Oriol Villaró Serrano

EXPLAINABILITY OF FUZZY RANDOM FORESTS BY MEANS OF A DOMINANCE ROUGH SET APPROACH

FINAL DEGREE PROJECT

Directed by Aïda Valls Mateu,
Najlaa Maaroof Wahib Al-Ziyadi
and Marcin Szeląg

Degree in Computer Engineering

UNIVERSITAT ROVIRA I VIRGILI

Tarragona

2021

Resum.

La retinopatia diabètica (DR) és una de les complicacions més esteses de Diabetus Mellitus, amb una incidència creixent. El seu símptome principal és la ceguesa, i té un gran impacte a nivell econòmic als sistemes de salut.

La detecció precoç de DR, a través d'un control periòdic dels símptomes, redueix significativament el cost financer dels tractaments i disminueix el nombre de pacients que desenvolupen ceguesa.

És difícil assolir una periodicitat breu en els controls degut al gran nombre de persones diabètiques, a la manca de recursos i al cost de les proves de detecció. És per això, que s'han utilitzat diferents mètodes de Machine Learning (ML) per predir quins pacients tenen més risc de desenvolupar la malaltia i poden beneficiar-se de proves més freqüents.

Molts d'aquests mètodes basats en ML es poden entendre com caixes negres, de manera que la construcció de sistemes que aportin explicabilitat a aquests models per justificar les seves decisions és un tema rellevant per a la investigació.

En aquest projecte utilitzem un Dominance-based Rough Set Approach (DRSA) per tal de justificar les decisions d'un model de ML basat en Fuzzy Random Forests per a la predicció de DR.

Resumen.

La retinopatía diabética (DR) es una de las complicaciones más extendidas de Diabetus Mellitus, con una incidencia creciente. Su síntoma principal es la ceguera, y tiene un gran impacto a nivel económico a los sistemas de salud.

La detección precoz de DR, a través de un control periódico de los síntomas, reduce significativamente el coste financiero de los tratamientos y disminuye el número de pacientes que desarrollan ceguera.

Es difícil lograr una periodicidad breve en los controles debido al gran número de personas diabéticas, a la falta de recursos y al coste de las pruebas de detección. Es por ello, que se han utilizado diferentes métodos de Machine Learning (ML) para predecir qué pacientes tienen más riesgo de desarrollar la enfermedad y pueden beneficiarse de pruebas más frecuentes.

Muchos de estos métodos basados en ML se pueden entender como cajas negras, por lo que la construcción de sistemas que aporten explicabilitat a estos modelos para justificar sus decisiones es un tema relevante para la investigación.

En este proyecto utilizamos un Dominance-based Rough Set Approach (DRSA) para justificar las decisiones de un modelo de ML basado en Fuzzy Random Forests para la predicción de DR.

Abstract.

Diabetic Rethinopathy (DR) is one of the more widespread morbidities of Diabetus Mellitus, with increasing incidence. It's main effect is blindness, with a large economic impact in healthcare.

The early detection of DR, by means of periodic screening and control, reduces significantly the financial cost of the treatments and decreases the number of patients who develop blindness.

Short periodicity is hard to achieve due to the large number of diabetic people, the lack of resources and the cost of screening test. For these reasons, different Machine Learning (ML) approaches have been used to predict which patients are at most risk and can benefit from more frequent screenings.

Many of these ML approaches may be considered as black boxes, so the construction of appropriate explanations for the decisions in those models is a relevant topic for research.

In this project we use a Dominance-based Rough Set Approach in order to justify the decisions of a ML model based on Fuzzy Random Forests for the prediction of DR.

Index

Index of Tables

Figure index

1. Introduction

This work is part of a research project between ITAKA research group at URV and the Ophtalmology department of Hospital Universitari Sant Joan de Reus. The main goal of this research project is to develop a clinical decision support system that assists the doctors in the early detection of signs of a disease called Diabetic Retinopathy.

Diabetic retinopathy (DR) is the most common complication of diabetes mellitus (DM). DR remains a leading cause of sight-loss among the working age population of industrialised regions. As of 2010, DR affected over 100 million patients worldwide, and is expected to affect mora than 190 million people by 2030 [1].

This project started about 8 years ago. It has been continuously funded by the Spanish Instituto de Investigación Carlos III. Currently, a software called Retiprogram is on testing at the hospital. This software uses information from the Electronic Health Care Record of a diabetic patient in order to predict the possibility of developing DR. This software was constructed using Machine Learning methods and it is based on Fuzzy Logic and Fuzzy Rules. It will be described in more detail later in this document.

Being Retiprogram a tool to be used by family doctors, who are not specialists in DR disease, it become a need to include some kind of automatic explanation about the prediction made by the system. This is the motivation of the work done in this Final Degree Bachelor Thesis.

1.1. Diabetic retinopathy

Diabetic Retinopathy (DR) is largely asymptomatic in its early stage, which is called non-proliferative diabetic retinopathy (NPDR). During this stage, different retinal pathologies can be detected, such as microaneurysms, hemorrhages and hard exudates. DR and diabetic macular edema (DME) are the major sight-threatening endpoints of diabetes. DME happens when blood vessels leak their content into the macular region, and can occur at any point of NPDR, causing blurred vision and darkened/distorted images, asi it can be seen in Figure 1.

Figure 1: Comparison between normal vision and suffering from DR

The second stage of DR is proliferative diabetic retinopathy (PDR) and is characterized by neovascularization. During this advanced stage, patients may suffer from severe vision impairment by vitreous hemorrage or when tractional retinal detachment is present [2]. The degree of vision impairment varies greatly, and it can go from the blood clearing in a few days, to the most extreme cases, where the patient may only be able to tell light from dark in that eye.

1.1.1. Usual diagnosis procedure

The diagnosis of DR is made by clinical manifestations of vascular abnormalities in the retina. Due to the asymptomatic nature of NPDR, its diagnosis requires regular screening for patients with diabetes in order to detect and manage the condition [2]. The initial diagnosis of DR can be based on changes in electroretinography (ERG), retinal blood flow and retinal blood vessel calibre [3] , but in practice the first clinical features of DR are evident in an ophtalmoscopy or eye fundus photography [4], which is included during a rotuine ophtalmologic examination worldwide.

