. The difference in these targets is also very small.

Figure 4.26 represents the percentage F1-score when using BN with iteration 1 versus

Figure 4.25: percentage increment in balanced accuracy from baseline when using BN with iteration 1 versus iteration 2



iteration 2. We can see from the figure that using BN after iteration 2 gives a higher F1-score when predicting all the targets in group M1 except *sarcopen* and *nbchu2* but the difference in these targets is also very small. Similarly, for targets in group M0, the F1-score is always better when using BN with iteration 2.

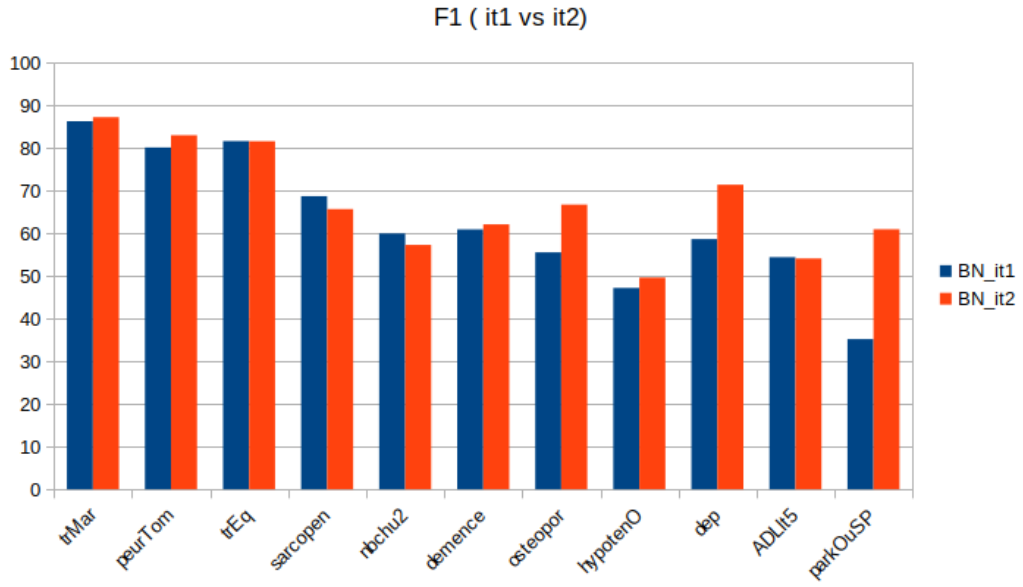
4.5 Conclusion

As mentioned in earlier chapters, fall prevention requires to provide a small number of recommendations that are selected depending on the risk factors present in a person. Thus the evaluation of risk factors is the base of fall prevention. To that aim, in this chapter, we used an iterative approach for analyzing our data. It is divided into two iterations. The objective for the first (second) iteration is to select the minimum (maximum) number of variables from the initial data set in order to evaluate the risk factors for falls. Here, in the first iteration, the goal was to provide a model with a reasonable size and in the second iteration to improve the results obtained by the first iteration and try to keep as many variables as possible.

We have presented the results using iteration 1 versus iteration 2 regarding the question: (1) should we use a complete set or a specific subset of variables to evaluate a given target risk factor? From the results presented, we see that there is no big difference in using either of the two. Also, in real-life situations, not all the information is available for a given patient; as a consequence, reducing the number of variables will further reduce the chances of getting the relevant information to predict a given target risk factor. So we decided to use the complete set of variables.

In addition, we have evaluated the interest in using imbalanced data versus balanced data in order to evaluate the target risk factor. We have presented the results using several oversampling

Figure 4.26: percentage F1-score when using BN with iteration 1 versus iteration 2



techniques namely: SMOTE, ADASYN, and SVM-SMOTE. From the results, we have decided to use the BN model with imbalanced data. Finally, we evaluated the target risk factors using both iterations and presented the difference in results.

Among the variables selected for this study, an arbitrary number of them can be observed, whether they are targets or not. Moreover, risk factors are not independent of each other, meaning that when one of them is observed, it should be used to improve the evaluation of the others, in addition to other observed features. That situation makes more difficult the use of usual classifiers because a new model would have to be learned for each target variable, and for each possible subset of observed variables. BN models allow overcoming that problem, since the same model can be used to evaluate any variable of the model, regarding any subset of observations. In addition, BNs allow a combination of general statistical knowledge and specific individual information, and to update belief on any node from incomplete observations. These features exactly answer the problem of predicting risk factors in real-life situations. Another advantage of BN is that the model can be built both from data and expert knowledge which is very interesting in the context of health. It is also very important to make the model interpreTable/ understandable by the final user (general practitioners) since it contributes to make the aiding system accepTable and augments the trust in results. So BN becomes the good choice to use because of the graphical representation that is easy to explain and understand.

Chapter 5

Conclusion and perspectives

In this chapter, we first summarize the main ideas from our proposed approaches, including relevant methodologies, results analysis, and discussion; then we describe the future improvements of this thesis in both short-term and long-term aspects. All details are presented as follows.

5.1 Conclusion

The use of machine learning algorithms in healthcare is an area of growing interest and has the potential to improve the quality and efficiency of healthcare services. These algorithms can be used as a way to aid in early illness identification, patient care, and community services as the amount of data in healthcare grows. However, the success of these algorithms heavily relies on the quality of the data used for training as well as the ability to handle and process the data effectively. Furthermore, building a good machine learning model requires a good data set. Also, working with data is more difficult than it may appear since it necessitates, careful handling of the data. With that aim, in our work, we have presented the basics of data preparation more specifically about handling missing data, the problem of imbalance in data, and the selection of relevant variables to build a good machine learning model. We also presented an overview of a literature review that shows the use of different machine learning algorithms in the healthcare domain and the challenges faced by researchers. Furthermore, in our work, we focus on the problem of the prevention of falls. In this context, machine learning algorithms can be used to detect health-related risks in patients, which can aid in the evaluation of risk factors for falls.

Fall prevention is a challenge to population aging, but it is one of the issues that require more attention. It assures that a large part of the elderly has a regular and efficient evaluation of their risk of fall and adopted recommendations to reduce their risk of fall. Since falls result from a complex interaction of risk factors, an important step in fall prevention is to detect the presence of these risk factors. Also, it has been shown that reducing the risk factors leads to a reduction in the risk of falls. Thus, fall prevention can be achieved by providing recommendations that help reduce the risk factors that are present for a given person (for instance: kinesiotherapy may improve balance and reduce the risk of muscular weakness). Among the risk factors for falls, some of them are reducible (or actionable), meaning that some actions can be carried out in order to reduce them. We focus on evaluating these risk factors. Furthermore, the evaluation of risk factors for falls remains a challenge since it requires time and expertise, and specific tests and devices may also be necessary. Moreover, the family physician, who is one of the main

actors in fall prevention, generally does not have a lot of time, whereas fall prevention requires a pedagogical and repeated approach. As a consequence, the collection of information for a complete evaluation of risk factors is not feasible regularly, and the risk factors for a person's fall should be assessed from an incomplete set of observations. When there is no direct information about a specific risk factor, it is however possible to get a sense of it from general knowledge about its frequency in a specific context, which is described by the other available information about the person. As an example, the experts know that a person who is afraid of falling and has neuropathy is much more likely to have balance problems than the average elderly. In this deduction, the expert combines general knowledge and reasoning with uncertainty.

This thesis is a contribution to the development of a knowledge-based system to evaluate risk factors to prevent falls. In that context, we aim to use a knowledge model with an inference engine that allows the updating of beliefs under uncertainty, the results and the model itself can be interpretable or explainable and have a good predictive performance. Furthermore, among the variables selected for this study, an arbitrary number of them can be observed, whether they are targets or not. Moreover, risk factors are not independent of each other, meaning that when one of them is observed, it should be used to improve the evaluation of the others, in addition to other observed features. That situation makes the use of usual classifiers more difficult because a new model would have to be learned for each target variable and for each possible subset of observed variables. With that in mind, we proposed the use of Bayesian networks (BN) which is a type of probabilistic graphical model. BN models allow for overcoming that problem since the same model can be used to evaluate any variable of the model, regarding any subset of observations. In addition, BNs allow a combination of general statistical knowledge and specific individual information to update beliefs on any node based on incomplete observations. These features exactly answer the problem of predicting risk factors in real-life situations. Another advantage of BN is that the model can be built both from data and expert knowledge, which is very interesting in the context of health. It is also very important to make the model interpretable/ understandable by the final user (general practitioners) since it contributes to making the aiding system acceptable and augments trust in the results. Hence BN becomes a good choice to use because of the graphical representation that is easy to explain and understand. So we constructed a BN model to automatically evaluate the main actionable risk factors using a real data set combined with expert knowledge and we compared the quality of results with other usual classifiers.

Data is an essential component of an AI or ML model. As data collection becomes simpler as a result of increased digitization, it is also becoming more widely accessible in the healthcare sector. The information is collected at hospitals, care centers, and other healthcare institutions. These data could include, among other things, information about biological processes, administrative processes, and insurance claims. The amount of data being used to solve healthcare-related issues is increasing, as is the use of data-driven methods.

In this work, we have presented the description of the data used which was collected at the Service of Fall Prevention, Hospital of Lille, France. The initial data includes 1810 patients who visited the service between January 2005 and December 2016, of whom 28% are male and 72% are female, with ages ranging from 51 years old to 100 years old, with an average age of 81 years old. In our study, we included persons who can walk and are 65 years of age or older. In addition, the methodology used to design an ontology for fall prevention, followed by the resulting ontology for the risk factors of falling, has also been presented. The goal of this ontology, which served as the foundation for the creation of the fall prevention software system, is to support the assessment of elderly individuals' risk factors for falls. It also provides a solid foundation for our knowledge of the variables in our data set.

To achieve that aim, in this work, we used an iterative approach for analyzing our data. It

is divided into two iterations. The objective of the first (or second) iteration is to select the minimum (or maximum) number of variables from the initial data set in order to evaluate the risk factors for falls. The idea here is to "begin small" during the first iteration to get a first return about the feasibility, the results, and the difficulties; then make a second iteration with the aim to improve the evaluation of the risk factors for falls by improving each step of the whole process. Furthermore, in order to identify the variables to use for our study, we first used the criterion of data quality to remove unusable variables and a second criterion was to remove variables that could not be used to evaluate any risk factors for falls. In pursuit of that aim, we removed variables related to recommendations, variables associated with the second appointment after 6 months, and administrative variables. After the variables guided by the first two criteria are removed, we also use a third criterion that is specific to each iteration of data cleaning and consists in providing a model with a reasonable size for the first iteration and improving each step of the first iteration for the second iteration. This cleaning led to a data set containing 45 variables in the first iteration and 90 variables in the second iteration. Additionally, we have identified 12 target risk factors (11 in the second iteration) to evaluate from our data. These risk factors have been selected as targets because they are actionable; are not frequently present in the patient's file; and are not easy to collect. All of them are with binary domains with prevalence ranges from 83% to 17% and can be divided into 2 groups: Group M0: targets having a majority class equal to 0; and Group M1: targets with a majority class equal to 1.

As mentioned earlier, fall prevention requires providing a small number of recommendations that are selected depending on the risk factors present in a person. Thus, the evaluation of risk factors is the basis of fall prevention. With that goal in mind, we first analyzed whether we should use a complete set or a specific subset of variables to evaluate a given target risk factor. In that aim, we compared the quality of the prediction for each target variable after using all variables, or a specific subset of variables selected thanks to one of the four following methods: chi-square, mutual information, and the intersection and union of these two subsets. From the results presented, we observed that there is no big difference between using either of the two. Also, in real-life situations, not all the information is available for a given patient; as a consequence, reducing the number of variables will further reduce the chances of getting the relevant information to predict a given target risk factor. So we decided to use the complete set of variables. In addition, we have evaluated the interest in using imbalanced data versus balanced data in order to evaluate the target risk factor. We have presented the results using several oversampling techniques, namely: SMOTE, ADASYN, and SVM-SMOTE. To achieve these results, we learned the structure and the parameters of Bayesian networks (thanks to the Greedy Hill Climbing algorithm with the BIC score), first based on the original imbalanced learning set, and second, based on the balanced learning set using the oversampling techniques. We compared the performance of these experiments using different metrics and we conclude that for this study, the use of a balancing method does not allow for improvement in the performance, except when focusing on the F1-score and only for the variables with imbalance classes in favor of the positive class (which is not the most frequent situation in real data sets). When focusing on highly imbalanced classes in favor of the negative class (which is a frequent case), the use of the original imbalanced data set is clearly better. When focusing on the metrics area under the precision-recall curve (AUC-PR) and area under the receiver operating characteristic curve (AUC-ROC), the results are mitigated except for highly imbalanced variables with a negative majority class. Based on these results, we have decided to use the BN model with imbalanced data.

Furthermore, we evaluated the target risk factors using both iterations and presented the difference in results. To achieve these results, we learned six usual classifiers based on SVMSMOTE balanced data: logistic regression, support vector machine, random forest, multinomial naive

Bayes classifier, artificial neural network and decision tree, and a Bayesian network based on the original imbalanced data set. The hyperparameters of all the classifiers have been tuned to maximize the performance. We compared the performance of these experiments using different performance metrics and focusing on balanced accuracy and F1-score. Despite the different measures showing some global similar trends, it appears that the choice of an appropriate measure is important together with a true analysis of the results regarding the performance metric since a quick look at the performance provided by a specific metric does not allow us to conclude easily on the results. In addition, we conclude that six out of the eleven target risk factors for falls are well predicted and the prediction of four other risk factors for falls shows lower performance whereas one risk factor is hardly predictable from our data set. Regarding the set of classifiers, six of them show similar performance with however important variations either for a target or for a metric and the ranking of the classifiers depends on the performance metric. Only one classifier (decision tree) provides regularly a less good evaluation.

