Fuzzy Sets in Probability Trees for Interpretable AI Decision Making

Ambags, E.L.

Award date:
2022

Department of Industrial Engineering & Innovation Sciences
Operations Management & Logistics
Information Systems Research Group

Master Thesis

Fuzzy Sets in Probability Trees for Interpretable AI Decision Making

Author:
E. L. Ambags (1504606)

Supervisors:
dr. L. Genga (TU/e supervisor)
dr. M. S. Nobile (TU/e supervisor)
I. D. C. Grau Garcia (TU/e supervisor)
prof. dr. P. Liò (external supervisor)
G. Capitoli (external supervisor)

Eindhoven, July 8, 2022

Abstract

The need for fully human-understandable models is increasingly being recognised as a central theme in AI research. The acceptance of AI models to assist in decision making in sensitive domains will grow when these models are interpretable, and this trend towards interpretable models will be amplified by upcoming regulations. One of the killer applications of interpretable AI is medical practice, which can benefit from accurate decision support methodologies that inherently generate trust. In this work, we propose a combination of fuzzy set theory and probabilistic decision trees to assist clinical practice. This allows for estimating uncertainties and analysing counterfactual statements, which will assist in decreasing the frequency of misdiagnoses. Specifically, the goal is create a model mimicking the reasoning of medical professionals. The approach is applied in a proof-of-concept to two real medical scenarios: the classification of malignant thyroid nodules, and the prediction of the progression of chronic kidney disease to kidney failure.

The results show that probabilistic fuzzy decision trees can effectively support clinicians in classifying thyroid nodules and, furthermore, the integration of fuzzy reasoning brings significant nuances that are lost when using the crisp thresholds set by probabilistic decision trees. In the case of chronic kidney disease, the results are less favorable, however, also two tested benchmark models (logistic regression and decision tree) do not achieve desirable results. Most importantly, the interpretability and the usability of the model for clinicians is discussed. A tool is presented that is developed to demonstrate the usability and comprehensibility for the users. It provides clinicians with naturally understandable statements written in the form of probabilities, that have been obtained from the underlying probabilistic tree.

Keywords— Interpretable AI (IAI), Fuzzy Probability Tree, Probability Tree, Fuzzy Set Theory, Clinical Decision Support System (CDSS)

Preface

This research project concludes my Master's degree at the Eindhoven University of Technology. I am pleased to have made the switch to the TU/e after completing my BSc elsewhere. I have learned a great deal during my time here. Throughout this research project, I have had the honor to receive the guidance of a (mostly) Italian team of researchers.
Marco, thank you for continuing to guide me although you left the TU/e in the early stages of this project. Furthermore, I would like to thank you for introducing me to your research and trusting me enough to introduce me to several of your research colleagues. Pietro, thank you for your valuable guidance, enthusiasm and endless burst of ideas. Giulia, you were always open to have chat, always took the time to listen and explain, and were very valuable in this project, and for that I would like to thank you. Laura, thanks for making all of this possible. It has been an unusual setup, and you were the key to making this all possible even though you weren't part of the project from the beginning. Isel, thank you for getting onto this project at the final stages to be my third TU/e assessor, you saved the day!
Furthermore, I have my friends and family to thank for supporting me. One big shout-out to my brother, who was always there for discussions.

Emma Laure Ambags, July 2022.

– This master thesis project is dedicated to my father, Michel Ambags, whom always had faith that I would get here eventually, and passed away shortly before starting this project. –

Contents

Acronyms

ACR American College of Radiology

AI Artificial Intelligence

AIA Artificial Intelligence Act

BMI Body Mass Index

BN Bayesian Network

CKD Chronic Kidney Disease

CDSS Clinical Decision Support System

CVD Cardiovascular Disease

DT Decision Tree

ESRD End Stage Renal Disease

EU European Union

FNA Fine Needle Aspiration

FPT Fuzzy Probability Tree

GFR Glomerular Filtration Rate

GDPR General Data Protection Regulation

GUI Graphical User Interface

IAI Interpretable Artificial Intelligence

ICD International Statistical Classification of Diseases and Related Health Problems

KNN K-Nearest Neighbors

MF Membership Function

ML Machine Learning

LR Logistic Regression

OOB Out-Of-Bag

PT Probability Tree

RASI renin-angiotensin-system inhibitors

SCM Structural Causal Model

TIRADS Thyroid Imaging Reporting and Data System

XAI Explainable Artificial Intelligence

List of Figures

List of Tables

1 Introduction

A Clinical Decision Support System (CDSS) is a health information system that helps healthcare providers make decisions in order to improve patient care. These systems provide assistance in the complex decision making processes of clinicians, by offering targeted clinical knowledge, care plan recommendations and other relevant health information at the point of care. Often these systems outperform human experts, as they mimic reasoning by medical professionals, but faster and less prone to human errors. As a result patient care can be significantly improved whilst simultaneously practice variability is reduced.

In spite of the proven success of Machine Learning (ML) and Artificial Intelligence (AI) algorithms in several other disciplines, the acceptance of such models to assist in highly sensitive domains (e.g., clinical environments) is still limited due to trust and transparency issues. In the clinical field, black boxes are unacceptable [58], for the clinical field is a sensitive domain in which the healthcare provider needs to be able to justify the rationale behind any decision. The Black Box Problem is a known shortfall of AI and ML based decision support systems; it refers to a system or program allowing the user to see its input and output, but is useless in explaining how it came to its output. Therefore, it is important that the basis of predictions and recommendations that are offered by a CDSS are understandable and interpretable to its users, improving the trust and acceptance of the generated suggestions.

The need for transparency and interpretability is increasingly being recognized as a central theme to be addressed by AI research. Even more so due to upcoming regulations. Such as the General Data Protection Regulation (GDPR), as imposed by the European Union (EU) in May 2016. Among other things these regulations are supposed to shield citizens from decision making based on black boxes [19]. Moreover, in April 2021, the Commission of the EU published a draft of what is called the Artificial Intelligence Act (AIA) [20]. The proposed rules are specifically aimed at regulating the development and use of AI. These proposed regulations impose AI systems to comply with a set of mandatory requirements for trustworthy AI to be allowed to be placed on the EU market, primarily to enforce transparency and human oversight obligations to high-risk AI systems, e.g., safety-critical systems such as systems deployed in clinical environments. CDSSs specifically, as they operate in the sensitive clinical field, will be subject to the strictest set of requirements [20].

The potential of CDSSs is evident, however, the need for these systems to be interpretable is ever growing. In practice, end-users (i.e., healthcare providers) of CDSSs are less likely to trust the recommendations of systems whose workings they do not understand [15]. Moreover, CDSSs will be challenged legally by having to comply with future requirements as imposed by the AIA.

The interpretable AI method on which this project builds is probabilistic decision trees. A method with highly self-explanatory semantics, that is able to accurately represent and structure a decision problem [51]. The presentation of a probability tree (PT) is descriptive and simple to understand, and intuitively visualizes conditional probabilities that build on Bayes' theorem [48]. Furthermore, the importance of evaluating uncertainty and vagueness in medical variables is stressed. Think of ill-defined concepts such as: 'young' *vs* 'old', 'small' *vs* 'large', or a concept such as having a 'high fever' at a body temperature of 39 °C (but what about a temperature of 38.9°?). The theory of fuzzy sets, as introduced by Zadeh [70], gives the possibility to formalise a partial membership of an element to a fuzzy set. Meaning that a variable can be in multiple states at once, to differing degrees. This concept allows for incorporating fuzzy relationships, and thus extending the possibilities of PTs.

This master thesis project presents an implementation for an interpretable decision making technique, with the aim to improve the compliance, acceptance, trustworthiness, and performance of future CDSSs. The proposed solution, to achieve interpretable decision making, is a novel technique in the field of AI and ML. Namely, the integration of probability trees, one of the simplest models for representing causal generative processes, and fuzzy logic. Within the project, the proposed methods are tested by carrying out two case studies. Furthermore, a tool is created to demonstrate how the proposed methods can be used to guide (clinical) decision making in an interpretable manner. The tool that is designed is directly implementable in the decision making process of clinicians. Figure 1 visualizes the process followed in the first of the two case studies that are carried out within this master thesis project, and shows where the tool comes into play. The purple rectangle contains the phases that have been completed within this research (modeling and the prediction tool), the first (non-purple) phase has been fulfilled outside of this project. The same setup is used for the second case study, however, no tool has been created.

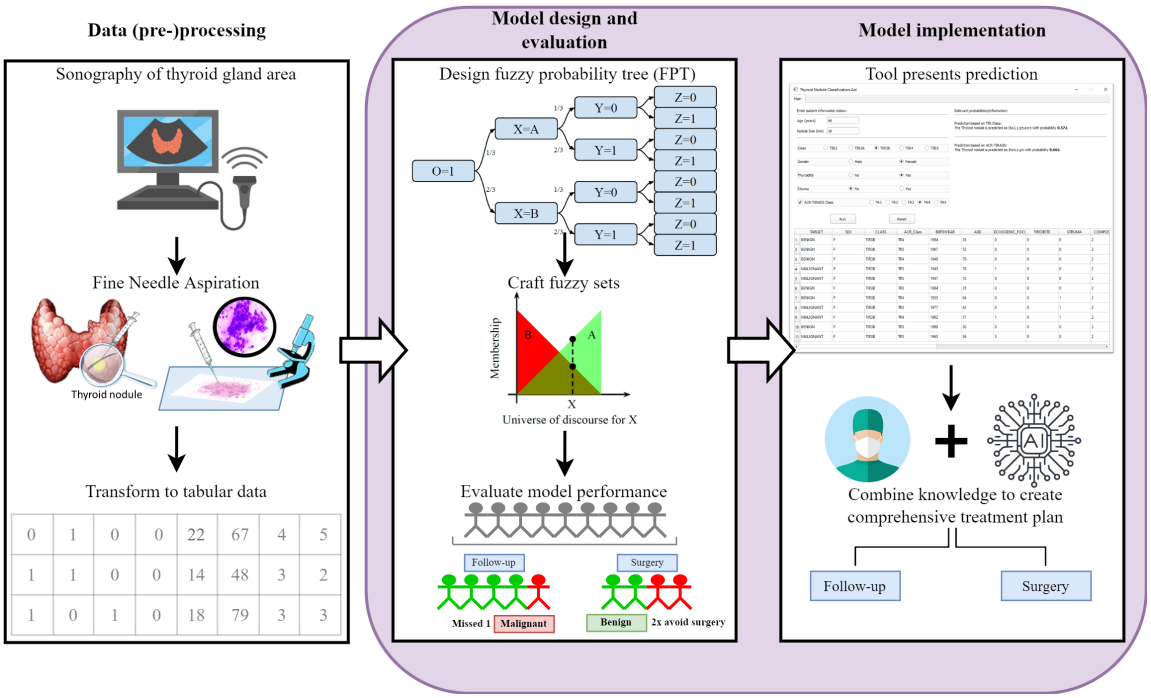


Figure 1: The summary of the approach to the decision support tool taken in this project for the first case study. The first phase has been completed outside of this project (the data is provided in tabular form). The phases in the purple rectangle have been carried out within this project. Firstly, the modeling phase is carried out (creation of the tree, crafting fuzzy sets, and assessing the performance). Secondly, a tool has been created to assist clinical decision making in an interpretable manner.

1.1 Research Problem

With the current extraordinary growth of data in the biomedical field, the need for a computational basis of medical practice is emerging. Medical practice everywhere could be leveraging on sharing information in order to assist general medical practice. Teams of experts should be able to integrate patient data, cohorts, observations and measurements in order to reduce biases from personal experience. Clearly the future of medicine is to develop computational and human/AI-in-the-loop methodologies that could leverage on machine learning analysis of large quantities of data and accurate diagnosis to create decision support systems that provide an estimate of the degree of uncertainties. According to Miller's law, the human mind is limited to consider 5 to 9 distinct pieces of information [39]; ML models can consider and compute thousands of dependencies across different types of data and databases. As ML is slowly integrated into medical decision making, we believe that interpretability is a cautious and wise ground to develop these classes of medical products on. Deep learning (i.e., AI methods that do not promote transparency) has achieved clinician-level expertise in many medical areas. Nevertheless, it is essential to establish a good amount of fairness between clinicians, patients and computational researchers, i.e., interpretability should be part of the contract.

This project aims to develop a computational framework that reasons like clinicians in order to assist clinical decision making, and also a tool that allows clinicians to interact with the automated reasoning. The proposed computational framework should support clinicians in their daily decision making processes, and can serve as a second opinion when the pressure for a decision mounts. By doing so with an explainable and interpretable method, we can introduce AI into the clinical decision making process. All the while, we must consider that the interplay of humans and AI is forced to comply with the guidelines of automated decision-making as prescribed in the GDPR [19].

1.1.1 Research Questions

The main objective of this master thesis project is to integrate discrete probability trees with fuzzy set theory. Such that the proposed methods can assist in (clinical) decision making processes in an human-comprehensible and interpretable manner, whilst considering the inherent uncertainty and vagueness that is common in medical concepts. Ideally, these methods should be easily generaliseable to other implementations in the medical field (or applications in any other sensitive domain that calls for interpretability).

The main research question this project aims to answer is:

RQ: *How can we develop a fuzzy probability tree method to support clinical decision-making in a human-comprehensible way?*

To answer the main research question, multiple sub-questions need to be answered. The considered sub-questions are:

SQ1: *What methods that exist in literature can be built upon to develop a fuzzy probability tree?*

SQ2: *What techniques need to be developed on top of the existing probability tree framework to model fuzzy probability trees?*

SQ3: *How does the proposed fuzzy probability tree perform in assisting clinical practice?*

SQ4: *How can fuzzy probability trees be effectively integrated into the workflow of clinicians?*

1.2 Research Design

This section discusses the performed steps to provide answers to the research questions. The setup of this project is split into two phases. The two phases and each of its steps are shown in Figure 2. The first phase addresses the background of the research, and which constructs are used and extended to create the proposed computational framework. The second phase revolves around two real world medical case studies, both of which are generally based on the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology [57].

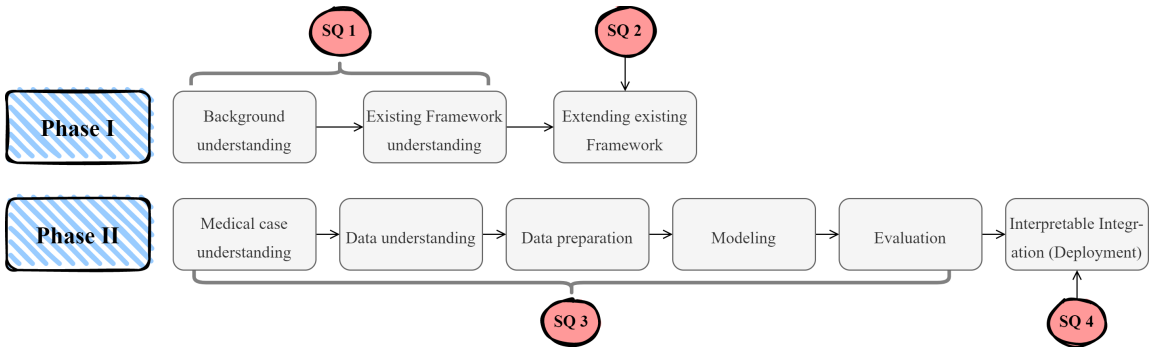


Figure 2: The two phases with corresponding steps and tasks specific to this project. The first phase focuses on extending the existing probability tree framework to a fuzzy probability tree. The second phase focuses on the implementation of the proposed methods in real world examples (leverages on parts of CRISP-DM and done for each of the two case studies), and on developing a tool that is directly integrable into to the workflow of clinicians in an interpretable manner.

Phase 1 The first phase answers the first two sub-questions. Firstly, the underlying constructs that are needed in this project are introduced and the existing framework for PT models is discussed. Secondly, the proposed extensions to this existing framework are formulated. These extensions are related to creating PT models more easily, obtaining the probabilities directly from data, making predictions based on the model, and assess the performances in doing so. Furthermore, the integration of PTs with fuzzy set theory is proposed and worked out.

Phase 2 The second phase handles the two real world case studies used to test the proposed fuzzy probability tree. Each case study is roughly set up based on the CRISP-DM framework [57]. The 'business understanding' stage is replaced by understanding the medical domain of the disease handled in each case study. Each case study will help in answering the third sub-question. For the deployment phase of the first case study a tool is developed, the tool is aimed at providing the ultimate end-users (clinicians) with an understandable and interpretable interface that they can interact with. The tool should utilize the inherent interpretability of the proposed methods, and translate the predictions into human-comprehensible statements. This represents the final stage of phase 2, and with that will answer the fourth and last sub-question.

1.3 Thesis Outline

This master thesis project consists of seven chapters. Chapter 2 discusses some fundamental constructs that are needed to have a proper understanding of the background on which this research rests. It includes general motivations about the need for interpretable AI, but also the algorithms and theories that underlie the proposed methods. Specifically, Chapter 2.2 details the existing probability tree framework that has been previously defined by Genewein et al. [28], which serves as the basis of this research. The additions that have been made throughout this research to extend this framework are presented and elaborated on with the use of examples in Chapter 3. Then, in Chapter 4, the proposed methods are tested in two real world examples with applications in the medical field. Firstly, the case of classifying cancerous from benign thyroid nodules is elaborately discussed. Secondly, the same procedure is repeated for the prediction of the risk for chronic kidney disease patients to progress to the last stage of their disease, which generally results in kidney failure. Thereafter, the results are discussed in terms of performance and interpretability, as well as some remaining opportunities of fuzzy probability trees that are yet to be explored. Chapter 5 presents the interface that has been developed for the deployment of the proposed methods in practice, and how it may be used to support the decision making process. Furthermore, it discusses some of the thoughts and feedback that clinicians addressed during a meeting in which the tool has been presented. A short discussion is presented in Chapter 6, in which an envisioned future of medicine is discussed, and how the proposed method/interface fits into this. Lastly, Chapter 7 will present the conclusion of this research by re-visiting and answering the research question, discuss the relevance for the literature on interpretable AI and the medical field, set out the limitations of this research, and finally, present suggestions for future research.

One more important element of the project are the code files created using the Python 3 programming language. These include files for conveniently creating probability trees from data, the algorithms developed for prediction making based on a fuzzy probability tree, and the framework created for the development of the graphical user interface.

2 Background

This chapter contains a background of the main constructs that are important in this master thesis project. Describing and clarifying these concepts will help with understanding the topics of this project. As aforementioned, this project addresses the need for interpretable AI decision support in sensitive domains. In light of this, a decision support method is developed that leverages probability trees and fuzzy set theory. Several important constructs will be introduced to provide important background knowledge that is the basis of the proposed methods. These constructs include: interpretable AI (including two IAI methods that are used as benchmarks in this project), probability trees, causal reasoning and fuzzy set theory.

2.1 Interpretable AI

With the rise of AI, machines are increasingly entrusted with high-impact decision making processes. This development has lead to the recent trend of research into transparent AI, aimed at increasing the user's trust in the AI system. Besides increasing trustworthiness of decisions made by an AI system, it may also provide insights into the way it makes decisions and consequently lead to new knowledge. It is important to know when a model will succeed, when it may fail, why a model makes a specific prediction and to what extent the model is reliable. These insights can be gained by deploying transparent AI systems, where transparency refers to providing users with relevant information about how the model makes its decisions [11]. There are two ways in achieving transparency in AI systems, namely we distinguish between eXplainable AI (XAI) and Interpretable AI (IAI). Although explainability and interpretability are often interchangeably used and their differences may seem subtle, in the field of AI there is an important difference between the two, as described by Basagaoglu et al. [7].

XAI methods use *post hoc* analysis of the decision making process, so that it provides insights in the way decisions have been taken by the model. Thus, XAI only peers inside the model after it has been created. Concretely, an XAI model will first involve building a black box model, after which it will help to dissect the internal mechanics of the black box model to understand the importance of various features and the decisions it can lead to. Such explainability provides global explanations, as it will tell the user something about the behavior of the entire model. Besides providing global explanations, XAI models may provide local explanations, giving insights into the prediction of a single instance. In this case, the XAI model is fed a sample, thereafter the XAI model will provide *post hoc* explanations projected on that specific sample. Therefore, a model is deemed explainable if it is capable of justifying the decisions made, and with that enhancing control and revealing potential new knowledge [7].
Two explanatory methods, SHaply Additive exPlanation (SHAP) [56] and Local Interpretable Model-agnostic Explanations (LIME) [52], have been implemented to provide transparency of the methods used in several CDSSs (SHAP in [4, 22, 63]; LIME in [21, 22]). These methods unveil the dependencies between predictor variables and the predictand, identify feature importances, reduce dimensionality of the input space, and developing explanations by training surrogate models to approximate the predictions of the underlying (black box) model. XAI methods are not fault proof, as they can be unstable or only provide a selection of the explanation. Therefore, XAI may be inappropriate in some situations where a full and precise explanation is (legally) required [40].

On the contrary, IAI creates models that are *a priori* interpretable by humans, i.e., human-interpretable from the beginning. The decision-making process of the system is directly observable. Interpretability is considered a broader term than explainability [7]. To provide a better understanding of interpretability, a highly interpretable model will be considered like the decision tree model. The path in a decision tree from the root node to a leaf can give us the rule that causes the model to make a particular prediction, and the same rules can be implemented by humans to make future predictions. Therefore, the rules of the model, that are either defined by domain experts or directly from data, are directly observable in the splits made by the decision tree. The interpretability of sequential decision rules and the cut-off values for splits make decision trees among the most popular algorithms used in clinical decision making [55]. Therefore, an interpretable system is where the user can not only see but also understand how inputs are mathematically mapped to outputs [23]. Other examples of IAI models are ordinary regression models such as linear regression (predicting the target as the weighted sum of feature inputs) and logistic regression (uses a logistic function to estimate the probabilities of an instance belonging to a certain class). Some popular IAI models include the Naive Bayes Classifier, which uses the Bayes' theorem of conditional probabilities, and K-Nearest Neighbors (KNN), which bases its predictions on neighboring data points. Furthermore, Fuzzy Inference Systems (FIS) are an interpretable and transparent type of model that reveal underlying relations

in data in terms of fuzzy rules. These fuzzy rules are clear IF-THEN rules that are based on linguistic variables and are comprehensible to human beings. Fuchs et al. [26] have developed a Python package, called pyFUME, that enables the automatic estimation of these fuzzy models from data. Lastly, Friedman and Popescu developed an interpretable model namely the RuleFit algorithm as presented in [25]. The RuleFit algorithm learns sparse linear models that automatically detect interaction effects between the features in the form of decision rules. Similarly to pyFUME, it constructs clear IF-THEN rules, but it does this by decomposing decision trees, as any path to a node in a tree can be converted to a decision rule.

The need for IAI is fueled by upcoming regulations imposed by the EU (GDPR [19] and AIA [20]). In some cases XAI is simply not (transparent) enough, and models that are intrinsically human-interpretable will be needed.

Two interpretable models that will be discussed in greater detail are used as benchmark models during this project. Benchmark models are used as reference points to compare the performance of the proposed methods to. The two benchmark models that are used and, thus, will be elucidated in this section are (1) Logistic Regression (LR), and (2) regular Decision Trees (DT).

Logistic Regression The main concepts in this paragraph are based on the work in [31].
The (binomial) Logistic regression (LR) is a statistical model which is well suited to model the relationship between a dichotomous outcome variable and one or more categorical or continuous predicting variables. A dichotomous variable is one that takes on one of two possible values, i.e., it only contains data encoded as 0 or 1 (which in this case refers to: benign/malignant). The LR makes predictions in the form of the probability of an event occurring, this probability can take values in $[0, 1]$. In order to do so, the LR model uses a logistic (sigmoid) function to map the output of a linear equation (weighted sum of input features plus a bias term) to $[0, 1]$. Therefore, the probability of any event is simply the logarithm of the odds. The LR is a good baseline model as it is simple to implement and understand. This causes it to be inherently interpretable, and therefore, a good comparison to our proposed interpretable model.
In [60] the authors have found that their logistic model outperformed five other (statistical) models – these are support vector machine, random tree, random forest, boosting, and artificial neural network models – in building discriminant functions to differentiate between malignant and benign thyroid nodules. The authors report that the results provided by their LR were stable, based on evaluation by 10-fold cross-validation.

Decision Tree The main concepts in this paragraph are based on the work in [1].
Regular decision trees (DTs) are a common decision support tool that uses a tree to model decisions and their possible consequences. The paths from root to leaf represent the classification rules computed by the DT. The tree is upside down, with its root at the top, and the end of each branch that does not split anymore is the leaf (the classification decision). A DT is the underlying model in the trees proposed in this research. Similarly, the tree is formed based on a set of hierarchical decision on features to divide the data into smaller subsets. When a subset only contains instances of a single class, then the subset is a pure partition, and no further splits can be done.
So these splits are made such that the purity of the leaves is maximized. This can be done using several measures, such as Gini index and entropy (information gain). Every time the tree makes a split, the one that yields the purest leafs will be chosen. A leaf is considered impure when the data points inside it belong evenly (50/50) to two different classes, therefore, a leaf is pure when it contains only data points belonging to one class. This process of splitting is repeated until the desired depth of the tree is reached, the leaves are pure, or splitting will no longer add value to the predictions. It may be that some subsets remain impure, this is acceptable, as otherwise it would likely lead to overfitting. DTs are considered interpretable because they can be linearized into very human-comprehensible decision rules [49]. The outcome of each rules is the content of the leaf node, and the conditions along the path in the tree (i.e., the total realization) form the if clause in this rule. The rules generally have the following form: "`if` condition x `and` condition y `and` condition z, ..., `then` outcome". Furthermore, DTs implicitly perform feature selection ensuring that relations can be observed easily. Similar to the trees in PT and FPT, DTs have low bias but high variance. This is directly dependent on the depth of the tree, and when the tree is too deep it may cause overfitting. Soni et al. [61] have tested several predictive methods in light of predicting heart disease in patients. Among other models, the authors chose to implement DTs because of their simplicity in both implementation and understanding, and its ability to deal with high dimensional data. They found that DTs often outperformed bayesian networks, KNN, neural networks, and clustering methods. Proving the DT to be a powerful yet comprehensible model.