Early signs of the disease include leaking blood vessels, retinal swelling, fatty diposits on the retina (exudates) and damaged nerve tissue (neuropathy), some of which can be seen in Figure 2. Since diabetes impairs the microcirculation and function of vital tissues throughout the body, alterations in the microvascular hemodynamics can be useful to diagnose and screen DR. Some of the descriptors used are vessel diameter, blood velocity, blood flow, wall shear ratea and wall shear stress [5].

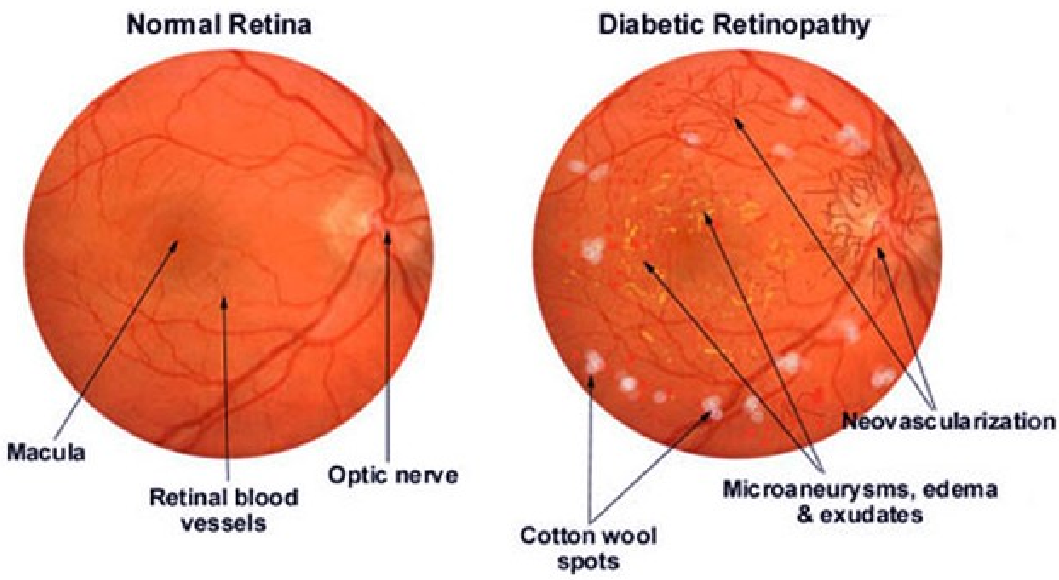


Figure 2: Comparison between a normal retina and one suffering from DR

This image-based screening is essential to detect cases that require examination and treatment to avoid permanent visual loss. The intervals of this screening tests among diabetic patients remains an area of debate, with different healthcare systems offering different solutions, in order to find a good cost-efectiveness ratio, although resources for nationwide screening programmes are scarce in many countries [6]. In Spain the test is made every two years, as it is not possible to make it more frequently, due to the lack of trained specialists and the resources needed in the hospitals.

The work in this project will center on the case of Diabetic Retinopathy. However, the method and its implementation could be used in other fields of application.

1.2. Machine learning

Machine learning (ML) is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy [7]. ML algorithms build a model based on sample data, known as «training data», in order to make predictions without being explicitly programmed to do so.

1.2.1. ML Approaches

Traditionally, there are two categories into which all ML approaches fall into, depending on the type of feedback available to the learning algorithm:

Supervised learning:

In supervised learning algorithms, the training set available contains both the inputs and the outputs, alsko known as «signal» or «feedback». The algorithm will build a mathematical model where each training example will be represented by an array and the training data is represented by a matrix. Through iterative optimization the algorithm calculates a function that can help predict the output for a set of inputs that were not part of the training set. Random forests, and subsequently Fuzzy random forests (FRF), like the ones used in this project, fall into this category.

Unsupervised learning:

Unsupervised learning algorithms take a training set of data without any label associated to each object. These methods aim to identify commonalities in this unlabeled data. The algorithm will judge all new inputs by the presence or absence of the commonalities it has detected in the training set [8].

The Retiprogram decision support System has been constructed using a supervised Machine learning method known as Fuzzy Random Forests. More details will be given in chapter 2.

1.2.2. Explainability in ML systems

Model explainability is one of the most important problems today in ML, and is one of the key aspects of this project. «Black box» models such as deep neural networks can run critical systems without even the developers knowing the reasons behind the decisions the algorithm is making. Different methods exist to shed some light on why a ML model makes a decision, or even which features have the most weight in a prediction, some of which we will introduce here [9] (Figures also taken from [9]

Partial dependence plots:

The objective of a partial dependence plots (PDP) is to show the effect of a feature on the outcome of a ML model. To do that we isolate the features that we are interested in their effect in the prediction into a partial function that only depends on the st of features w want to know.

PDP assumes that the features we marginalize are not correlated with the set we want to know the outcome of. If this assumption is not correct the average calculated for the PDP will include unlikely or impossible data points.

In Figure 3 we can see an example of PDPs for th prediction of bycicle rental, with the factors being plotted are temperature, humidity and wind speed. With these plots it can be observed that the feature that has the biggest effect is the temperature: up until 20ºC the hotter it gets the more bikes will get rented. On the x-axis of the plot we can see the data distribution.