5.2 Future perspectives

In this section, we present the future perspectives of our thesis. We mentioned some perspectives when we discussed our contributions and their implications. The objective of this section is to develop these perspectives and aggregate them. In this regard, we present five perspectives: causal BN graph, reasoning with partial observations, data collection, reasoning based on temporal information, and application for the general physician as follows.

Causal BN graph

In this study, we aim to get a causal graph since it contributes to the understandability of the graph by experts. To achieve that aim, we worked with domain experts and defined some mandatory arcs to be included in the structure of the BN model. But we had limited time with domain experts and the number of arcs learned by any given algorithm is very high. Due to this constraint, we focused on the possible arcs around each target risk factor. In future work, it will be interesting to build a fully causal graph. As a reminder, one of the main objectives of our work is to support the general practitioner in evaluating the risk of falls. With that aim, we want to provide a model that can be understood by the final user so that the results can be explained. So, we want to focus on the relationship between the variables in our BN model to make the graph more understandable and thus, augment the trust of final users in the results of a future fall prevention application.

Reasoning with partial information

As a reminder, the family physician, who is one of the main actors in fall prevention, generally does not have a lot of time, whereas fall prevention requires a pedagogical and repeated approach. As a consequence, the collection of complete information for the evaluation of risk factors is not feasible regularly by the general practitioner, and the risk factors for a person's fall should be assessed from an incomplete set of observations. In our work, we have only investigated this part during iteration 1 which is when the number of selected variables was less. The results obtained in the first iteration showed that for some target variables, the quality of the evaluation augment with the proportion of available information whereas for other variables, the evaluation was not better (and not good). However, this result depends also on the performance metric considered. We expect the results of the risk factor evaluation from partial observation to be better in the second iteration, because of the larger set of variables, and also because of the improvements of

the results in the second iteration (due to the careful consideration of each step of the whole process). Due to the time constraints, it was not feasible to investigate this. Hence in future work, it will be very interesting to check the quality of prediction based on available information for a given person (partial or complete).

Data collection

The data for this work was provided by the service of Fall Prevention, Hospital of Lille, France which includes 1810 patients who visited the service between January 2005 and December 2016. As the size of our dataset was limited, it will be very interesting to evaluate the quality of our model when learned using a larger data set. Also, the data set used does not represent the general elderly population because the patients who visited the service are already at high risk of falls. It will also be interesting to consider a more representative group of the elderly when building a prediction model to evaluate the risk factors for falls. Another perspective is to gather information about any elderly person based on a large panel of different sources such as automatic extraction from the Electronic Health Record, and from the medical database of hospitals where the patient has been received, different sensors, and manual input from different stakeholders in fall prevention: the elderly person, family members and caregivers, nurses, physiotherapist, any physician and specialists related with the elderly person, and also medical staff in a hospital where the person has been. This larger number of sources of information about the patient would allow for getting a more complete set of information about the patient, which is very important for the evaluation of a repeated risk factor for falls. We recall that except in a fall prevention service where the information of the patient is completely and carefully collected, the amount and the quality of information available on a person is a key point for fall prevention. Such a multisource and continuous process of data collection about a given person with the aim of fall prevention could provide a partial set of dated information, associated with the source of each piece of information.

Reasoning based on temporal information

A large part of the information of a person changes with time. In real life, when information is required immediately for a given person, such as for fall prevention by general practitioners, the available information can be seen as a partial time-stamped observation set as explained above. Beyond this thesis, a perspective is to allow an assessment of the risk factors for falls based on a partial set of dated information. This perspective raises the first question which is to evaluate the current state of a variable given a dated observation (or a set of dated observations). For this purpose, we need knowledge about the dynamics of the considered variables, as well as a sufficient temporal data set that allows to reason about the changes regarding a person's features over a long period of time. Faced with the difficulty of finding such a data set, we have proposed an algorithm to simulate such a data set, based on real static data provided by the service of fall prevention at Lille's hospital. We have selected five *persistent* variables, meaning that their value may change at most once. An example of such a variable is Parkinson's disease: when we consider a binary variable, the initial value is the absence of the disease, and it may change toward the positive value at most once in the life of a person, and then it never changes again. We call this variable a positive persistent variable. The algorithm to simulate a temporal data set for a subset of persistent variables is based on assumptions regarding the temporal evolution of each contextualized variable, as defined by a Bayesian network learned on the real static data set. The temporal data set simulated thanks to the proposed algorithm is evaluated by the comparison of the temporal distribution of each contextualized variable with the functions

obtained by linear interpolation from the real data set. Since our static data set includes the age of the person, we analyzed the proportion of positive values for each age group. From this analysis, we built a dynamic BN that was used to validate the simulated temporal data.

Further information about this process can be found in appendix A. However, given the scale of the question and time constraints we did not go further in this direction, and in future work, it will be very interesting to explore the way to manage dated observation sets.

Application for the General Physician

As a reminder, the main objective of this work was to help the general physician evaluate the main risk factors for falls. But generally, they are no experts in using different inference engines. With that in mind, it would be very helpful to have an application that can be easily used by the general physician as well as the other actors such as the nurse or the person himself to get the evaluation of risk factors for falls and associated recommendations for fall prevention. In this direction, we have co-supervised a master's student project whose aim was to develop a first draft of an application which can be found at `https://chutepa.uphf.fr/`.

Bibliography

[1] Mohammad Adibuzzaman, Poching DeLaurentis, Jennifer Hill, and Brian D Benneyworth. "Big data in healthcare–the promises, challenges and opportunities from a research perspective: A case study with a model database". In: *AMIA Annual Symposium Proceedings*. Vol. 2017. American Medical Informatics Association. 2017, p. 384.

[2] Harshita Agrawal, Prateek Jain, and Amit M. Joshi. "Machine learning models for non-invasive glucose measurement: towards diabetes management in smart healthcare". en. In: *Health and Technology* 12 (5) (Sept. 2022), pp. 955–970. ISSN: 2190-7188, 2190-7196. DOI: 10.1007/s12553-022-00690-7. URL: https://link.springer.com/10.1007/s12553-022-00690-7 (visited on 02/14/2023).

[3] Tanvir Ahammad. "Risk factors identification for stroke prognosis using machine learning algorithms". en. In: *Jordanian Journal of Computers and Information Technology* (0) (2022), p. 1. ISSN: 2413-9351. DOI: 10.5455/jjcit.71-1652725746. URL: https://www.ejmanager.com/fulltextpdf.php?mno=35136 (visited on 02/14/2023).

[4] Samah Alajmani and Hanan Elazhary. "Hospital readmission prediction using machine learning techniques". In: *International Journal of Advanced Computer Science and Applications* 10 (4) (2019).

[5] Tahani Aljuaid and Sreela Sasi. "Proper imputation techniques for missing values in data sets". In: *2016 International Conference on Data Science and Engineering (ICDSE)*. IEEE. 2016, pp. 1–5.

[6] Ricardo de Almeida Falbo. "SABiO: Systematic Approach for Building Ontologies." In: *ONTO. COM/ODISE@ FOIS*. 2014.

[7] Mohammed F. Alrifaie, Zakir Hussain, Asaad Shakir, and Modhi Lafta. "Using Machine Learning Technologies to Classify and Predict Heart Disease". en. In: *International Journal of Advanced Computer Science and Applications* 12 (3) (2021). ISSN: 21565570, 2158107X. DOI: 10.14569/IJACSA.2021.0120315. URL: http://thesai.org/Publications/ViewPaper?Volume=12&Issue=3&Code=IJACSA&SerialNo=15 (visited on 02/14/2023).

[8] Huda M. Alshanbari, Tahir Mehmood, Waqas Sami, Wael Alturaiki, Mauawia A. Hamza, and Bandar Alosaimi. "Prediction and Classification of COVID-19 Admissions to Intensive Care Units (ICU) Using Weighted Radial Kernel SVM Coupled with Recursive Feature Elimination (RFE)". en. In: *Life* 12 (7) (July 2022), p. 1100. ISSN: 2075-1729. DOI: 10.3390/life12071100. URL: https://www.mdpi.com/2075-1729/12/7/1100 (visited on 02/14/2023).

[9] Emily Grace Armitage, Joanna Godzien, Vanesa Alonso-Herranz, Ángeles López-Gonzálvez, and Coral Barbas. "Missing value imputation strategies for metabolomics data". In: *Electrophoresis* 36 (24) (2015), pp. 3050–3060.

[10] Hiba Asri, Hajar Mousannif, Hassan Al Moatassime, and Thomas Noel. "Using machine learning algorithms for breast cancer risk prediction and diagnosis". In: *Procedia Computer Science* 83 (2016), pp. 1064–1069.

[11] Janna Beling and Margaret Roller. "Multifactorial intervention with balance training as a core component among fall-prone older adults". In: *Journal of Geriatric Physical Therapy* 32 (3) (2009), pp. 125–133.

[12] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Vol. 4. 4. Springer, 2006.

[13] H. Bourdessol and S. Pin. *inpes Edition. collection Refrentiels*. 2014.

[14] L Breiman. *Random forests machine learning, vol. 45*. 2001.

[15] Henrik Brink, Joseph Richards, and Mark Fetherolf. *Real-world machine learning*. Simon and Schuster, 2016.

[16] Brandon Butcher and Brian J Smith. *Feature Engineering and Selection: A Practical Approach for Predictive Models*. 2020.

[17] Nurheri Cahyana, Siti Khomsah, and Agus Sasmito Aribowo. "Improving imbalanced dataset classification using oversampling and gradient boosting". In: *2019 5th International Conference on Science in Information Technology (ICSITech)*. IEEE. 2019, pp. 217–222.

[18] Luca Cattelani, Federico Chesani, Luca Palmerini, Pierpaolo Palumbo, Lorenzo Chiari, and Stefania Bandinelli. "A rule-based framework for risk assessment in the health domain". In: *International Journal of Approximate Reasoning* 119 (2020), pp. 242–259.

[19] Kabalan Chaccour, Rony Darazi, Amir Hajjam El Hassani, and Emmanuel Andres. "From fall detection to fall prevention: A generic classification of fall-related systems". In: *IEEE Sensors Journal* 17 (3) (2016), pp. 812–822.

[20] Cheng-Min Chao, Ya-Wen Yu, Bor-Wen Cheng, and Yao-Lung Kuo. "Construction the model on the breast cancer survival analysis use support vector machine, logistic regression and decision tree". In: *Journal of medical systems* 38 (10) (2014), pp. 1–7.

[21] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. "SMOTE: synthetic minority over-sampling technique". In: *Journal of artificial intelligence research* 16 (2002), pp. 321–357.

[22] Min Chen, Yixue Hao, Kai Hwang, Lu Wang, and Lin Wang. "Disease prediction by machine learning over big data from healthcare communities". In: *Ieee Access* 5 (2017), pp. 8869–8879.

[23] Alexander S Chiu, Raymond A Jean, Matthew Fleming, and Kevin Y Pei. "Recurrent falls among elderly patients and the impact of anticoagulation therapy". In: *World journal of surgery* 42 (12) (2018), pp. 3932–3938.

[24] Nello Cristianini, John Shawe-Taylor, et al. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[25] P Dargent-Molina and G. Bréart. "Epidemiology of falls and fall-related injuries in the aged". In: *Revue d'Épidémiologie et de Santé Publique* 43 (1) (1995), pp. 72–83.

[26] Jonathan J Davis and Andrew J Clark. "Data preprocessing for anomaly based network intrusion detection: A review". In: *computers & security* 30 (6-7) (2011), pp. 353–375.

[27] Jip de Vries, Michiel H.S. Kraak, Richard A. Skeffington, Andrew J. Wade, and Piet F.M. Verdonschot. "A Bayesian network to simulate macroinvertebrate responses to multiple stressors in lowland streams". In: *Water Research* 194 (2021), p. 116952. ISSN: 0043-1354.

[28] Yueng Santiago Delahoz and Miguel Angel Labrador. "Survey on fall detection and fall prevention using wearable and external sensors". In: *Sensors* 14 (10) (2014), pp. 19806–19842.

[29] V. Delcroix, E. Grislin-Le Strugeon, and F. Puisieux. "A knowledge based system for the management of a time stamped uncertain observation set with application on preserving mobility". In: *International Journal of Approximate Reasoning* 134 (2021), pp. 53–71.

[30] Véronique Delcroix, Fatma Essghaier, Kathia Oliveira, Philippe Pudlo, Cédric Gaxatte, and Fran Puisieux. "Towards a fall prevention system design by using ontology". In: *en lien avec les Journées francophones d'Ingénierie des Connaissances, Plate-Forme PFIA* (July 2019).

[31] Véronique Delcroix, Gulshan Sihag, Emmanuelle Grislin-Le Strugeon, Xavier Siebert, Sylvain Piechowiak, and François Puisieux. "Prédiction des facteurs de risque de chute chez les personnes âgées à partir d'observations partielles à l'aide d'un réseau bayésien". In: *Colloque francophone sur la chute de la personne âgée*. 2021.