2.2 Probability Trees

A Probability Tree (PT) is a graphic model describing probabilities that comprises of nodes and arcs to represent all possible outcomes of an event. Each node represents a potential state of the process, and the arcs indicate the probabilistic transitions between the nodes. The end node of every branch is denoted as a leaf. The graphical component is descriptive and simple to understand, making PTs an inherently interpretable model. A generic PT is presented in Figure 3. By convention, the root node is bound to $\mathcal{O} = 1$. Every path in the tree is a unique combination of decision variables and the values that these variables can take. Every transition probability (θ) in the PT is the result of a conditional probability: $P(Y = 0 \mid X = 0) = \theta_{0,0}$. The probabilities on the arcs are conditional probabilities to go from one node to the next, these probabilities build on the Bayes' theorem [48]. PTs can be used as causal models, where the arcs do not only represent the probabilistic transitions, but also causal dependencies between the nodes [28]. Similarly to Bayesian Networks (BNs), they can be used to model causal relationships and perform inference. However, differently from BNs, and thanks to the fact that PTs are not represented as a directed acyclic graphs, they allow to model multiple alternative scenarios where variables do not necessarily follow a partial order.

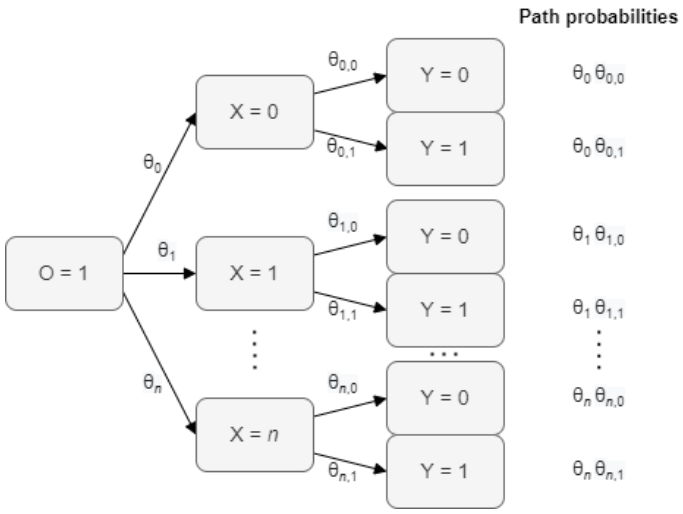


Figure 3: A generic probability tree for a combinatorial decision problem capturing a sequence of two variables. The thetas (θ_i) represent the transition probabilities for each value i of the decision variable.

Formal definition A probability tree is formally defined as the following by Genewein et al. [28].
We define a *node* $n \in \mathcal{N}$ in the three as a tuple $n = (u, \mathcal{S}, \mathcal{C})$ where: $u \in \mathbb{N}$ is a unique numerical identifier for the node within the tree; S is a list of statements such as '$X = 0$' and '$W =$ rainy'; and C is a (possibly-empty) ordered set of transitions $(p, m) \in [0, 1] \times \mathcal{N}$ where p is the transition probability to the child node m. We will represent statements such as '$X = 1$' as a tuple $(X, 1) \in \mathcal{X} \times \mathcal{V}_X$ where $\mathcal{X}$ is the set of variables and $\mathcal{V}_X$ is the range of the variable X. Obviously, the transition probabilities must sum up to one. The root is the unique node with no parents, and a leaf is a node with an empty set of transitions. A *(total) realization* in the probability tree is a path from the root to a leaf, and its probability is obtained by multiplying the transition probabilities along the path (Figure 3 under the header 'Path probabilities'); and a *partial realization* is any connected sub-path within a total realization. When entering a node, the process binds the listed random variables to definite values.
An *event* is a collection of total realizations that we can filter using propositions about a random variable. Therefore, an event is used to describe a set of all realizations that contain a node with the specific statement, for instance the event '$X = 0$'. Furthermore, we can use logical connectives of negation (`Not`, $\neg$), conjunction (`And`, $\wedge$) and disjunction (`Or`, $\vee$) [28]. With this we can state composite events, such as '$\neg(X = 0 \wedge Y = 1)$'. We can also describe events that meet a causal condition using precedence (`Prec`, $\prec$).

Concretely, a probability tree has weighted directed edges such that for every internal vertex, the sum of the weights of the outgoing edges equals 1 (the weights refer to the transition probabilities). Leaf vertices

do not have any outgoing edges. Each level of a tree is a sample space, where the total probability of all possible next steps has a total probability of 1. Therefore, each node and its respective children form a probability space [12].

Constructing a PT can be done using induction (from data to tree; e.g., through data mining; calculating the information gain) as proposed in [10] and seen in [35, 72], or using deduction (based on domain knowledge) as seen in [6, 13, 34, 41]. Drawing a PT is a way to visualize all of the possible total realizations (i.e., all possible combinations of variables) present in the data. PTs have been used in practice and are an active field of research, however, PTs have not yet received the same attention as BNs have, in spite of their simplicity and expressiveness. For instance, an agency of the US government have created a PT for the development of early warning programs to assess the risk of the eruption of volcanoes [6]. The proposed PT is a combination of possible states in which a volcano may be, and the corresponding risk of eruption. PTs may also be used to predict the movement of people, Leng et al. have used a PT to predict the destination station of people using the Beijing subway system [35]. The PT that has been developed is based on the results from data mining. Furthermore, Zhu et al. have used PTs to predict the movement of communities of people [72], also this PT has been based on patterns that were mined from data.

This project will build on the PT framework as designed by Genewein et al. [28]. Therefore, we need to establish some definitions and illustrate the way in which the PT framework, proposed in [28], works. Figure 4 will be used as an example tree throughout this section and it will be used as reference on several instances.

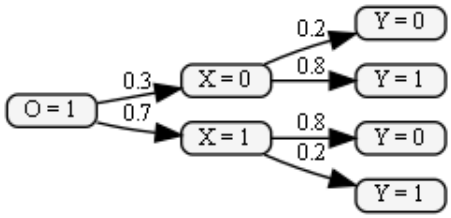
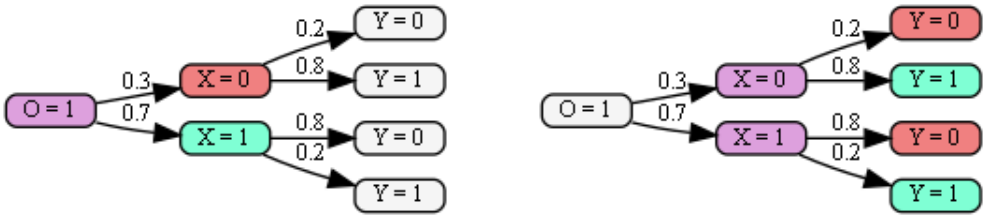


Figure 4: Probability tree. Leading example throughout this chapter.

Previously, we have defined a *total realization* as a path in a tree from the root to a leaf node. Therefore, the probability for any given total realization is obtained by multiplying the transition probabilities along the path. Furthermore, we have defined an *event* as a collection of total realizations. For instance, the event '$Y = 0$' is the set of all total realizations that traverse a node with the statement '$Y = 0$'. In Figure 4 this includes two total realizations. Furthermore, we can use common logical connectives such as negation (`Not`, $\neg$), conjunction (`And`, $\wedge$), and disjunction (`Or`, $\vee$) to state events such as '$\neg(X = 0 \wedge Y = 0)$'. We assume that every event is *well-formed*, that is, every total realization must contain only one node within its path where an event is true.

Events can be represented using *cuts* – a collection of nodes that are mutually exclusive and with probabilities summing up to 1. In particular, we are interested in *min-cuts* (minimal cuts). A min-cut of an event collects the smallest number of nodes in the probability tree that resolves whether an event has occurred or not [28]. This allows us to distinguish between the nodes that render the event true from the nodes that render the event false. So events can be described using a cut $\delta = (\mathcal{T}, \mathcal{F})$, where the true set $\mathcal{T}$ and the false set $\mathcal{F}$ contain all the nodes where the event becomes true or false respectively. Figure 5 shows examples of min-cuts for the simple events '$X = 1$' (Fig. 5a) and '$Y = 1$' (Fig. 5b). For any event we can identify the nodes where the very next transition will determine whether a given event will occur or not, these are the *critical nodes*. A *critical set*, therefore, refers to all critical nodes. These are highlighted in purple in Figure 5.



(a) The min-cut for the event '$X = 1$'.

(b) The min-cut for the event '$Y = 1$'.

Figure 5: Min-cuts (red and green nodes) and critical sets (purple nodes). The red nodes correspond to the false set $\mathcal{F}$, and the green nodes to the true set $\mathcal{T}$.

Ideally, to deduce meaningful (conditional) probabilistic relationships from the PT, there must exist some causality between consecutive nodes. However, also consecutive nodes may follow a temporal order based on predictive causality. Meaning that there is a relationship of precedence between the nodes. This implies that in a tree where node X directly precedes node Y with an edge connecting the two, node X *forecasts* node Y. This relationship may exist rather than a classical causal relationship where node X *causes* node Y. This principal of predictive causality is based on Granger causality as introduced in [29]. Hence, we can formalize a longitudinal diagnosis process as a PT.

2.3 Causal Reasoning

Causal reasoning is the process of identifying causal relationships, and is an important universal human capacity. A causal relationship describes a relationship between two variables such that one caused the other to occur. This relationship is stronger than correlation [5], which just describes co-movement patterns between two variables. Once a causal model is available, either by external domain knowledge or a learning process, causal reasoning allows to draw conclusions on the effects of interventions, counterfactuals and potential outcomes [54]. These refer to the three fundamental operations of the causal hierarchy [46]: 1. Association 2. Intervention, and 3. Counterfactuals. Table 1 presents an overview of the causal hierarchy, providing typical questions associated to every level of causation. The questions associated to each level can only be answered if information from the lower levels is available.

Table 1: The causal hierarchy and typical associated questions.

Level	**Form**	**Typical Question**
1. Association	$P(\mathcal{A}\|\mathcal{B})$	What is the probability of event $\mathcal{A}$ given that event $\mathcal{B}$ is true?
2. Intervention	$P(\mathcal{A}\|do(\mathcal{B}))$	What is the probability of event $\mathcal{A}$ given that event $\mathcal{B}$ was *made* true?
3. Counterfactuals	$P(\mathcal{A}_\mathcal{C}\|\mathcal{B})$	Given that $\mathcal{B}$ is true, what would the probability of $\mathcal{A}$ be if $\mathcal{C}$ *were* true?

At the first level there is association, which is the most basic level of causation. It invokes purely statistical relationships, meaning no causal information is needed yet. It is important to distinguish between marginal and conditional associations, as conditional associations are a key building block for causal inference. Marginal association is simply the co-movement (i.e., correlation) between two variables $\mathcal{A}$ and $\mathcal{B}$ [46].

The second level of causation are interventions, as the word suggests, this involves intervening and changing what is observed. An intervention forces a subset of variables to attain fixed values, causing the definition of a new distribution over the remaining variables [46].

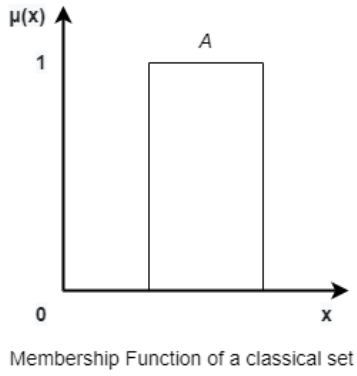
The top level of the causal hierarchy are counterfactuals. The most distinctive characteristic of functional models is the analysis of counterfactuals [46]. If we want to achieve human level reasoning in machines, they need to be equipped with counterfactual thinking [47]. Counterfactual problems reason about why things happened, imagining the consequences of different actions in hindsight. A counterfactual statement is a statement about a *subjunctive*, which is a possible or imagined event that could have happened had the stochastic process taken a different course [28]. Using this reasoning, alternate realities can be tested and the actions that lead to the desired outcome can be determined. A typical question associated with counterfactuals, as presented in Table 1, uses three variables $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$. $\mathcal{A}_\mathcal{C}$ denotes the subjunctive event $\mathcal{A}$ under the counterfactual assumption that event $\mathcal{C}$ has occurred, therefore it is a potential response.

Where event $\mathcal{B}$ is the *indicative*, which is the factual assumption [28]. Meaning that $\mathcal{B}$ refers to the things that have actually happened.
Genewein et al. [28] have presented concrete algorithms for causal reasoning (association, intervention, and counterfactuals) in PTs.

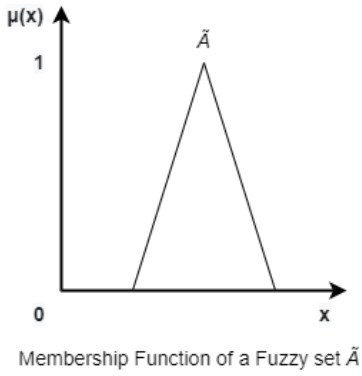
Causal inference techniques can go beyond traditional machine learning techniques that rely on statistical relations alone. Causality will allow technology to reason and make choices like humans do.

2.4 Fuzzy Set Theory

The word 'fuzzy' refers to vagueness and ambiguity. Fuzziness occurs when the boundaries of a piece of information are not crisp/clear-cut. To include uncertainty that stems from the values in a variable itself (rather than statistical uncertainty), Zadeh introduced an extension to classical set notation in 1965 [70]. Classical set theory allows elements to either belong fully to a set or not at all, thus, the membership of elements in a set are in binary terms. An encoding approach that falls short in human-centered systems. Fuzzy set theory allows for membership functions that can be valued anywhere in the interval $[0, 1]$, where 0 = do not belong, and 1 = fully belong. The use of fuzzy sets provides a representation mechanism that improves the ability and flexibility for dealing with data that is associated to complex concepts. The graphical representations of the two membership functions are shown in Figure 6, where Figure 6a depicts a membership function in traditional set theory, and Figure 6b in fuzzy set theory.



(a) Membership function for a classical set A. The degree of membership is either 0 or 1.



(b) Membership function for a fuzzy set $\widetilde{A}$. The degree of membership is a real number in the interval $[0, 1]$.

Figure 6: Graphical representation of a traditional crisp set and a fuzzy set.

Thus, a fuzzy set has gradual boundaries, in contrast to classical sets, which contain discrete borders. The universe of discourse (U) is the set of possible values that a variable u can take on. We can mathematically define a fuzzy set $\widetilde{A}$ as:

$$\widetilde{A} = \{(u, \mu_{\widetilde{A}}(u)) \mid u \in U\} \tag{1}$$

Here, $\mu_{\widetilde{A}}(u)$ is the degree of membership of u in the fuzzy set $\widetilde{A}$, assuming $\mu_{\widetilde{A}}(u) \in [0, 1]$. Fuzzy membership functions can have several shapes, some typical shapes for the membership functions of fuzzy sets include: linear, triangular, trapezoidal, Bell or Gaussian, and S-shaped. Figure 7 presents illustrations for these membership functions.
An important element in fuzzy set theory is the concept of linguistic variables. A linguistic variable is a variable whose values are words or sentences in a natural language [71], which can be represented as fuzzy sets. These linguistic variables serve as an approximate characterisation of phenomena, which are too complex or too ill-defined to be represented in precise terms. Human reasoning often involves linguistic variables of ill-defined concepts. For example, 'age' can be represented as a linguistic variable when it is defined in terms that are linguistic rather than numerical, e.g., young *vs* old. These terms are vague and ambiguous, and therefore, we can use fuzzy sets to quantify these imprecise terms.

Integration of Probability Trees and Fuzzy Sets We propose the integration of PTs and Fuzzy set theory, therefore, calling the method a Fuzzy Probability Tree (FPT). This allows for using the existing

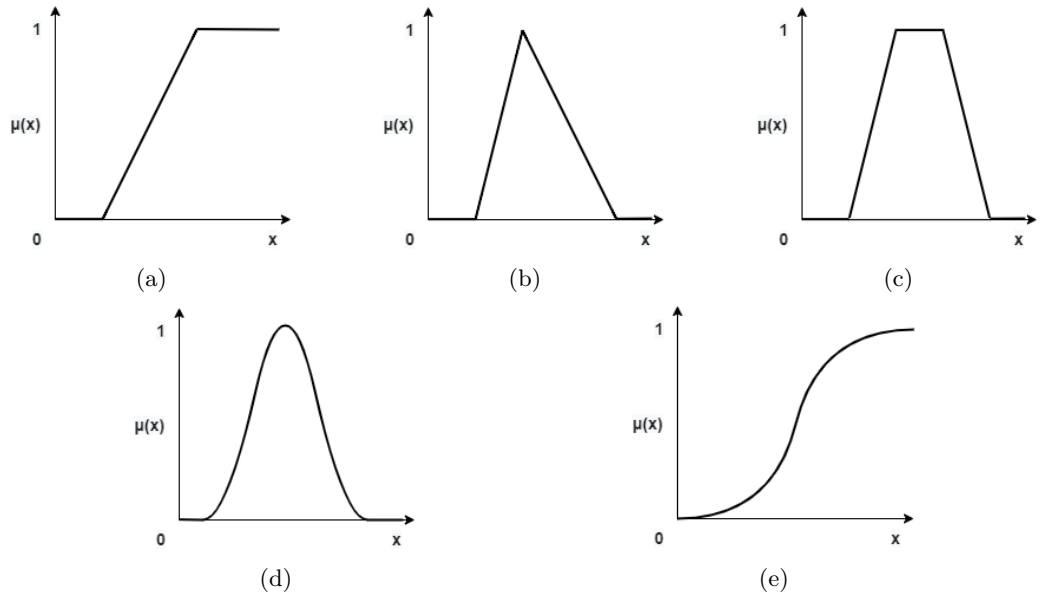


Figure 7: Typical shapes for fuzzy set membership functions: (a) linear, (b) triangular, (c) trapezoidal, (d) Bell or Gaussian, and (e) S-shape. The y-axis shows the degrees of membership, the x-axis represents the universe of discourse for variable x (i.e., the values that the variable may take on).

method of PTs, while incorporating uncertainty about the data, or allowing for a flexible description of vague variables. Thus, enabling us to incorporate human expert knowledge in the form of fuzzy membership functions to probabilistic trees.

While PTs require well defined discrete concepts and events, this is seldom the case in real world scenarios. Especially in the medical field, where the variables are often fuzzy, vague or ambiguous. By using the FPT approach – and by means of carefully crafted fuzzy sets – we can incorporate the inherent uncertainty in variables to balance the probabilities (e.g., when we define a 'high fever' as temperature $\geq 39°$; what about a temperature of 38.9°?). The integration of PTs and fuzzy sets, leading to an FPT, will provide an AI method that is aligned with the way humans reason.

3 Additions to the Existing Framework for Probability Trees

This project builds on the framework provided by Genewein et al. [28]. In which the authors have presented concrete algorithms for causal reasoning in PTs. Having established the potential of integrating PTs and fuzzy set theory, we will discuss the implementation of fuzzy reasoning and some other additions onto this PT framework. The PT framework as provided by Genewein et al. [28] is created in the Python 3 programming language. The additions presented in this chapter are also developed using the Python 3 programming language. Throughout this chapter the example probability tree of Section 2.2 (Figure 4)) will be used for the purpose of explaining.

Several additions have been made to the existing PT framework as presented in [28] and discussed above. These include:

- Calculate transition probabilities directly from data;
- Simplification of the process to create a PT and automation to create the tree directly from data (given that the data is provided as a Pandas DataFrame [38]);
- Implement functionalities to make predictions based on the PT;
- Integration of fuzzy variables into the functionality to make predictions;
- Make predictions for an entire data set (Pandas DataFrame) and determine performance of the PT or FPT based on several performance metrics.

The additions will be discussed briefly below.

Transition Probabilities Directly from the Data The nodes in a tree are connected by arcs, and these arcs have transition probabilities that represent the probability of occurrence of an event when in a certain state of the process. These transition probabilities can be deduced directly from the data, and they are basically a frequency count of the occurrence of each state present in the data set. The arcs leaving any node always sum to 1. Thus, the data set can be conditioned (i.e., filtered) to represent only the data points that correspond to the state of the process that is factual in that node. Then for all selected data points, we determine the transition probability to the next node by counting the occurrences of each possible event (i.e., the value the variable in the next node can take on).

Creating the PT Previously, in order to create a PT every separate node had to be defined with a probability and a list of its children (whom also all have lists of child nodes). This is tedious work, and takes a lot of time as the tree increases in size, some examples can be found in the interactive tutorial as provided by [28]. The process of creating PTs has been automated, such that any size of tree can be created using just two lines of code. The user will only need to specify a list of strings containing the names of the variables to be used in the tree, in the order in which the variables should occur in the tree. This implementation relies on Pandas [38], and thus, the data needs to be represented as a Pandas DataFrame.

Predicting new data points using the PT The PTs all start with a root node ($\mathcal{O} = 1$) and end with one or two leaf nodes. The leaf nodes contain the possible target values (i.e., positive (1)/negative (0); malignant/benign). The PT will calculate the probability that the target variable is positive and that it is negative ($1 - P$(positive); these sum to 1). By default the threshold for predicting positive is at 0.50, and therefore, the majority class in the leaf nodes will be the final prediction. Every prediction will come with a probability, which indicates the degree of certainty of the prediction. For instance, when there are ten data points in the leaf nodes of which 6 are positive and 4 are negative: the prediction will be positive with a probability of 0.60. The prediction is positive because the probability for a positive target variable is higher than the threshold ($0.60 > 0.50$). The probability will always be compared to the threshold, meaning that the threshold can be altered to suit different situations.
As the PT is created based on the data points present in the data set, a situation may arise where not every possible combination of variables is represented in the PT (because this particular combination does not occur in the data). In this case, the PT does not contain a full realization (path from root to leaf) that corresponds to the characteristics of the – to be predicted – data point. When this occurs, a small algorithm has been composed that backtracks through the characteristics of a patient to find the largest set of characteristics that has an actual node in the tree. Then the weighted average of all data points that follow from this node is determined to create the prediction. This means that the newly introduced, and unrepresented, data point in the tree is predicted based on the weighted average of all *similar* data points in the data set. This entails that when there is a missing value for a variable that is in one of the first

level of nodes in the tree, the variables that follow will not be considered. In the example PT as shown in Figure 4 this means that in the case that a value for the variable 'X' is missing, but '$Y = 0$', then the value for 'Y' will not be considered. When the algorithm backtracks through the tree to find the largest set of existing conditions it will return an empty list, as variable 'X' is the first node in the tree. Due to this simplification the model performs best for patient profiles that are complete. In the case studies presented in this research project, all patients have complete profiles and thus this situation will not be encountered and cause errors in the results. This has been implemented to cover the base case when a user does not provide a full patient profile to the tool that has been designed and will be presented later in Chapter 5.

Predicting new data points using the FPT When fuzziness is incorporated into the predictions, the tree that is used still relies on the concepts of PTs. Meaning that the tree does not change when a prediction is made using the FPT method with respect to the tree that is used when a prediction is made using the PT method. When a variable is fuzzy, it means that it can be in multiple (fuzzy) sets at once, and therefore, it can be in multiple nodes – of the same depth – in the tree at once. Consequently, a prediction is made based on multiple total realizations (paths from root to leaf) in the tree, where each total realization is considered according to the degree of membership of the variable to the fuzzy set. An elaboration of the implementation and the algorithm developed to make predictions based on the FPT is provided in Section 3.1.

Performance Evaluation of PT and FPT In order to evaluate the performance of any given tree, we need to be able to make predictions for (part of) a data set. Again, the implementation builds on Pandas, and thus the data should be of a Pandas DataFrame type. A basic implementation has been made such that a user can build a PT based on part of the data set, and make predictions for the residual of the data set. Allowing for the quantification of the tree's performance.

3.1 The Fuzzy Probabilistic Prediction Algorithm

In this section, we will discuss the algorithm underlying the decision making in FPTs. For which the pseudo-code is presented in Algorithm 1. The method `predict` expects as input a PT, a root node, and a list of statements that contains all characteristics of a patient, as well as a copy of this list of statements. Each characteristic in the list of statements is defined as a '`Condition`'. This refers to a class object that has been developed for the purpose of implementing FPTs. Each `Condition` has two fields, (1) a string indicating the variable it refers to, and (2) a dictionary containing the possible values that that variable may have and the degree of membership that it has for this value. For example, when we have a person whom is considered to be 30% in the (fuzzy) class *young* and 70% in the class *old*, the condition be represented as: `var:Age, mf:{young:0.3, old:0.7}`. The conditions in the list of statements follow the same order as the tree. In the case that the variables do not have fuzziness, the condition will look like the following: `var:Age, mf:{young:0, old:1}`. We will see how this is useful.

Algorithm 1: predict (Fuzzy Probabilistic Prediction)

```
Input  : tree: PT, node: tree node, statements: list, c_statements: list
Output: probability for malignancy (for one patient)
if node is leaf then
    return ratio of labels with value 1
if statements = ∅ then
    return conditionalProbability(tree, findExistingConditions(tree, statements))
nextCondition = statements[0]
probability = 0
if nextCondition ∈ node.children then
    for value, mf ∈ nextCondition do
        if child exists with value then
            // predict probability for this child
            probability += mf * predict(tree, child, statements[1 :], c_statements)
        else
            // compute weighted probability over all children
            probability += mf * conditionalProbability(tree, findExistingConditions(tree,
              c_statements)
else
    mf = 1/len(node.children)
    for child ∈ node.children do
        probability += mf * predict(tree, child, statements, c_statements)
return probability
```

The algorithm descends down the tree and every time that it reaches a leaf node, it will return the ratio of labels with value 1 to the labels with value 0 (positive labels/negative labels). Since `predict` is a recursive function, this ratio will be determined for every leaf node that is reached in the total realization (that is, the path from root to leaf). In the case of a variable that has a degree of membership in multiple fuzzy sets, the process of determining the ratio of positive/negative labels will be carried out for all total realizations that result from these fuzzy sets.