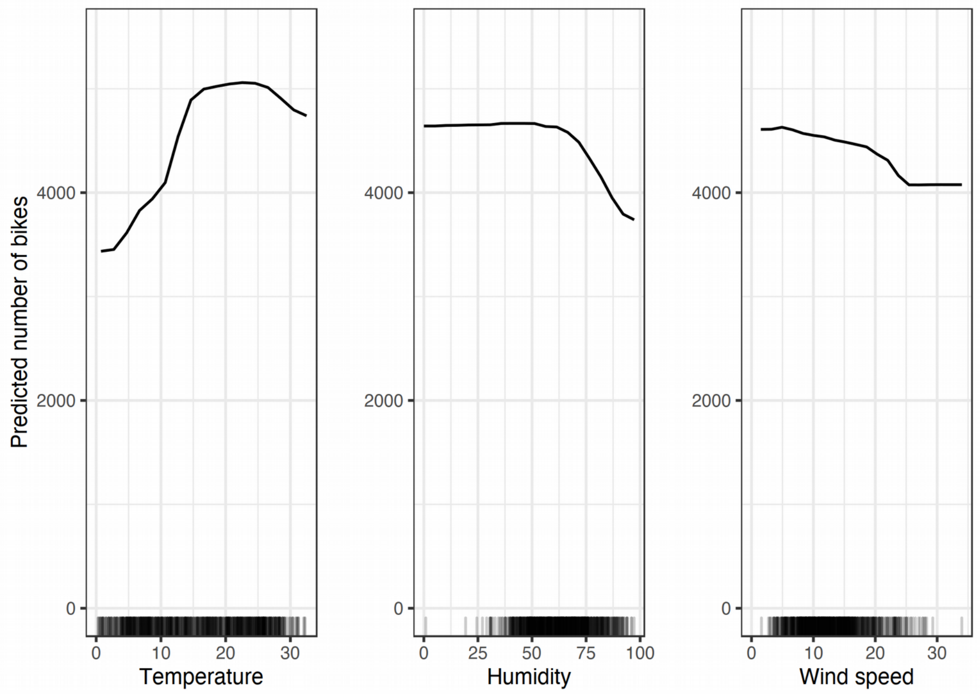


Figure 3: Partial dependence plots example

Permutation feature importance:

Permutation feature importance measures the increase in the prediction error of the model after we permuted th feature’s values, which breaks the relationship between the feature and the true outcome. With this method, a feature is considered as «important» if shuffling its value incrases the model error, since the model relied on that particular feature to make the prediction. If this model error does not incrase in the particular case, the feature is considered as «uninportant».

In Figure 4 we can see an example of permutation feature importance, keeping with the last situation of bike rental. After calculating and sorting the features by the error value they produce aftr being permutated, we can see that the model sees temperature and days since 2011 as the most important features.

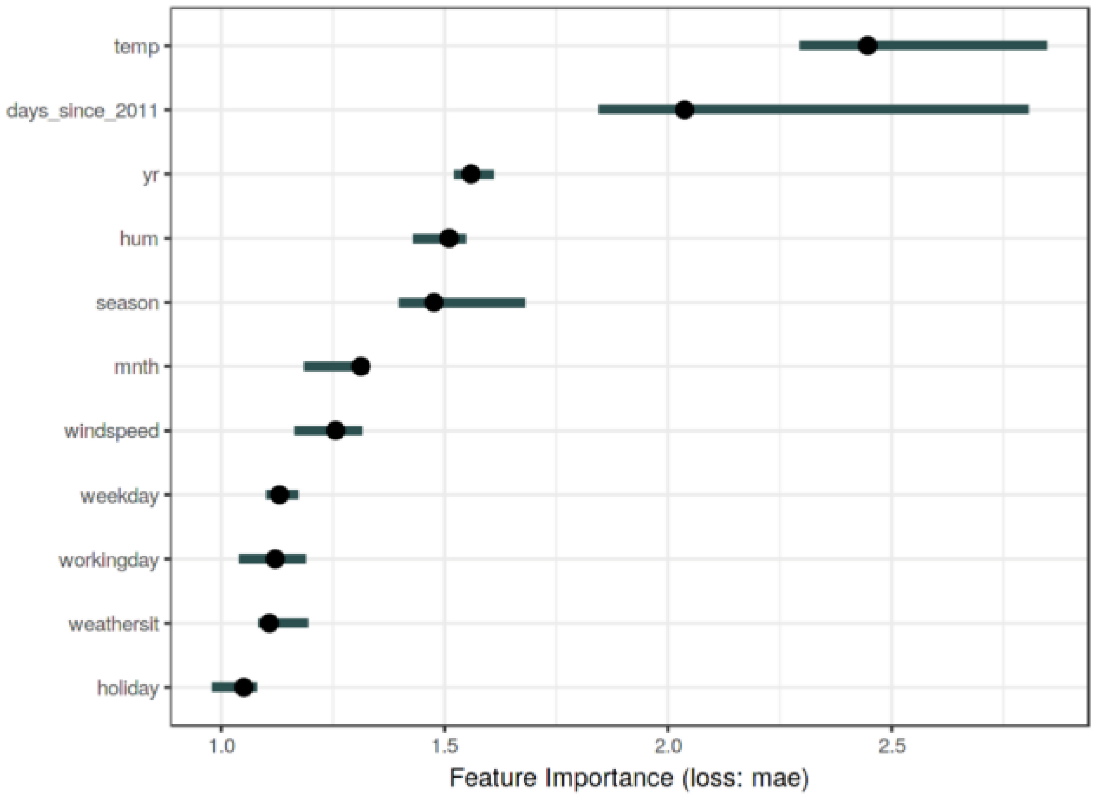


Figure 4: Permutation feature importance plot example

Accumulation local effects plot:

Accumulated local effects (ALE) [10] describe how features influence th prediction of a machine learning model on average. ALE plots are faster and unbiased when compared to PDPs. ALE plots fix the issue PDPs have when the features are correlated, by calculating the differnce in predictions instead of averages. Instead of calculating what the model predicts on average when each data instance has a value v for a specific feature, even if it does not make sense, ALE plots calculate how the model predictions change in a small «window» of the feature aroung the value v for data instances in that window.

Individual conditional expectation:

Individual conditional expectation (ICE) plots dispaly one line per data instance, which shows how the instance's prediction changs when a feature changes. A PDP is the average of all the lines of an ICE plot. The advantage ICE plots have over the global method of PDP is that PDPs can overlook obscure heterogneous relationship created by interactions, which can get lost in the average of all other data.

In Figure 5 we can see an example of 3 ICE plots, which use the same data as the PDP example. Since all the curves seem to follow the same course, we can conclude that there are no furthr obvious interactions, and that th PDP is already a good summary for the explainability of the model.