[32] Samrat Kumar Dey, Khandaker Mohammad Mohi Uddin, Hafiz Md. Hasan Babu, Md. Mahbubur Rahman, Arpita Howlader, and K.M. Aslam Uddin. "Chi2-MI: A hybrid feature selection based machine learning approach in diagnosis of chronic kidney disease". en. In: *Intelligent Systems with Applications* 16 (Nov. 2022), p. 200144. ISSN: 26673053. DOI: 10.1016/j.iswa.2022.200144. URL: https://linkinghub.elsevier.com/retrieve/pii/S2667305322000813 (visited on 02/14/2023).

[33] Ivo D Dinov. "Volume and value of big healthcare data". In: *Journal of medical statistics and informatics* 4 (2016).

[34] A Rogier T Donders, Geert JMG Van Der Heijden, Theo Stijnen, and Karel GM Moons. "A gentle introduction to imputation of missing values". In: *Journal of clinical epidemiology* 59 (10) (2006), pp. 1087–1091.

[35] Gaspard Ducamp, Philippe Bonnard, Anthony Nouy, and Pierre-Henri Wuillemin. "An Efficient Low-Rank Tensors Representation for Algorithms in Complex Probabilistic Graphical Models". In: *International Conference on Probabilistic Graphical Models, PGM 2020, 23-25 September 2020, Aalborg, Denmark*. Ed. by Manfred Jaeger and Thomas Dyhre Nielsen. Vol. 138. Proceedings of Machine Learning Research. PMLR, 2020, pp. 173–184.

[36] Patricia C Dykes et al. "Fall prevention in acute care hospitals: a randomized trial". In: *Jama* 304 (17) (2010), pp. 1912–1918.

[37] I Eekhout. "Don't Miss Out!: Incomplete data can contain valuable information". In: *PhD Dissertation at VU medical center Amsterdam, the Netherlands* (2014).

[38] Muhammad Faisal, Andy Scally, Robin Howes, Kevin Beatson, Donald Richardson, and Mohammed A Mohammed. "A comparison of logistic regression models with alternative machine learning methods to predict the risk of in-hospital mortality in emergency medical admissions via external validation". en. In: *Health Informatics Journal* 26 (1) (Mar. 2020), pp. 34–44. ISSN: 1460-4582, 1741-2811. DOI: 10.1177/1460458218813600. URL: http://journals.sagepub.com/doi/10.1177/1460458218813600 (visited on 02/14/2023).

[39] A Famili, Wei-Min Shen, Richard Weber, and Evangelos Simoudis. "Data preprocessing and intelligent data analysis". In: *Intelligent data analysis* 1 (1) (1997), pp. 3–23.

[40] Bahar Farahani, Farshad Firouzi, Victor Chang, Mustafa Badaroglu, Nicholas Constant, and Kunal Mankodiya. "Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare". In: *Future Generation Computer Systems* 78 (2018), pp. 659–676.

[41] Susan M Fox-Wasylyshyn and Maher M El-Masri. "Handling missing data in self-report measures". In: *Research in nursing & health* 28 (6) (2005), pp. 488–495.

[42] Sagayaraj Francis, Panem Prasad, and s Zahoor-Ul-Huq. "Medical Data Classification Based on SMOTE and Recurrent Neural Network". In: *International Journal of Engineering and Advanced Technology* 9 (Feb. 2020). DOI: 10.35940/ijeat.C5444.029320.

[43] Salvador Garcıa, Julián Luengo, and Francisco Herrera. *Data preprocessing in data mining*. Vol. 72. Springer, 2015.

[44] C Gaxatte, E Faraj, O Lathuillerie, Julia Salleron, Vincent Deramecourt, V Pardessus, MH Destailleur, Eric Boulanger, and Francois Puisieux. "Alcohol and psychotropic drugs: risk factors for orthostatic hypotension in elderly fallers". In: *Journal of human hypertension* 31 (4) (2017), pp. 299–304.

[45] C Gaxatte, T Nguyen, F Chourabi, Julia Salleron, Vinciane Pardessus, I Delabrière, André Thevenon, and F Puisieux. "Fear of falling as seen in the multidisciplinary falls consultation". In: *Annals of physical and rehabilitation medicine* 54 (4) (2011), pp. 248–258.

[46] Ramin Ghorbani, Rouzbeh Ghousi, Ahmad Makui, and Alireza Atashi. "A New Hybrid Predictive Model to Predict the Early Mortality Risk in Intensive Care Units on a Highly Imbalanced Dataset". en. In: *IEEE Access* 8 (2020), pp. 141066–141079. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.3013320. URL: https://ieeexplore.ieee.org/document/9153559/ (visited on 02/14/2023).

[47] Yoav Gimmon, Avi Barash, Ronen Debi, Yoram Snir, Yair Bar David, Jacob Grinshpon, and Itshak Melzer. "Application of the clinical version of the narrow path walking test to identify elderly fallers". In: *Archives of Gerontology and Geriatrics* 63 (2016), pp. 108–113.

[48] M. Gokiladevi, Sundar Santhoshkumar, and Vijayakumar Varadarajan. "MACHINE LEARNING ALGORITHM SELECTION FOR CHRONIC KIDNEY DISEASE DIAGNOSIS AND CLASSIFICATION". en. In: *Malaysian Journal of Computer Science* (Mar. 2022), pp. 102–115. ISSN: 01279084. DOI: 10.22452/mjcs.sp2022no1.8. URL: https://ejournal.um.edu.my/index.php/MJCS/article/view/35978 (visited on 02/14/2023).

[49] Sunila Gollapudi. *Practical machine learning*. Packt Publishing Ltd, 2016.

[50] Luise Gootjes-Dreesbach, Meemansa Sood, Akrishta Sahay, Martin Hofmann-Apitius, and Holger Fröhlich. "Variational Autoencoder Modular Bayesian Networks for Simulation of Heterogeneous Clinical Study Data". In: *Frontiers in Big Data* 3 (2020), p. 16. ISSN: 2624-909X. DOI: 10.3389/fdata.2020.00016. URL: https://www.frontiersin.org/article/10.3389/fdata.2020.00016.

[51] Hina Gull, Gomathi Krishna, May Issa Aldossary, and Sardar Zafar Iqbal. "Severity Prediction of COVID-19 Patients Using Machine Learning Classification Algorithms: A Case Study of Small City in Pakistan with Minimal Health Facility". en. In: *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*. Chengdu, China: IEEE, Dec. 2020, pp. 1537–1541. ISBN: 978-1-72818-635-1. DOI: 10.1109/ICCC51575.2020.9344984. URL: https://ieeexplore.ieee.org/document/9344984/ (visited on 02/14/2023).

[52] Eren Gultepe, Jeffrey P Green, Hien Nguyen, Jason Adams, Timothy Albertson, and Ilias Tagkopoulos. "From vital signs to clinical outcomes for patients with sepsis: a machine learning basis for a clinical decision support system". In: *Journal of the American Medical Informatics Association* 21 (2) (2014), pp. 315–325.

[53] Julian Hamm, Arthur G Money, Anita Atwal, and Ioannis Paraskevopoulos. "Fall prevention intervention technologies: A conceptual framework and survey of the state of the art". In: *Journal of biomedical informatics* 59 (2016), pp. 319–345.

[54] Emrana Kabir Hashi, Md Shahid Uz Zaman, and Md Rokibul Hasan. "An expert clinical decision support system to predict disease using classification techniques". In: *2017 International conference on electrical, computer and communication engineering (ECCE)*. IEEE. 2017, pp. 396–400.

[55] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[56] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. "ADASYN: Adaptive synthetic sampling approach for imbalanced learning". In: *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*. IEEE. 2008, pp. 1322–1328.

[57] Haibo He and Edwardo A Garcia. "Learning from imbalanced data". In: *IEEE Transactions on knowledge and data engineering* 21 (9) (2009), pp. 1263–1284.

[58] David Heckerman, Dan Geiger, and David M Chickering. "Learning Bayesian networks: The combination of knowledge and statistical data". In: *Machine learning* 20 (3) (1995), pp. 197–243.

[59] Mark L. Homer, Nathan P. Palmer, Kathe P. Fox, Joanne Armstrong, and Kenneth D. Mandl. "Predicting Falls in People Aged 65 Years and Older from Insurance Claims". In: *The American Journal of Medicine* 130 (6) (2017), 744.e17–744.e23.

[60] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*. Vol. 398. John Wiley & Sons, 2013.

[61] Rachael A Hughes, Jon Heron, Jonathan AC Sterne, and Kate Tilling. "Accounting for missing data in statistical analyses: multiple imputation is not always the answer". In: *International journal of epidemiology* 48 (4) (2019), pp. 1294–1304.

[62] Gazi Mohammed Ifraz, Muhammad Hasnath Rashid, Tahia Tazin, Sami Bourouis, and Mohammad Monirujjaman Khan. "Comparative Analysis for Prediction of Kidney Disease Using Intelligent Machine Learning Methods". en. In: *Computational and Mathematical Methods in Medicine* 2021 (Dec. 2021). Ed. by Osamah Ibrahim Khalaf, pp. 1–10. ISSN: 1748-6718, 1748-670X. DOI: 10.1155/2021/6141470. URL: https://www.hindawi.com/journals/cmmm/2021/6141470/ (visited on 02/14/2023).

[63] Stephen P Juraschek, Natalie Daya, Lawrence J Appel, Edgar R Miller III, Kunihiro Matsushita, Erin D Michos, B Gwen Windham, Christie M Ballantyne, and Elizabeth Selvin. "Subclinical cardiovascular disease and fall risk in older adults: results from the atherosclerosis risk in communities study". In: *Journal of the American Geriatrics Society* 67 (9) (2019), pp. 1795–1802.

[64] Anastasiia Kabeshova. "Prédire la chute de la personne âgée : apports des modèles mathématiques non-linéaires. Médecine humaine et pathologie." PhD thesis. France: Université d'Angers, 2015.

[65] Sidney Katz, Amasa B. Ford, Roland W. Moskowitz, Beverly A. Jackson, and Marjorie W. Jaffe. "Studies of illness in the aged. The index of ADL: a standardized measure of biological and physical function." In: *journal of American Medical Association* 185 (1963), pp. 914–9.

[66] Matloob Khushi, Kamran Shaukat, Talha Mahboob Alam, Ibrahim A. Hameed, Shahadat Uddin, Suhuai Luo, Xiaoyan Yang, and Maranatha Consuelo Reyes. "A Comparative Performance Analysis of Data Resampling Methods on Imbalance Medical Data". en. In: *IEEE Access* 9 (2021), pp. 109960–109975. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2021.3102399. URL: https://ieeexplore.ieee.org/document/9505667/ (visited on 02/14/2023).

[67] Hafsa Binte Kibria and Abdul Matin. "The severity prediction of the binary and multi-class cardiovascular disease A machine learning-based fusion approach". en. In: *Computational Biology and Chemistry* 98 (June 2022), p. 107672. ISSN: 14769271. DOI: 10.1016/j.compbiolchem.2022.107672. URL: https://linkinghub.elsevier.com/retrieve/pii/S1476927122000524 (visited on 02/14/2023).

[68] David G Kleinbaum and Mitchel Klein. *Survival analysis*. Vol. 3. Springer, 2010.

[69] SB Kotsiantis, Dimitris Kanellopoulos, and PE Pintelas. "Data preprocessing for supervised leaning". In: *International Journal of Computer Science* 1 (2) (2006), pp. 111–117.

[70] Oliver Kuss. "Global goodness-of-fit tests in logistic regression with sparse data". In: *Statistics in medicine* 21 (24) (2002), pp. 3789–3801.

[71] Rosanne M Leipzig, Robert G Cumming, and Mary E Tinetti. "Drugs and falls in older people: a systematic review and meta-analysis: I. Psychotropic drugs". In: *Journal of the American Geriatrics Society* 47 (1) (1999), pp. 30–39.

[72] Rosanne M Leipzig, Robert G Cumming, and Mary E Tinetti. "Drugs and falls in older people: a systematic review and meta-analysis: II. Cardiac and analgesic drugs". In: *Journal of the American Geriatrics Society* 47 (1) (1999), pp. 40–50.

[73] Mariano Fernández López, Asunción Gómez-Pérez, Juan Pazos Sierra, and Alejandro Pazos Sierra. "Building a chemical ontology using methontology and the ontology design environment". In: *IEEE Intelligent Systems and their applications* 14 (1) (1999), pp. 37–46.

[74] Anas Maach, Jamila Elalami, Noureddine Elalami, and El Houssine El Mazoudi. "An Intelligent Decision Support Ensemble Voting Model for Coronary Artery Disease Prediction in Smart Healthcare Monitoring Environments". en. In: *International Journal of Advanced Computer Science and Applications* 13 (9) (2022). arXiv:2210.14906 [cs]. ISSN: 21565570, 2158107X. DOI: 10.14569/IJACSA.2022.0130984. URL: http://arxiv.org/abs/2210.14906 (visited on 02/14/2023).

[75] A. Marier, L.E. Olsho, W. Rhodes, and W.D. Spector. "Improving prediction of fall risk among nursing home residents using electronic medical records". In: *Journal of the American Medical Informatics Association* 23 (2) (2016), pp. 276–82.

[76] Simone Marini, Emanuele Trifoglio, Nicola Barbarini, Francesco Sambo, Barbara Di Camillo, Alberto Malovini, Marco Manfrini, Claudio Cobelli, and Riccardo Bellazzi. "A Dynamic Bayesian Network model for long-term simulation of clinical complications in type 1 diabetes". In: *Journal of Biomedical Informatics* 57 (2015), pp. 369–376. ISSN: 1532-0464. DOI: https://doi.org/10.1016/j.jbi.2015.08.021. URL: https://www.sciencedirect.com/science/article/pii/S1532046415001896.