For every condition we check that the node – in which we currently find ourselves – has (a) child node(s) for the condition variable. If this is the case, then we check whether any of the child nodes has the associated value. If so, then we increment the probability by the predicted probability that the data point will have a positive label × the degree of membership ('mf' in Algorithm 1). Otherwise the child is not found, in which case we will increment the probability by the predicted probability × mf, for every child node that follows from the node. As the node does not exist we consider all other children equally, and we can find the last existing node in the tree by calling the function `findExistingConditions()` (the workings of this function will be discussed shortly below). If none of the child nodes contain the variable associated to the next condition, then we know that there is a gap in the list of conditions (in Figure 3 this is the case when you try to go directly from '$\mathcal{O} = 1$' to '$Y = 0$', i.e., information on the variable 'X' is missing). In this case we will end up the last `else`-statement in Algorithm 1, where every total realization leaving the current node is considered equally (due to $mf = \frac{1}{len(\text{node.children})}$). This means that we will not consider any more conditions from the list of statements, after the point where we have found a gap (in Figure 3 this would mean that if we do not specify any value for 'X', then it will not consider the value for 'Y' as it is located behind the gap in the conditions). We adopt this as a simplification, and therefore, our algorithm expects to be given complete data points in order to predict them, for which it will function best.

In Algorithm 1 we see the line `probability += mf · conditionalProbability(...)` multiple times. This ensures that the final probability that is calculated consists of the probabilistic prediction for every total realization multiplied by the membership function (mf). So whenever a variable is crisp (i.e., not fuzzy) then the mf will be either 0 or 1. Therefore, the probability will not be incremented when the mf equals 0 (meaning that it is not in that set). Thus, we only consider the total realizations that are relevant. We also have the case where the list of statements is empty after a few recursive calls. This implies that the list of statements is shorter than the depth of the tree, and thus we have not yet reached a leaf node. In which case, we find the last existing node in the tree – for our set of conditions – by using the function `findExistingConditions()`. This function backtracks through the list of statements to find the largest set of conditions for which a data point exists. This is where the copied list of statements comes in. Because

the main function (`predict`) calls itself recursively and the list of statements is only passed through to the next call from the element at index 1 on-wards (`statements[1:]`). In order to backtrack through the list a complete version of the list of statements is needed. Furthermore, the function `conditionalProbability()` is used inside the algorithm. This function returns the probability for a given conditional statement (e.g., $P(Y = 1 \mid X = 0) = \theta_{0,1}$ (Fig. 3)). This conditional probability is determined directly from the tree, based on methods developed by Genewein et al. [28]. This is done in the following way: we find the min-cut in the tree for the series of 'events' (nodes) we are interested in. Recall that a min-cut is a minimal representation of an event in terms of the nodes of a probability tree (i.e., the smallest number of nodes at which it is resolved whether an event has occurred or not, e.g., the event $X = 0$). We find the min-cut in the tree for the series of 'events' we are interested in. A min-cut is a minimal representation of an event in terms of the nodes of a probability tree (i.e., the nodes that represent the conditional state, which in this case is 'Gender = Female'). Meaning that a min-cut of an event collects the smallest number of nodes in the probability tree that resolves whether an event has occurred or not [28]. We can then create another min-cut that corresponds to the event of $Y = 1$, and determine the probability of the occurrence of this min-cut given that $X = 0$.

The prediction method as presented in Algorithm 1 works for fuzzy variables with any number of classes. However, in the rest of this thesis project we will only work with fuzzy variables that can only belong to two sets. Thus, although the variable age may be divided into three (or more) linguistic variables, for example: young *vs* middle-aged *vs* old. The codes created in light of this thesis project currently only support variables that are divided into two linguistic variables, such as: young *vs* old. Although the prediction algorithm as presented in Algorithm 1 already supports fuzzy variables that belong to more than two sets, the way in which the '`Conditions`' are created will need to be expanded slightly to support this option as well. During this thesis project there were no fuzzy variables that belonged to more than two sets, and therefore, this has not yet been worked out. Although, it can be done relatively easily.

3.2 A Demonstration: Predicting Patients using PT and FPT

We consider the small dummy data set presented in Table 2, for which the corresponding PT is visualized in Figure 8. The dummy data set contains the information on seven COVID-19 patients, for whom the gender, age, status of vaccination, and whether Long Covid (LC) has been diagnosed. The transition probabilities on the arcs represent the frequency count of specific patients in the data set (e.g., total of 7 people where 4 are vaccinated; $P(\text{Vaccinated} = \text{F}) = 0.571$). Note that in the tree the feature 'Age' is transformed to a crisp variable indicating whether a person is 50 years or older, this is necessary in order to make splits. We can obtain useful probabilities intuitively from this tree. For instance, to determine the probability of developing LC symptoms when contracting COVID-19 while having received vaccination ($P(\text{LC} = \text{Y} \mid \text{Vaccinated} = \text{Y}) = 0.25$) and compare this to the probability of developing LC when a patient is not vaccinated (i.e., $P(\text{LC} = \text{Y} \mid \text{Vaccinated} = \text{N}) = 0.667$). The order of the features presented in the PT (Figure 8) is based on the relevance of these features. Placing a feature at the beginning of the tree that is better able to effectively split the data set will improve the predictive performance of the tree.

ID	Gender	Age	Vaccinated	Long Covid
1	F	46 (50−)	Y	N
2	M	67 (50+)	Y	N
3	M	51 (50+)	N	Y
4	F	34 (50−)	N	Y
5	F	71 (50+)	Y	N
6	F	53 (50+)	N	N
7	M	56 (50+)	Y	Y

Table 2: Dummy data set for prediction of long Covid symptoms.

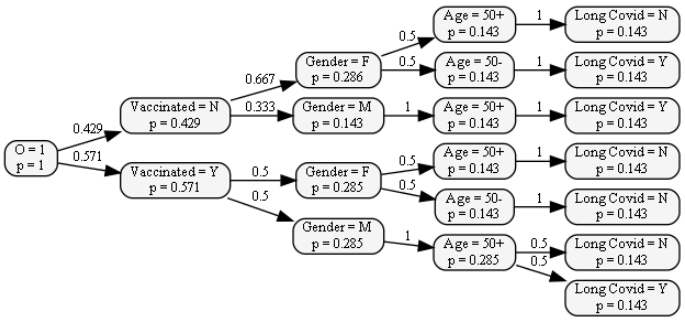


Figure 8: PT for the dummy data set (LC=Long Covid). The probabilities depicted in the nodes correspond to the overall probability that the process ends up in the state represented by the node. This PT contains one fuzzy variable (Age). We can make predictions based on this tree using the (crisp) PT methods (next paragraph) or based on the FPT methods (last paragraph of this chapter).

Classifying patients (crisp variables) and using counterfactual statements Furthermore, we can use the tree to classify new patients in order to assess their risk of having LC symptoms after contracting the COVID-19 virus. We introduce a new patient with the following characteristics: [Male; Age = 50+; Vaccinated = N]. The probability for this patient to develop LC symptoms is $P(\text{LC} \mid [\text{Not Vaccinated}, \text{Male}, 50+]) = 1$. Counterfactual statements allow clinicians to investigate probabilities associated with an 'alternate reality'. We distinguish between the indicative which is the factual situation (things that have actually happened) and the subjunctive (things that could have happened in an alternate reality). Counterfactuals are particularly interesting when there are actual actions that can be taken, such as to receive a vaccination. For demonstrating the implementation of counterfactuals we will consider the patient to indeed have developed LC symptoms. Now, using counterfactuals, we can investigate the probability of this patient developing LC symptoms *had* he been vaccinated. So the indicative premise is that the patient has not been vaccinated, and the subjunctive premise is if the patient had been vaccinated in an alternate reality. This situation is described using the following equation:

$$\begin{aligned} P(\text{LC}_{\text{vaccinated}} \mid [\text{Not Vaccinated,LC, Male}, 50+]) &= P(\text{LC} \mid [\text{Vaccinated}, \text{Male}, 50+]) \\ &= 0 \cdot 0.50 + 1 \cdot 0.50 \\ &= 0.50 \end{aligned}$$

This translates to the following in text:

"Given the patient [*not vaccinated, male,* 50+*, LC*], what would be the probability of LC if the patient *were* vaccinated?"

The reasoning underlying this counterfactual statement is the following:

- The patient has not been vaccinated and has developed LC symptoms;
- This implies that, most likely, the patient is at a greater risk of developing LC symptoms;
- Thus, knowing that the patient has developed LC whilst he was not vaccinated, this patient is probably still more likely to have had developed LC *had* he been vaccinated, compared to other vaccinated patients. Therefore, the counterfactual statement considers the (increased) susceptibility of a specific patient to developing LC symptoms. Thus, the probability for developing LC may be increased even if the patient had in fact received a COVID-19 vaccination.

Constructing a counterfactual will take into account that the patient has developed LC symptoms in the factual situation (where the patient has not been vaccinated). Thus, it considers an increased risk of developing LC in the subjunctive situation (where the patient *is* vaccinated) based on the factual situation.

Predicting patients not (yet) represented in the tree In real world situations there exist many different types of patients, all with a possibly unique combination of features. It may occur that a new patient has a unique combination of features that is not yet represented in the tree as visualized in Figure 8. For demonstrative purposes, we consider another patient with the following characteristics: [Vaccinated = Y, Gender = Male, Age = 50−]. The current tree does not yet include male patients that are younger

than 50 years of age. When predicting the likelihood of LC for such patients, we have implemented the tree to classify the patient based on similar patients. The tree will backtrack through the list of statements (i.e., the patient characteristics), to find the largest set of conditions for which a data point (and thus a node in the tree) exists, removing the conditions back to front. It will then take the weighted average of all similar data points to make its prediction for the patient. In this example, the tree will see that no data points exist for the given combination of conditions, but after eliminating the 'Age' feature, a node exists. The prediction for the specified patient will be the weighted average of all data points with the following conditions: [Vaccinated = Y; Gender = Male]. Among these patients the development of LC symptoms is 50/50, and therefore, the resulting probability $P(\text{LC} \mid [\text{Vaccinated} = \text{Y}, \text{Gender} = \text{Male}, \text{Age} = 50-] = 0.50$.

Classifying patients (fuzzy variables) In the previous paragraphs we have converted the fuzzy concept of 'Age' to a crisp concept, by implementing a threshold at 50 years. This prevents the proper handling of uncertainty in the data (e.g., what about an age of 49 years?). To maintain the vagueness and uncertainty in the data, we formalise a membership function where we allow the possibility for having a partial membership in an element to a fuzzy set. We create linguistic variables, variables whose values are words in a natural language, to create two fuzzy sets. For this example we consider the two fuzzy sets ('young' *vs* 'old') as depicted in Figure 9. The tree as presented in Figure 8 remains the same, and thus, utilizes crisp variable thresholds and is based on the concepts of a traditional PT. However, as new patients are introduced and classified using the tree, fuzzy variables will be treated as such. For this example we consider a patient with the following characteristics: [Vaccinated = N, Gender = Female, Age = 48]. As a results, based on the membership function depicted in Figure 9, this patient belongs to both the young and old classes to some degree. Namely, with a membership of 0.20 to the young class, and 0.80 to the old class. Therefore, the classification of this patient will be based on the weighted average of both groups, weighted according to the degrees of membership. This results in a probability for LC as presented in Equation 2.

$$\text{Fuzzy prediction: } P(\text{LC} = \text{Y}) = \text{mf}_{\text{young}} \cdot P(\text{LC} = \text{Y}) + \text{mf}_{\text{old}} \cdot P(\text{LC} = \text{Y}) = 0.2 \cdot 1 + 0.8 \cdot 0 = 0.2 \quad (2)$$

The probabilities $P(LC = Y)$ and $P(LC = N)$ should always sum to 1, to demonstrate that this is true the probability for $LC = N$ is presented in Equation 3.

$$\text{Fuzzy prediction: } P(\text{LC} = \text{N}) = \text{mf}_{\text{young}} \cdot P(\text{LC} = \text{N}) + \text{mf}_{\text{old}} \cdot P(\text{LC} = \text{N}) = 0.2 \cdot 0 + 0.8 \cdot 1 = 0.8 \quad (3)$$

The PT, as depicted in Figure 8, classifies the patient to incur LC with a probability of 0. However, implementing the concepts of FPT, the patient is predicted to incur LC symptoms with a probability of 0.2. This prediction essentially being the weighted (according to the degree of membership) average of several paths in the tree. This demonstrates the important nuances that incorporating fuzziness in feature values may bring. Considering a patient aged 48 to belong fully to the group of patients that are 50 years or less, leads to a possible loss of important information. As this patient is very close to the age of 50, he/she should be considered accordingly.

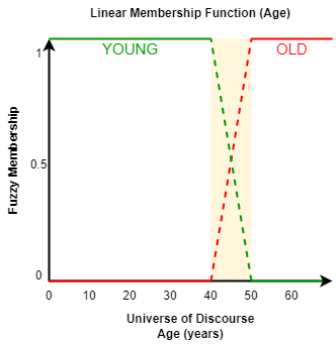


Figure 9: Linear membership functions for the linguistic fuzzy variables 'young' and 'old', that represent the age of a patient ('0'=green, '1'=red, yellow area=where the sets overlap and thus become fuzzy.

4 Real World Medical Examples: Implementing FPT

In this chapter the proposed FPT method is implemented to make predictions in two real medical scenarios. Firstly, a case of classifying malignant and benign thyroid nodules will be handled. Including detailed descriptions of the process of designing the tree, the inner workings of the tree, and setting out the differences between classifying using regular PTs and the FPT method. Thereafter, we discuss the case of predicting the 2-year risk of patients suffering from Chronic Kidney Disease (CKD) to progress to the most advanced stage of this disease, namely End Stage Renal Disease (ESRD), also known as kidney failure. The second case study will serve to substantiate the generalisability of the proposed methods. Furthermore, it is interesting to incorporate two unrelated case studies as they consider two different types of diseases, and the performance of the proposed methods can be examined more carefully. The two diseases are different in terms of the classifications introduced by the International Statistical Classification of Diseases and Related Health Problems (ICD). Thyroid cancer falls under the category of neoplasms (ICD-10 Code: C73) and CKD belongs to the diseases of the genitourinary system (ICD-10 Code: N18) [45].

4.1 Case Study I: Thyroid Nodules

The medical field concerned with the diagnosis of thyroid nodules is in need of an effective yet interpretable method for distinguishing malignant (cancerous) and benign thyroid nodules.

4.1.1 Clinical Question & Aim

Thyroid nodules are lumps within the thyroid gland, they are exceedingly common and a frequent find on neck sonography. These nodules are of importance as they bear the risk of being malignant, and several types of carcinomas may occur. As much as half of all people are found to have at least one thyroid nodule by the age of 60 [24]. However, only 5 to 10% of thyroid nodules are found to be cancerous [9, 42, 59]. How to distinguish between benign and malignant thyroid nodules is a great clinical challenge, one in need of solving in order to perform the appropriate surgery (lobectomy) with the correct indication. Patients that undergo surgery to remove (part of) the thyroid gland, run the risk of needing lifelong hormone replacement therapy (thyroid hormones control functions such as body temperature, digestion and heart functions). However, many patients still needlessly undergo surgery in response to their thyroid nodules. Which is burdensome for the patient, as well as it leading to high healthcare costs for society.
Since the risks are high in the clinical field of classifying thyroid nodules, an effective yet interpretable model to assist in the complex decision making process of clinicians is considered very valuable.

Disease Detection Workflow The detection of thyroid nodules follows a clear workflow, we will discuss the order of events with corresponding features that represent each stage in the data set. There are four main steps defined in the process of thyroid nodule diagnosis:

1. Echographic exam: thyroid nodules may be discovered in various ways, but typically with the use of ultrasound. Based on the results of the echographic exam, scores are assigned for each of the five ultrasound features, shown in the top row in Figure 10. After which the thyroid nodule is assigned a classification, indicating the level of suspiciousness. Each patient present in the data set has the assigned number of points for each ultrasound feature, as well as the total score as predicted by both the EU and ACR TIRADS system.
2. Biopsy: based on the classification of the EU and ACR TIRADS there is an indication to perform the FNA (biopsy) or not. The thresholds maintained by ACR TIRADS for these indications are reported in Figure 10.
3. Cytological evaluation: this refers to the analysis of body cells under a microscope. This process is performed by pathologists, looking at clusters of cells present in the biopsy to classify the patient with the correct THY score. The THY scores are depicted in Figure 11, these are similar to the classifications of the TR scores as presented in Figure 10. However, the THY scores are based on the cytological evaluation, while the TR scores are determined on the basis of echographic examination. All scores are reported in the data set.
4. Histopathological/FUP evaluation: from the cytological evaluations two groups of patients are formed. Firstly, patients with highly suspicious nodules will undergo thyroidectomy. Thyroidectomy is the surgical removal of all or part of the thyroid. After this surgery, the pathologist is able to classify the nodule to be benign or malignant with certainty, by means of histological evaluation. The second group of patients, those that have nodules that have been classified as benign or pre-malignant,

may be followed at a follow-up (FUP) during which their final diagnosis is done. The results from the histopathological evaluations are reported in the dataset as a "Final" classification that can be "Benign" or "Malignant". A number of patients in the data set have not received a final diagnosis, this is due to these patients being lost to FUP. Usually these patients went to another hospital for surgery or FUP, caused by the long waiting lists.

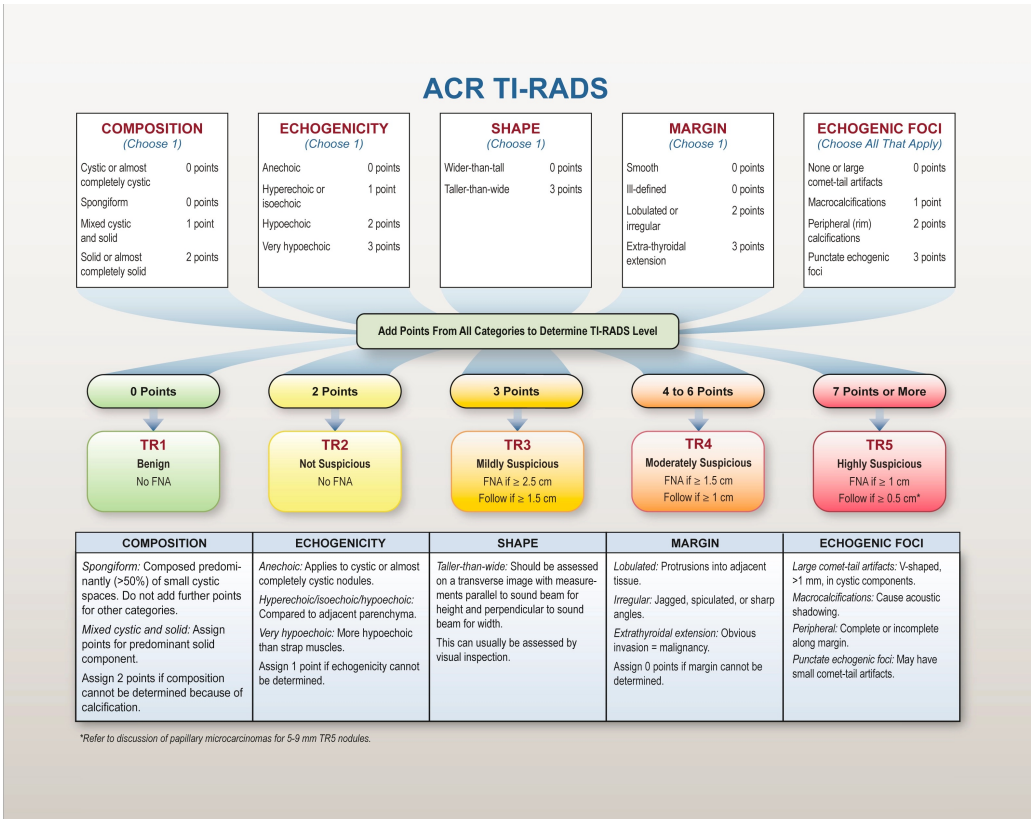


Figure 10: ACR TIRADS classification chart [65]. The scores of a thyroid nodule in the five echographic features are quantified to determine the final ACR TIRADS risk class.

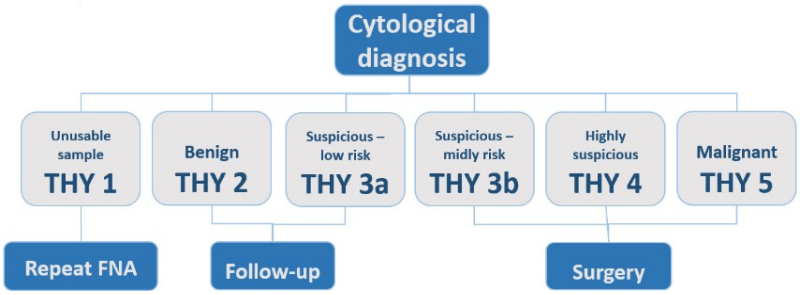


Figure 11: THY scores and recommendations based on Cytological evaluation.

Terminology This paragraph will elucidate some of the terminology and abbreviations commonly used in the medical field and specifically in the area of thyroid nodules. For the identification of nodules that warrant biopsy or surgery, systems have been founded to assist in the classification of thyroid nodules based on their degree of suspicion. There are two commonly used systems to assess thyroid nodules. Firstly, the EU-TIRADS (Thyroid Imaging Reporting and Data System) as presented by the European Thyroid Association [53]. Secondly, the ACR-TIRADS, as presented by the American College of Radiology

[65]. Both the EU-TIRADS and ACR-TIRADS are practical, useful and generally good classification systems. The two classification systems share common characteristics, but also differ in several aspects of the classification process. Often leading to slight differences in their final diagnoses. Both classification systems are represented in the data set. To provide some further insights, the ACR system is depicted in Figure 10. It divides thyroid nodules into five categories, each with a follow-up recommendation:

- TR1: Benign nodule, no biopsy needed (risk of malignancy is 0.3%);
- TR2: Not suspicious, no biopsy needed (risk of malignancy is 1.5%);
- TR3: Mildly suspicious, follow-up if the nodule has a size larger than 1.5 cm, biopsy if the size exceeds 2.5 cm (risk of malignancy is 4.8%);
- TR4: Moderately suspicious, follow-up if nodule size is larger than 1 cm, biopsy if the size exceeds 1.5 cm (risk of malignancy is 9.1%);
- TR5: Highly suspicious, follow-up if nodule size is larger than 0.5 cm, biopsy if the size exceeds 1 cm (risk of malignancy is 35%).

The thyroid nodules are categorized based on the amount of 'points' they score for five ultrasound features, the points per characteristic of the nodules are depicted in Figure 10. These features are composition, echogenicity, shape, margin, and punctate echogenic foci. Biopsy is performed by means of Fine-Needle Aspiration (FNA).
Furthermore, for each thyroid nodule it is relevant whether there is a case of Thyroiditis (inflammatory state of the thyroid) and/or Struma (an enlarged thyroid). Nodules may additionally be labeled as NIFTP, which means that the nodule is in a pre-malignant stage, or as Adenoma, which are benign nodules but often transform to become malignant.

4.1.2 The Data Set

This study includes a real medical case data set, including 448 patients who underwent Fine Needle Aspiration (FNA), guided by the United States (US) Thyroid Imaging Reporting and Data Systems (TIRADS). The patients were under treatment at the interventional radiology clinic, ASST Monza, Italy between the months of January and August 2019. The vast majority of the patients is of Caucasian descent. The original data set is of tabular form and contains the information on a total of 480 thyroid nodules that were subjected to FNA, each record represents a thyroid nodule case in a specific patient containing the relevant information for 33 features. Table 3 presents an overview of the relevant features and their types. From the 480 thyroid nodules, a total of 79 were excluded: 13 due to FNA exams that resulted in unusable samples and no repeating procedures were performed; and 66 nodules that did not receive definitive diagnosis. Resulting in a total of 401 subjects to be considered in developing the model.

The study has been approved by the ASST Monza Ethical Board (Associazione Italiana Ricerca sul Cancro-AIRC-MFAG 2016 Id. 18445, HSG Ethical Board Committee approval October 2016, 27102016) and study participants signed an informed consent.

Data Pre-processing The following steps have been taken during the pre-processing of the data set.

- A few minor errors in typing are corrected manually (e.g., errors where "N0" and "N" are written instead of "No", error in the data where "11/1271965" is written instead of "11/12/1965".);
- To clean the data, columns including trivial information are removed;
- The data has been filtered for patients that have received a definitive diagnosis. This is needed to serve as the target variable;
- New (linguistic) features have been deduced to represent continuous variables. This will be elucidated later within this Section (Paragraph Feature Engineering & Feature Selection);
- The disease risk classifications have been encoded as categorical variables;
- The data is divided in training and testing data, with the use of stratification. The split that is made and other characteristics will be elucidated in Section 4.1.3.

Table 3: Explanation of relevant features.

Description	Type
Date of birth	Ordinal
Gender	Binary
Date of FNA	Ordinal
Thyroid suspiciousness class	Ordinal
Final classification	Ordered
Thyroiditis indication	Binary
Struma indication	Binary
Composition of thyroid nodule	Categorical
Echogenicity of thyroid nodule	Categorical
Shape of thyroid nodule	Categorical
Margins of thyroid nodule	Categorical
Measure presence of echogenic foci	Categorical
EU TIRADS indication	Categorical
ACR TIRADS indication	Categorical
Thyroid nodule dimensions	Quantitative

Feature Engineering & Feature Selection The data set contains a considerable amount of features. Not all features will and can be used in creating the tree, as this would lead to blowing up the size of the tree and many leaves will be very specific, which leads to very few data points per leaf. Otherwise, this could lead to overfitting and the generalisability of the model is lost. Meaning that the model will not achieve a desirable performance on new data points. Furthermore, several of the features are continuous, and thus need to be amended to be represented in a binary manner. In order to do so, linguistic features are deduced from the data (linguistic variable is a variable whose values are words or sentences in a natural language [71]). We can use these linguistic features to create binary variables, but most importantly linguistic variables are often deployed in fuzzy logic to make the expression of fuzzy rules and sets easier. Two continuous variables are transformed to linguistic features such that these can be modeled as fuzzy variables.