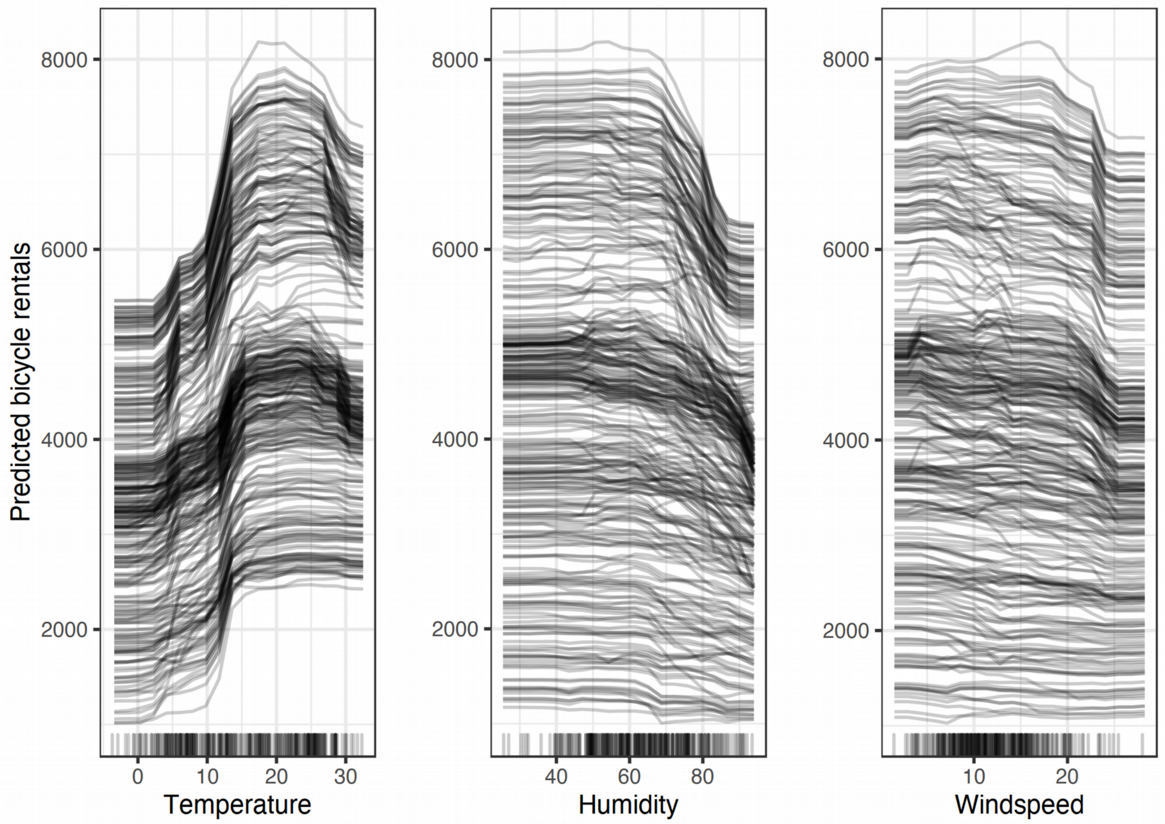


Figure 5: ICE plots example

Surrogate models:

Surrogate models are interpretable models (i.e., decision trees or linear models) that are trained to approximate the prediction of a more complex black box model. By interpreting the surrogate model's prediction, we can better understand the black box model's decision.

In this work, we have focused on this last approach: surrogate models. The complex decision model to explain is Retiprogram, formed by hundreds of trees, each one with hundreds of fuzzy rules. A surrogate model is constructed each time that a new instance has been classified with the Retiprogram, in order to have a simple model that explains the outcome obtained for a certain input case. The surrogate model will consists on a set of decision conjunctive rules of the form:

If attribute1 <= valueX **and** attribute2 > valueY **and** ... **then** class = C

1.3. Project motivation and objectives

The main aim of this project is to generate a small number of rules that explains the reasons why the Retiprogram has assigned the label 0 (no risk of DR) or 1 (risk of DR) to a certain patient.

ITAKA group is studying the generation of explanations with a method based on the generation of a set of "fictive" people similar to the patient studied (called neighbours). From this set of neigbours we need to find rules that differentiate the neighbours of class 0 from the ones of class 1.

The novelty of this project is that we will study if the method kown as Dominance Rough Set Approach (DRSA) is capable of generating good rules for explanation. These rules should help doctors and patients to understand the factors that have a bigger impact for the presence or abscence of DR in each individual case.

To achieve this objective, the following tasks will be performed:

1. Develop an algorithm to generate neighbours for a given patient that belong to the same predicted class..
2. Find an exemple of an instance of the opposite class to the one assigned to the patient.
3. Generate neighbours for the opposite instance that belong to the same class.
4. Use a knowledge base to fuzzify the values of all the neigbours created.
5. Use a DRSA to extract the rules for each of the patients.
6. Validate the rules generated with DRSA in comparison to other ways of generating rules.

2. Data overview and requirements

In this chapter of the document we will introduce some the major parts of the system, starting from the data to use and the requirements of the system to develop.

2.1. Data about diabetic patients

The data used in this project is that of the training set for the Retiprogram. This data accounts patients of type 2 DM (T2DM), defined by the following risk factors, and their diagnosis of DR [11] :

- Current age
- Biological sex
- Body mass index (BMI)
- Duration of T2DM
- Type of treatment of T2DM
- Control of arterial hypertension
- Hemoglobin A1c (HbA1c) level
- Estimated glomerular filtration rate, calculated from plasma creatinine using chronic kidney disease epidemiology collaboration equation (CDK-EPI equation)
- Microalbuminuria

In order to use this information, a file with metadata was supplied, referred from this point on as the knowledge base. The knowledge base contains the following attributes of all the risk facotrs mentioned:

- **Name**
- **Type:** the two types of features from the knowledge base are *numerical* and *categorical.* Numerical features can be increased or discreased, while categorical features must be changed according to the situation.
- **Minimum and maximum**
- **Step:** the value that must be added or removed when changing numerical features. For categorical features it is present as one, although it is never used.
- **Categories:** each categories defined by the values. It includes the name of the ctagory for explainability, and due to the fuzzy method of the algorithm it also defines the two limits for all fuzzy categories.

2.2. General functional requirements

For the design of the explainability method for Retiprogram, there are two main general requirements:

2.2.1. Reduced number of rules

One of the main requirements of this project is that the number of rules it generates must be low enough in order to help people understand the reason behind the FRF model's prediction. FRF already predict based on thousand of rules, but this is not intuitive to human beings, so this DRSA-based approach is required to be explainable, that is, to present its result in a way that is easy to understand. We also want to generate rules with a short number of conditions, which make them easier to be interpreted by the users.

2.2.2. Numerical and fuzzy versions

The rules should consider two diferent formats: numerical and fuzzy. The data of the diabetic patients [11] is in a numerical form, however Retiprogram transforms them into fuzzy lingüístic labels by means of membership functions defined by the experts.