[77] Michael Marschollek, Mehmet Gövercin, Stefan Rust, Matthias Gietzelt, Mareike Schulze, Klaus-Hendrik Wolf, and Elisabeth Steinhagen-Thiessen. "Mining geriatric assessment data for in-patient fall prediction models and high-risk subgroups". In: *BMC Medical Informatics and Decision Making* 12 (19) (2012), pp. 1–6.

[78] Scott McLachlan, Kudakwashe Dube, Graham A Hitman, Norman E Fenton, and Evangelia Kyrimi. "Bayesian networks in healthcare: Distribution by medical condition". In: *Artificial Intelligence in Medicine* 107 (2020), p. 101912. ISSN: 0933-3657.

[79] Tom M Mitchell and Tom M Mitchell. *Machine learning*. Vol. 1. 9. McGraw-hill New York, 1997.

[80] Michelle E. Mlinac and Michelle C. Feng. "Assessment of Activities of Daily Living, Self-Care, and Independence". In: *Archives of Clinical Neuropsychology* 31 (6) (Aug. 2016), pp. 506–516.

[81] Israa Mohamed, Mostafa M. Fouda, and Khalid M. Hosny. "Machine Learning Algorithms for COPD Patients Readmission Prediction: A Data Analytics Approach". en. In: *IEEE Access* 10 (2022), pp. 15279–15287. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2022.3148600. URL: https://ieeexplore.ieee.org/document/9701336/ (visited on 02/14/2023).

[82] Siti Fairuz Mohd Radzi, Mohd Sayuti Hassan, and Muhammad Abdul Hadi Mohd Radzi. "Comparison of classification algorithms for predicting autistic spectrum disorder using WEKA modeler". en. In: *BMC Medical Informatics and Decision Making* 22 (1) (Nov. 2022), p. 306. ISSN: 1472-6947. DOI: 10.1186/s12911-022-02050-x. URL: https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-022-02050-x (visited on 02/14/2023).

[83] Manuel Montero-Odasso, Marcelo Schapira, Enrique R Soriano, Miguel Varela, Roberto Kaplan, Luis A Camera, and L Marcelo Mayorga. "Gait velocity as a single predictor of adverse events in healthy seniors aged 75 years and older". In: *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences* 60 (10) (2005), pp. 1304–1309.

[84] Mário WL Moreira, Joel JPC Rodrigues, Antonio MB Oliveira, Ronaldo F Ramos, and Kashif Saleem. "A preeclampsia diagnosis approach using Bayesian networks". In: *2016 IEEE International Conference on Communications (ICC)*. IEEE. 2016, pp. 1–5.

[85] Ali Ben Mrad, Véronique Delcroix, Sylvain Piechowiak, Philip Leicester, and Mohamed Abid. "An explication of uncertain evidence in Bayesian networks: likelihood evidence and probabilistic evidence - Uncertain evidence in Bayesian networks". In: *Appl. Intell.* 43 (4) (2015), pp. 802–824.

[86] Kevin P Murphy. "Dynamic bayesian networks: Representation, inference and learning". PhD thesis. University of California, Berkeley, 2002.

[87] Kevin P Murphy. *Machine learning: a probabilistic perspective*. Prentice Hall Press, 2012.

[88] P. Naım, P.H. Wuillemin, P. Leray, O. Pourret, and A. Becker. *Réseaux bayésiens*. Algorithmes. Eyrolles, 2011. ISBN: 9782212047233. URL: https://books.google.fr/books?id=7d%5C_Jq2ehb0oC.

[89] Thomas Dyhre Nielsen and Finn Verner Jensen. *Bayesian networks and decision graphs*. Springer Science & Business Media, 2007.

[90] Natalya F Noy, Deborah L McGuinness, et al. *Ontology development 101: A guide to creating your first ontology*. 2001.

[91] Ravi B Parikh et al. "Machine learning approaches to predict 6-month mortality among patients with cancer". In: *JAMA network open* 2 (10) (2019), e1915997–e1915997.

[92] James L Peugh and Craig K Enders. "Missing data in educational research: A review of reporting practices and suggestions for improvement". In: *Review of educational research* 74 (4) (2004), pp. 525–556.

[93] CA Pfortmueller, Gregor Lindner, and AK Exadaktylos. "Reducing fall risk in the elderly: risk factors and fall prevention, a systematic review". In: *Minerva Med* 105 (4) (2014), pp. 275–81.

[94] Wayne A Ray, Marie R Griffin, William Schaffner, David K Baugh, and L Joseph Melton III. "Psychotropic drug use and the risk of hip fracture". In: *New England Journal of Medicine* 316 (7) (1987), pp. 363–369.

[95] Irina Rish et al. "An empirical study of the naive Bayes classifier". In: *IJCAI 2001 workshop on empirical methods in artificial intelligence*. Vol. 3. 22. 2001, pp. 41–46.

[96] Laurence Z Rubenstein. "Falls in older people: epidemiology, risk factors and strategies for prevention". In: *Age and ageing* 35 (suppl_2) (2006), pp. ii37–ii41.

[97] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. MIT press, 2009.

[98] Haute Autorité de Santé. *Mise en oeuvre de l'éducation thérapeutique dans le cadre de l'expérimentation PAERPA*. 2014.

[99] Shirish Krishnaj Shevade and S Sathiya Keerthi. "A simple and efficient algorithm for gene selection using sparse logistic regression". In: *Bioinformatics* 19 (17) (2003), pp. 2246–2253.

[100] Gulshan Sihag, Veronique Delcroix, Emmanuelle Grislin, Xavier Siebert, Sylvain Piechowiak, and Francois Puisieux. "Prediction of Risk Factors for Fall using Bayesian Networks with Partial Health Information". In: *Globecom AIdSH Workshop*. IEEE. 2020, pp. 1–6.

[101] Gulshan Sihag, Veronique Delcroix, Emmanuelle Grisllin, Sylvain Piechowiak, and Xavier Siebert. "Using oversampling with a Bayesian network when data is imbalanced: study on a real case". *Accepted to 11èmes Journées Francophones sur les Réseaux Bayésiens et les Modèles Graphiques Probabilis*. 2023.

[102] Gulshan Sihag, Veronique Delcroix, Emmanuelle Grisllin, Xavier Siebert, and Sylvain Piechowiak. "Machine Learning Approaches to Identifying Risk Factors for Falls Among Elderly People: A Comparative Study". *Accepted to 11èmes Journées Francophones sur les Réseaux Bayésiens et les Modèles Graphiques Probabilis*. 2023.

[103] Gulshan Sihag, Véronique Delcroix, Emmanuelle Grislin-Le Strugeon, Xavier Siebert, and Sylvain Piechowiak. "Temporal Data Simulation based on a real data set for fall prevention". In: *10èmes Journées Francophones sur les Réseaux Bayésiens et les Modèles Graphiques Probabilis*. 2021.

[104] Gulshan Sihag, Véronique Delcroix, Emmanuelle Grislin-Le Strugeon, Xavier Siebert, Sylvain Piechowiak, Cédric Gaxatte, and François Puisieux. "Evaluation of risk factors for fall in elderly using Bayesian networks: A case study". In: *Computer Methods and Programs in Biomedicine Update* 1 (2021), p. 100035.

[105] Gulshan Sihag, Pankaj Yadav, Vivek Vijay, Veronique Delcroix, Xavier Siebert, Sandeep Yadav, and FranÇois Puisieux. "Advantages of oversampling techniques: a case study in risk factors for fall prediction". *Submitted as a book chapter in Springer*. 2022.

[106] Gulshan Sihag., Pankaj Yadav., Veronique Delcroix., Vivek Vijay., Xavier Siebert., Sandeep Yadav., and FranÇois Puisieux. "Evaluation of Risk Factors for Fall in Elderly People from Imbalanced Data using the Oversampling Technique SMOTE". In: *Proceedings of the 8th International Conference on Information and Communication Technologies for Ageing Well and e-Health - ICT4AWE*, INSTICC. SciTePress, 2022, pp. 50–58. ISBN: 978-989-758-566-1. DOI: 10.5220/0011041200003188.

[107] Hugo Silva and Jorge Bernardino. "Machine Learning Algorithms: An Experimental Evaluation for Decision Support Systems". en. In: *Algorithms* 15 (4) (Apr. 2022), p. 130. ISSN: 1999-4893. DOI: 10.3390/a15040130. URL: https://www.mdpi.com/1999-4893/15/4/130 (visited on 02/14/2023).

[108] Jaemun Sim, Jonathan Sangyun Lee, and Ohbyung Kwon. "Missing values and optimal selection of an imputation method and classification algorithm to improve the accuracy of ubiquitous computing applications". In: *Mathematical problems in engineering* 2015 (2015).

[109] American Geriatrics Society, Geriatrics Society, American Academy Of, and Orthopaedic Surgeons Panel On Falls Prevention. "Guideline for the prevention of falls in older persons". In: *Journal of the American Geriatrics Society* 49 (5) (2001), pp. 664–672.

[110] Marina Sokolova, Nathalie Japkowicz, and Stan Szpakowicz. "Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation". In: *Australasian joint conference on artificial intelligence*. Springer. 2006, pp. 1015–1021.

[111] Ingo Steinwart and Andreas Christmann. *Support vector machines*. Springer Science & Business Media, 2008.

[112] Sudeep Tanwar, Jayneel Vora, Shriya Kaneriya, Sudhanshu Tyagi, Neeraj Kumar, Vishal Sharma, and Ilsun You. "Human arthritis analysis in fog computing environment using Bayesian network classifier and thread protocol". In: *IEEE Consumer Electronics Magazine* 9 (1) (2019), pp. 88–94.

[113] Joanne K Taylor, Iain E Buchan, and Sabine N Van Der Veer. "Assessing life-space mobility for a more holistic view on wellbeing in geriatric research and clinical practice". In: *Aging clinical and experimental research* 31 (4) (2019), pp. 439–445.

[114] David Thesmar, David Sraer, Lisa Pinheiro, Nick Dadson, Razvan Veliche, and Paul Greenberg. "Combining the power of artificial intelligence with the richness of healthcare claims data: opportunities and challenges". In: *PharmacoEconomics* 37 (6) (2019), pp. 745–752.

[115] Paul Thottakkara, Tezcan Ozrazgat-Baslanti, Bradley B Hupf, Parisa Rashidi, Panos Pardalos, Petar Momcilovic, and Azra Bihorac. "Application of machine learning techniques to high-dimensional clinical data to forecast postoperative complications". In: *PloS one* 11 (5) (2016), e0155705.

[116] Mary E Tinetti. "Preventing falls in elderly persons". In: *New England journal of medicine* 348 (1) (2003), pp. 42–49.

[117] Mary E Tinetti, John T Doucette, and Elizabeth B Claus. "The contribution of predisposing and situational risk factors to serious fall injuries". In: *Journal of the American Geriatrics Society* 43 (11) (1995), pp. 1207–1213.

[118] Samantha Turner, Rupert Kisser, and Wim Rogmans. *Factsheet for falls among older adults in the EU-28*. Available at https://eupha.org/repository/sections/ipsp/Factsheet-falls-in-older-adults-in-EU.pdf. Accessed: 2020-05-15. 2015.

[119] UN. *World Population Prospects, Department of Economic and Social Affairs, Population Division*. https://population.un.org/wpp/. Accessed: 2020-05-15. 2019.

[120] Andrea Ungar, Martina Rafanelli, Iacopo Iacomelli, Maria Angela Brunetti, Alice Ceccofiglio, Francesca Tesi, and Niccolò Marchionni. "Fall prevention in the elderly". In: *Clinical Cases in mineral and bone metabolism* 10 (2) (2013), p. 91.

[121] Mike Uschold and Michael Gruninger. "Ontologies: Principles, methods and applications". In: *The knowledge engineering review* 11 (2) (1996), pp. 93–136.

[122] Alfredo Vellido. "The importance of interpretability and visualization in machine learning for applications in medicine and health care". In: *Neural computing and applications* 32 (24) (2020), pp. 18069–18083.

[123] Akbar K Waljee, Ashin Mukherjee, Amit G Singal, Yiwei Zhang, Jeffrey Warren, Ulysses Balis, Jorge Marrero, Ji Zhu, and Peter DR Higgins. "Comparison of imputation methods for missing laboratory data in medicine". In: *BMJ open* 3 (8) (2013), e002847.

[124] Akbar K Waljee et al. "Predicting hospitalization and outpatient corticosteroid use in inflammatory bowel disease patients using machine learning". In: *Inflammatory bowel diseases* 24 (1) (2018), pp. 45–53.

[125] Kung-Jeng Wang, Jyun-Lin Chen, and Kung-Min Wang. "Medical expenditure estimation by Bayesian network for lung cancer patients at different severity stages". In: *Computers in Biology and Medicine* 106 (2019), pp. 97–105. ISSN: 0010-4825.

[126] Ahrou Wassim, Elalaouy Elarbi, and Rhoulami Khadija. "Application of Machine Learning Approaches in Health Care Sector to The Diagnosis of Breast Cancer". en. In: *Journal of Physics: Conference Series* 2224 (1) (Apr. 2022), p. 012012. ISSN: 1742-6588, 1742-6596. DOI: 10.1088/1742-6596/2224/1/012012. URL: https://iopscience.iop.org/article/10.1088/1742-6596/2224/1/012012 (visited on 02/14/2023).