The first (linguistic) feature that is created represents the age class of patients. There are two age classes, and therefore it can be represented as a binary variable. The feature is called '*50Plus*', and it represents a crisp boundary for a patient's age being under or over the age of 50. This split results in two classes of patients: young *vs* old patients. The reasoning for defining the age of 50 to be the distinction is based on several studies that have found that the occurrence of thyroid nodules increases as patients get older [33, 68]. Another reason for creating an informative variable representing the age of the patient, is due to the decrease in likelihood that thyroid nodules are malignant as age increases. Kwong et al. [33] show that younger patients, in the age categories from 20 to 50 years old, when they have thyroid nodules, are considerably more likely to be of malignant nature. Moreover, within the group of 'young' patients, being patients between the ages of $20 - 50$, the risk of malignancy further increases as the patients are younger. Kwong et al. [33] report that the youngest group (ages $20 - 29$) of patients had a 14.8% risk of malignancy per thyroid nodule at diagnosis. Whilst the oldest group (ages ≥ 70) only showed a 4.8% risk of malignancy per nodule. Therefore, one could even consider splitting the age up into multiple categories (e.g., young *vs* middle-aged *vs* old).

The second linguistic variable is created to represent the size of the thyroid nodule. Again, we divide the size of a nodule to fall into one of two classes. The feature is called '*Large Nodule*', and the threshold defining whether a nodule will be referred to as large is when it has a dimension of at least 20 millimeters (mm). The reasoning underlying setting the threshold at 20 mm is twofold. Firstly, the mean size of a thyroid nodule in the data set is 20.66 mm. Secondly, several studies have reported the detection of a threshold at a nodule size of 20 mm [2, 17]. The highest risk for malignancy was found in nodules smaller than 20 mm, and no increase in malignancy risk is found for nodules larger than 20 mm. In fact, the bigger the nodule, the lower the prevalence of cancer in nodules [2, 17]. Thus, the size of a thyroid nodule seems to be inversely related to the risk of malignancy, however, nodule size on its own seems to be a poor predictor for malignancy [17]. Due to this division, we indirectly create two classes that make up the size of a nodule: small *vs* large, where small refers to any nodule with dimensions up to (but not including) 20 mm, and large nodules to any nodules of 20 mm or larger.

Data Exploration Exploring the data to understand the contents of the data set and its characteristics is important in order to design a tree reflecting both the data and domain knowledge. This paragraph discusses some important trends in the data and their (possible) underlying clinical explanations. Which will stress the importance of these features and the underlying reasoning of incorporating them into the final tree.

Thyroid nodules are nearly three times more common to occur in females than they are in males [50]. This is likely to be due to the production of hormones taking place in the thyroid gland, which are different for males and females. The data set reflects this, with a predominance of thyroid nodules occurring in female patients, namely as much as 73.3%, as can be seen in Table 4. However, although females are more likely to develop thyroid nodules, males more frequently appear to develop malignant thyroid nodules. Table 4 shows that 7.5% of thyroid nodules in women are found to be cancerous, whilst 11.2% of thyroid nodules in men are cancerous. The authors in [8] found similar results; they stated that although patients were more frequently female than they were male, the frequency of cancer was significantly lower in female patients than they were in male patients.

Table 4: Distribution of female *vs* male patients, the number of malignant cases per gender category, and the percentage of malignancy found in both genders.

	Female	**Male**	**Total**
Population	294 (73.3%)	107 (26.7%)	401
Nr. of malignant cases	22 (64.7%)	12 (35.3%)	34
Malignancy %	7.5%	11.2%	8.5%

Size has been found to be inversely related to the risk for malignancy [2, 17]. A similar trend can be observed in the data. Table 5 shows that, while small nodules (dim. $<$ 20 mm) make up 55.1% of all nodules, they account for 67.7% of all malignant nodules. The probability for malignancy in small thyroid nodules is 10.4%, which is substantially higher than in the case of large nodules (dim. $\geq$ 20 mm) where the probability for malignancy is observed to be 6.1%.

Table 5: Distribution of nodule sizes (small *vs* large), the number of malignant cases per size category, and the percentage of malignancy found in each size categories.

	Small	**Large**	**Total**
Population	221 (55.1%)	180 (44.9%)	401
Nr. of malignant cases	23 (67.6%)	11 (32.4%)	34
Malignancy %	10.4%	6.1%	8.5%

As both the age of a patient and the size of the thyroid nodule are inversely related to the risk of malignancy, this could indicate that larger nodules occur more frequently in older people (i.e., there is dependence between the two variables age and size). However, based on Table 6, these variables seem to be independent of each other. Two variables are independent when the occurrence of one does not influence the probability of occurrence of the other, which is the case when the following equation holds:

$$P(A \cap B) = P(A)P(B) \tag{4}$$

Equations 5 and 6 show this formula applied to the variables age and large nodule size (with a large nodule being >20 mm). The resulting probabilities show that the two variables are indeed independent of each other. Thus, indicating that both variables independently influence the risk for malignancy.

$$P(50+ \cap \text{Large}) = \frac{138}{401} \approx 0.344 \tag{5}$$

$$P(50+)P(\text{Large}) = \frac{309}{401} \cdot \frac{180}{401} \approx 0.346 \tag{6}$$

Classifying the risk profiles of thyroid nodules based on the information obtained during the echographical examination (and the biopsy) is very important for deciding which patients need to undergo surgery. Two systems are commonly used for guidance in this process, namely the EU and ACR TIRAD systems. Both systems assign points for specific characteristics and ultimately determine a risk profile based on the total number of points and give an indication for needing to do a biopsy or not. The performances of these two systems are compared in Table 7. The performances of these systems are very comparable. Besides

Table 6: Distribution of 50+ *vs* 50− patients and the number of large nodules per age category.

	50+	**50-**	**Total**
Population	309 (77.1%)	92 (22.9%)	401
Nr. of large nodules	138 (76.7%)	42 (23.3%)	180

providing an indication whether to perform biopsy or not, the EU and ACR TIRADS systems categorize each nodule to a risk profile. Only the risk classifications as made by the ACR TIRADS will be considered here. The clinicians indicated that this systems has a better overall performance and, therefore, generally produces more reliable results. Table 8 provides some insights on the ability of ACR TIRADS to categorize the patients according to their risk profiles. The malignant cases are spread over the classes TR3 to TR5.

Table 7: Performance of EU TIRADS *vs* ACR TIRADS based on given indications to do biopsy (FNAB) or not. Row 1: indicates the number of biopsy indications have been given per system, Row 2: indicates how many of these cases were malignant, Row 3: indicates the number of malignant cases that these systems have missed.

	EU FNAB	**ACR FNAB**	**Total**
Population	203	180	401
Malignant cases, FNAB = yes	24 (70.6%)	23 (67.6%)	34
Malignant cases, FNAB = no	10 (29.4%)	11 (32.4%)	

Table 8: Occurrence of benign and malignant cases per ACR TIRADS risk class.

	TR1	**TR2**	**TR3**	**TR4**	**TR5**	**Total**
Benign cases	29	35	111	165	27	367
Malignant cases	0	0	5	11	18	34
All cases	29	35	116	176	45	401

However, the stage at which we desire to support clinician decision making, is when deciding whether or not to surgically remove (part of) the thyroid gland. At this point the results from the biopsy will already be available. Based on the observations made from the biopsy the risk profile for the nodule is assessed. This risk classification is generally a better indication of malignancy, as can be seen in Table 9 where the frequency of malignancy is presented per risk class.

Table 9: Occurrence of benign and malignant cases per risk class (classification based on biopsy).

	TIR2	**TIR3A**	**TIR3B**	**TIR4**	**TIR5**	**Total**
Benign cases	334	24	8	1	0	367
Malignant cases	0	2	5	10	17	34
All cases	334	26	13	11	17	401

We compare the individual explanatory power of the different features to each other. Table 10 shows the individual impact of each individual feature on the probability of a thyroid nodule to be malignant. Each probability represented in Table 10 is the result of a conditional probability, which can be written in the form $P(\text{Malignant} \mid \text{Age} < 50) = 0.105$. This shows that the nodule class is a strong predictor of malignancy, as the probability for malignancy steadily increases for higher classes.

4.1.3 Training Procedure

The data is split into train and test subsets, in order to verify the performance of the proposed model. The available data set contains information on 480 thyroid nodules, of which 79 are excluded (discussed in Section 4.1.2). Resulting in a total of 401 thyroid nodules to be considered. The data set is split into cuts of 75/25, 75% for training and 25% is used for testing. The data set is relatively small, but also we must consider that the classes are imbalanced. Only 34 of these 401 thyroid nodules are diagnosed as malignant. Therefore, during the training process in order to consider this imbalance in the classes we make use of

Table 10: Impact of each individual feature on the probability of malignancy in a thyroid nodule.

Features	Value	Probability of Malignancy
Age	< 50	0.105
	≥ 50	0.075
Dimensions	< 20 mm	0.101
	≥ 20 mm	0.059
Gender	Female	0.074
	Male	0.106
ACR TIRADS	TR1	0
	TR2	0
	TR3	0.04
	TR4	0.06
	TR5	0.4
Nodule Class	TIR2	0
	TIR3A	0.077
	TIR3B	0.382
	TIR4	0.91
	TIR5	1

stratification. Furthermore, bootstrapping is used to determine the accuracy of the predictions made by the model, by providing 95% confidence intervals.

Stratification When the data is split into train and test sub sets, stratification is used in order to create subsets that are representative of the entire patient data set. This will ensure that the classes of patients (i.e., patients with malignant nodules *vs* benign nodules) are evenly distributed over the train and test subsets. The process of stratification is very important in classification problems, even more so when there is skewness in the classes.

Bootstrapping Bootstrapping is a statistical techniques that creates many different bags of random samples, drawn from the data set with replacement. During every bootstrap the model is trained on the bootstrapped data and then tested on the out-of-bag (OOB) data. The portion of the data that is in the OOB has never been selected in any of the random samples, and thus, the model has not yet seen this data. Therefore, the model's performance can be well tested using this OOB test data set. Using bootstrapping we can construct confidence intervals. In order for these confidence intervals to give a good estimate of the variance in the results, the number of bootstraps needs to be sufficiently large. As the number of resamples (i.e., number of bootstraps) increases, the results will increasingly resemble a normal distribution. This method is based on the Central Limit Theorem (CLT). In this case study, we will repeat the bootstrapping procedure 1000 times, meaning that 1000 different random samples of the data set are drawn to create a model and predict the unseen data.

4.1.4 Performance Evaluation

The performance of the PT and FPT are evaluated with the help of several performance metrics. We will be using a combination of accuracy, specificity, sensitivity, and precision.

This study focuses on imbalanced binary classification. In supervised binary classification we classify observations into one of four categories: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). We visualize these categories using a confusion matrix, as shown in Figure 11.

Table 11: The confusion matrix: measuring performance.

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (**TP**)	False Negative (**FN**)
	Negative	False Positive (**FP**)	True Negative (**TN**)

Accuracy is the ratio of correctly predicted observations to the total observations. It is, therefore, a highly intuitive performance measure. Accuracy is a great measure when the distribution of classes are symmetrical and the costs of FNs and FPs are equal. Which is not the case in the medical data sets at hand. Thus, although accuracy is a very intuitive measure, it is not the most important measure used in this study. Accuracy is computed by means of Equation 7.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \tag{7}$$

Specificity (also known as 'True Negative rate') represents the model's ability to correctly identify the patients without a condition. Concretely, it is the ratio of correctly predicted negative observations to all observations predicted as negative. The formula for computing the specificity is presented in Equation 8.

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{8}$$

Sensitivity (also often referred to as 'Recall' and 'True Positive rate') represents the fraction of positive observations that were predicted correctly. In other words, it is the probability of a positive test given that the patient has the disease. So in the field of medicine, sensitivity refers to the ability to correctly detect ill patients [3]. This is an important metric for CDSSs. Sensitivity will not be distorted by the imbalanced classes present in the data sets at hand. The sensitivity can be mathematically expressed using Equation 9.

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{9}$$

Precision is the ratio of correctly predicted positive observations to the total of positively predicted observations, where sensitivity is the fraction of relevant observations that have been retrieved. Thus, high precision relates to a low FP rate. It can be computed using the formula presented in Equation 10.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{10}$$

The overall performance of the model will be assessed based on these four metrics. However, not every metric is equally important. As we must consider the context of the study being the medical field, where the cost of the misclassification of an ill patient to be not ill is higher than classifying a healthy patient as ill. Although, the model must find a balance between the two, because an overly careful model (i.e., predicting many patients as positive) might lead to a high FP rate and will decrease the trustworthiness of the predictions made by the model. Furthermore, we will need to consider that often in medicine the data is imbalanced, as the positive class (i.e., the ill patients) is generally the minority class.

4.1.5 An FPT model for the prediction of thyroid nodules

This section discusses the modeling choices made in creating the tree, its variables, and crafting the fuzzy sets of the previously introduced linguistic variables.

Creation of the Tree In this project, the trees are constructed using both induction and deduction. The features used and the order of the features in the tree are based on domain knowledge (deduction), however, the transition probabilities in the trees are based on the data (induction). Often PTs are used to represent temporal processes, in which each nodes represents a state of the process, in this case it is important to respect the temporal order of features in the tree.

The tree will reflect all possible patient profiles/hypotheses that are present in the data set. When a path has a probability of zero (i.e., no data points exist for this path) it will not be visualized. No strict temporality is observed in the data set at hand, therefore, the creation of the tree is an iterative process. Several combinations and orders of features are tried. Ideally, this process is done in collaboration with domain experts, to understand the way in which the different features influence each other and the workflow that is followed in practice to reach a diagnosis. Moreover, Section 4.1.2, paragraph 'Data Exploration', has provided us insights on the importances of several features and some of the trends in the data. Furthermore, it is important to be wary of the depth of the tree. This is directly related to the leaf size, being the average number of data points in each leaf. If the tree grows too deep, each total realization may only represent a single data point (recall that a total realization is a path from root to leaf). In this case the model will overfit and not generalise well to new data. Ultimately, the tree for the detection of thyroid nodules is composed of a combination of variables containing general information on the patient (age and gender), information

on the nodule (dimensions, conditions struma and thyroiditis), and the risk profile that resulted from the biopsy. A schematization of the probability tree is visualized in Figure 12. It shows the possible values for all variables in the model, the arcs represent the conditional probabilities for the entire data set. This figure does not show every possible total realization as a separate path in the probability tree, as we have seen in the examples handled in Section 3.2. Visualizing the tree as we have seen in previous sections will blow up the image, due to the large number of paths. The tree as presented in Figure 12 will – on average – have 2.5 data points in each total realization up to a leaf (thus, excluding the leaf nodes). Although in reality, some patient profiles are more common and the data points tend to clump together while others may not even occur at all. The average number of data points per total realization is determined by calculating the possible number of combinations of feature values and dividing the total number of data points by this number ($\frac{401}{5 \cdot 2^5} = 2.51$).

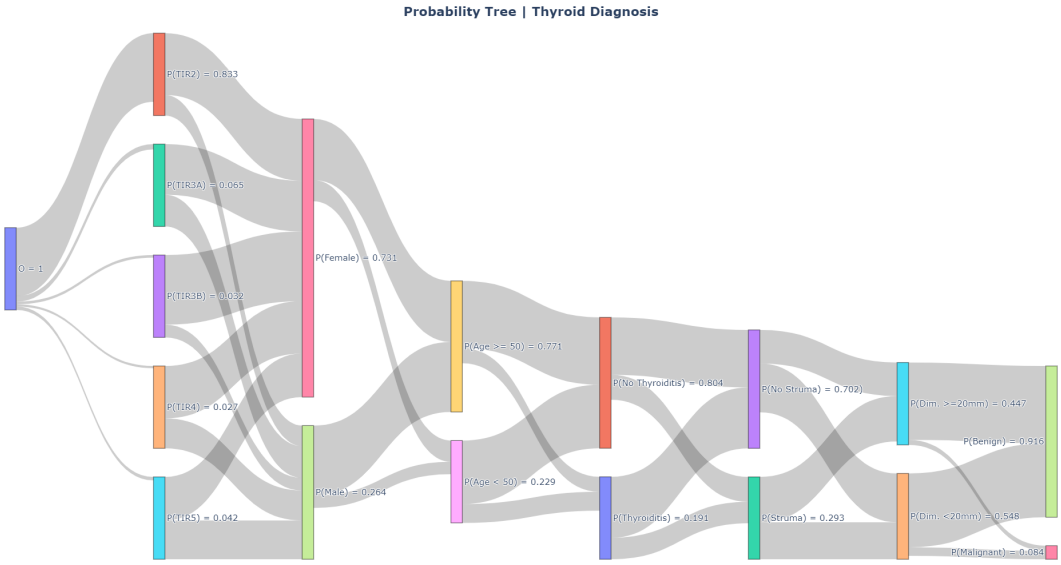


Figure 12: PT developed for the classification of Thyroid Nodules. Each arc represents the transition probability from one node (variable) to the next, the size of the arcs are representative of the corresponding transition probabilities. The PT has been constructed based on the thyroid nodule data set. As a path is followed along the tree, it will end up in benign or malignant leaf nodes. The majority class in the leaf nodes will be the final prediction (in the case 'threshold = 0.5' is used).

Furthermore, a tree is created to allow for the possibility to include the risk profile based on the echographical features of the nodule (shape, margins, echogenicity, etc.). This information is indirectly included in the tree when incorporating the classifications made by the ACR TIRADS systems (classes TR1 to TR5). These are not included in the primary tree, as it will lead to a substantial increase of the size of the tree, and thus, a lower amount of data points in each total realization. Furthermore, it decreases the performance of the tree. Different variations of the tree are introduced to improve the usability of the tool that is developed and will be presented in Chapter 5. The variations of the tree that are created are the following: (1) a tree containing only the risk classification based on the biopsy results (Figure 12); (2) a tree containing both risk profile classifications made by the ACR TIRADS and resulting from the biopsy (Figure 13; and (3) a tree containing only the risk classification based on the ACR TIRADS. Tree (2) respects the temporal order that is followed in practice, where first patients are given an ACR TIRADS score based on the results of their echographical examination, after which a biopsy may be done to assign patients a 'TIR' risk profile. This tree is presented in Figure 13. The additional trees, (2) and (3), aim to assist clinicians as much as possible in situation where a clinician does not have all the information, or would like to see the influence of adding the ACR TIRADS classification.

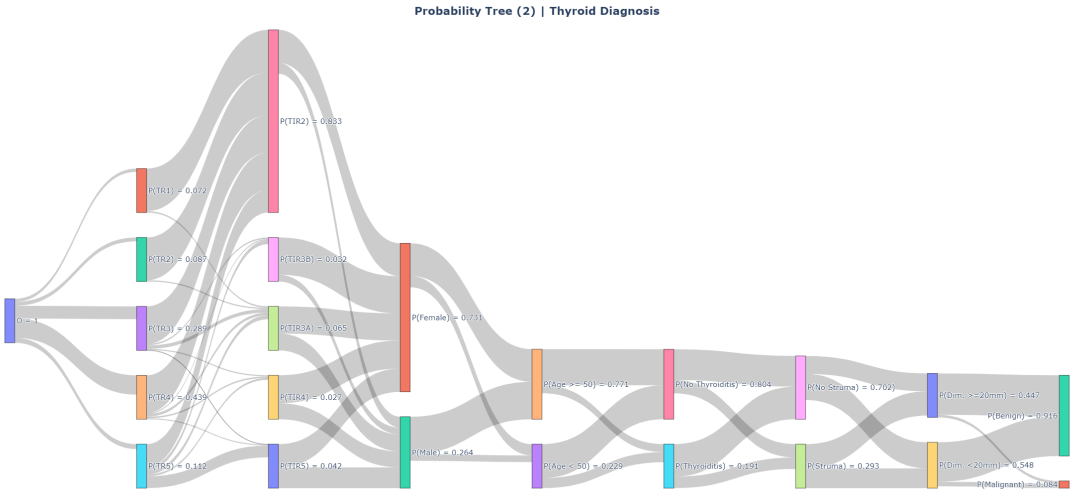
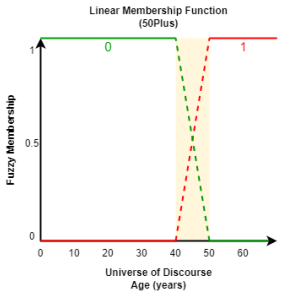


Figure 13: PT II developed for the classification of Thyroid Nodules. Each arc represents the transition probability from one node (variable) to the next, the size of the arcs are representative of the corresponding transition probabilities. This tree contains additional information on ACR TIRADS risk classifications.

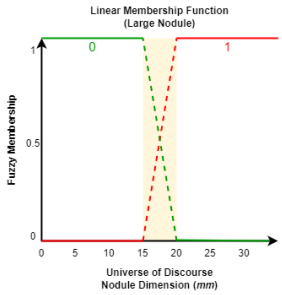
Crafting the Fuzzy Sets The fuzzy sets are carefully crafted based on domain knowledge. Fuzzy sets are created to represent the linguistic terms '50Plus' and 'Large Nodule'. The aim of creating fuzzy sets for these two terms is to create gradual boundaries.

We let the set '50Plus' be two fuzzy sets, and therefore, the membership degree $\mu_{50+}(x)$ represents the proximity of x to the elements of 50+ ('50Plus = 1'), and the membership degree $\mu_{50-}(x)$ represents the proximity of x to the elements of 50− ('50Plus = 0'). The fuzzy set is depicted in Figure 14a. An element x is completely in the fuzzy set 50− while it is smaller than 41. Starting from the value 41, the $\mu_{50+}(x)$ will be incremented with steps of 0.10 for every 1 increase in x, up to the point where the $\mu_{50-}(x) = 0$ and $\mu_{50+}(x) = 1$.

Similarly, the linguistic variable Large Nodule (LN) is represented by two fuzzy sets; with membership degrees $\mu_{LN=0}(x)$ and $\mu_{LN=1}(x)$. The gradual transition between the two fuzzy sets starts from the value 16 (mm), where the $\mu_{LN=0}(x) = 0.80$ and $\mu_{LN=1}(x) = 0.20$. From the value 20 (mm), the element x will be completely in the fuzzy set $LN = 1$. This fuzzy set is depicted in Figure 14b.



(a) We represent the two fuzzy membership sets and membership functions for the linguistic variable 50Plus.



(b) We represent the two fuzzy membership sets and membership functions for the linguistic variable Large Nodule.

Figure 14: Linear membership functions for the fuzzy variables (Age and Nodule Dimensions) ('0'=green, '1'=red, yellow area= where the sets overlap and thus become fuzzy).

Demonstration: Show Inner Workings of the FPT This paragraph will present a demonstration of the (F)PT developed for the thyroid nodule case study. In order to do so, a synthetic patient is introduced, for whom the feature values are presented in Table 12. The processes of classifying the synthetic patient based on PT and FPT can be observed in Figure 15a and Figure 15b respectively. These figures show only the fraction of the tree that is relevant for the classification of the presented patient.

Table 12: Synthetic patient characteristics.

Feature	**Value**
Class	TIR3B
Gender	Female
Age	48
Thyroiditis	No
Struma	No
Nodule Dimensions	18mm

In this demonstration we consider the feature age as the linguistic feature '50Plus' and the feature nodule dimensions as the linguistic feature 'Large Nodule', where both are modelled to be fuzzy. The corresponding membership functions are depicted in Figure 14a and Figure 14b. In the original PT these variables have crisp thresholds (Age: 50+; Dimensions: $\geq$ 20 millimetres (mm)).

The PT is depicted in Figure 15a, in which the probabilities on the arcs represent the transition probabilities to go from one node to a following node. For the FPT implementation, visualised in Figure 15b, the tree and its transition probabilities do not change compared to the PT. However, when we make a prediction using the FPT method, we consider the fuzziness in the fuzzy variables. Then a prediction may be based on several total realization (paths from root to leaf) in the tree, where each realization is considered to the degree of membership for the concerning fuzzy variable. In Figure 15b, the nodes corresponding to fuzzy variables are depicted as yellow node boxes.

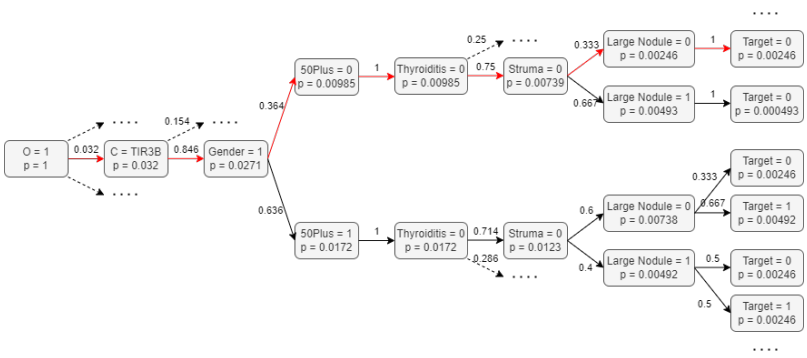
The PT (Figure 15a) classifies the synthesised patient to have a benign nodule ('Target = 0') with a probability of 1. This prediction is made by simply following the path corresponding to the patient's information in the tree, which is denoted by the red arrows (in Figure 15a). The FPT (Figure 15b) considers all paths as the patient is partially in the set of both fuzzy variables ($0 < mf < 1$). Thus, all arcs leaving the yellow fuzzy nodes are depicted in red. The FPT classifies the synthesised patient to have a nodule that is benign with a probability of 0.427. This demonstrates the important nuances that incorporating fuzziness in feature values may bring. Considering a patient aged 48 to belong fully to the group of patients that are 50 years or less, leads to a possible loss of important information. As this patient is very close to the age of 50, the patient should be considered accordingly. Therefore, the FPT considers the feature 'nodule dimensions' to be a fuzzy variable. In the PT implementation, a large nodule

is defined as any nodule with dimensions strictly larger than or equal to 20mm (crisp threshold). The FPT implements a linear membership function, herein a nodule with dimensions of 18mm belongs to the class of 'large nodules' to a (membership) degree of 0.80. The written out calculations, showing the integration of the membership values, to obtain the classification for the synthetic patient as made by the FPT is provided in Equation 11.