We will constuct a system that can work in both modalities. This is done with the objective to see wether or not the same rules will be generated, if the number of rules will remain the same or if one method will yield stronger rules than the other.

2.3. System requirements

The requirements that were specified before the design and implementation of the explainability system are the following ones:

1. Generate a certain number of neighbours from an initial given patient by changing some of the values of his/her risk factors. The number will be an input parameter.
2. The neigbours must be similar enough to the initial patient.
3. Distinguish between neigbours that belong to the same predicted class for the initial patient by the Retiprogram, and the ones of the opposite class.
4. The number of neighbours of each class must be the same.
5. The change of the values must be done automatically using information from a knowledge base that defines the attributes' scales and direction
6. The system should be able to transform the numerical values into linguistic terms using some predefined fuzzy sets.
7. The DRSA method must be used to generate rules.
8. Some confidence scores about the quality of the rules must be obtained.
9. The rules must be saved in text files.
10. The programs generated must be applicable to other datasets.
11. A user guide will be written for the proper future use of the programs generated.

3. System design

In this chapter I present the architecture of the software system that has been constructed. Its components are also explained in the different sections, separating the ones that have been designed for this project from the existint ones.

3.1. System architecture

In Figure 6 we can see a diagram of the whole process that takes as input the data about a patient ang generates rules as a result. This process uses two existing programs: Retiprogram[1] and RuleStudio[2]. My work consisted in the design and implementation of the

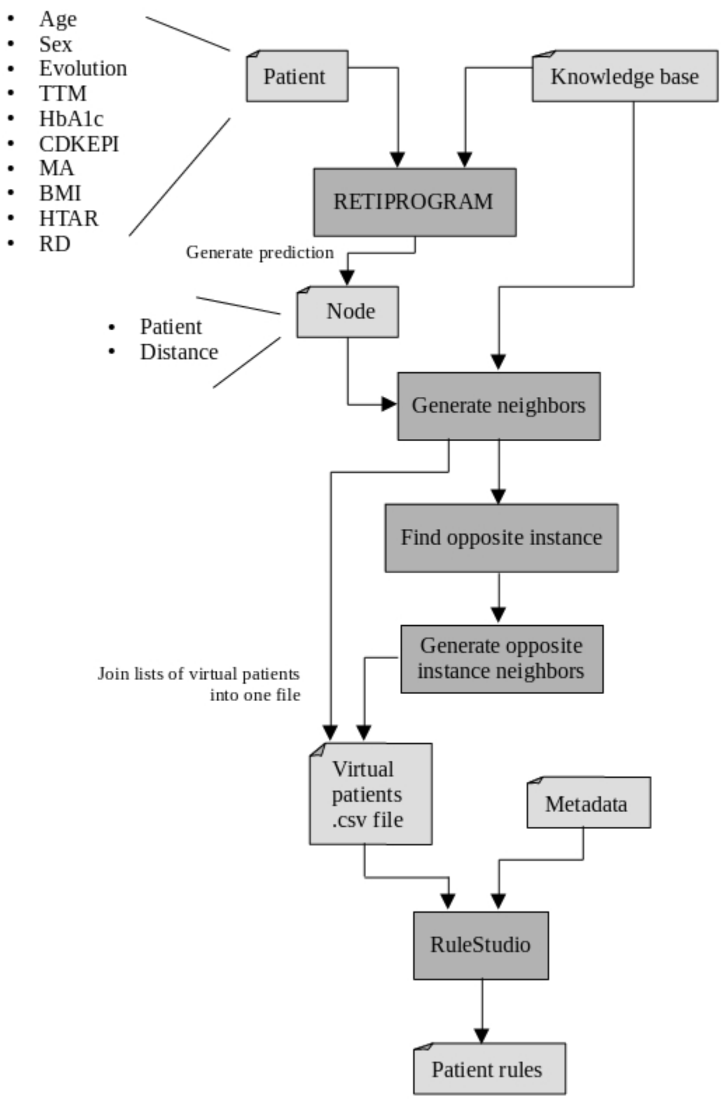


Figure 6: Algorithm pipeline

1 **Retiprogram:** http://www.neverblind.ai/
2 **RuleStudio:** https://github.com/dominieq/rule-studio

rest of the modules in this diagram, related to neighbours generation and to the preparation and processing of input and output files that are required to run Retiprogram and RuleStudio.

3.2. Retiprogram

Retiprogram is a Clinical Decision Supposrt System that helps to discriminate between diabetic people that may develop DR and others who do not have any risk. This computer program focuses on patients that suffer from T2DM and uses 9 input risk factors that can be obtained from the Electronic Health Record of the patient, or introduced by a physician. Then, the system uses a set of fuzzy classification rules and a procedure of aggregation to determine the degree of risk of developing DR.

The users of this system consist mostly of family physicians or ophtalmologists, who receive a prediction of DR risk together with a degree of confidence for the result. Given that information, the system is able to calculate the most appropriate date for the next control visit of each patient. The system also helps to guarantee that human, material and economic resources are efficiently employed.

Retiprogram has been constructed with with Artificial Intelligence techniques and is the result of an extensive multidisciplinary research since 2007, including ophtalmologists' experts in retina [11]–[15].

This project uses the Retiprogram as the ML algorithm that predicts the possibility of a patient developing DR, and it works using FRF.