[127] Stephen F Weng, Jenna Reps, Joe Kai, Jonathan M Garibaldi, and Nadeem Qureshi. "Can machine-learning improve cardiovascular risk prediction using routine clinical data?" In: *PloS one* 12 (4) (2017), e0174944.

[128] Jenna Wiens and Erica S Shenoy. "Machine learning for healthcare: on the verge of a major shift in healthcare epidemiology". In: *Clinical Infectious Diseases* 66 (1) (2018), pp. 149–153.

[129] World Health Organization. *WHO global report on falls prevention in older age*. https://apps.who.int/iris/handle/10665/43811. 2008.

[130] Tung-Kuang Wu, Shian-Chang Huang, and Ying-Ru Meng. "Evaluation of ANN and SVM classifiers as predictors to the diagnosis of students with learning disabilities". In: *Expert Systems with Applications* 34 (3) (2008), pp. 1846–1856.

[131] Chengyin Ye et al. "Identification of elders at higher risk for fall with statewide electronic health records and a machine learning algorithm". In: *International Journal of Medical Informatics* 137 (2020), p. 104105.

[132] Mitchell Yuwono, Steven W Su, and Bruce Moulton. "Fall detection using a Gaussian distribution of clustered knowledge, augmented radial basis neural-network, and multilayer perceptron". In: *7th International Conference on Broadband Communications and Biomedical Applications*. IEEE. 2011, pp. 145–150.

[133] Xiaoru Zheng. *SMOTE variants for imbalanced binary classification: heart disease prediction*. University of California, Los Angeles, 2020.

[134] Xiaojin Zhu and Andrew B Goldberg. "Introduction to semi-supervised learning". In: *Synthesis lectures on artificial intelligence and machine learning* 3 (1) (2009), pp. 1–130.

Appendix A

Temporal Data Simulation based on a real data set for fall prevention

Outline of the current chapter

A.1 Introduction

Data are the basis of a lot of work in artificial intelligence, often related to learning and reasoning. A temporal data set includes a series of values over time for a set of variables and a set of samples. With regard to data on people, describing, for example, their abilities, environment, behavior, etc, longitudinal studies allow for a collection of repeated observations over time of a phenomenon

and/or a sample of individuals. However, such data collection is very costly and it often concerns either a short period of time, and/or a small number of persons, and/or a limited number of observations for a given variable.

In order to palliate that difficulty, we aim to simulate a complete temporal data set that includes the values of all variables at each time step for a long period (several decades) and for a large number of elderly.

When dealing with static data, it is frequent to simulate data from a Bayesian network [50, 27]. Temporal data sets can also be simulated with dynamic Bayesian networks (DBN) [86]. An example is given in [76] where a cohort is simulated for fifteen years, thanks to a DBN learned from a longitudinal study. In a DBN, variables are related to each other over two or more time slices. From a theoretical point of view, in order to consider sequences of arbitrary length, a solution is to consider that the probability distributions describing the temporal dependencies are time-invariant. In that way, the relations defined between two time slices can be easily deployed (unrolled) for a particular number of steps. However, when we create a junction tree from an unrolled DBN, the cliques tend to be very large, often making exact inferences tends to become intractable [35, 86].

In this work, we propose another approach to simulate a temporal data set that allows us not to make the strong assumption that temporal evolution is time-invariant. We present an algorithm to simulate a temporal data set on the basis of a real static data set. including information collected during the multidisciplinary consultation for fall prevention in Lille's Hospital. The real data set includes only one observation per person and per variable. We first present the context and the motivation to simulate a temporal data set, followed by the real static data set from Lille's hospital. Second, we explain the algorithm and related assumptions and definitions. We also provide some elements to evaluate the quality of the simulated data set. Finally, we give some perspective about how this data set could be used in the context of the prediction of risk factors for fall from a partial time-stamped data set.

A.2 Context and motivation

This work takes place in the context of fall prevention. We present below our collaboration with Lille's hospital and our motivation to simulate a temporal data set from a real static data set provided by that service.

One-third of people aged 65 and over living at home fall every year. This is the case for half of those over 85 years of age [25]. Falls account for 40% of all injury deaths [96]. According to the World Health Organization [129], falls and consequent injuries are major public health problems that require frequent medical attention. Falls prevention is a challenge to population aging, but it is one of the issues that have not been given sufficient attention. Since falls result from a complex interaction of risk factors, an important step in fall prevention is to detect the presence of risk factors for falls.

At the hospital in Lille, patients are received in a day hospital for a multidisciplinary evaluation of the risk factors for falls. This leads to the selection of a small number of adapted recommendations. Most part of the time this specialized consultation consists of data collection by different specialists, using specific types of equipment and tests. It provides a picture of the person's current state, behavior, and environment, incorporating past events that can help to assess risk factors for falls.

However, outside the context of a specialized consultation on falls, such a complete data collection is not possible because of a lack of time, expertise, and equipment. Though, there are many potential actors in the prevention of falls, and furthermore, it is possible to have almost

instantaneously a partial set of information on a person from his or her personal medical record (electronic health record). These records regroup a collection of reports and information over the patient's life and are increasingly being used. It is therefore possible to extract quickly dated information about a person. But for some variables, the person's current condition may have changed making the information useless, even misleading.

In real life, when information is required immediately for a given person, such as for fall prevention by general practitioners, the available information can be seen as a partial time-stamped observation set. Beyond this study, our overall objective is to allow an assessment of the risk factors for falls, based on a partial set of dated information [29]. For this purpose, we need knowledge about the dynamics of the considered variables, as well as a sufficient temporal data set in a number of patients, covering a long period of time with a fine time step. For these reasons, we aim to simulate a realistic temporal data set about features of interest in fall prevention.

The simulated data will make it possible to simulate a partially dated observation set and to build and evaluate some models or algorithms to predict fall risk factors from a partially dated observation set. The prediction of the unobserved risk factors for falls contributes to fall prevention since adequate actions may reduce those risks and in turn, reduce the risk of falls. In this work, the variables about the loss of autonomy (*ADLinf5*) and dementia (*demence*) are two target risk factors for falls.

Below, we present a brief overview of some existing methods to predict fall risk, then, we present the static data set provided by Lille's hospital that we use to simulate temporal data.

A.3 Overview on some methods to predict fall risk

Several recent articles focus on the prediction of fall or fall risk based on models learned from large data sets [77, 75, 59, 131] which comes from the population for which the data collection is facilitated (in-patients [77], nursing home residents [75], or people with a specific program of given health insurance, ensuring complete claims coverage [59]. Despite this favorable data source, in [77], the authors mentioned the limitation to generalizing their findings due to the significant amount of missing data for some sub-items. Our work is also motivated by this aspect with the final objective to propose a way to help in fall prevention for the whole elderly population, on the basis of their available information, even if it is very partial. Furthermore, existing Electronic Medical Record (EMR) systems do not provide an easy mechanism to synthesize and summarize information on changing risk variables collected in various portions of the EMR to support clinical decision-making [75]. This point brings us to our second consideration, which is determining how old data can be used to assess present risk. Finally, all of those articles are concerned with assessing fall risk, whereas we focus on the evaluation of risk factors for falls.

A.4 Lille's data set and variable selection

As a reminder, the real data set from the multidisciplinary consultation for fall prevention of Lille's Hospital includes personal data from about 1810 persons, collected between 2005 and 2016. In that study, we keep only the 1752 cases with ages between 65 and 95.

The original file contains more than 400 columns, among which we have first selected 65 variables for a previous study about the prediction of the main risk factors for fall [100]. We now present the five variables selected for this work.

	Short name	Description	Persistent variable
G	GUGOgt20	true when the result of the Get Up and GO test is greater than 20 seconds	positive
C	conduit	true when the person still drives her car	negative
A	ADLinf5	true when the score of Activities of Daily Living (ADL) is less than 5	positive
D	demence	true when a dementia is probable or confirmed	positive
M	maisRet	true when the person lives in a retirement home	positive

Table A.1: The persistent variables selected for that study

A.4.1 Variables selection from the real data set

We had several interviews with Pr. Puisieux, from the multidisciplinary consultation on fall prevention at Lille's hospital about the way the 65 variables previously selected evolve with time. As a result, we have identified a subset of variables whose temporal behavior is simple and that we name (positive) *persistent* variables. They are binary variables, whose value is most often false for young people and that can change at most once during the life of a person. Table A.1 presents the 5 persistent variables that we select for the purpose of the current study.

The variables *demence* and *ADLinf5* are important predisposing risk factors for fall [30]. The variable *ADLinf5* is an indicator of loss of autonomy. ADL measurements and scales can vary significantly [80]. The Katz Index of independence in ADLs [65] is one of the most commonly used tools to asses basic ADLs (bathing, dressing, toileting transferring, continence, and feeding). Both *demence* and *ADLinf5* are important to be predicted because information related with these risk factors can be difficult to collect and because they are modifiable, meaning that specific actions can be conducted to reduce them.

The get-up and go test (*GUGOgt20*) is related to gait disorder. When its score is greater than 20 seconds, it is considered a risk factor for fall [30].

The four variables *GUGOgt20*, *maisRet*, *ADLinf5* and *demence* are positive persistent: when they become true for a given person, there is no chance that they become false again later. The variable *conduit* is negative persistent since it is generally true for adults and becomes false when the capacities of the elderly decrease, while people who did not drive as adults will not drive as elderly.

A.5 Definitions and assumptions

Before presenting our algorithm to simulate a temporal data set that represents information over time on a set of persons, we first introduce some notations, definitions, and assumptions used for the temporal data simulation.

A.5.1 Notations

Here is a list of some notations:

- $\mathbf{X}$: main set of variables,
- $X, X_i \in \mathbf{X}$: some random variables,
- $Dom(X)$: domain of the variable X,
- $Dom(\mathbf{Y}) = Dom(Y_1) \times \ldots \times Dom(Y_m)$, where $\mathbf{Y} = \{Y_1, \ldots, Y_m\} \subset \mathbf{X}$
- $x \in Dom(X)$, $x_i \in Dom(X_i)$: a value of X or X_i [1],
- $\mathcal{T} = \{t_0, t_1, \ldots, t_p\}$ with $t_{i+1} = t_i + \Delta t$: period of time over which information is simulated and Δt is the length of a step,
- t, $t_k \in \mathcal{T}$: different times,
- $x^t \in Dom(X)$, $x_i^t \in Dom(X_i)$: values of the variables X and X_i at the time t for a given person.
- N: size of the population (number of samples),
- n: index of a specific person,
- $\mathcal{D}_{\mathcal{T}}$: complete temporal data set over the period $\mathcal{T}$,

We now present the definitions and related assumptions regarding the variables and their temporal evolution in a context defined by the Bayesian network.

A.5.2 Variables, observations and temporal data set

In this study, we consider only binary variables. For any variable X, $Dom(X) = \{0, 1\}$, where 1 is called the positive value. We also consider only hard observation (see [85] about uncertain observations and [29] about their use in fall prevention). Let's precise definitions and notations regarding dated information.

Definition 1 ***Time stamped observation***
A time stamped observation o on a variable X for a given person n is a tuple $o = (X, x, t, n)$ where $x \in Dom(X)$ is the value of X observed at t.

An time stamped observation $o = (X, x, t, n)$ of a binary variable X is said to be *positive* when the observed value is positive ($x = 1$).

Definition 2 ***Complete temporal data set***
A time stamped data set $\mathcal{D}$ on the set of variables $\boldsymbol{X}$ and a set of persons indexed by $[1..N]$ is said to be complete over a period $\mathcal{T}$ when the set $\mathcal{D} = \{(X, x, t, n), X \in \boldsymbol{X}, t \in \mathcal{T}, n \in [1..N]\}$ includes exactly one value for each element $(X, t, n) \in \boldsymbol{X} \times \mathcal{T} \times [1..N]$.

[1] We do not use a specific notation to distinguish the different values of X in $Dom(X)$

When we consider a specific ordered subset of variables $\mathbf{X_J} = (\dots, X_j, \dots)$ and one of its a possible setting $\mathbf{v} = (\dots, v_j, \dots)$, we write that $(\mathbf{X_J}, \mathbf{v}, t, n) \in \mathcal{D}$ to denote that for each variable $X_j \in \mathbf{X_J}$ and its value v_j, the element (X_j, v_j, t, n) belongs to the temporal data set $\mathcal{D}$.

Furthermore, in order to set the way each variable evolves with time, it appears useful to take into account the value of some other variables. Indeed, for a given variable, different schemas of temporal changes can be defined depending on the values of some other variables.

In that aim, we use a Bayesian network to define the dependence between variables [88]. We denote $pa(X)$ as the parents of the variable X in the graph of the Bayesian network.

In this work, we assume that the Bayesian network graph does not change with time. As a consequence, each variable is associated with a *context* defined by the values of its parents in the graph.

We denote $(X, \mathbf{v})$ a variable X in a context $\mathbf{v}$, where $\mathbf{v}$ is one of the possible combinations of values of the parents of X in the graph of a Bayesian network $\mathcal{B}$: $\mathbf{v} \in Dom(pa(X))$. We name such a couple a *contextualized variable*, and say that $\mathbf{v}$ is one of the context of X in $\mathcal{B}$.

In order to simplify the notation, when X has no parent in the graph of the Bayesian network, $(pa(X) = \emptyset)$, the couple $(X, \mathbf{v})$ represents the variable X.