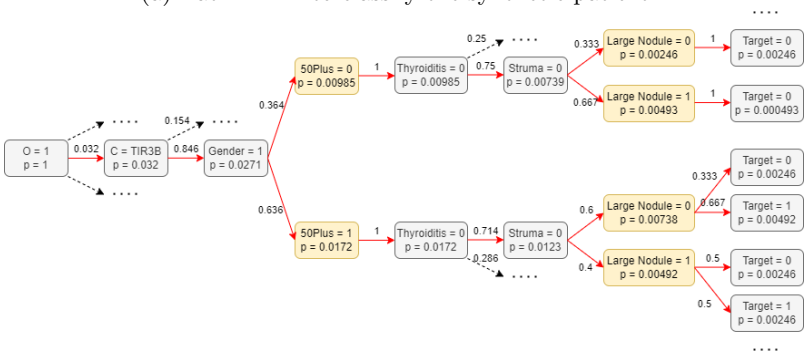
$$\begin{aligned} P(\text{Malignant} \mid \text{ synthetic patient (fuzzy)}) = \\ & 0.2 \cdot \big((0.2 \cdot 1 \cdot 0) + (0.8 \cdot 1 \cdot 0)\big) + \\ & 0.8 \cdot \big(0.2 \cdot (0.333 \cdot 0 + 0.667 \cdot 1) + \\ & 0.8 \cdot (0.5 \cdot 0 + 0.5 \cdot 1)\big) = 0.427. \end{aligned} \tag{11}$$

The probabilities for a thyroid nodule to be malignant and benign always sum to 1. For the written out calculation for the probability that the same patient has a benign thyroid nodule, refer to Equation 12. And as stated before, the probabilities in the tree do not change, and thus the probability distribution remains as before (all leaf nodes sum to 1).

$$\begin{aligned} P(\text{Benign} \mid \text{ synthetic patient (fuzzy)}) = \\ & 0.2 \cdot \big((0.2 \cdot 1 \cdot 1) + (0.8 \cdot 1 \cdot 1)\big) + \\ & 0.8 \cdot \big(0.2 \cdot (0.333 \cdot 1 + \cdot 0.667 \cdot 0) + \\ & 0.8 \cdot (0.5 \cdot 1 + 0.5 \cdot 0)\big) = 0.573. \end{aligned} \tag{12}$$



(a) Path in PT to classify the synthetic patient.



(b) Path in FPT to classify the synthetic patient.

Figure 15: Results of classifying the same patient using PT vs FPT. The grey nodes denote crisp variables, where yellow boxes denote fuzzy variables. The path that is taken in the tree to classify the synthetic patient is highlighted by the red arrows. In (b) FPT, all arrows are red as all 'paths' are considered to some degree for the final classification. Only the part of the tree relevant to classifying the example is shown here, the complete tree will be substantially larger.

4.1.6 Results

The performance in terms of accuracy, specificity, sensitivity, and precision along with corresponding confidence intervals (95%) for the PT and FPT implementations are presented in Table 13 and Table 14 respectively. The performances are reported for the two trees that have been introduced PT I and PT II. PT I refers to the tree that only includes the risk classifications based on the biopsy, as depicted in Figure 12. PT II refers to the tree that includes both the risk classification according to the ACR TIRADS and the biopsy, as depicted in Figure 13. The predictions are made using the default threshold, which is set at a probability 0.50. The predicted probability is compared to this threshold, when it is lower than the threshold the predicted classification will be the negative class, when it is higher or equal to the threshold it will be classified as positive. This threshold can be adjusted according to the situation. For example, it may be increased in times of low capacity when only the most urgent cases can be assessed, i.e., in situations where prioritization is required.

The performance of the PT and FPT are determined based on 1000 bootstraps. Using this technique, confidence intervals are constructed, presenting a range in which the results will fall with a certainty of 95%. As previously discussed, the number of bootstraps needs to be sufficiently large, such that the results will resemble a normal distribution. In order to test the number of bootstraps that is used, the results are plotted in histograms in Figure 21 for PT I and in Figure 22 for PT II, which can both be found in Appendix A.

The performance of the PT and FPT are very similar for both trees. This is to be expected, as the models are very similar. The models base their predictions on the same tree, the difference being the handling of the fuzzy variables (age and nodule dimensions). For PT I, the FPT implementation slightly outperforms the PT implementation. When comparing Table 13 to Table 14 it can be observed that the accuracy is negligibly higher, however, the confidence interval does show that this accuracy is achieved with less variability. The same is true for the specificity and the precision. As to the sensitivity, the FPT obtains a performance that is 1 percent point higher than the PT implementation. The differences between the implementations of the two models are very small, as there are only two fuzzy variables. Therefore, the results show that the FPT is better able to capture the fuzziness in these two variables and seems to perform better for data points that fall within this fuzzy region. Furthermore, it performs more steadily with regards to the precision (as indicated by the lower bound of the CI). This combined with the one percent point increase in sensitivity are the two most important metrics in any healthcare situation. As it indicates a model's ability to identify the positive cases, i.e., the patients with a malignant thyroid nodule.
The performance of the two models for PT II is slightly reduced compared to the performance in PT I. Mainly it performs worse with regards to sensitivity and precision, which are evidently the two most important metrics. As can be seen, it is not beneficial to include the ACR TIRADS classifications into the tree. The results achieved by the PT and FPT in the case of PT II are quite similar to each other.

Table 13: PT implementation results on accuracy, specificity, sensitivity and precision (95% Confidence Intervals (CIs)). Based on 1000 bootstrapped data sets. Threshold = 0.50.

	Accuracy (95% CI)	**Specificity (95% CI)**	**Sensitivity (95% CI)**	**Precision (95% CI)**
PT I	96.8% [93.1 – 99]	98.2% [94.6 – 100]	82.2% [55.6 – 100]	84.1% [58.3 – 100]
PT II	96.4% [93.1 – 99]	98.3% [94.6 – 100]	76.5% [44.4 – 100]	83.6% [60 – 100]

Table 14: FPT results on accuracy, specificity, sensitivity and precision (95% Confidence Intervals (CIs)). Based on 1000 bootstrapped data sets. Threshold = 0.50.

	Accuracy (95% CI)	**Specificity (95% CI)**	**Sensitivity (95% CI)**	**Precision (95% CI)**
PT I	96.9% [94.1 – 99]	98.3% [95.7 – 100]	83.2% [55.6 – 100]	84.2% [63.6 – 100]
PT II	96.4% [93.1 – 99]	98.3% [95.7 – 100]	76.7% [44.4 – 100]	83.3% [60 – 100]

4.1.7 Benchmark Models (Results Comparison)

The results obtained by the PT and FPT are compared to the two previously introduced benchmark models that belong to the class of IAI methods, namely LR and DT.
The performances based on accuracy, specificity, sensitivity, and precision for both the LR and DT model are presented in Table 15, with corresponding 95% confidence intervals. Predictions are made and performance is assessed for both combinations of features: PT I (Fig. 12) and PT II (Fig. 13). Both the LR and the DT are able to handle continuous predictor variables, therefore, the original variables are used instead of the linguistic variables that have been introduced for the age and dimensions ('50Plus' and 'Large Nodule'). The rest of the variables are either categorical or binary, and thus, are not changed.
Comparing the performance of the PT (Table 13) and FPT (Table 14) to the performances of the LR and DT in Table 15, it can be concluded that the model LR with predictor variables as presented in PT II is the best performing model. This is noticeable as the performance of all other models is better for PT I than it is for PT II. When analysing the performance, mainly we are interested in maximizing the performance for the metrics sensitivity and precision. Based on this, we can conclude that the FPT model in combination with the variables as presented in PT I reports the second best performance. Overall, the performances of all models are quite comparable and they struggle in correctly classifying the same data points. These are the data points that represent patients that are in the TIR3A or TIR3B class (classes based on the biopsy). As there are very few data points present in the data set that have a class TIR3A (2 data points) or TIR3B (5 data points) and end up having a malignant nodule, it is logical that the models generally struggle with classifying these data points.

Table 15: LR (no penalty, LBFGS solver) and DT results on accuracy, specificity, sensitivity and precision (95% Confidence Intervals (CIs)). Based on 1000 bootstrapped data sets.

		Accuracy (95% CI)	**Specificity (95% CI)**	**Sensitivity (95% CI)**	**Precision (95% CI)**
LR	PT I	97.1% [94.1 – 100]	98.5% [95.7 – 100]	82.7% [55.6 – 100]	85.8% [66.7 – 100]
	PT II	97.5% [94.1 – 100]	98.8% [95.7 – 100]	84.3% [55.6 – 100]	88.3% [66.7 – 100]
DT	PT I	96.3% [93.1 – 99]	98% [94.6 – 100]	79.8% [55.6 – 100]	81.5% [57.1 – 100]
	PT II	96.2% [93.1 – 99]	97.8% [93.5 – 100]	80% [55.6 – 100]	80.3% [57.1 – 100]

4.2 Case Study II: Nephrology

Nephrology refers to the study of kidneys. This case study specifically aims to support clinicians in the prediction of the occurrence of End Stage Renal Disease (ESRD) in patients that are suffering from Chronic Kidney Disease (CKD). In other words, we aim to predict the risk of kidney failure (the last stage of CKD) in patients suffering from CKD.

4.2.1 Clinical Question & Aim

CKD is a condition in which the kidneys are damaged and cannot filter blood the way that they should. Because of this, excess fluid and waste from blood remain in the body and may cause other health problems, such as heart disease and stroke. Prevalence and incidence of CKD have almost doubled in the past two decades [69]. We aim to predict whether patients will progress to ESRD, which is the most advanced and harmful stage of kidney disease. In this stage, patients often need dialysis and/or a kidney transplant. Dialysis is a procedure to remove waste products and excess fluid from the blood when the kidneys stop working properly. Thus, it is a therapy that replaces the normal blood-filtering function performed by the kidneys. Determining the probability of kidney failure may be useful for the communication between the clinicians and patients, furthermore, it will be leading in the triage and management of nephrology referrals and the timing of dialysis and kidney transplants. Reliable and interpretable prediction tools are needed to identify patients with CKD that are at greater risk of developing ESRD. The progression is often hard to predict, as the disease does not progress in the same rate for all patients. Moreover, it is said that a substantial portion of patients suffering from CKD do not follow a predictable pattern of disease progression.

Disease Detection Workflow Nephrology clinics keep clinical information records of every patient coming in with kidney disease. This consists of information on general records (such as age, gender, Body Mass Index (BMI), cigarette smoking habits), medical history (any diseases such as diabetes, coronary artery disease, congestive heart failure, and cerebrovascular and peripheral vascular disease), record of medication, results of 24-hour urine collection (levels of protein, kalium, etc.), and results of blood tests (levels of creatinine, hemoglobin, etc.). Based on some of this information a patient is classified with a stage of kidney disease. The six stages of kidney disease are depicted in Figure 16, and are directly related to a patient's GFR levels. Within this case study, we will only consider patients that suffer from CKD in the stages 3A or higher. If desired, the clinicians can perform a kidney biopsy in order to examine the tissue under a microscope for signs of damage or disease. However, this information in not included in the data set at hand. At this point, the clinician can decide on the appropriate care for the patient. For instance, renin-angiotensin-system inhibitors (RASI) can be prescribed to slow down the progression of CKD. Although there are risks, as RASI may potentially induce hyperkalemia (a condition where you have alarmingly high levels of potassium in your blood) which can lead to cardiac arrest and eventually death. One year after a patient's first visit to the clinic, the patient is called for a second examination, during which the same clinical information is recorded. Some patients do not make it to the second visit due to the earlier occurrence of an event. This event can be the patient reaching the end stage of kidney disease or death. In this case, these events are registered instead. After the two visits, periodic updates were planned at 18, 24, 30, and 36 months after the study. These records are not included in the data set.
When a patient reaches ESRD, which is the fifth and final stage of kidney disease, kidney failure will occur and patients will need to undergo dialysis or even a kidney transplant to survive.

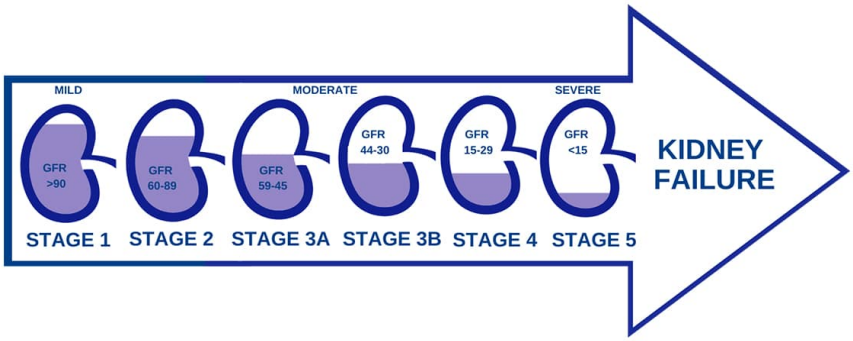


Figure 16: Stages of Kidney Disease.

Terminology This paragraph will discuss some of the important terminology and corresponding abbreviations used in the field of nephrology.

- Hyperkalemia: a condition where there is an elevated level of serum potassium (kalium in the blood). Normal potassium levels for adults lie between 3.5-5.0 mmol/L. Hyperkalemia occurs when levels go above 5.5 mmol/L.
- Glomerular Filtration Rate (GFR): describes the flow rate of filtered fluid through the kidney. A GFR test estimates how much blood passes through these filters each minute. It is one of the key measures of measuring kidney function, but GFR alone is not sufficient for clinical decision making [32]. When a patient has a GFR ≤ 60, the patient is diagnosed with kidney disease. As GFR drops below a value of 15 this is usually associated with kidney failure.
- Proteinuria: the protein excretion rate per 24 hours (in the urine). When a patient has increased protein levels present in the urine it is worrisome. Protein is needed in the body, so when high levels occur in the urine, it leaves the body which is not healthy. This condition can be a sign of kidney damage. It is considered a risk factor when protein excretion > 0.5 g/24h.
- Serum creatinine: the concentration of creatinine in the blood, measured in mg/dL. Creatinine levels in the blood are a marker of kidney functioning, and high levels of serum creatinine indicates improper functioning of the kidneys. Typically, a high creatinine level is anything above 1.3 mg/dL.
- Anemia: a condition that occurs when there are not enough red blood cells to carry adequate oxygen to body tissue. This is caused by low hemoglobin, which is the case when hemoglobin levels are

< 13 g/dL for males and < 12 g/dL for females. Often these patients are already under treatment for anemia and taking Erythropoietin Stimulating Agents (ESA). Thus, patients taking ESA are considered anemic regardless of their hemoglobin levels.

- Phosphate: high phosphate levels are often a sign of kidney damage. The body needs some phosphate to strengthen bones and teeth, produce energy, and build cell membranes. However, in larger than normal amounts, phosphate can cause bone and muscle problems and increase the risk for heart attacks and strokes. It is considered high phosphate when phosphate > 4.6 mg/dL.
- Cardiovascular Disease (CVD): is a term for the group of disorders affecting the heart and blood vessels.
- Diabetes: a chronic disease that affects the way in which the body turns food into energy. It occurs when the pancreas fails turn sugar into insuline or the body cannot make good use of the insulin it produces. Kidney damage is a very common complication of diabetes (both type I and type II diabetes). Both types are present in the data set, no distinctions are made between the two types in this case study.

4.2.2 The Data Set

The data set is the result of a multi-centre prospective study pooling data from 46 established hospitals and clinics located in the EU. The studies have been conducted for a total of 20 years, starting in 1995 and continuing up to 31 December 2015. The data set is of tabular form and contains the records of 3.278 patients, each consisting of 131 different features. Table 16 presents an overview of the relevant features and their types. From the 3.278 records, a total of 674 were excluded. A number of 236 duplicate patients have been removed, as well as 242 patients that had GFR scores that are no indication for CKD. Furthermore, 202 patient files have been excluded due to missing information about the levels of potassium (kalium) and/or protein present in the urine. This leads to a final data set containing the records of 2.599 patients with CKD. All study participants have signed an informed consent, more details can be provided upon request.

Table 16: Explanation of relevant features.

Description	Type
Date of birth	Ordinal
Gender	Binary
BMI	Quantitative
Diabetes	Binary
Smoking	Binary
CVD	Binary
Kalium (mmol/l)	Binary
GFR (mg/mmol)	Binary
Stadium of GFR	Ordinal
Serum Creatinine (mg/dL)	Quantitative
Proteinuria (g/24h)	Quantitative
Hemoglobine (g/dl)	Quantitative
Phosphate (mg/dl)	Quantitative
RASI medication	Binary
Date of visits	Ordinal
Death pre-dialysis	Binary
ESRD	Binary

Data Exploration In this paragraph the data set is explored for meaningful features, insights and trends which may be useful in the creation of the probabilistic tree.

We investigate the influence of gender on the occurrence of CKD and its progression. CKD is found to be more common in women than in men [16], however, the data set does not reflect this. Male patients tend to progress to the final stages of CKD (i.e., ESRD) more commonly and faster than female patients [16]. This is observed in the data set. Table 17 shows these trends; from the total data set 2.5% is female of

whom 25% have progressed to ESRD, 57.5% of the population is male of whom 26.9% have progressed to ESRD. However, the differences between males and females progressing to ESRD are not notable.

Table 17: Distribution of female vs. male patients, the number of ESRD cases per gender category, and the percentage of ESRD occurrence in both genders.

	Female	**Male**	**Total**
Population	1106 (42.5%)	1493 (57.5%)	2599
ESRD cases	276 (40.7%)	402 (59.3%)	678 (26.1%)

Another variable that is significantly associated with CKD and – possibly with ESRD – is the age of a patient. CKD is more common to occur in older people [43], due to GFR levels that decrease as age increases, even in people without kidney disease. Among patients with similar levels of GFR, older patients have higher rates of death and lower rates of ESRD than younger patients [43]. In the data set the average age is 66.5 years (Table 18). Figure 23 (Appendix B.1) depicts the relationship researched between GFR and age in the data set. A clear trend of diminishing GFR as age increases is observable. Table 18 shows the numbers from the data set for the average age of patients that have progressed to ESRD to be 61.4 years, and the average age of patients that have died prior to undergoing dialysis is 75.3 years. Therefore, age is an important factor in both predicting the progression of ESRD and death before dialysis.

Table 18: The average age (in years) of all patients, patients that have progressed to ESRD, and patients that have died prior to dialysis.

	Total	**ESRD**	**Death (pre dialysis)**
Average age (years)	66.5	61.4	75.3

The most common causes for CKD are diabetes and hypertension (high blood pressure), responsible for up to two-thirds of all the cases [64]. Hypertension is one the of the most predominant risk factors for the development of several cardiovascular diseases (CVDs). Therefore, using the data set at hand we will evaluate the effects of diabetes and CVD on the development and progression of CKD. Table 19 shows that roughly one-third of the CKD patients suffer from diabetes and/or CVD. Similarly, Table 19 shows that a little under one-third of the ESRD cases have underlying conditions diabetes and/or CVD. Although diabetes and CVD are strong predictors of CKD, on their own they do not predict the progression to ESRD.

Table 19: Prevalence of diabetes and CVD in patients suffering from CKD and their progression to ESRD.

	Diabetes	**CVD**	**Total**
Nr. of patients	760 (29.2%)	809 (31.1%)	2599
ESRD cases	188 (27.7%)	177 (26.1%)	678

Anemia occurs when a patient does not have enough red blood cells, this is tested through measuring the hemoglobin levels in a blood test. The thresholds of healthy hemoglobin levels differ for men and women. Male patients are diagnosed with anemia when their hemoglobin levels are < 13 g/dL, for female patients this threshold lies at < 12 g/dL. Patients that are currently under treatment for anemia and thus are taking ESA – which may increase hemoglobin levels up to a seemingly healthy level – are still considered as being anemic. Table 20 shows the results of analysing the data set for anemia based on the thresholds for hemoglobin levels that have been established. 69.8% of patients that progress to ESRD have some level of anemia, with a total of 57% of patients in the data set having anemia. Therefore, anemia seems to be an indicator of CKD progressing to ESRD.

Table 20: Prevalence of anemia in patients suffering from CKD and their progression to ESRD.

	Anemia	**Total**
Nr. of patients	1482 (57%)	2599
ESRD cases	473 (69.8%)	678

Urine and blood test values are very indicative of the functioning of a patient's kidneys. Among other values, a clinician will look at the values of serum creatinine, proteinuria, phosphate, and potassium (indicator of

hyperkalemia). Typically, serum creatinine values are considered high when the concentration in the blood exceeds 1.3 mg/dL. However, the vast majority of patients in the data set have serum creatinine levels generously exceeding 1.3 mg/dL. A histogram that visualizes the distribution of serum creatinine levels over the data set is presented in Figure 24 in Appendix B.2. As a result, the threshold value of 1.3 mg/dL as a splitting condition is not very informative, as the majority of the data points will belong to the positive class. Therefore, a more informative threshold is identified. Hypercreatinemia (when serum creatinine is high) is considered mild with serum creatinine values up to 1.8 mg/dL, moderate between values of 1.8 - 3.5 mg/dL, marked between 3.5 - 7 mg/dL, and severe in the case of values higher than 7 mg/dL [18]. We choose a threshold for splitting the serum creatinine at a value of 2.15 mg/dL. The reasoning is tripartite: 1. we consider cases that are at least moderate; 2. Figure 24 (Appendix B.2) shows that the majority of data points fall in the bins ≤ 2; and 3. 2.15 mg/dL showed the highest performance compared to other values that are close to 2 mg/dL. Table 21 presents the results of analysing the entire data set for patients with CKD and compare these with the values corresponding to patients that progressed to ESRD. Hyperkalemia occurs in 34.4% of the cases, however, it is more frequent in patients that progressed to ESRD. Furthermore, alarming values of proteinuria, serum creatinine, and phosphate seem to be equally common. However, when looking at the cases of patients that have progressed to ESRD we see that in the first place serum creatinine values are very often too high, similarly, proteinuria values are generally too high, only phosphate is less frequently an issue.

Table 21: Urine and blood test values (Potassium ($>$ 5.5 mmol/L); Proteinuria ($>$ 0.5 g/24h); Serum Creatinine ($>$ 2.15 mg/dL); Phosphate ($>$ 4.6 mg/dL)).

	Hyperkalemia	**Proteinuria**	**Serum Creatinine**	**Phosphate**	**Total**
Nr. of patients	893 (34.4%)	1213 (46.7%)	1213 (46.7%)	1213 (46.7%)	2599
ESRD cases	281 (41.4%)	496 (73.2%)	553 (81.6%)	163 (24%)	678

Furthermore, we investigate the influence of smoking and body weight on the development of CKD to ESRD. Table 22 shows that 9.7% of the population with CKD have a habit of smoking. However, the patients progressing to the final stage of the disease are slightly more common to have a smoking habit, namely 10.8% of the patients smokes. Body weight is analysed using two categories; overweight (when BMI $>$ 25) and obesity (when BMI $>$ 30). Table 22 shows that a substantially large portion of the patients suffers from overweight and also obesity is relatively common. However, these patients do not necessarily seem to be more likely to progress to ESRD, which is not in accordance with the results reported in [30, 67]. Which may be due to the slightly higher rate of deaths prior to undergoing dialysis for patients suffering from overweight and obesity (Table 23).

Table 22: Occurrence of smoking, overweight (BMI $>$ 25) and obesity (BMI $>$ 30) in all CKD patients vs ESRD patients.

	Smoking	**Overweight**	**Obesity**	**Total**
Population	252 (9.7%)	1788 (68.8%)	705 (27.1%)	2599
ESRD cases	73 (10.8%)	417 (61.5%)	156 (23%)	678

Table 23: Rate of death (pre-dialysis) for all patients vs patients suffering from overweight (BMI $>$ 25) and patients suffering from obesity (BMI $>$ 30).

	Overweight	**Obesity**	**Total**
Deaths pre-dialysis	311 (70.8%)	113 (25.7%)	439

The GFR is a key indicator of kidney functioning, and therefore, a strong predictor in the deterioration of a patient's kidney function for the following 2-year period. Table 24 presents an overview of the probability of progressing to ESRD in the following two years for every stage of CKD, as well as the occurrence of each of these stages in the data set. These probabilities are the result of conditional probabilities, thus, $P(\text{ESRD} \mid \text{CKD Stage} = 5) = 0.720$.

Table 24: Occurrence of CKD stages and impact on the probability of progressing to ESRD in a 2-year period.

CKD Stage	Population	Probability of ESRD
3A	526 (20.2%)	0.065
3B	878 (33.8%)	0.113
4	871 (33.5%)	0.358
5	324 (12.5%)	0.720

Data Pre-processing The following steps have been taken during the pre-processing of the data set.

- Dealing with missing values:
  - Patients with `NaN`-values for the feature ESA – indicating whether a patient is currently under medication to deal with the condition of anemia – are assigned 0 values. Clinicians indicated that in the case of a missing value in this feature, this indicated that the patient was in fact not taking any ESA medication;
  - All rows that contained missing values for gender, age, BMI, GPR, serum creatinine, kalium, protein, or the target variable (ESRD) are removed from the data set.
- New (linguistic) features have been deduced to represent continuous variables. This will be elucidated later within this Section (Paragraph Feature Engineering & Feature Selection);
- The disease stages have been encoded as categorical variables;
- The data is divided in training and testing data, with the use of stratification. The splits made and other characteristics will be elucidated in Section 4.2.3.

Feature Engineering & Feature Selection A selection of the features present in the data set are used, to keep the tree moderate in size (i.e., depth) so it will be generaliseable to new data. Some of these features are not directly suited to be used in the tree and will be amended. Furthermore, many features present in the data set are derived from levels of certain elements in blood and urine, and therefore, these values are in a continuous range. These features will be amended to be represented as binary features.