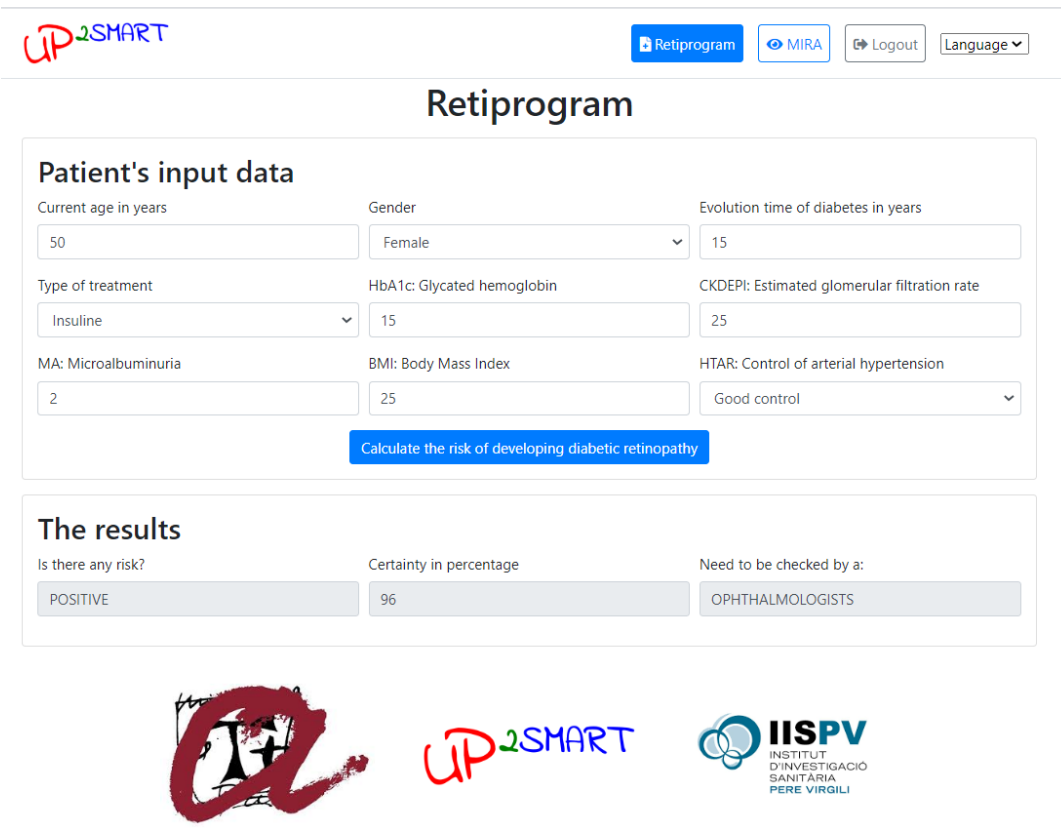


Figure 7: Screenshot of the Retiprogram web interface

3.2.1. Fuzzy random forests

To understand the ML used to develop the classification model that this project is based on, Retiprogram, we must first introduce some concepts.

A decision tree is a schematic tree-shaped diagram used to determine a course of action or to show a statistical probability. The attributes are used to create partitions of a set of examples. The tree nodes are the names or identifiers of the attribute associated to the node. The leaf nodes are sets of classified examples . The tree structure shows how one choice leads to the next, and the use of branches indicates that each option is mutually exclusive, which means that two options cannot occur at the same time.

The strategy to build the tree consists of selecting, in each momentm the attribute that is potentially more useful for the classification, meaning the attribute that will help in the creation of a better tree from that moment on.

Once the tree is built, when an object is received for classification, we must start from the root node to the leaf node that will classify the object. The process of going from one node to one of its childs is recursive, and since the options are mutually exclusive, once a node has been selected the others can be discarded.

Decision trees as a model are not perfect. One of their weaknesses is their tendency to extract some residual variation from the training data, i.e. the noise, and understand it as underlying model structure. Because of that, while the model may perform very well with this training set, it may fail to fit additonal data or predict future observations reliably [16]. The model used in this project uses random decision forests instead, which usually outperform decision trees.

Random forests randomly choose samples from the training data and then build a decision tree for each of the samples. Ones all the trees have been generated, we average the predictions of each tree to come up with a final model, as we can observe in Figure 8. By doing this, the final model typically has less variability and results in a lower test error rate, although this comes at the price of transparency. While decision trees were easy to understand, since the outcome only depended on a single diagram, random forests depend on the average of all the structures created. As such, they are frequently referred as «blackbox» models.

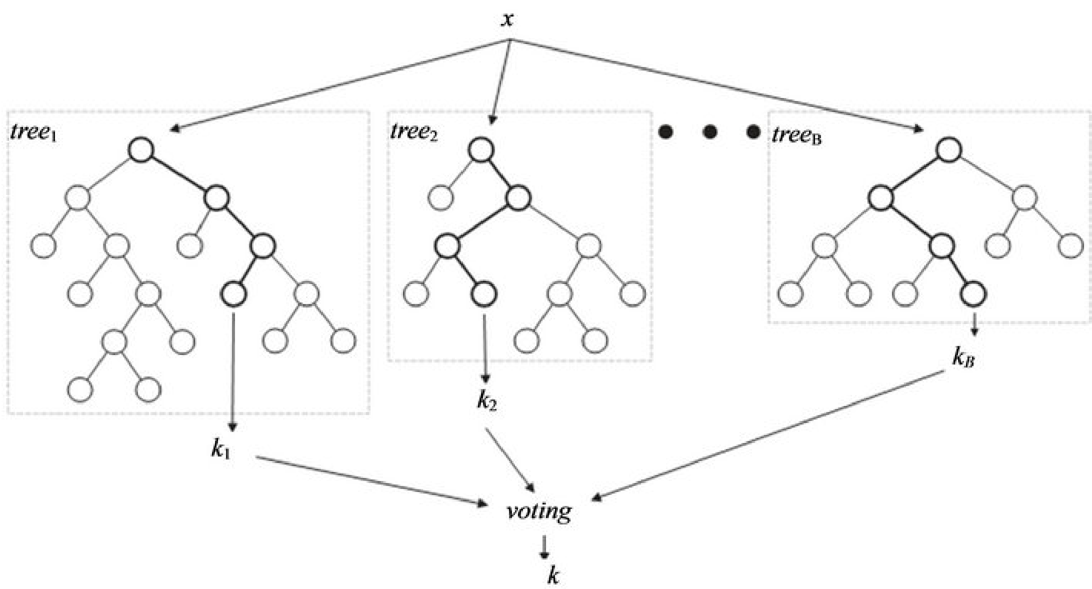


Figure 8: General Architect of random forst

3.2.2. Fuzzy logic

Fuzzy logic provides appropiate methods to treat uncertainties in the form of ambiguity and vagueness. A concept may be considered vague if its extension is unclear or imprecise, due to the uncertainty about the objects that belong to the concept, or has difficulty in defining precise boundaries for some domains of interest [17]. It is the latter definition the one we will focus in this project, since the inputs the ML model takes make more sense as fuzzy categories than as exact numerical values, due to the possible overlap between different categories and the gradual changes between one category and the next.

In Figure 9 we can see 5 different types of fuzzy logic, by the domains the data is divided into. In this project trapezoidal fuzzy logic is used.