In the following, we present a schema of temporal evolution for each contextualized variable $(X, \mathbf{v})$.

A.5.3 Persistent variable

We got a better understanding of the way the variables change with time thanks to the interviews with Professor Puisieux. It appears that variables can be classified into several classes regarding the characteristics of their change over time. Except for constant variables that never change, such as sex, we define the concept of persistent variables as the simplest class regarding temporal evolution. A variable is called persistent if its value never changes once it becomes true. for example, once a person has developed Parkinson's disease he never gets recovered.

Definition 3 ***Positive (resp. negative) persistent variable***

A binary variable X with $Dom(X) = \{0, 1\}$ is said to be positive persistent in a temporal data set $\mathcal{D}$ when its value never changes after the value becomes 1 for a given person indexed by n :

$$\forall t, t' \in \mathcal{T}, \text{ with } t' > t, (X, 1, t, n) \in \mathcal{D} \Rightarrow (X, 1, t', n) \in \mathcal{D}$$

Respectively, the value of a negative persistent variable never changes after it becomes zero.

As a consequence, when we consider a population composed of a group of persons, the proportion of persons with X being positive increases with the age of the persons. Thus, when a variable X is positive persistent, the function $f(age) = P(X = 1 \mid age)$ is an increasing function, where $P(X = 1 \mid age)$ denotes the probability for a variable X to be positive among the given age group[2].

In this work, we consider only persistent variables. Figure A.1 shows the graph of the Bayesian network for the five variables that we consider in this article. To get it, we first learned a Bayesian network from the real data set, then we removed the arc $conduit \rightarrow demence$ so that every node has at most two parents. Indeed, the number of combinations of the values of the

[2] In our simulated temporal data set, the number of persons is constant whatever the age group. On the contrary, in the real static data set, the distribution regarding the age is not constant, making it important to consider the conditional probability and not the joint probability.

parents is higher with three parents, making it possible that some cases have no representing sample in the data set.

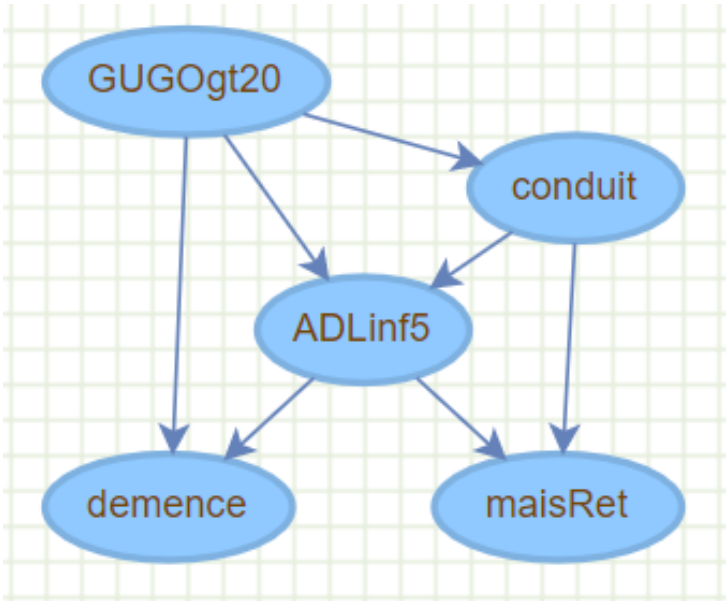


Figure A.1: Graph of the Bayesian network.

Since our real data set includes the age of the persons, we use that information to extract the temporal behavior of the variables: we assume that the distributions of each variable in function of the age on the whole population allow us to derive the evolution of the variables for a given person regarding her age. In that aim, we plot the distribution of each variable regarding the age of the persons from the real data set (see Figure A.2).

In order to simulate temporal series for each of these variables, we consider a linear interpolation for each curve. This approximation, combined with the feature of persistent variables, is used to compute the probability of a variable to become positive at a defined time step when it was negative at the previous time step. It is important to note that more complex interpolation functions could be used in the algorithm that we propose.

In addition, we also want that the simulated data set reflects the dependencies between variables, such as described by a Bayesian network. With that aim, we make further assumptions described below.

A.5.4 Parent-persistent contextualized variable

Our goal is now to combine the information given by dependencies between variables in the Bayesian network and information from the distribution of positive value in function of the age, in order to simulate a temporal data set. In that aim, we introduce a concept related to the evolution of a variable in the context of its parents' values, which extends the concept of a persistent variable.

Consider a Bayesian network $\mathcal{B}$ on a set of variables including a binary variable X, and let $\mathbf{v} \in Dom(pa(X))$ be a context of X in $\mathcal{B}$.

Definition 4 ***Positive (resp. negative) parent-persistent contextualized variable***

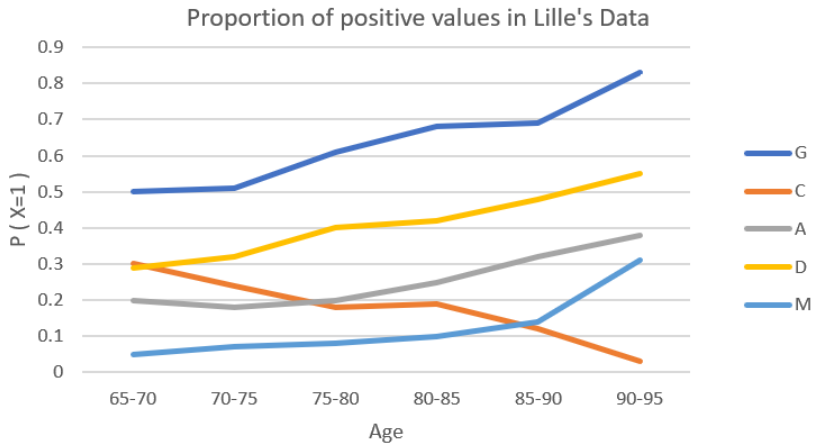


Figure A.2: Proportion of positive values for each variable according to the age in Lille's real data set.

A contextualized variable $(X, \mathbf{v})$ *is said to be positive parent-persistent in a temporal data set* $\mathcal{D}$ *when its value in the context* $\mathbf{v}$ *never changes after the value becomes 1 for a given person indexed by* n*:*

$$\begin{aligned} &\forall t, t' \in \mathcal{T}, \text{ with } t' > t, \\ &\text{if for each couple}(Y, y) \text{ with } Y \in pa(X) \\ &\text{and } y \text{ is the value of } Y \text{ in } \mathbf{v}, \\ &(Y, y, t, n) \in \mathcal{D} \text{ and } (Y, y, t', n) \in \mathcal{D}, \text{ then} \\ &(X, 1, t, n) \in \mathcal{D} \Rightarrow (X, 1, t', n) \in \mathcal{D} \end{aligned} \tag{A.1}$$

Respectively, we consider also negative parent-persistent contextualized variables.

Remark: If a variable X is positive persistent in a data set $\mathcal{D}$, then for any context $\mathbf{v}$, the contextualized variable $(X, \mathbf{v})$ is positive parent-persistent in $\mathcal{D}$.

In this work, we thus assume that all contextualized variables are parent-persistent. For convenience, we also speak about contextualized variables when the set of parents is empty.

Figures A.3 to A.6 shows the distribution of each contextualized variable in function of the age, computed from our real static data set. These plots are based on intervals of five years for the age. For each contextualized variable $(X, \mathbf{v})$, we plot the proportion

$$\frac{\#samples(X = 1, Pa(X) = \mathbf{v}, Age = a_l)}{\#samples(Pa(X) = \mathbf{v}, Age = a_l)}$$

where $Dom(Age) = a_1, \ldots a_l, \ldots$, computed from the real static data set.

More data is needed to obtain smoother curves.

A.5.5 Linear assumption

Figures A.3 to A.6 show the distribution of contextualized variables regarding the age of patients. These curves are based on a discretization of Lille's data set with intervals of 5 years, which is a compromise between information quality and statistical quality.

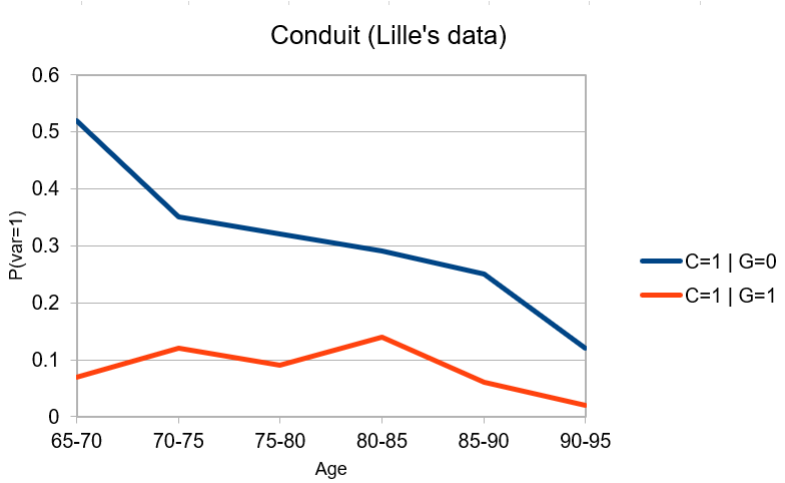


Figure A.3: Proportion of positive values of Conduit variable in function of the age.

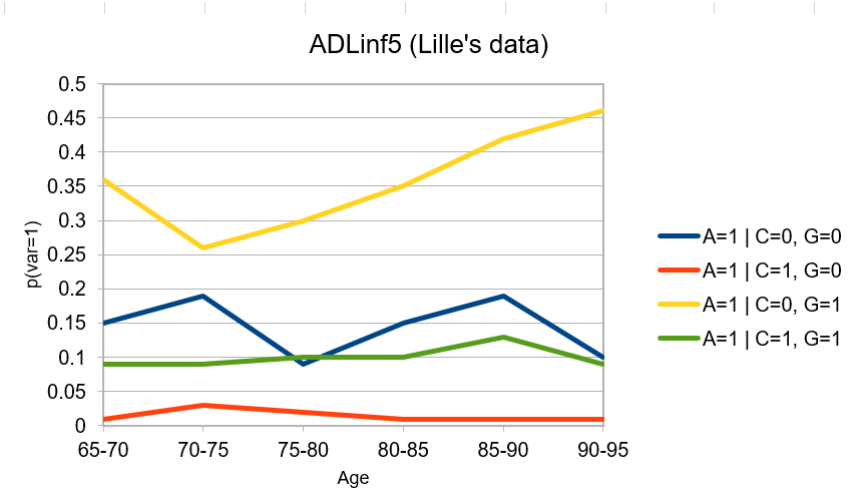


Figure A.4: Proportion of positive values of Activity of daily living variable in function of the age.

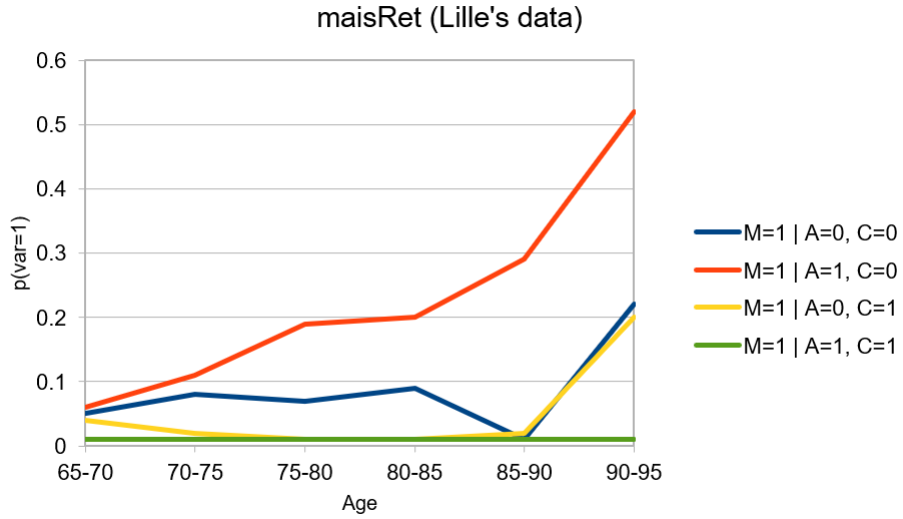


Figure A.5: Proportion of positive values of lives in retirement home variable in function of the age.

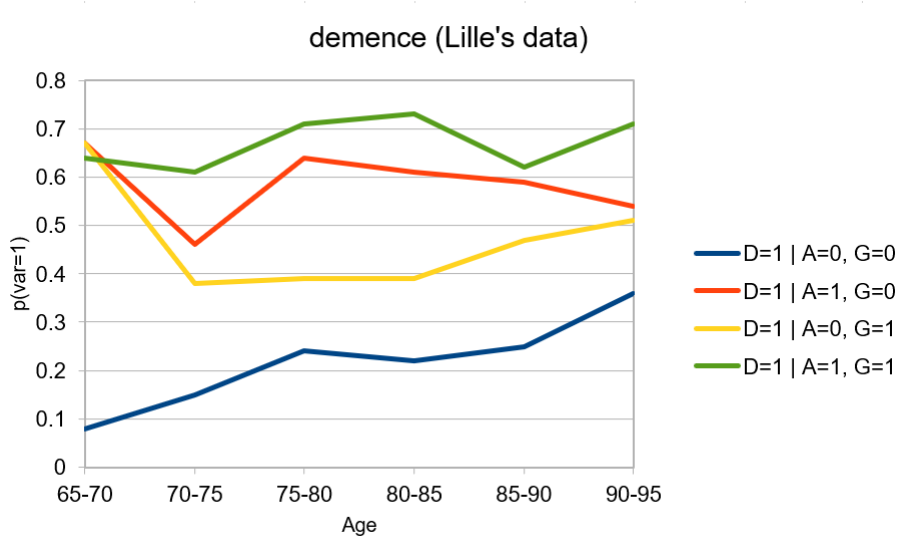


Figure A.6: Proportion of positive values of Dementia variable in function of the age.