A feature is created that represents the stage of the kidney disease ('CKD_stage'). These stages are based on the pre-defined stages as depicted in Figure 16.
Several features are selected that resulted from the blood and urine tests. The resulting features all fall in a continuous range, and therefore, need to be adjusted to be able to be represented as binary variables. These include anemia (hemoglobin), hyperkalemia (potassium), proteinuria, serum creatinine, and phosphate. For each of these variables threshold values are determined, whenever the value surpasses the assigned threshold it will be assigned a '1'. The thresholds are identical to the ones in the Paragraph on 'Data Exploration'. These thresholds are all crisp, in Section 4.2.5 we will discuss the fuzzy sets created for these variables. For the feature 'anemia' specifically, a patient is assigned a '1' when either the value surpasses the threshold or the patient is currently under treatment for anemia and taking ESA medication.
Furthermore, we introduce a feature '65PLUS', which indicates whether a patient is 65 years of age or older. This threshold has been chosen based on the information gain for several thresholds. The thresholds that have been explored are 65+/70+/75+, these are based on typical meaningful thresholds identified in the literature on CKD [43]. Moreover, the average age in the data set is 66.5 years, whilst the average age of patients progressing to ESRD is 61.4 years and the age of patients that die pre-dialysis is 75.3 years (Table 18). This substantiates testing setting the thresholds at the ages of 65, 70, and 75.

4.2.3 Training Procedure

The training procedure is designed similar to the procedure followed in the previous case study focused on thyroid nodules (Section 4.1.3). The data is split into a 75/25 cut, 75% is used for training and 25% for testing. As the classes are imbalanced (678 positive class *vs* 1.921 negative class) we make use of stratification. Furthermore, bootstrapping is used to test the model's performance on randomly drawn samples (with replacement) of the data set. The bootstrapping procedure is repeated 250 times, which allows for the determination of confidence intervals. The bootstrapping procedure is repeated 'only' 250 times due to the substantial increase in size of the population as opposed to the population in the data set used in the previous case study.

4.2.4 Performance Evaluation

The performance of the PT and FPT will be evaluated using the performance metrics as discussed in Section 4.1.4. We will thus be calculating the performance according to the following metrics accuracy, specificity, sensitivity, and precision. The chosen metrics are not all equally important, most weight is put on correctly identifying the positive class (i.e., the CKD patients that progress to ESRD). Therefore, the sensitivity – that is the ability to identify the positive class – is the most important performance metric. Although, in combination with the precision to ensure that the model does not achieve high sensitivity due to it being very generous with predicting the positive class. However, the accuracy and specificity are not trivial, as they will tell us something on the overall performance and the ability to predict the negative class.

4.2.5 The Model

This section discusses the modeling choices made in creating a tree to classify the risk of ESRD in CKD patients. Furthermore, fuzzy sets are crafted for the (linguistic) variables used in this model.

Creation of the Tree The process is repeated that was followed in the previous case study. Namely, the trees are constructed using both deduction (i.e., domain knowledge) and induction to determine the transition probabilities based on the data. Designing the tree is done based on the conversations with clinicians and based on previous research to predictive models in CKD patients [64]. There is no temporality between the features in the data set, and therefore, creating the tree is an iterative process. Multiple combinations and orders of features are tried. The included features are chosen based on the findings from the paragraph 'Data Exploration' in Section 4.2.2 and features that are deemed important by the clinicians (domain knowledge). The number of variables in the tree is carefully chosen, as we are considerate about the depth of the tree.
A schematization of the probability tree is presented in Figure 17, and consists of features that contain general patient information (age, diabetes), features based on blood tests values (serum creatinine, anemia (hemoglobin), GFR), and features based on urine testing values (proteinuria, phosphate). On average, the tree will have about 5 data points in each total realization up to a leaf. Although, some patient 'profiles' will be more common than others. This is determined by dividing the total number of patients by the number of possible combinations of feature values: $\frac{2599}{4 \cdot 2^7} = 5.08$.

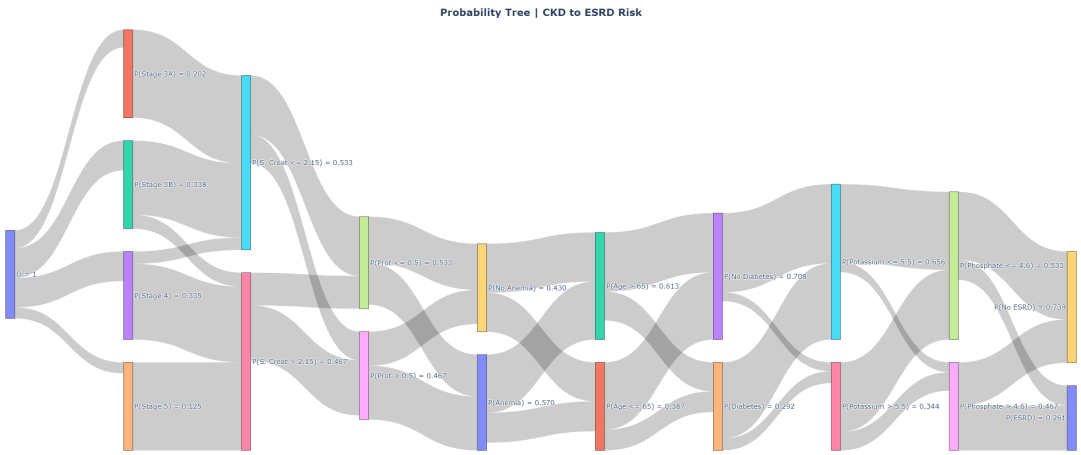
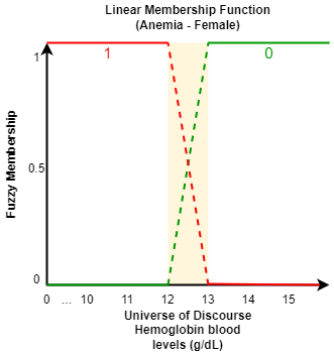


Figure 17: PT developed for predicting the risk of CKD patients progressing to ESRD. Each arc represents the transition probability from one node (variable) to the next, the size of the arcs are representative of the corresponding transition probabilities.

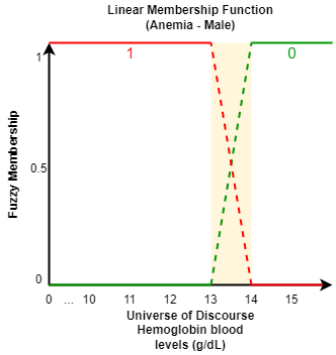
Crafting the Fuzzy Sets The following fuzzy sets have been crafted carefully, and in consultation with clinicians. Fuzzy sets are created to represent the variables: serum creatinine, anemia, proteinuria, hyperkalemia, phosphate and the age. The aim of creating these fuzzy sets is to create a gradual boundary between the positive and negative class. These fuzzy sets will are presented in Figure 18 and each are

discussed briefly.

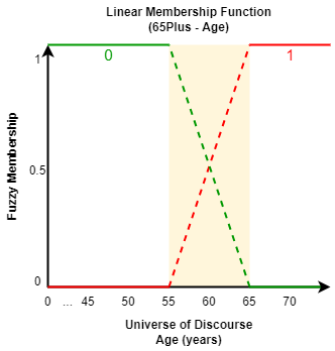
There are two sets representing anemia, one for female patients and one for male patients, as the threshold values differ for the two gender groups. Each anemia set is represented by two fuzzy sets, where $\mu_1(x)$ represents the proximity of x to the elements of the set '1'. The same applies to the set '0'. The fuzzy sets are depicted in Figure 18a for females, and in Figure 18b for males. The hemoglobin values that are considered to point towards anemia differ between males and females, thus, these groups are separated. Furthermore, anemia may be already be under treatment with the use of ESA, and therefore, anemia may be present but not observable in the hemoglobin levels. Therefore, the fuzzy rule for anemia has two components: "`IF` hemoglobin levels $< 12/13$ `OR` ESA $= 1$ `THEN` Anemia $= 1$". Otherwise, there is a gradual linear increase in degree of membership to the fuzzy set 'anemia' as the hemoglobin levels decrease.
The variable that has been created to represent the age is represented as a fuzzy set, imposing a gradual boundary to classify a patient as 'old' (in this case 'old' corresponds to an age of 65 years or more). The gradual transition between the two fuzzy sets starts from an age of 56, and from an age of 65 and onwards the element will be fully in the fuzzy set 'old'. The fuzzy sets are depicted in Figure 18c.
The prognosis of hyperkalemia is assigned when potassium levels increase above 5.5 mmol/L, however, any patient with potassium levels between 5.0 and 5.5 will now also be considered somewhat hyperkalemic (Figure 18d). Similarly, the sets for proteinuria, serum creatinine, and phosphate are fuzzified. As a result, the degree of membership to proteinuria follows a gradual linear increase between the values of 0.4 and 0.5 g/24h (Figure 18e. Serum creatinine increases gradually between values of 1.3 and 2.15 mg/dL (Figure 18f). Phosphate increases linearly between values of 4.1 and 4.6 mg/dL (Figure 18g).



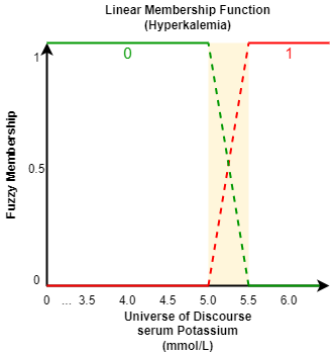
(a) Fuzzy sets for the variable Anemia (females).



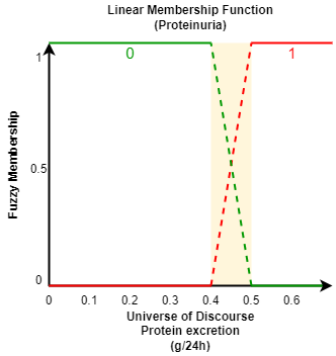
(b) Fuzzy sets for the variable Anemia (males).



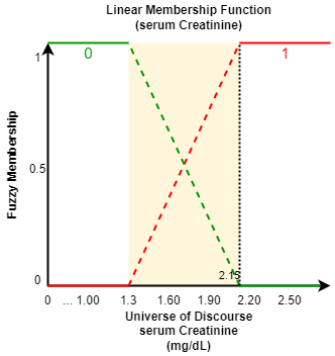
(c) Fuzzy sets for the variable age (65Plus).



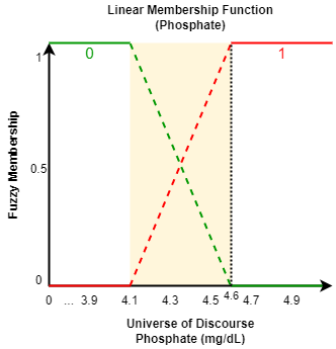
(d) Fuzzy sets for the variable hyperkalemia (serum potassium).



(e) Fuzzy sets for the variable proteinuria.



(f) Fuzzy sets for the variable serum creatinine.



(g) Fuzzy sets for the variable Phosphate.

Figure 18: Fuzzy sets and linear membership functions for the variables: anemia (males and females), age, hyperkalemia, proteinuria, serum creatinine, and phosphate ('0'=green, '1'=red, yellow area= where the sets overlap and thus become fuzzy).

4.2.6 Results

The performance of both the PT and FPT implementations for the prediction of the progression of CKD is presented in Table 25. The performance is measured by the same four metrics, where the importance is on the ability to detect the positive class. Meaning that the model's performance in the metrics sensitivity and precision is of main interest, and in particular the sensitivity metric. Comparing the performance of the PT and FPT based on the reported metrics presented in Table 25 shows that the main difference between the two models lies in the sensitivity scores. Where the FPT outperforms the PT by 5.5%. Furthermore, the overall accuracy is slightly improved compared to the PT. However, the PT demonstrates slightly better performance for the specificity metric. Nonetheless, it indicates that the FPT is better able to identify

the CKD cases that progress to ESRD in the following two year period (the positive class) than the PT. Regarding the precision, the two models achieve very comparable scores (difference of 0.2 percentage points in favor of the PT). This indicates that the FPT's increase in sensitivity is not achieved simply by the model being more generous with predicting the positive class. However, it must be noted that both models obtain performance that is generally perceived as low. Despite the relatively low scores in sensitivity and precision, the accuracy and specificity do obtain more promising results. However, these scores alone do not tell the whole story, as the majority of the data points belong to the negative class. Conclusively, both PT and FPT obtain decent accuracy and specificity scores, and thus, in general perform decently in predicting the progression of CKD in the following two years. However, the sensitivity and precision scores indicate that the models fall short on predicting the CKD cases that progress to ESRD. Although the FPT is an improvement to the PT, it still performs relatively poor.

Table 25: PT and FPT results on accuracy, specificity, sensitivity and precision (95% Confidence Intervals (CIs)). Based on 250 bootstrapped data sets. Thresholds used = 0.50.

	Accuracy (95% CI)	**Specificity (95% CI)**	**Sensitivity (95% CI)**	**Precision (95% CI)**
PT	76.9% [73.7 – 80]	84.1% [78.4 – 89.2]	56.7% [47.6 – 66.4]	56.1% [49.8 – 63.2]
FPT	77.2% [73.8 – 80.2]	82.5% [77.3 – 87.3]	62.2% [52.9 – 72.3]	55.9% [50 – 62.2]

4.2.7 Benchmark Models (Results Comparison)

The performance of the PT and FPT are compared to two benchmark models. The same two models as in previous sections are used as reference to compare the results to, namely logistic regression (LR) and decision tree (DT). The same combination of features has been used in creating the benchmark models as the (F)PT. However, LR and DT are both capable of handling continuous predictor variables, therefore, the continuous values of the variables are used. As opposed to the (fuzzy) binary variables that were created for the PT and FPT, that divided the data points into two divisions using splits. The performances for the two benchmark models, based on the four performance metrics, are presented in Table 26. Comparing the two benchmark models, the LR is the best performing model on all metrics except for sensitivity. The LR seems to be well capable in predicting the CKD patients that do not progress to ESRD, which concerns the majority of the patients present in the data set, hence, the high accuracy score. The patients that the LR does predict as the positive class, it predicts quite accurately, as indicated by the precision score of 74.1%. However, as stressed before, the greatest weight should be put on the performance in terms of sensitivity, as it is most important to identify the patients that are at risk of progressing to ESRD. The DT achieves a slightly higher performance in terms of sensitivity than the LR does, with a sensitivity score of 53.1%. On all other metrics, the LR outperforms the DT.
Next, we compare the performance of the LR and DT to the proposed PT and FPT. The LR performs superior to the FPT on all metrics except for the sensitivity. Overall, the LR improves the accuracy by 5.5% and the specificity by 11% compared to the FPT. Thus, it is significantly more accurate in making its predictions, but it seems to be somewhat limited to the data points that belong to the negative class. The higher performance of the FPT compared to LR in terms of sensitivity (10% increase) indicates that the FPT may actually be the preferred model. Although, it must be noted that all models are limited in their performance, and may not display the performance desired in a clinical environment. We will discuss this matter a bit further in Section 4.3.

Table 26: LR (no penalty, LBFGS solver) and DT results on accuracy, specificity, sensitivity and precision (95% Confidence Intervals (CIs)). Based on 250 bootstrapped data sets.

	Accuracy (95% CI)	**Specificity (95% CI)**	**Sensitivity (95% CI)**	**Precision (95% CI)**
LR	82.7% [80.6 – 84.8]	93.5% [91.5 – 95.6]	52.2% [45.3 – 59.4]	74.1% [67.9 – 80.8]
DT	74.6% [71.7 – 77.2]	82.2% [78.3 – 85.6]	53.1% [44.7 – 61.1]	51.5% [46.2 – 56.8]

4.3 Case Study Results: an Important Discussion on the Methods

Having implemented and evaluated the proposed methods in two unrelated case studies in the clinical field that handle different complex diseases, we will briefly discuss some of the key points and remarks that result from this. Furthermore, we discuss the models in terms of interpretability. Lastly, some unused yet potentially promising aspects of FPTs are mentioned, and we will discuss why this might be interesting to doctors in the future.

The performance of the (F)PT and benchmark models Across both case studies, the fuzzy probability tree (FPT) outperformed the regular PT. Including the fuzziness, that many medical variables inherently have, consistently improves the model performance. Although, for the first case study (thyroid nodules) the differences in performance between PT and FPT were relatively small. In the second case study (progression of CKD) the differences in performance were more notable. This is due to the small number of fuzzy variables in the first case study. Only two variables were fuzzy (age and nodule dimensions), as opposed to six variables that were fuzzy in the second case study. Therefore, the two models (PT and FPT) were quite similar in the first case study, and so the differences in performance were relatively small. Although there were in fact differences, and in both case studies the FPT achieved higher performance than the PT.
The performances of the proposed methods were measured against two benchmark models, and generally, the LR model seemed to outperform the FPT. Although, in the CKD case study the FPT outperformed the LR model on an important measure, namely, the ability to identify positive labels (i.e., the sensitivity).

Regarding the CKD case study, there are concerns as to the performance of all models. The FPT achieved a sensitivity of 62.2%, indicating that it is able to correctly predict the positive class more than 60% of the time. Although, the FPT model had the highest sensitivity score compared to all other models, it is still too low for clinicians to be able to use and trust the predictions of the model. Trustworthiness must come from both interpretability (i.e., transparency of the decision making process) and performance. If one of the two lacks, it is unlikely that the decision making process can be supported effectively by any model.

The interpretability of the (F)PT and benchmark models Besides comparing the (F)PT to LR and DT in terms of performance, the methods should also be compared in terms of interpretability. Although there is no concrete mathematical definition or measurable metric for interpretability [36], it was proposed to split interpretability into two main categories. Namely, an interpretable model should (1) give insights into how the model works in a way that is understandable to a human, and (2) reveal potential new knowledge [37]. The interpretability of a regular DT and the FPT are both based on a rule system that is induced from the tree. Where the edges are connected by 'AND', meaning that we obtain a rule specifying all conditions in a specific branch of the tree. As the depth of a tree increases, the rule system may become more complicated (the length of each rule will be equal to the depth of the tree, i.e., the number of nodes that are in the total realization). The FPT has an important advantage over DTs, namely that its outputs are presented in the form of probabilities. The prediction is an estimate of the probability of the event occurring, as opposed to a binary classification that is presented by a DT in classification problems. LRs also present their outcomes as a probability of the event occurring. As a LR tries to maximize its log odds (natural logarithm of the odds) function by optimizing the parameters. The log odds can be difficult to make sense of, however, the LR presents its estimates as an odds ratio (i.e., probability of the event occurring), which eases the interpretation of the results. Furthermore, it is able to quantify the amount a prediction will change when a certain feature is increased by a value of 1. As a result, FPTs (and DTs) are more inherently comprehensible for humans, as their outputs can be translated to natural language in the form of rules. Whereas, the inner workings of an LR may not be as inherently comprehensible, however, the outputs it presents are interpretable and it is capable of presenting clear feature 'weights', indicating the effect on the target variable after a change in a certain feature value.

The potential of the FPT method (counterfactuals) It is important to note that the full potential of the fuzzy probability tree may be yet to be discovered. As probability trees are causal models that use nodes to represent potential states of a process, therefore, the arcs between nodes indicate the probabilistic transition, but preferably also the causal dependency between two states of the process [28]. For the two implementations discussed within this project, the trees did not reflect clear processes wherein every node represents an actual state of the process that is a measurable moment in time. Consequently, it is hard to prove any causality between nodes. Possibly, the full potential of (F)PT will be used when there is a clear temporal aspect in the process that the tree aims to model, and the nodes represent actual states

of a process.
Furthermore, an important power of (F)PTs, that has been mentioned several times, lies in its ability to consider counterfactual statements. Counterfactuals allow for the testing of alternate realities, meaning that a doctor may test hypotheses that represent slight alterations to the factual situation. An example question that may be answered with the help of counterfactual statements in the CKD case study is: "*Given a patient that developed kidney failure and did not take any medication. What would be the probability of this patient developing kidney failure if he/she had in fact taken RASI medication?*". However, careful consultation with clinicians is necessary in creating such counterfactual statements, as RASI medication may induce hyperkalemia, which is again a potentially deadly condition. Nonetheless, this allows clinicians to reason about how the situation may have unfolded had other courses of action been taken. This ability to think in counterfactuals is natural to humans and it is what sets out human intelligence from other animals. When implementing (F)PTs, this human way of reasoning can be leveraged in AI methods that are also able to consider large amounts of data. The functionality to support the testing of counterfactuals is yet to be developed for the FPT implementation (Section 7.4 (Future research)).

5 Developing the Graphical User Interface

Interpretability is very important for decision support in clinical environments. However, the probability trees grow substantially large and complicated, impeding the comprehensibility of the methods for clinicians. Generally, clinicians lack a background in machine learning, probability theory and statistics. Part of the strength of the proposed FPT method is that it is able to present the certainty of its prediction. Therefore, a tool is created with the aim of allowing clinicians to interactively obtain relevant and naturally understandable statements. These statements are written in the form of probabilities, obtained from the underlying probabilistic decision tree that has been generated based on past patient records. In order to create an effective tool, able to assist in the complex decision making processes of clinicians, its information must be well conveyed.

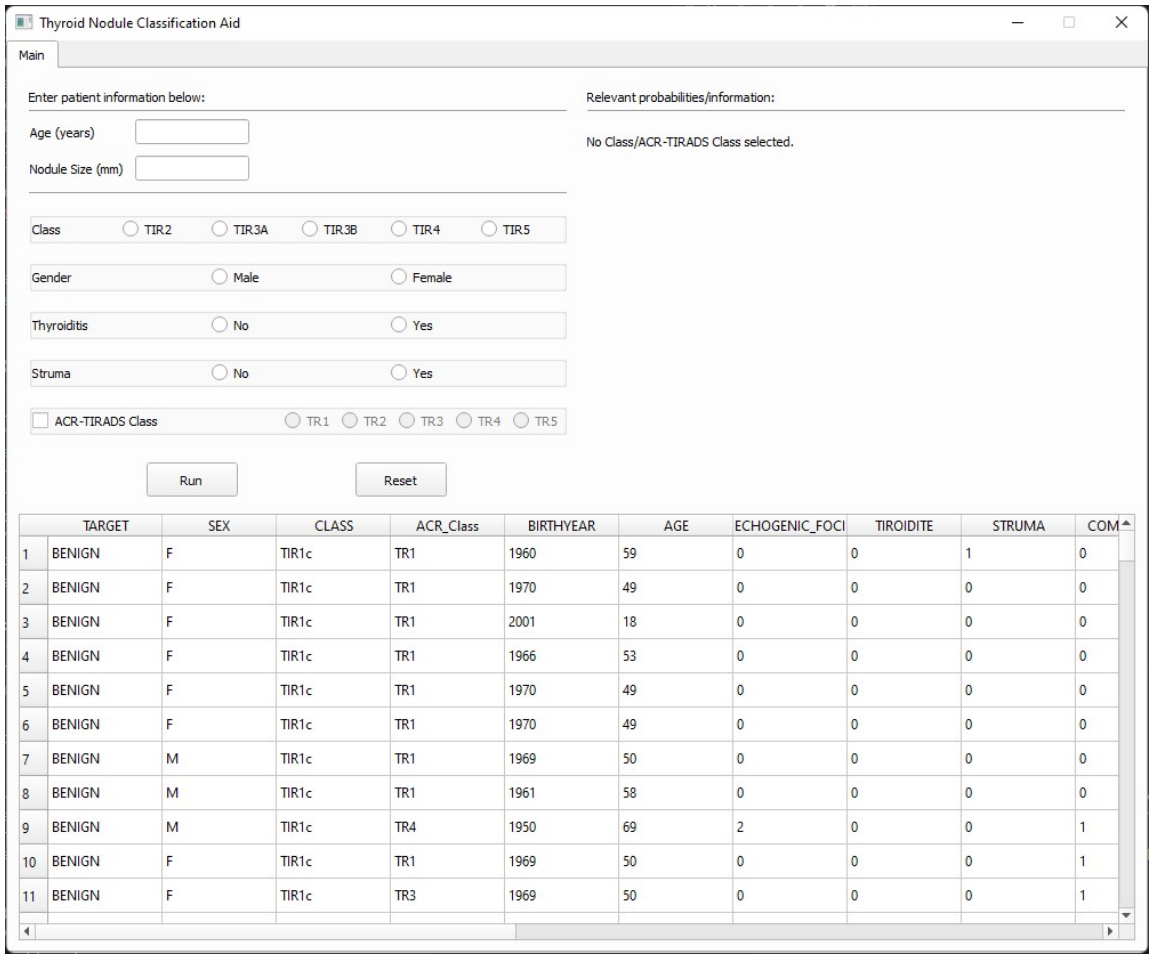
It is important to assess the interpretability of the proposed methods and its predictions. In order to do so, we must interface with clinicians to demonstrate the functioning of the model through the tool and coincide on the usability and interpretability of the model and the way the tool presents its predictions. This allows for the opportunity to effectively gather feedback on the method, its performance and ultimately the tool.

The rest of this chapter will demonstrate the tool that has been created, what information can be obtained by clinicians using the tool, the results from a feedback session held with clinicians, and the programs and frameworks used in creating the tool in order to promote the generalizability of the GUI setup, such that it can be generalized to other purposes.

5.1 Introducing the Tool

A tool in the form of a Graphical User Interface (GUI) is developed specifically for the thyroid nodule case study. The GUI is shown in Figure 19, which depicts the start-up screen when launching the GUI. The GUI consist of three components: (1) section where patient information is entered (top left), (2) a table containing all (relevant) data points (bottom half), and (3) a section where the prediction probabilities are presented (top right). When a patient's data is entered into the GUI, the "Run" button can be clicked to generate a prediction. In the case that a user would like to clear all fields of information, the "Reset" button should be clicked.