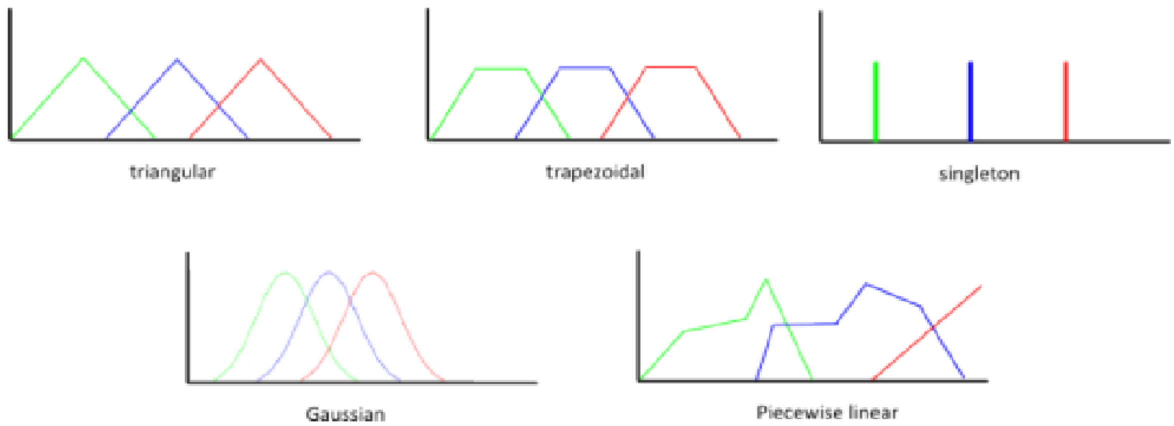


Figure 9: Fuzzy logic types

In summary, in this project the ML model that will classify the virtual patients by their probability of developing DR, is based on the average of predictions of trees built with randomised sets of the training data, which is defined in fuzzy sets. The downside of this approach when it comes to the explainability of the decision is that it depends on hundreds or thousands or rules simultaneously, all derived from the trees generated.

3.3. Generation of neigbours

In order for the algorithm to be able to find rules regarding the highest risk factor for DR in a patient we need to generate a high enough number of neighbours (instances derived from the original patient), knowing the prediction the FRF model used gives each one of them.

Neighbours are defined as instances similar to the patient generated by adding a change into one of the categories that define a patient, as seen in «Data about diabetic patients». Neighbours are generated from the original patient or from one of the neighbours and share the same prediction made with the model, in this case wether or not the patient will develop DR.

Neighbours are implemented as a Patient, which is in turn defined as the set of 10 features of the diabetic patients, plus their respective distance from the original patient, or root.

In this project, when a neighbour presents the opposite prediction than the patient it is based on we refer to them as opposite instances. This opposite instances will be used as the seed to generate neighbours that will share its prediction, and we will use this two sets of data to generate the rules using a DRSA. In order to conclude which type of opposite instance works best we will take both the closest opposite instance to the root, and also the furthest one. Neighbours will be generated for each one of these seeds, and then a third list which a half part of each one will be creating, to have a third.

3.4. Initial algorithm

The algorithm [18] that this project uses as a basis is explained below:

Algorithm 1: Guided-Lore
input: x_i: an instance to explain, T: an auxiliary set, b: a black-box model, L: maximum level of exploration, and $K\ B$: knowledge base.
output: E: explanation of the decision of b on x

1. $y_i \leftarrow b(x_i);$
2. $D^+ \leftarrow GetNeighbours(x_i,\ y_i,\ b,\ L,\ K\ B);$
3. $x^-,\ y^- \leftarrow FindDiffExample(x_i,\ y_i,\ b,\ T);$
4. $D^- \leftarrow GetNeighbours(x^-,\ y^-,\ b,\ L,\ K\ B);$
5. $D \leftarrow D^+\ U\ D^-;$
6. $t \leftarrow BuildTree(D);$
7. $r = (p \rightarrow y) \leftarrow ExtractRule(x_i,\ t);$
8. $\Theta \leftarrow ExtractCounterfactuals(x_i,\ r,\ t);$
9. $E \leftarrow (r,\ \Theta);$

Algorithm 1, given an instance x_i and its outcome y_i, produced by a black box system *b*, the Retiprogram, has the objective to generate an explanationtowards the decision of b.

In step 1 of this algorithm we predict the outcome yi of a patient xi using the black box model b. In step 2 a set D+ is constructed with fictive patients similar to xi with the same label yi. Step 3 consists of finding an instance x- similar to xi within the auxiliary set T. The outcome y- has to be the opposite than yi. In step 4 the set D- of virtual patients similar to x-. Then both sets are combined to form D in step 5. This set is finally used to build the decision trees in step 6, which the algorithm will extract rules and the explainabilityin steps 7 to 9.

3.5. Changes to the algorithm

In this project a different approach will be taken for step 3, with the objective of finding an opposite instance without the use of an auxiliary set *T.* To accomplish this, all the virtual patients created will be saved, no matter the prediction. Once the total number of neigbours desired has been created, the algorithm will look for instances created with the opposite prediction from the instance to explain and one of them will be selected as y- (the different exemple, seed for constructing D-. It that way we do not need to use the model b nor the dataset T.

Moreover, from step 6 on, instead of using Guided-Lore and build a new decision tree, we will substitute these steps by the generation of rules will be generated with Dominance-based rough set approach (DRSA).

Therefore, the new proposed algorithm Guided-Lore-DRSA is the following:

```
Algorithm 2: Guided-Lore-DRSA
input: xi: an instance to explain, b: a black-box model, L: maximum
      level of exploration, and K B: knowledge base.
output: E: explanation of the decision of b on x
   1. yi ← b(xi);
   2. D+ ← GetNeighbours(xi, yi, b, L, K B);
   3. x-, y- ← FindDiffExample(xi, yi, D+);
   4. D- ← GetNeighbours(x-, y-, b, L, K B);
   5. D ← D+ U D-;
   6. E ← BuildDRSArules(D);
```

3.6. RuleStudio

RuleStudio[3]. is an open source application that makes use of the ruleLearn library[4] to make its core funcionalities available in a user-friendly manner. The rule models are induced according to sequential covering algorithms [19]. The system is based on a model of rule generation known as DRSA: Dominance-based Rough Set Approach. It is a well known method in the Operational Research field but not really widespread in the field of Artificial Intelligence yet.