In order to generate temporal data with any desired temporal granularity, while remaining faithful to the real data set, we replace these curves by interpolated functions. We choose linear interpolation:

- the functions $f(age) = P(X = x \mid age)$ are linear functions, for all $X \in \mathbf{X}$,
- the functions $f(age) = P(X = x \mid \mathbf{X_J} = \mathbf{v}, age)$ are linear functions, for all $\mathbf{v} \in Dom(\mathbf{X_J})$ where $\mathbf{X_J} = pa(X)$.

From the distributions plotted in Figure **??**, we have defined a linear function associated with each contextualized variable.

A.5.6 About Survival Analysis

The functions shown in Figures A.2 and from A.3 to A.6 are very similar to survival functions and hazard rate conditional [68]. We show the evolution of the risk whereas survival functions usually show the chances that a person survives. In our case, the event of interest is the change of value of a risk factor, from absent (0) to present (1). Some methods to estimate the Survival function are based on the assumption that data follows some distribution (such as exponential, gamma, weibull, log-normal, etc.) and then we calculate its parameters. Other methods such as 'Kaplan-Meier' estimator do not have any prior assumptions. However, estimating survival function from data supposes that data include information about the response for each subject. In that kind of data, the subject is always "alive" when the study period starts, and the event of interest may or not occur before the end of that period. When the event does not occur, the survival time is labeled as 'Censored'.

In our case, our data from Lille's Hospital are very different since it corresponds to a single moment of observation for each subject, and we do not know when the risk occurs. At the moment of the observation, the risk is present for some people and absent for others. Because we do not have information about the time when the event of interest occurs, we take benefit from the fact that the observed population involves persons of different ages, and we assume that the proportion of persons with a risk factor at a given age may give us a way to estimate the survival function.

A.5.7 Assumptions regarding the period of time over which data are simulated

In order to simplify the simulation of the data set, we assume that the period $\mathcal{T}$ starts at time t_0 with all persons being 65 years old. This assumption can be easily removed later by shifting each data row randomly in time. This would allow us to get a data set in which people of any age over 65 are considered at time t_0.

Second, we simulate data for all persons during the complete period, meaning that we do not consider the death of people. When we want to remove that second assumption, the age of death could be simulated on the basis of general knowledge about the distribution of the age of death.

In addition, let's precise that we consider $\Delta t = 6$ months.

A.6 Algorithm to simulate temporal data set from a static data set

The objective is to generate a temporal data set. The real data set includes the age of the person.

We now describe an algorithm to generate a temporal data set on a set of variables $\mathbf{X}$, on the basis of a real data set that includes a set of observations collected only once for a given person. We follow two main ideas: (1) respect the proportion of positive values for each variable given the age of the patient, (2) respect the dependence between the variables as described by a Bayesian network $\mathcal{B}$ learned on $\mathbf{X}$.

Let $\mathbf{X} = \{X_1, \ldots X_i, \ldots\}$ such that the order of the variables in $\mathbf{X}$ is compatible with the partial order defined by the graph of the Bayesian network $\mathcal{B}$.

The algorithm `simulateTDS` generates a value for each variable X_i at each time on a given period, based on the values of the parents of X_i in the graph of the Bayesian network at the same time and on the value of X_i at the previous time.

As explained above, we extract the temporal behavior of each contextualized variable regarding the age of the person.

Two basic functions are used by the algorithm to simulate the temporal data set: `parents(X)` and `linearF(X,v,a)`. These functions are based on a Bayesian network $\mathcal{B}$ and the set of linear functions associated with each contextualized variable $(X, \mathbf{v})$, where X is a variable associated with a node of $\mathcal{B}$, and $\mathbf{v}$ is a vector of values of the parents of X in $\mathcal{B}$. The function `parents(X)` returns the list of variables that are parents of the variable X in the graph of the Bayesian network. The function `linearF(X,v,a)` returns the value of the linear function associated with the contextualized variable $(X, \mathbf{v})$ for the age a. In addition, the function `generate(p)` returns 0 or 1 with probability distribution $(1-p, p)$ where the parameter p is a value in $[0,1]$.

In order to simplify the presentation of the algorithm, we assume that we have only positive persistent variables. Indeed, a negative persistent variable can be replaced by a positive persistent variable by exchanging values 0 and 1. The temporal data are generated with a regular time step for a given number of iterations and a given number of samples. The `simulateTDS` algorithm generates each new value and fills gradually a 3D table whose dimensions correspond to the samples, the variables, and the time (lines 1–3).

The values are generated following a partial order of the variables so that the values of the parents of a given variable can be used to generate the value of this variable, and following the temporal order, so that the previous value of a variable can be used to generate its next value.

The operations to generate a value for a given person (or sample), a given variable, and a given time are detailed in the `simulateOne` algorithm. At first, the context of the variable is identified by extracting the value of its parents from the data already simulated (lines 1–2). In order to generate a value of a variable at the first time step (age = 65), one generates randomly 0 or 1 with a uniform probability corresponding to the value of 65 for the linear function associated with the contextualized variable (lines 3–5). When a previous value has already been generated for a given variable, the value to be generated depends on it: When the previous value is 1, the new value has to remain 1, by definition of a positive persistent variable (lines 6–7). When the previous value is 0, one generates randomly 0 or 1 with a uniform probability corresponding to the increase of the linear function associated with the contextualized variable during one step of time and reduced to the negative cases (lines 9–12). Remark that this step is based on the interpolated functions, but does not require these functions to be linear.

A.7 Evaluation of the simulated temporal data set

Using this algorithm and the real static data set of Lille, we have simulated a temporal data set of 2000 cases, over a period of 30 years, with a time step of 6 months.

In order to evaluate the quality of the simulated temporal data set, we plot the proportion of positive values for each variable in the simulated data set (Figure A.7). In comparison to

Algorithme 1 : simulateTDS($\mathbf{X}, K, \Delta t, N$)

Input : $\mathbf{X}$ ▷ *an ordered set of variables*
Input : K ▷ *number of temporal iterations*
Input : Δt ▷ *length of the time step*
Input : N ▷ *number of samples*
Output : D ▷ *a 3-dimension table containing the simulated temporal data set on $\mathbf{X}$ over the period $\mathcal{T}$.* The cell $D[n,i,t]$ contains the simulated value of sample n for the variable X_i, at time t.

```
1 D ← 0                                              ▷ initialize the 3D array to zero
2 foreach k ∈ [1..K] do                              ▷ generate data at time t_k
3     foreach person n ∈ [1..N] do                   ▷ generate N samples
4         foreach variable X_i ∈ X (in topological order) do   ▷ generate value of X_i at t_k
5             D[n,i,k] ← simulateOne(n, i, t_k, Δt, D)
6 return D
```

Algorithme 2 : simulateOne($n, i, t_k, \Delta t, D$)

Input : n ▷ *sample index*
Input : i ▷ *variable index*
Input : t_k ▷ *time to be simulated*
Input : Δt ▷ *length of the time step ($t_k - t_{k-1}$)*
Input-Output : D ▷ *a 3-dimension table containing the already simulated temporal data set on $\mathbf{X}$ over the period $\mathcal{T}$*

Data : a Bayesian network $\mathcal{B}$
Data : Linear functions associated with each contextualized variable $(X, \mathbf{v})$, regarding $\mathcal{B}$

```
1  X_J ← parents(X_i)
2  v ← values of X_J generated at t_k                ▷ a context of X_i
3  if k = 0 then                                     ▷ first time step t_0
4      p ← linearF(X_i, v, 65) D[n,i,0] ← generate(p)
5  else                                              ▷ generate data at time t_k from value at t_{k-1}
6      if D[n,i,k-1] = 1 then                        ▷ previous value
7          D[n,i,k] ← 1                              ▷ positive persistent variable
8      else                                          ▷ compute the probability to become 1 among the negative cases
9          c ← linearF(X_i, v, t_k) − linearF(X_i, v, t_{k-1})
10         p ← c/(1 − linearF(X_i, v, t_{k-1}))
11         D[n,i,k] ← generate(p)
```

Figure A.2, the result clearly shows that the linear assumption is faithfully reproduced in the simulated data set when considering each variable separately.

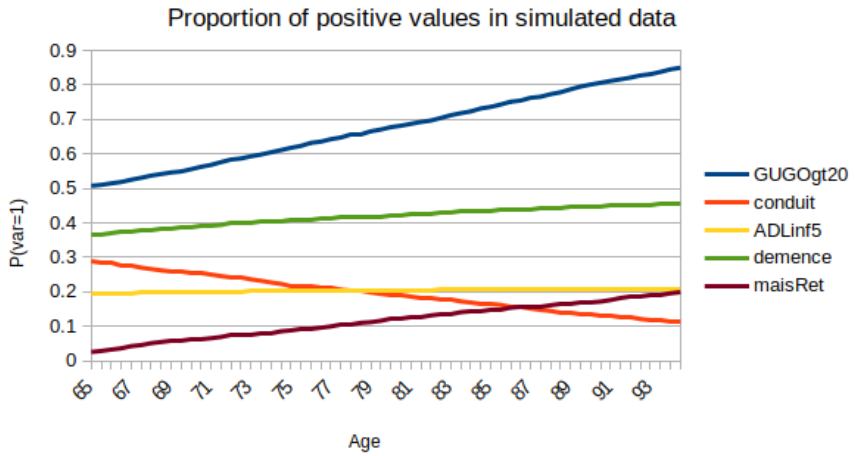


Figure A.7: Proportion of positive values of each variable in simulated data.

In addition, the algorithm of data simulation is also based on information provided by the Bayesian network learned on the real static data set, by taking into account the way each variable changes in a given context. In order to evaluate that second point, we show in Figures A.8 to A.11 the proportion of positive values for each variable in a given context along with time in the simulated data set, and we compare with the linear functions computed for each contextualized variable defined from the static data set (based on Figures A.3 to A.6).

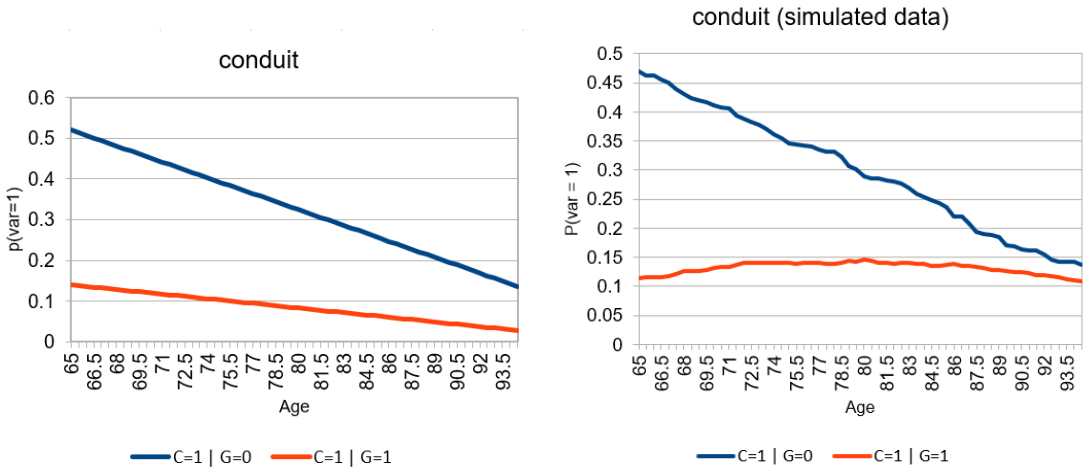


Figure A.8: Linear functions associated with conduit variable (left) and Proportion of positive value of conduit in simulated data (right).

The comparison of the linear functions and the plot from simulated data shows that in most cases, the proportion of positive values in the simulated data is faithful with the linear functions.

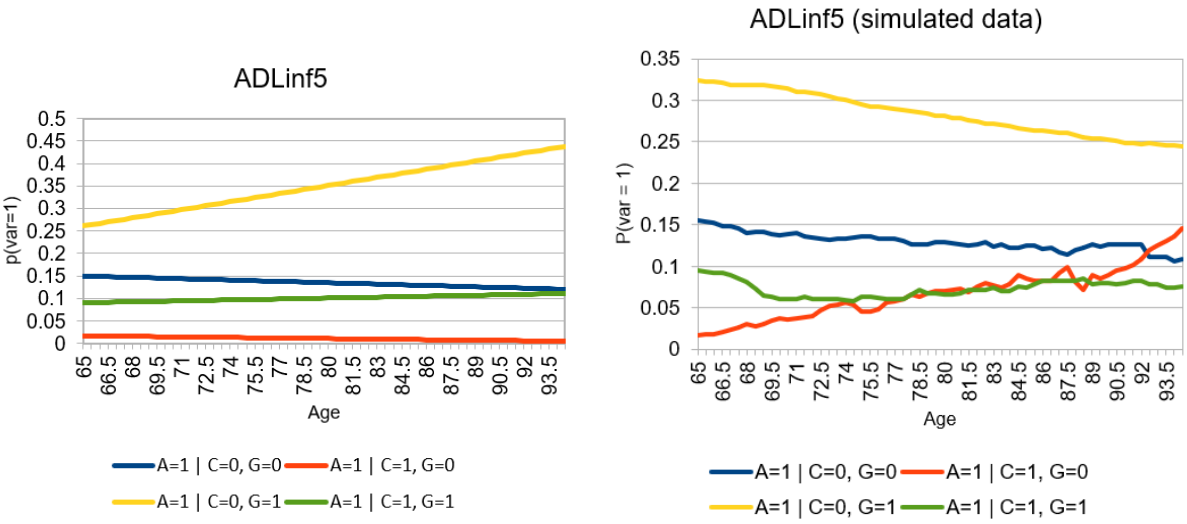


Figure A.9: Linear functions associated with activities of daily life variable (left) and Proportion of positive value of activities of daily life variable in simulated data (right).