When entering the patient information, there are two options to specify the 'Class'. The first 'Class' variable refers to the TIR class (derived from the biopsy results), the second variable 'ACR-TIRADS Class' is derived from the ACR-TIRADS. The user has three options: (1) only specify a value for the TIR Class (this was referred to as PT I during the case study), (2) only specify a value for the ACR-TIRADS Class, or (3) specify a value for both (this was referred to as PT II during the case study). This means that another, third, tree is created to support the GUI, containing only the ACR-TIRADS Class. When the user wants to include the ACR-TIRADS classification, the check-box can be checked and an ACR-TIRADS class can be chosen. When the check-box is unchecked, the ACR-TIRADS classification will no longer be considered as shown in Figure 26 (Appendix C).
The GUI requires the user to at least select one 'class' before the model can make any predictions, but preferably a complete patient profile is entered. This is because the class features are the variables that correspond to the first nodes in the PT. The reason why no predictions can be made otherwise has been discussed in Section 3, and will be handled in the recommendations for future research in Section 7.4. When a full patient profile is entered containing a value for both 'class' variables, as shown in Figure 20, then two prediction probabilities are presented in the top right section of the GUI. The first prediction is based on PT I (only considers TIR class) and the second prediction is based on PT II (which considers both TIR and ACR-TIRADS class). Furthermore, the table will be filtered to show only the data points that are considered in the prediction making process of PT I. The table in Figure 20 shows a total of 11 patients, this means that the prediction made using the tree that contains only the TIR Class (i.e., PT I) is derived from those 11 patients. To highlight this and be transparent about the number of samples underlying the prediction made, the number of samples that supported the prediction is stated explicitly (Fig. 20). However, the degree to which each data point influences the final prediction probability may differ, due to the fuzzy implementation. Initially, the table contains the records for all 401 patients (as is shown in Figure 25 in Appendix C). As mentioned earlier, the table and message indicating the number of data points used in the prediction only reflects the patients that have been considered by PT I (i.e., the tree that only considers the TIR class).
As the clinicians enter a patient's information into the GUI, it may occur that faulty values are entered or



	TARGET	SEX	CLASS	ACR_Class	BIRTHYEAR	AGE	ECHOGENIC_FOCI	TIROIDITE	STRUMA	COM
1	BENIGN	F	TIR1c	TR1	1960	59	0	0	1	0
2	BENIGN	F	TIR1c	TR1	1970	49	0	0	0	0
3	BENIGN	F	TIR1c	TR1	2001	18	0	0	0	0
4	BENIGN	F	TIR1c	TR1	1966	53	0	0	0	0
5	BENIGN	F	TIR1c	TR1	1970	49	0	0	0	0
6	BENIGN	F	TIR1c	TR1	1970	49	0	0	0	0
7	BENIGN	M	TIR1c	TR1	1969	50	0	0	0	0
8	BENIGN	M	TIR1c	TR1	1961	58	0	0	0	0
9	BENIGN	M	TIR1c	TR4	1950	69	2	0	0	1
10	BENIGN	F	TIR1c	TR1	1969	50	0	0	0	1
11	BENIGN	F	TIR1c	TR3	1969	50	0	0	0	1

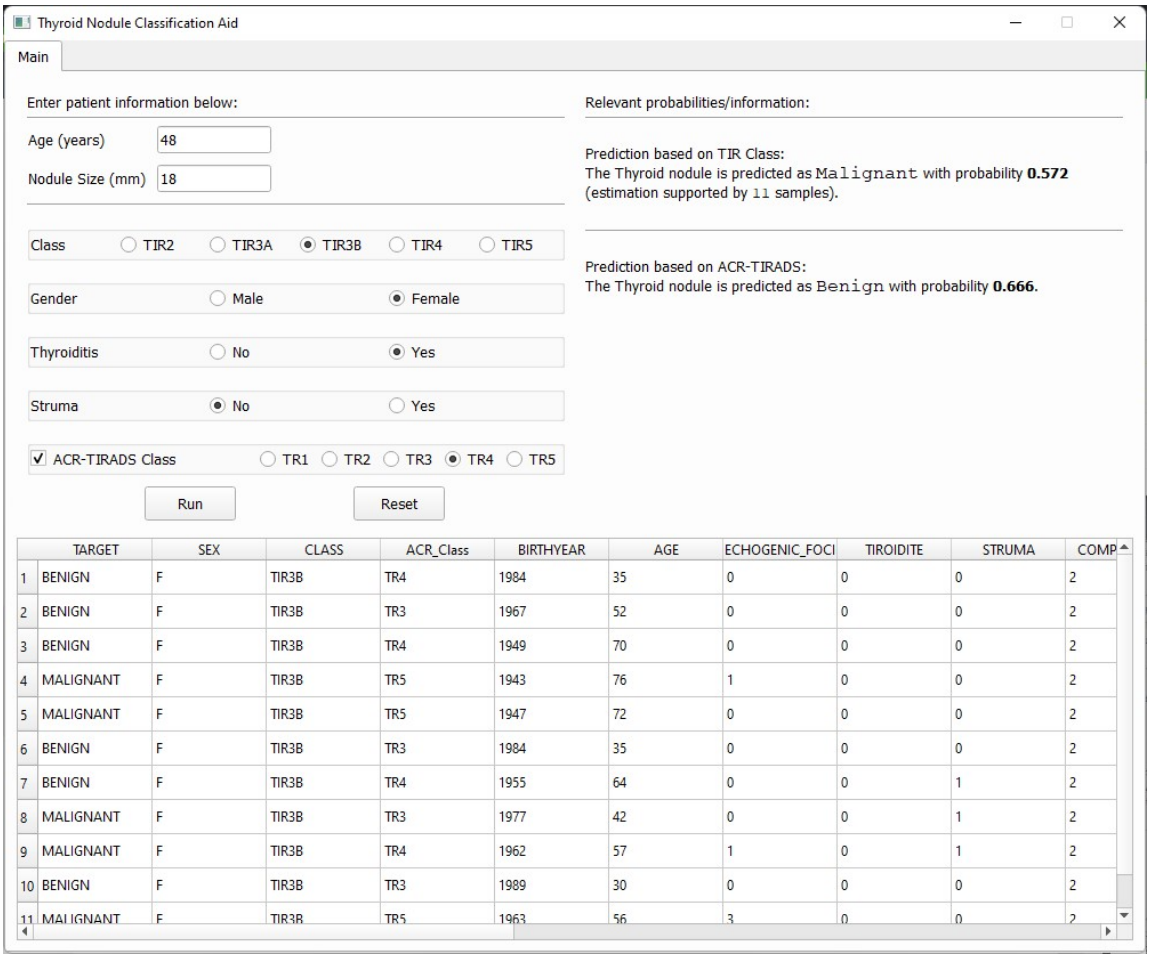
Figure 19: GUI start-up screen. The interface has three components. Select at least a Class/ACR-TIRADS Class to obtain a probabilistic prediction.

incomplete patient profiles cause the algorithm to generate less accurate predictions. Basic error messages are implemented to serve as warnings when unusual values are entered for the continuous variables or patient information is missing. Figures 27a, 27b and 28 (Appendix C) depict these error messages.

5.1.1 Usability for Clinicians

The GUI is aimed at providing decision support on the spot to any healthcare worker. The idea is the following:

- The healthcare worker will be able to interact with the GUI and enters a patient's information;
- A prediction for this patient will be presented, together with the probability of this prediction being true based on the corresponding PT. This is part of the strength of the method, it shows the degree of certainty of its prediction;
- A table is filtered to only show the relevant data points, i.e., the data points on which the prediction has been based. The data points presented in the table will be 'similar' to the patient whose information is entered into the tool. Indirectly, the amount of data points shown in the table serves as a measure of confidence; the larger the number of data points is on which the prediction is based, the more confident the model is about its prediction. Therefore, the GUI explicitly states the number of samples that support the prediction;
- Using the table of similar patients, a healthcare worker can easily identify past patients with a similar profile. This includes all patients that are present in the data set, not only the patients that a healthcare worker has treated. As the data set grows each decision will be further substantiated;



	TARGET	SEX	CLASS	ACR_Class	BIRTHYEAR	AGE	ECHOGENIC_FOCI	TIROIDITE	STRUMA	COMP
1	BENIGN	F	TIR3B	TR4	1984	35	0	0	0	2
2	BENIGN	F	TIR3B	TR3	1967	52	0	0	0	2
3	BENIGN	F	TIR3B	TR4	1949	70	0	0	0	2
4	MALIGNANT	F	TIR3B	TR5	1943	76	1	0	0	2
5	MALIGNANT	F	TIR3B	TR5	1947	72	0	0	0	2
6	BENIGN	F	TIR3B	TR3	1984	35	0	0	0	2
7	BENIGN	F	TIR3B	TR4	1955	64	0	0	1	2
8	MALIGNANT	F	TIR3B	TR3	1977	42	0	0	1	2
9	MALIGNANT	F	TIR3B	TR4	1962	57	1	0	1	2
10	BENIGN	F	TIR3B	TR3	1989	30	0	0	0	2
11	MALIGNANT	F	TIR3B	TR5	1963	56	3	0	0	2

Figure 20: GUI shows prediction for a given patient.

- If the data file linked to the GUI is updated with new patients, the added data points will be considered the next time that the GUI is launched. As the data set increases in size, the model will represent a more complete overview of the patient pool, and its performance will likely increase. The time it takes to train the tree is dependent on the size of the data set and the depth of the tree (i.e., number of features in the tree). For the data set at hand the time to train the tree is negligible.

As discussed before, the GUI makes use of three trees. PT I (TIR class), PT II (TIR & ACR-TIRADS class) and PT III (ACR-TIRADS class). PT I achieves the highest performance, thereafter, PT II. Lastly, PT III achieves the lowest performance. The table that is presented on the bottom of the GUI (Figure 25) does not always show every data point that is used in the calculation of the prediction probability. This is due to the combination of the fuzzy implementation and the way the model handles missing data points (i.e., when there is no total realization in the tree that corresponds to the data point to be predicted). This occurs as sometimes we need to interpolate/aggregate between data points. This can be fixed, but it is slightly advanced for a prototype GUI. Moreover, this problems resolves itself when the data set increases in size, as we will no longer encounter having to interpolate between data points.

5.1.2 Feedback from Clinicians

The decision support solution and the corresponding prototype version of the tool have been presented to clinicians active in the field of thyroid nodule detection, whom are the intended users of the tool. During this meeting several important aspects have been covered, we will address these one-by-one.

Scope and performance of the model The clinicians were interested to see the performance of the model in another cohort and indicated that this would be necessary before the possible implementation of the tool in practice. Another data set will be needed for this, but it may present an interesting direction for future research.
Furthermore, it was indicated by clinicians that the main struggle in thyroid nodule prediction is the identification of the patients that have cancerous nodules and correspond to the TIR3A and TIR3B classes. However, the model performs worst for these classes. In order to improve the model for these cases, substantially more data points are required. There are currently only two patients in the class TIR3A and four patients in the class TIR3B that have been observed to have cancerous thyroid nodules. It is a relevant scope for the tool to possibly focus on in the future, and even separate trees could be introduced for this purpose. However, substantially more data will be required for this.

The tool's design Overall, the clinicians appreciated the tool and its design. The probabilistic statement was naturally understandable, and the table showing the list of patients that were used in the decision making process was helpful and appreciated. The actions needed for filling out the conditions of a patient were natural and easy to perform.

Some thoughts from clinicians to improve the tool:

- Clinician indicated the desire for a clear follow-up plan, i.e., when the probability is below certain threshold then we do follow-up, between certain margins we do a(nother) biopsy, above a certain threshold we should definitely do surgery (thyroidectomy). This means, different decision should be set for different probabilities. However, another clinician contradicted this by stating that every clinician has his/her own beliefs and experiences. When the tool gives a one-size-fits-all plan, this will be unlikely to be followed. Moreover, this is outside the scope of the current research, and its implementation would require thorough domain knowledge;
- It was suggested to allow clinicians to construct their own fuzzy sets. This may only require minor explanations for clinicians to understand the concept of fuzzy sets and its use in this tool. However, it will increase customizeability of the tool and perhaps generalizeability of the methods to other cohorts;
- Lastly, it has been suggested to rank the table of similar patients. Meaning that the patients that were most influential in the prediction process are represented at the top of the table.

5.2 Generalizability of the Tool

The tool is created in the Python 3 programming language and depends on PyQT [66], Pandas [38], NumPy [44], Simpful [62] and the probability tree algorithms as proposed by Genewein et al. [28]. The tool can be launched in Python 3. Although the interface is made for the thyroid nodule implementation, the program is created to allow for customisation towards other purposes. The tool is built according to the Model-View-Controller architecture [14], which enhances the readability and re-usability of the code, favouring the creation of similar tools – based on the same GUI framework – for other purposes. In developing the tool, it was deemed important that the tool would be easily generalizeable to other purposes.

6 Discussion

Clearly the future of medicine lies in the use and development of human/AI interactive intelligence systems that leverage on machine learning techniques that are able to analyse large quantities of data to create decision support systems. Therefore, medical practice everywhere should be leveraging on sharing information and data in order to improve general medical practice. Implementing such systems will reduce the variability of healthcare among doctors, hospitals and countries. Deep learning has achieved expert-level expertise in many medical areas, nevertheless, it is essential to aim for interpretability in these systems. Moreover, several regulations imposed by the EU enforce that interpretability is part of the contract. The GDPR [19] states that: "*... individuals should not be subject to a decision that is based solely on automated processing (such as algorithms) and that is legally binding or which significantly affects them.*". Furthermore, regulations that have been drafted by the EU in the AIA [20] aim to specifically regulate the development and use of AI in safety-critical systems (such as the field of healthcare). Any decision support system active in the clinical field will be subject to a strict set of requirements. Conclusively, interpretability must be part of the deal.

The goal of this master thesis project has been just that; making interpretability part of the deal. The results imply that IAI that mimics human reasoning has a fair shot at supporting clinicians in decision making. Furthermore, allowing the fuzzy, vague and ambiguous medical variables to have fuzzy boundaries will only improve the model and make it even more human-like, while being able to handle large quantities of data. However, more research is needed to identify for which cases the methods work best. As the proposed methods did not achieve consistent results in both case studies presented in this master thesis project. However, the method is successful in it being inherently interpretable, as well as the tool that has been designed to convey its predictions.

7 Conclusion

This last chapter presents the conclusions of this master thesis project. Firstly, the research questions as described in Section 1.1.1 are revisited and answered. Secondly, the relevance of the research, for both the scientific community and the field of healthcare, is discussed. Thirdly, the limitations of this project are discussed. Lastly, suggestions for future research are made.

7.1 Revisiting the Research Questions

Throughout this master thesis research project a novel interpretable and fuzzy AI method is proposed, background knowledge needed for composing such a method is reviewed, the proposed novel AI technique is used for modeling and its performances are evaluated for two case studies, and a tool is developed for the integration of such techniques into the daily workflow of clinicians. As described in Section 1.1.1, the main research question for this research to answer is formulated as:

RQ: *How can we develop a fuzzy probability tree method to support clinical decision-making in a human-comprehensible way?*

To answer this question, four sub-questions are formulated and answered below.

SQ1: *What methods that exist in literature can be built upon to develop a fuzzy probability tree?*

To answer this question, a review of the relevant literature has been conducted. Several concepts underlie the proposed fuzzy probability trees. These include: IAI, probability trees (PTs), causal reasoning and fuzzy set theory. But most importantly, the probability tree causal reasoning framework as proposed by Genewein et al. [28] was the basis of this project. Developing the ultimate fuzzy probability tree required combining causal probability trees and fuzzy set theory. Furthermore, we investigated ways to determine the optimal tree structure for (F)PTs. PTs can be constructed leveraging several different techniques (similar to regular decision trees). The trees may be formulated based on induction and deduction. Induction refers to a tree being created from data, e.g., using data mining techniques or creating an optimal tree structure based on maximizing the information gain at each split. Deduction, on the other hand, bases its tree design on domain knowledge. A combination has been applied in this research project, as the structure of the tree is based on domain knowledge, but the transition probabilities are determined from the data. Both methods have also been leveraged by several authors throughout the literature, and the appropriate (combination of) method(s) will vary across different implementations.

SQ2: *What techniques need to be developed on top of the existing probability tree framework to model fuzzy probability trees?*

The PT framework created by Genewein et al. [28] has been extended on several aspects, namely to allow for the creation of probability trees directly from data, make predictions based on the constructed trees, and incorporate fuzzy set theory into its prediction making methods. Furthermore, this project has introduced methods to calculate the transition probabilities directly from data and has automated the prediction process for an entire data set. The main contribution of this project to the existing PT framework is the inclusion of (linguistic) variables that belong to fuzzy sets, rather than the traditional crisp sets that are naturally used in tree-like structures. To briefly explain, the implementation is such that the PT is created using crisp boundaries (e.g., we divide patients by age using a crisp threshold of 50 years creating two classes 50− and 50+), however, when a prediction is made for a patient of the age 49 this patient will be considered partially in both classes (in accordance to the membership function). Meaning that the prediction of the patient will be based on multiple total realizations (that is; a path from root to leaf), where the contribution of each total realization to the final prediction is directly proportional to the degree of membership of the variable. For example, if a patient of age 49 belongs to the class of 50+ to a degree of 0.90, then the final prediction will be the sum of 0.10× the prediction for 50− and 0.90× the prediction for 50+. This method has been named FPT. Allowing variables to be fuzzy (i.e., ambiguous or vague) brings important nuances, as shown in the example handled in Section 4.1.5.

SQ3: *How does the proposed fuzzy probability tree perform in assisting clinical practice?*

The FPT methods that have been developed are put to the test in two different medical case studies. The two case studies are different in terms of the classifications of diseases as introduced by the ICD. First, a case study centered around identifying thyroid cancer is conducted, which falls under the ICD category of neoplasms [45]. Second, a case study is performed to assess the risk for a chronic kidney disease (CKD) patient to progress to the final stages of the disease (that is; ESRD) in the following 2-year period. CKD falls under the disease of the genitourinary system [45]. The results of the FPT showed that the FPT implementation outperformed the traditional PTs in both case studies. However, in the thyroid nodule case study the differences between the PT and FPT were relatively small, due to the small differences between the models. Only two variables in the tree were fuzzy, and therefore, did not often result in differing predictions. In the CKD case study six of eight variables were fuzzy, and therefore, the performances of the PT and FPT diverged slightly more. Furthermore, the overall performance obtained in the two case studies differed. The performance of the models in the thyroid nodule case study are substantially higher than the results obtained in the CKD case study. This is largely due to the inherent randomness in the progression of the CKD disease in patients. Substantiated by the fact that the benchmark models – deployed to compare the results of the PT and FPT to – also performed significantly less in the CKD case study than they did in the thyroid nodule case. The performance of each model was tested using four performance metrics (accuracy, specificity, sensitivity, precision) and for a large number of bootstraps to report a degree of certainty to the performance of each model. Comparing the FPT to the two benchmark models that are used (logistic regression (LR) and regular decision tree (DT)), the FPT method performed best in terms of sensitivity (for a specific design of the tree in the thyroid nodule case study, and in the CKD case study). In the context of disease detection, sensitivity is arguably the most important metric. However, in most other metrics the LR outperformed the FPT. Therefore, typically the FPT was the second best performing model, despite the FPT performing best in some important aspects. Conclusively, the proposed FPT performs competitively to popular interpretable ML and statistical methods. Although the performance of the FPT may not be desirable yet (for the CKD case), it presents a promising starting point for the methods.

SQ4: *How can fuzzy probability trees be effectively integrated into the workflow of clinicians?*

Clinicians are the intended users of the proposed methods. And generally, clinicians lack a background in machine learning, probability theory and statistics. In order to create an effective tool, able to assist in complex decision making processes, its information must be well conveyed. Therefore, a tool is created with the aim of allowing clinicians to interactively obtain relevant and naturally understandable statements. These statements are written in the form of probabilities, obtained from the underlying probabilistic decision tree that has been generated based on past patient records. Rather than allowing the clinicians insight into the actual tree, it is thought better to create a tool that translates the reasoning of the tree to the clinicians. Therefore, the tool is in the form of a graphical user interface (GUI), designed such that the clinician can interact with the tool to represent a specific patient in real-time, and by changing or filtering the conditions,

changes in the probabilities can be observed. This allows the clinician to produce counterfactual statements by themselves, and see what the probability for malignancy would be in an alternate reality. Furthermore, for each prediction the data set is filtered to show only the data points (i.e., patients) that have been considered in the decision making process. This allows for easy identification of past patients that had a similar clinical profile. This supports clinicians in the complex decision making process, and allows for each decision to be based on a much larger patient data base than can be achieved from the experience of a single clinical practitioner. As the data set grows, each decision is expected to be increasingly accurate as more data will underlie the prediction.

Altogether, these sub-questions help answer the main research question of this project:

RQ: *How can we develop a fuzzy probability tree method to support clinical decision-making in a human-comprehensible way?*

The proposed FPT methods alone are a step towards complying with the increasing need for interpretable decision making in clinical environments. Although, in order to support clinical decision making in a human-comprehensible way, and specifically in a way that is comprehensible to clinicians, an effective and easily manageable tool is required. The tool needs to leverage from the inherent interpretability of the underlying methods, whilst simplifying its predictions and messages in order to be easily implemented into the daily workflow of clinicians. A first prototype for such a tool has been developed during this research project.

7.2 Relevance

The contents of this research present relevant scientific contributions as well as some practical implications.

Scientific contributions There is a growing need in the scientific literature for interpretable AI/ML methods, and this project aims to be a valid addition to this field of research.The methods as proposed by Genewein et al. [28] have been implemented in a real life scenario and put to the test to investigate the appropriateness in clinical decision making. Furthermore, the performance has been compared to existing benchmark models. Moreover, the existing methods have been extended and improved to allow for easy implementation and leveraging the models for prediction making in general and in situations where variables may be vague, ambiguous and fuzzy.

Practical implications Besides the development of an interpretable fuzzy tree model, this research proposes a corresponding tool that is interpretable and comprehensible in conveying the prediction making process to clinicians. The tool is programmed in Python 3 and its relatively easily adjustable framework may serve as the basis for more CDSS tools.

7.3 Limitations

This section contains the limitations of the conducted master thesis project.

Size and imbalance of data set The first two related limitations of the research arise from the data sets used in the case studies. The size of the data sets used is relatively small, but mainly for the thyroid nodule case study the data set is considerably small (Thyroid nodule: 401 cases, CKD: 2.599 cases). Moreover, the data is imbalanced. The majority of the patients belong to the negative class, while the positive class is the most important class to predict accurately. Especially, in the thyroid nodule case study this posed a problem for the accurate prediction of malignant nodules in patients that belong to the TIR3A and TIR3B class. As to the CKD case study, as there is substantial variance in the progression of CKD across patients, more data that is needed for the identification of relevant prediction patterns.

Design of the tree/fuzzy sets Within this research the construction of the probabilistic trees and fuzzy sets has bee (partially) done based on domain knowledge obtained from experts and relevant literature. However, this is up to the human researcher to conclude, and a different design will result in a difference in performance. The performance of the model is dependent on the design of the tree, and it is done according to the beliefs of the researcher/domain experts. Therefore, it is subject to variance and subjectivity.

Feature selection Following on the previous consideration, the features that are used in the trees are chosen based on expert knowledge and relevant literature. Also, the number of features (i.e., the depth of the tree) is chosen by the researcher. It has been attempted to include as many relevant features into the tree as possible, without causing the model to overfit to the data set (which may occur when the number of data points in each leaf is substantially small).

Biased data Both data sets contain bias. Firstly, the thyroid nodule data set has been collected in a clinic in Milano, Italy. Therefore, the vast majority of the patients is Caucasian and lives in the Northern part of Italy. Secondly, the data set for CKD is gathered from several clinics throughout Western Europe, and therefore, consists mainly of Caucasian patients. These selections of patients may not be representative of the populations in other parts of the world, and their models may not generalize well to other cohorts that are pre-dominantly non-Caucasian. All the while, ethnicity has been found to be a counting factor in the clinical environment [27].

Unique combination of features It may occur that a patient is introduced to the tree that is not yet represented in the tree. Meaning that there is not yet a total realization that reflects the patient profile at hand. In this case the models will aggregate between similar data points, in ways described in detail in Section 3.1. However, this entails that when this occurs the patient will be predicted based on patients that are 'similar', however, different in some features. However, as the data set increases in size, the occurrence of a unique combination of conditions will arise less frequently.

Missing value When a data point contains a missing value for which a node exists in the tree, we refer to this as a missing 'condition' in a patient profile. The handling of such missing conditions in the prediction algorithm is done by placing uniform conditional probabilities over every realization leaving the last 'known' node (for more details on implementation, refer to Section 3.1). Therefore, any known conditions that occurs after this missing condition, will not be considered. This implementation is arbitrary and made for the sake of simplicity. Whenever this occurs the GUI will present the user with a warning (as shown in Figure 28 in Appendix C). This has introduced no error in the results, as all data points reflected complete patient profiles. This has solely been implemented to improve the useability of the tool, to ensure it can make predictions even if the patient profile may not be complete.

7.4 Future Research

This section discusses several recommendations for future research.

Missing value The first suggestion continues on the last limitations discussed in the previous section. The current implementation may be improved by introducing methods to only include the realizations that correspond to the full set of conditions for a specific patient. Right now the full potential of a data point containing missing values is not able to be used. However, this will only have any effects if patient profiles are not complete, which is unlikely in practice.

Non-binary target The current implementation of (F)PT can be tested for non-binary classification targets. For instance, a PT may be designed for CKD patients with three outcome variables: no event, ESRD (i.e., kidney failure), death pre-dialysis.

More complex fuzzy sets This research project only considers fuzzy sets with two possible values (e.g., young *vs* old). A point of future research may be implementing more complex fuzzy sets, where fuzzy sets can take more than just two possible values (e.g., young *vs* middle-aged *vs* old). Furthermore, the use of fuzzy reasoning and aggregation functions can be investigated, such that multiple variables may influence the outcome of a fuzzy rule.

Tool upgrade: craft fuzzy sets The tool may be upgraded to support the creation of fuzzy sets inside the GUI. Allowing the clinician to craft his/her own fuzzy sets that will be implemented into the FPT directly, making it convenient for clinicians to change the fuzzy sets and to have them reflect their own beliefs.

Leveraging counterfactual statements The existing framework for (F)PT can be leveraged to research the use of counterfactual statements. And with this the effects of certain interventions on the outcome (target variable) can be tested. For instance, counterfactual statements could be leveraged to identify the effects of RASI drugs on the progression of CKD. Testing alternate scenario hypotheses such as "*Given that this patient has progressed to ESRD, if this patient had taken RASI, what would then be the probability of progressing to ESRD?*". Leveraging counterfactuals in FPTs allows for the integration of human-like reasoning in terms of counterfactual statements, human-like reasoning in terms of fuzzy variables, and large amounts of data.