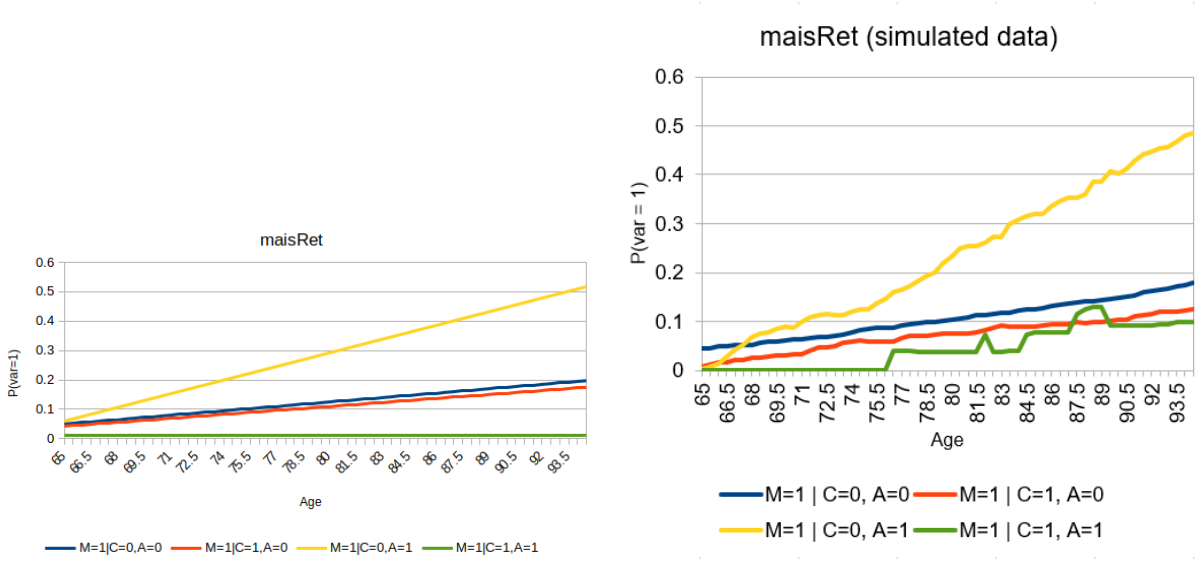


Figure A.10: Linear functions associated with lives in retirement home variable (left) and Proportion of positive value of lives in retirement home variable in simulated data (right).

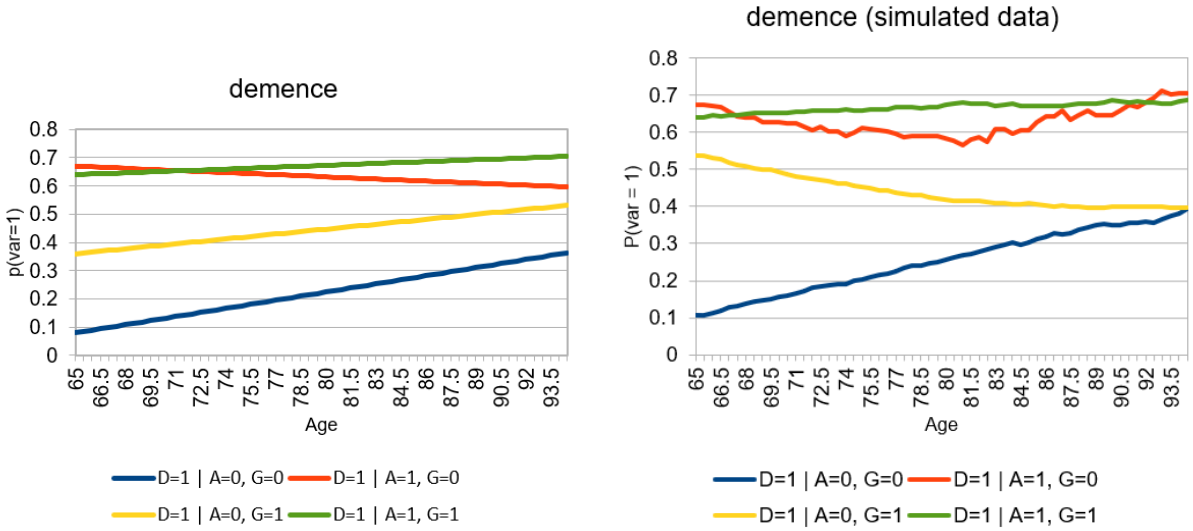


Figure A.11: Linear functions associated with dementia variable (left) and Proportion of positive value of dementia variable in simulated data (right).

A.8 Perspective and conclusion

This article proposes a first attempt to simulate temporal data using a Bayesian network with the aim to complete a real static data set to be applied in the context of fall prevention for elderly people.

We combine several assumptions and expert knowledge in order to provide a temporal data set that is faithful with the real data set. In that aim, we select a small number of variables of the real data set regarding their schema of temporal evolution. We focus on a subset of persistent variables whose value may evolve only once in the life of a person, e.g., from zero to one for the positive persistent variables. This concept emerged during discussions with experts about temporal changes of a large set of variables of interest for fall prevention.

The persistent feature of the selected variables is visible on the plot of the distributions of positive values as functions of the age. We assume that these distributions can be used as a basis for the evolution of the associated variables for a given person. We also consider a set of possible contexts for each variable, as defined by a Bayesian network learned on the static real data set. Finally, we use linear interpolation to get a simple model of the proportion of positive values for each contextualized variable.

On this basis, we propose an algorithm to simulate a temporal data set. The results are evaluated through the comparison of the temporal distributions in the temporal data set generated thanks to the algorithm and the linear functions computed from the real data set.

We are aware that the data are generated on the basis of two strong assumptions: the persistence of the concerned variables and the linear approximation of their distribution according to the age. However, the data generation algorithms and their first results consist of a first step toward the necessary filling of the data gaps in our health application context.

As a perspective, we now intend to exploit this data set in the context of fall prevention. More precisely, the objective is to predict some risk factors for falls based on a partial set of time-stamped observations. In this problem, the challenge is first to take benefit from old

observations, meaning that the value of some variable observed in the past may have changed, and second to reason with a number of observations that can be arbitrarily small.

About perspectives on data simulation, it could be interesting to compare our data simulation with the one obtained by a dynamic Bayesian network when linear functions are used to approximate the dynamic of contextualized variables since it makes changes time-invariant. Other perspectives concern the use of non-linear functions for interpolation, the inclusion of variables with other temporal schema, such as semi-persistent variables, and variables with larger domains.

Appendix B

Full Bayesian Graph

Figure B.1 represents the full BN graph learned in our study. This graph includes the mandatory (causal) arcs related to the target risk factors for falls but the rest of the graph is not completely causal.

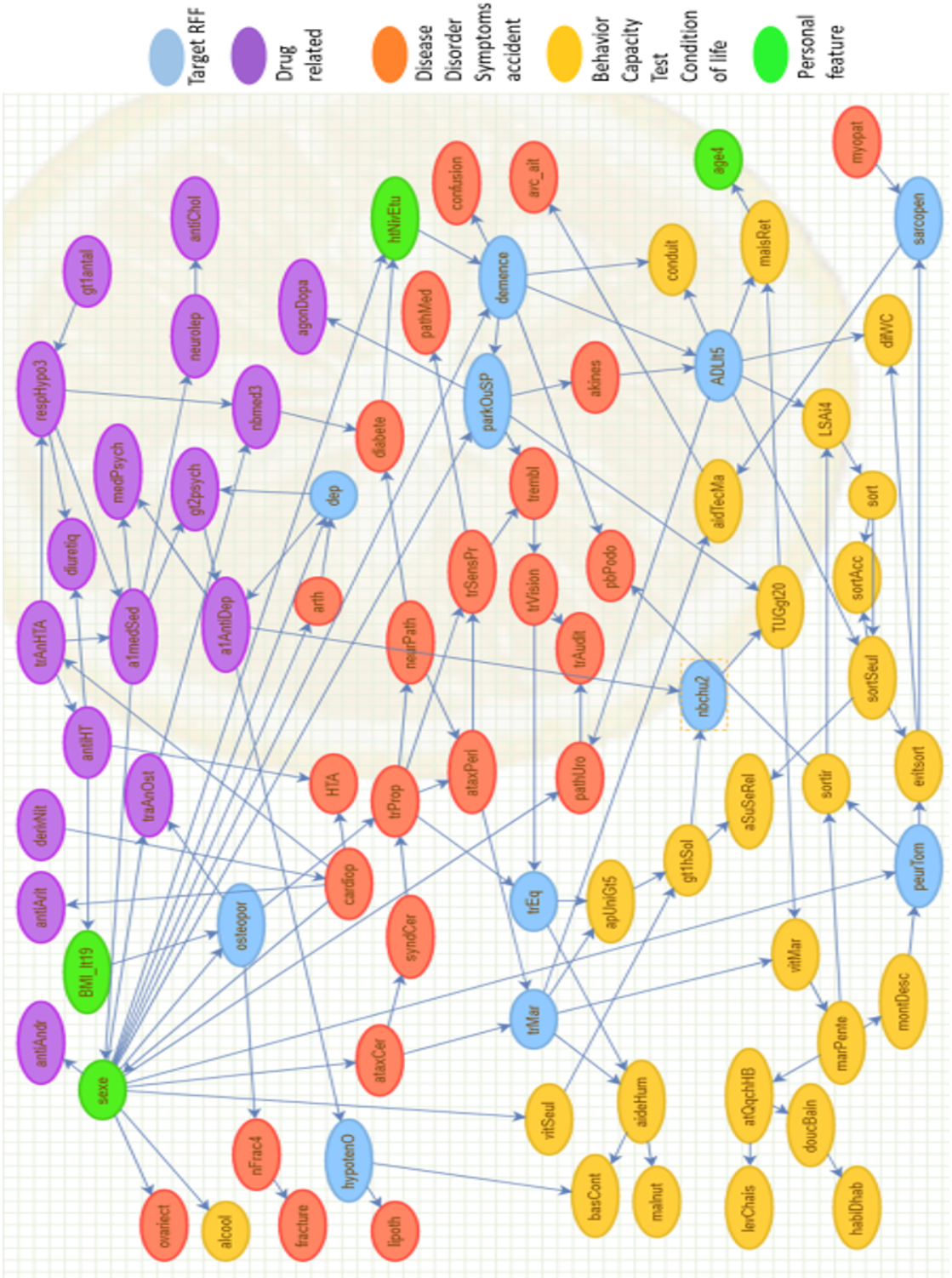


Figure B.1: Full BN graph learned

Appendix C

Publications

Published:

- Gulshan Sihag et al. "Evaluation of risk factors for fall in elderly using Bayesian networks: A case study". In: *Computer Methods and Programs in Biomedicine Update* 1 (2021), p. 100035
- Gulshan Sihag et al. "Prediction of Risk Factors for Fall using Bayesian Networks with Partial Health Information". In: *Globecom AIdSH Workshop*. IEEE. 2020, pp. 1–6
- Gulshan Sihag. et al. "Evaluation of Risk Factors for Fall in Elderly People from Imbalanced Data using the Oversampling Technique SMOTE". in: *Proceedings of the 8th International Conference on Information and Communication Technologies for Ageing Well and e-Health - ICT4AWE,*. INSTICC. SciTePress, 2022, pp. 50–58. ISBN: 978-989-758-566-1. DOI: 10.5220/0011041200003188
- Gulshan Sihag et al. "Temporal Data Simulation based on a real data set for fall prevention". In: *10èmes Journées Francophones sur les Réseaux Bayésiens et les Modèles Graphiques Probabilis*. 2021
- Véronique Delcroix et al. "Prédiction des facteurs de risque de chute chez les personnes âgées à partir d'observations partielles à l'aide d'un réseau bayésien". In: *Colloque francophone sur la chute de la personne âgée*. 2021
- Gulshan Sihag et al. "Using oversampling with a Bayesian network when data is imbalanced: study on a real case". *Accepted to 11èmes Journées Francophones sur les Réseaux Bayésiens et les Modèles Graphiques Probabilis*. 2023
- Gulshan Sihag et al. "Machine Learning Approaches to Identifying Risk Factors for Falls Among Elderly People: A Comparative Study". *Accepted to 11èmes Journées Francophones sur les Réseaux Bayésiens et les Modèles Graphiques Probabilis*. 2023

Accepted:

- Gulshan Sihag et al. "Advantages of oversampling techniques: a case study in risk factors for fall prediction". *Submitted as a book chapter in Springer*. 2022