Temporal data The PT framework as presented by Genewein et al. [28] focuses on causal reasoning, and ideally there must exist some causality between consecutive nodes in a PT. Therefore, a point of future research is the construction of a FPT for temporal data, where the nodes represent stages of the process as actual points in time. For instance, leveraging a FPT for the detection of volcano eruptions as is done in [6].

References

[1] Alfred V Aho and John E Hopcroft. *The design and analysis of computer algorithms*. Pearson Education India, 1974.

[2] Hadi Afandi Al-Hakami, Raneem Alqahtani, Asim Alahmadi, Dakheelallah Almutairi, Mohammed Algarni, and Talal Alandejani. Thyroid nodule size and prediction of cancer: a study at tertiary care hospital in saudi arabia. *Cureus*, 12(3), 2020.

[3] Douglas G Altman and J Martin Bland. Diagnostic tests. 1: Sensitivity and specificity. *BMJ: British Medical Journal*, 308(6943):1552, 1994.

[4] Anna Markella Antoniadi, Miriam Galvin, Mark Heverin, Orla Hardiman, and Catherine Mooney. Development of an explainable clinical decision support system for the prediction of patient quality of life in amyotrophic lateral sclerosis. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, pages 594–602, 2021.

[5] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[6] Willy Aspinall and Roger M Cooke. Expert judgement and the montserrat volcano eruption. In *Proceedings of the 4th international conference on probabilistic safety assessment and management PSAM4*, volume 3, pages 13–18. Springer New York, 1998.

[7] Hakan Başağaoğlu, Debaditya Chakraborty, Cesar Do Lago, Lilianna Gutierrez, Mehmet Arif Şahinli, Marcio Giacomoni, Chad Furl, Ali Mirchi, Daniel Moriasi, and Sema Sevinç Şengör. A review on interpretable and explainable artificial intelligence in hydroclimatic applications. *Water*, 14(8):1230, 2022.

[8] Antonino Belfiore, Giacomo Lucio La Rosa, Gianfranco Antonio La Porta, Dario Giuffrida, Giovanni Milazzo, Lorenzo Lupo, Concetto Regalbuto, and Riccardo Vigneri. Cancer risk in patients with cold thyroid nodules: relevance of iodine intake, sex, age, and multinodularity. *The American journal of medicine*, 93(4):363–369, 1992.

[9] LJ Bessey, NK Lai, NE Coorough, H Chen, and RS Sippel. The incidence of thyroid cancer by fna varies by age and gender. *Journal of Surgical Research*, 2(172):188, 2012.

[10] Alina Beygelzimer, John Langford, Yuri Lifshits, Gregory Sorkin, and Alexander L Strehl. Conditional probability tree estimation analysis and algorithms. *arXiv preprint arXiv:1408.2031*, 2014.

[11] Umang Bhatt, Alice Xiang, Shubham Sharma, Adrian Weller, Ankur Taly, Yunhan Jia, Joydeep Ghosh, Ruchir Puri, José MF Moura, and Peter Eckersley. Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 648–657, 2020.

[12] Phillip G. Bradford, Himadri N. Saha, and Marcus Tanque. Chapter 7 - knowledge representation for causal calculi on internet of things. In Gurjit Kaur, Pradeep Tomar, and Marcus Tanque, editors, *Artificial Intelligence to Solve Pervasive Internet of Things Issues*, pages 125–145. Academic Press, 2021.

[13] Mario P Brito, David A Smeed, and Gwyn Griffiths. Analysis of causation of loss of communication with marine autonomous systems: A probability tree approach. *Methods in Oceanography*, 10:122–137, 2014.

[14] Steve Burbeck. Applications programming in smalltalk-80 (tm): How to use model-view-controller (mvc). *Smalltalk-80 v2*, 5:1–11, 1992.

[15] Jenna Burrell. How the machine 'thinks': Understanding opacity in machine learning algorithms. *Big Data & Society*, 3(1), 2016.

[16] Juan Jesus Carrero, Manfred Hecking, Nicholas C Chesnaye, and Kitty J Jager. Sex and gender disparities in the epidemiology and outcomes of chronic kidney disease. *Nature Reviews Nephrology*, 14(3):151–164, 2018.

[17] Allison Cavallo, Daniel N Johnson, Michael G White, Saaduddin Siddiqui, Tatjana Antic, Melvy Mathew, Raymon H Grogan, Peter Angelos, Edwin L Kaplan, and Nicole A Cipriani. Thyroid nodule size at ultrasound as a predictor of malignancy and final pathologic size. *Thyroid*, 27(5):641–650, 2017.

[18] Ferruccio Ceriotti, James C Boyd, Gerhard Klein, Joseph Henny, Josep Queralto, Veli Kairisto, Mauro Panteghini, IFCC Committee on Reference Intervals, and Decision Limits (C-RIDL). Reference intervals for serum creatinine concentrations: assessment of available data for global application. *Clinical chemistry*, 54(3):559–566, 2008.

[19] European Commission. General data protection regulation, 2016.

[20] European Commission. Laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts, 2021.

[21] Francesco Curia. Cervical cancer risk prediction with robust ensemble and explainable black boxes method. *Health and Technology*, pages 1–11, 2021.

[22] Francesco Curia. Features and explainable methods for cytokines analysis of dry eye disease in hiv infected patients. *Healthcare Analytics*, 1:100001, 2021.

[23] Derek Doran, Sarah Schulz, and Tarek R Besold. What does explainable ai really mean? a new conceptualization of perspectives. *arXiv preprint arXiv:1710.00794*, 2017.

[24] Mary C Frates, Carol B Benson, J William Charboneau, Edmund S Cibas, Orlo H Clark, Beverly G Coleman, John J Cronan, Peter M Doubilet, Douglas B Evans, John R Goellner, et al. Management of thyroid nodules detected at us: Society of radiologists in ultrasound consensus conference statement. *Radiology*, 237(3):794–800, 2005.

[25] Jerome H Friedman and Bogdan E Popescu. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3):916–954, 2008.

[26] Caro Fuchs, Simone Spolaor, Marco S Nobile, and Uzay Kaymak. pyfume: a python package for fuzzy model estimation. In *2020 IEEE international conference on fuzzy systems (FUZZ-IEEE)*, pages 1–8. IEEE, 2020.

[27] T Galm, S Minhas, RJ Cullen, and H Griffiths. Thyroid cancer: is ethnicity relevant? *The Journal of Laryngology & Otology*, 125(8):816–819, 2011.

[28] Tim Genewein, Tom McGrath, Grégoire Delétang, Vladimir Mikulik, Miljan Martic, Shane Legg, and Pedro A Ortega. Algorithms for causal reasoning in probability trees. *arXiv preprint arXiv:2010.12237*, 2020.

[29] Clive WJ Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: journal of the Econometric Society*, pages 424–438, 1969.

[30] Michael E Hall, Jussara M do Carmo, Alexandre A da Silva, Luis A Juncos, Zhen Wang, and John E Hall. Obesity, hypertension, and chronic kidney disease. *International journal of nephrology and renovascular disease*, 7:75, 2014.

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

[32] Arsh K Jain, Ian McLeod, Cindy Huo, Meaghan S Cuerden, Ayub Akbari, Marcello Tonelli, Carl Van Walraven, Rob R Quinn, Brenda Hemmelgarn, Matt J Oliver, et al. When laboratories report estimated glomerular filtration rates in addition to serum creatinines, nephrology consults increase. *Kidney international*, 76(3):318–323, 2009.

[33] Norra Kwong, Marco Medici, Trevor E Angell, Xiaoyun Liu, Ellen Marqusee, Edmund S Cibas, Jeffrey F Krane, Justine A Barletta, Matthew I Kim, P Reed Larsen, et al. The influence of patient age on thyroid nodule formation, multinodularity, and thyroid cancer risk. *The Journal of Clinical Endocrinology & Metabolism*, 100(12):4434–4440, 2015.

[34] Erkki K Laitinen and Teija Laitinen. A probability tree model of audit quality. *European Journal of Operational Research*, 243(2):665–677, 2015.

[35] Biao Leng, Jiabei Zeng, Zhang Xiong, Weifeng Lv, and Yueliang Wan. Probability tree based passenger flow prediction and its application to the beijing subway system. *Frontiers of Computer Science*, 7(2):195–203, 2013.

[36] Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. Explainable ai: A review of machine learning interpretability methods. *Entropy*, 23(1):18, 2020.

[37] Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[38] Wes McKinney et al. pandas: a foundational python library for data analysis and statistics. *Python for high performance and scientific computing*, 14(9):1–9, 2011.

[39] George A Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81, 1956.

[40] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

[41] C Newhall and Rhttps Hoblitt. Constructing event trees for volcanic crises. *Bulletin of Volcanology*, 64(1):3–20, 2002.

[42] Bülent Öcal, Mehmet Hakan Korkmaz, Demet Yılmazer, Tuğba Taşkın Türkmenoğlu, Ömer Bayır, Güleser Saylam, Emel Çadallı Tatar, Sevilay Karahan, and Erman Cakal. The malignancy risk assessment of cytologically indeterminate thyroid nodules improves markedly by using a predictive model. *European thyroid journal*, 8(2):83–89, 2019.

[43] Ann M O'Hare, Andy I Choi, Daniel Bertenthal, Peter Bacchetti, Amit X Garg, James S Kaufman, Louise C Walter, Kala M Mehta, Michael A Steinman, Michael Allon, et al. Age affects outcomes in chronic kidney disease. *Journal of the American Society of Nephrology*, 18(10):2758–2765, 2007.

[44] Travis E Oliphant. Python for scientific computing. *Computing in science & engineering*, 9(3):10–20, 2007.

[45] World Health Organization. International classification of diseases (icd) 10th revision., 2019.

[46] Judea Pearl. *Causality*. Cambridge university press, 2009.

[47] Judea Pearl. The seven tools of causal inference, with reflections on machine learning. *Communications of the ACM*, 62(3):54–60, 2019.

[48] Judea Pearl et al. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19, 2000.

[49] J. Ross Quinlan. Simplifying decision trees. *International journal of man-machine studies*, 27(3):221–234, 1987.

[50] Reza Rahbari, Lisa Zhang, and Electron Kebebew. Thyroid cancer gender disparity. *Future Oncology*, 6(11):1771–1779, 2010.

[51] Howard Raiffa. Decision analysis: Introductory lectures on choices under uncertainty. 1968.

[52] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[53] Gilles Russ, Steen J Bonnema, Murat Faik Erdogan, Cosimo Durante, Rose Ngu, and Laurence Leenhardt. European thyroid association guidelines for ultrasound malignancy risk stratification of thyroid nodules in adults: the eu-tirads. *European thyroid journal*, 6(5):225–237, 2017.

[54] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

[55] Tavpritesh Sethi, Anushtha Kalia, Arjun Sharma, and Aditya Nagori. Chapter 1 - interpretable artificial intelligence: Closing the adoption gap in healthcare. In Debmalya Barh, editor, *Artificial Intelligence in Precision Health*, pages 3–29. Academic Press, 2020.

[56] L Shapley. Quota solutions op n-person games1. *Edited by Emil Artin and Marston Morse*, page 343, 1953.

[57] Colin Shearer. The crisp-dm model: the new blueprint for data mining. *Journal of data warehousing*, 5(4):13–22, 2000.

[58] Edward H Shortliffe and Martin J Sepúlveda. Clinical decision support in the era of artificial intelligence. *Jama*, 320(21):2199–2200, 2018.

[59] R Smith-Bindman, P Lebda, VA Feldstein, D Sellami, RB Goldstein, N Brasic, C Jin, and J Kornak. Risk of thyroid cancer based on thyroid ultrasound imaging characteristics: Results of a population-based study. jama intern med. 2013; 173 (19): 1788-96. pmid: 23978950. pmcid: Pmc3936789, 2013.

[60] Gesheng Song, Fuzhong Xue, and Chengqi Zhang. A model using texture features to differentiate the nature of thyroid nodules on sonography. *Journal of Ultrasound in Medicine*, 34(10):1753–1760, 2015.

[61] Jyoti Soni, Ujma Ansari, Dipesh Sharma, and Sunita Soni. Predictive data mining for medical diagnosis: An overview of heart disease prediction. *International Journal of Computer Applications*, 17(8):43–48, 2011.

[62] S Spolaor, C Fuchs, P Cazzaniga, U Kaymak, D Besozzi, and MS Nobile. Simpful: fuzzy logic made simple. *Int J Comput Intell Syst*, 13(1):1687–98, 2020.

[63] Jungyo Suh, Sangjun Yoo, Juhyun Park, Sung Yong Cho, Min Chul Cho, Hwancheol Son, and Hyeon Jeong. Development and validation of an explainable artificial intelligence-based decision-supporting tool for prostate biopsy. *BJU international*, 126(6):694–703, 2020.

[64] Navdeep Tangri, Lesley A Stevens, John Griffith, Hocine Tighiouart, Ognjenka Djurdjev, David Naimark, Adeera Levin, and Andrew S Levey. A predictive model for progression of chronic kidney disease to kidney failure. *Jama*, 305(15):1553–1559, 2011.

[65] Franklin N Tessler, William D Middleton, Edward G Grant, Jenny K Hoang, Lincoln L Berland, Sharlene A Teefey, John J Cronan, Michael D Beland, Terry S Desser, Mary C Frates, et al. Acr thyroid imaging, reporting and data system (ti-rads): white paper of the acr ti-rads committee. *Journal of the American college of radiology*, 14(5):587–595, 2017.

[66] Phil Thompson. Pyqt5 – pypi.

[67] Stephen MS Ting, Harikrishnan Nair, Irene Ching, Shahrad Taheri, and Indranil Dasgupta. Overweight, obesity and chronic kidney disease. *Nephron Clinical Practice*, 112(3):c121–c127, 2009.

[68] Zhihong Wang, Chirag M Vyas, Olivia Van Benschoten, Matt A Nehs, Francis D Moore Jr, Ellen Marqusee, Jeffrey F Krane, Matthew I Kim, Howard T Heller, Atul A Gawande, et al. Quantitative analysis of the benefits and risk of thyroid nodule evaluation in patients 70 years old. *Thyroid*, 28(4):465–471, 2018.

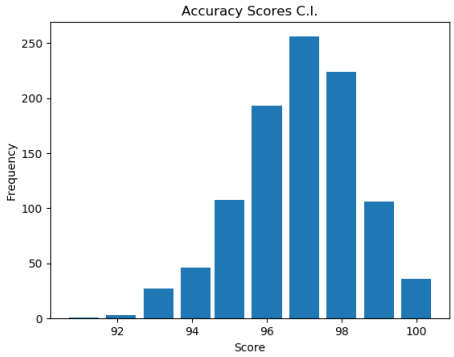
[69] Yan Xie, Benjamin Bowe, Ali H Mokdad, Hong Xian, Yan Yan, Tingting Li, Geetha Maddukuri, Cheng-You Tsai, Tasheia Floyd, and Ziyad Al-Aly. Analysis of the global burden of disease study highlights the global, regional, and national trends of chronic kidney disease epidemiology from 1990 to 2016. *Kidney international*, 94(3):567–581, 2018.

[70] Lotfi A Zadeh. Information and control. *Fuzzy sets*, 8(3):338–353, 1965.

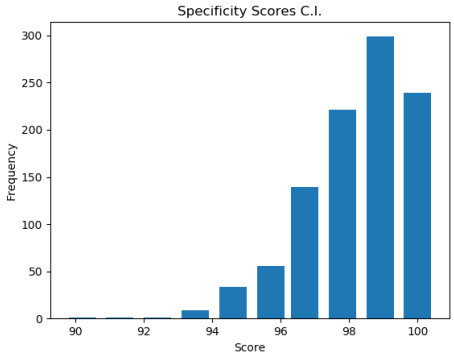
[71] Lotfi A Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on systems, Man, and Cybernetics*, (1):28–44, 1973.

[72] Wen-Yuan Zhu, Wen-Chih Peng, Chih-Chieh Hung, Po-Ruey Lei, and Ling-Jyh Chen. Exploring sequential probability tree for movement-based community discovery. *IEEE Transactions on Knowledge and Data Engineering*, 26(11):2717–2730, 2014.

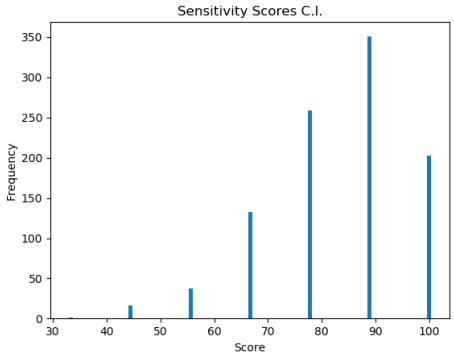
Appendix A Bootstrap Histograms Thyroid Case Study



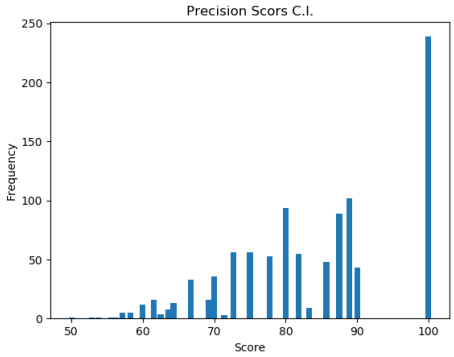
(a) Bootstrap histogram: Accuracy.



(b) Bootstrap histogram: Specificity.

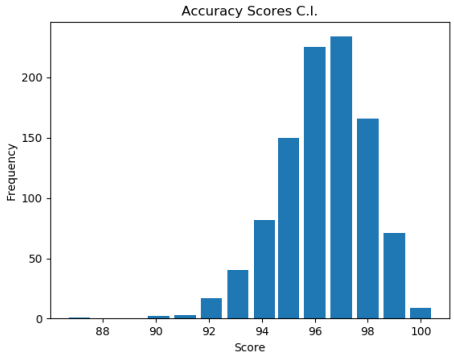


(c) Bootstrap histogram: Sensitivity.

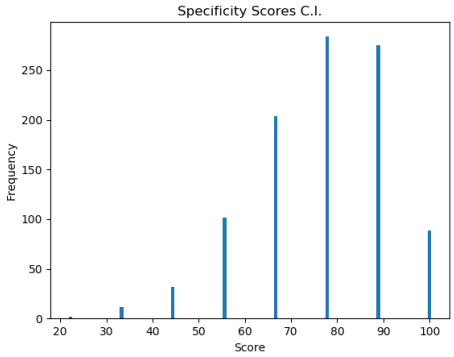


(d) Bootstrap histogram: Precision.

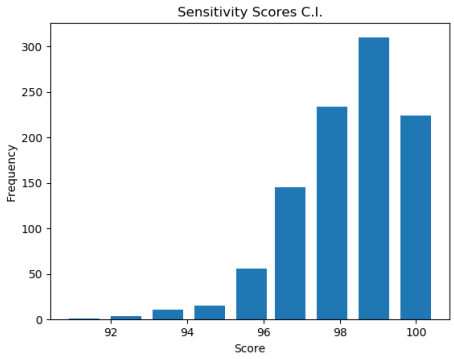
Figure 21: The bootstrap distributions for the four performance metrics. The histograms for accuracy and specificity resemble a normal distribution. Due to the low number of positive samples in the data set, the sensitivity and precision have great variance. However, it begins to approach a normal distribution.



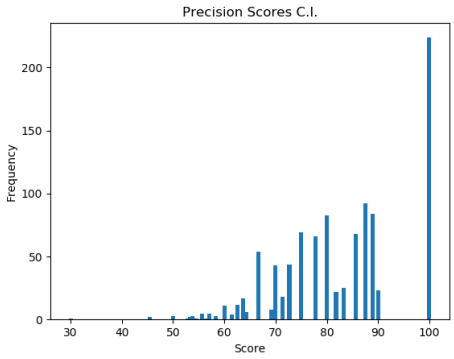
(a) Bootstrap histogram: Accuracy.



(b) Bootstrap histogram: Specificity.



(c) Bootstrap histogram: Sensitivity.



(d) Bootstrap histogram: Precision.

Figure 22: The bootstrap distributions for the four performance metrics. The histograms for accuracy and specificity resemble a normal distribution. Due to the low number of positive samples in the data set, the sensitivity and precision have great variance. However, it begins to approach a normal distribution.

Appendix B Data Exploration CKD

B.1 Nephrology: GFR *vs* Age

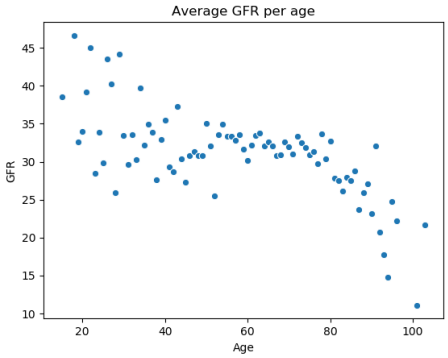


Figure 23: Relationship between Glomerular Filtration Rate (GFR) and age.

B.2 Histogram Serum Creatinine

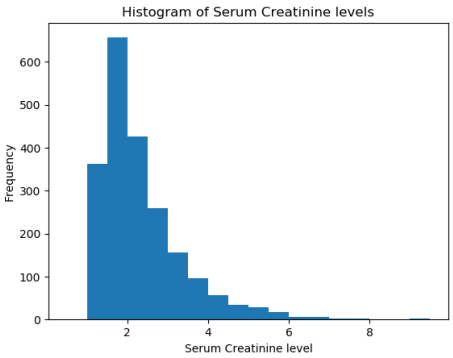


Figure 24: Histogram visualizing the distribution of serum creatinine levels in the data set.

Appendix C Graphical User Interface

	TARGET	SEX	CLASS	ACR_Class	BIRTHYEAR	AGE	ECHOGENIC_FOCI	TIROIDITE	STRUMA	COM
391	MALIGNANT	F	TIR5	TR4	1957	62	0	1	0	2
392	MALIGNANT	F	TIR5	TR4	1931	88	0	0	0	2
393	MALIGNANT	M	TIR5	TR5	1954	65	1	0	0	2
394	MALIGNANT	F	TIR5	TR4	1960	59	0	0	0	2
395	MALIGNANT	M	TIR5	TR4	1951	68	0	0	1	2
396	MALIGNANT	F	TIR5	TR5	1963	56	0	1	0	2
397	MALIGNANT	F	TIR5	TR5	1936	83	3	0	0	2
398	MALIGNANT	M	TIR5	TR5	1987	32	0	1	0	2
399	MALIGNANT	F	TIR5	TR5	1965	54	3	0	0	2
400	MALIGNANT	M	TIR5	TR5	1969	50	3	0	0	2
401	MALIGNANT	M	TIR5	TR3	1969	50	0	0	0	2

Figure 25: The table in the GUI visualizes all patients in the data set when no patient information is entered. When patient information is entered, the table will be filtered to show only the data points on which the prediction is based. Meaning that it shows the patients that are in one of the total realizations in the PT that are used in the making of the prediction.

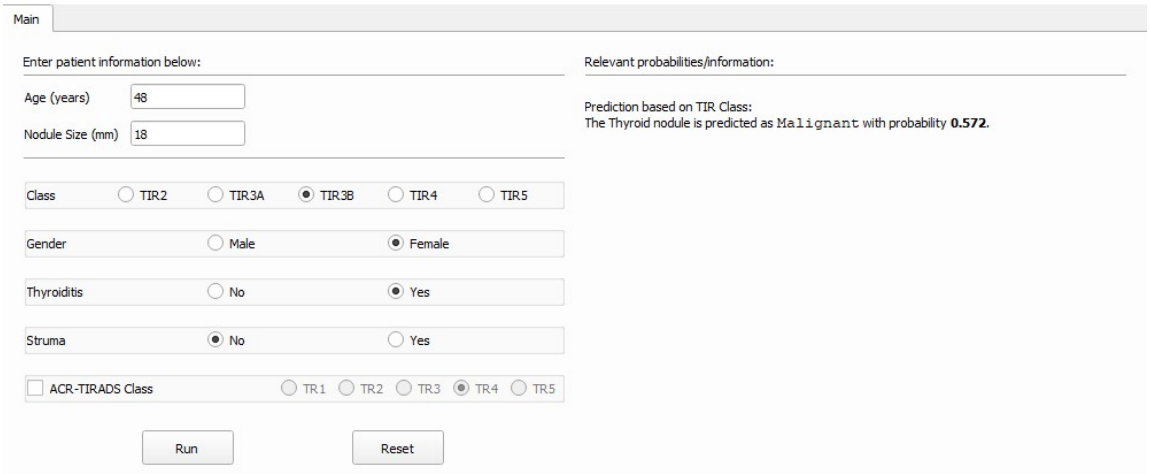
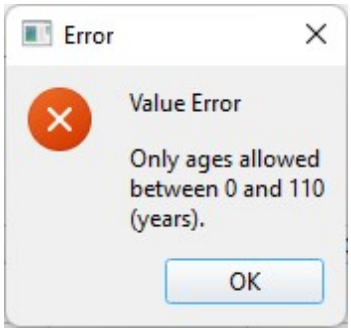
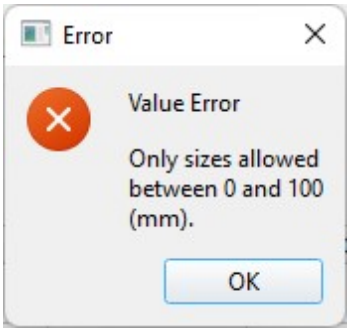


Figure 26: After entering patient information, and hitting the 'Run' button, the GUI will calculate the predictions. Entering a class for the ACR-TIRADS classification system is optional, and can be turned on and off by clicking the check-box. When the check-box is switched off, only a prediction based on the TIR class is presented. Hitting the 'Reset' button will empty out all fields containing information.



(a) GUI Error message: age.



(b) GUI Error message: nodule dimensions.

Figure 27: GUI Error messages to constraint the values that can be filled in on the GUI. This will assist in avoiding mistakes made when filling in the values.

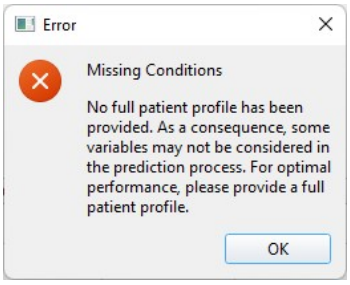


Figure 28: GUI Error message to warn the user when conditions are missing.