This method for constructing rules is based on techniques of approximate reasoning (rough sets) and in the preference models defined by the different attributes used when taking a decision. Data sets analyzed by ruleLearn, and by definition by RuleStudio, are represented as decision tables, with some input data and a unique output variable. DRSA methodology includes the analysis for the consistency of data.

3.6.1. Dominance-based Rough Set Approach

In this section we will do an introduction to the topics necessary to begin to understand DRSA, the basis of the program used in this project to generate rules, RuleStudio.

Rough set theory is a formal theory derived from fundamental research on logical properties of information systems. The main goal of the rough set analysis is the approximation of concepts, and it offers mathematical tools to discover patterns hidden in data. Some of its uses are feature and pattern extraction, data reduction and decision rule generation. It can aso identify partial or total dependencies in data, among other things.

In rough set, the data is stored in a table, which is called an Information Table. Each row represents a fact or an object and often the facts are not consistent with each other. Information Systems are defined as a pair (U,A), where U is a non-empty finite set of objects and A is a non-empty finite set of attributes, called conditional attributes.

3 **RuleStudio:** https://github.com/dominieq/rule-studio
4 **RuleLearn:** https://github.com/ruleLearn/rulelearn

In the DRSA method, the set of attributes *X* used to calculate the output decision variable *D* are ordinal attributes. Then, the DRSA rules constructed represent different kinds of relationships between *X* and *D*. A positive relationship means that the greater the value of the condition attribute, the higher the class label (i.e. the value of the decision attribute), and a negative relationship means that the greater the value of the condition attribute, the lower the class label. Using DRSA, we can generate rough approximations of each decision class D_k . We assume that the set of classes in *D* are ordered from the lowest to the highest. The number of classes can be more than 2. In the case of Diabetic Retinopathy, we only have two classes: 0 for absence of DR and 1 for indicating presence of DR.

Rules are constructed using elementary building blocks, known as dominance cones, with origins in each example in the attribute space. Based on the rough set concept, rules for lower and upper approximation of each decision class are obtained from the training observations [20].

The choice of DRSA for explainability in the Diabetic Retinopathy disease is motivated by the fact that the values of the attributes are mainly ordinal, where its change from one value to another is an indicator of the risk of developing DR. Moreover, DRSA method generates the minimum number of rules possible for distinguishing between the two possible decision outputs, which is one of the aims of a surrogate model (i.e. as less rules as possible).

Rules represent knowledge discovered from data and we distinguish mainly two types of rules:

- D≥-*decision rules*, providing lower profile descriptions for objects belonging at least to class D_k (so they belong to D_k or a better class, D_{k+1}, D_{k+2}, …).

  if $x_1 \geq v_1$ *and* $x_2 \geq v_2$ *and* … $x_n \geq v_n$ *then* $y \geq D_k$

- D≤-*decision rules*, providing upper profile descriptions for objects belonging at most to class D_k (so they belong to D_k or a lower class, D_{k-1}, $D_{k.2}$, …).

  if $x_1 \leq v_1$ *and* $x_2 \leq v_2$ *and* … $x_n \leq v_n$ *then* $y \leq D_k$

In this notation, we must take into account that all attributes in *X* are considered to be maximization functions (the highest the value, the more related to a highest class in *D*), which are called Gain attributes. In case the relation is a minimization, the case is called Cost attribute, and the lowest its value, the more related to a highest class in *D*.

3.6.2. RuleStudio configuration

The metadata in RuleStudio represents the attributes of the project and it is the key point for building the set of rules. It defines the role of each atribute, the preference direction and other paràmetres. The design of the DRSA model is done with this information. For each attribute, the following information must be provided:

- Name
- If it influences the rules or not
- In which direction does it influence the decision
- If it is a condition or the decision
- The type of the data

The attributes used by RuleStudio for each of the categories described can be found in Table 1. By the design, when an attribute has no direction, that is, there is no clear relation to «more of the attribute always means more or less chance of reaching a decision», RuleStudio offers to «Transform» the attribute. Transforming an attribute means separating it into two separate but equal attributes, each one for a direction, and then try to extract rules separately. In this project we will do this for the CDKEPI, the BMI and the EVOL.

Table 1: Atributes used by RuleStudio

Name	Direction	Type	Type of data	
			Numerical	Fuzzy
CKDEPI	None	Condition	Real	Ennumeration
MA	Gain	Condition	Real	Ennumeration
BMI	None	Condition	Real	Ennumeration
HbA1c	Gain	Condition	Real	Ennumeration
HTAR	Gain	Condition	Real	Ennumeration
TTM	Gain	Condition	Real	Ennumeration
Age	Gain	Condition	Real	Ennumeration
EVOL	Gain	Condition	Real	Ennumeration
Sex	None	Condition	Real	Ennumeration
DR	Gain	Decision	Real	Ennumeration

4. Implementation

For the implementation of the explanation system shown in Chapter 3, 3 different programs were implemented.

4.1. Neighbour generation algorithm

The Guided Neighbours Generator algorithm [18] has been developed in Java, as this was the language used to develop the Retiprogram. This Algorithm 3 has been implemented as it appears in the cited paper:

```
Algorithm 3: Guided Neighbours Generator
input: An example x_0, its output y_o, a black-box model B, the maximum
      level of exploration L, and a knowledge-base KB.
output: The set of neighbours, N.
initialization;
root ← node(x_i,NULL,0);
root.label = y_0 ← B(x_0);
q ← [root];
N ← [];
while Not need to stop do
   n ← head[q];
   if n.label != root.label or n.level > L then
      continue;
   else
      foreach feature f in KB do
         if f is Nominal then
            //choose action
            x_c ← copy(n.x);
            x_c[f] ← random(KB[f][range]) n_c ← node(x_c,n,n.level+1);
            n_c.label ← B(x_c);
            n_c.d ← distance(x_c,x_0);
            add n_c to q and N;
         else
            step ← KB[f][step];
            max_value ← KB[f][max];
            min_value ← KB[f][min];
            if n.x[f] + step <= max_value then
               //forward action
               x_1 ← copy(n.x);
               x_1[f] ← n.x[f]+step;
               n_1 ← node(x_1,n,n.level+1);
               n_1.label ← B(x_1);
               n_1.d ← distance(x_1,x_0);
               add n_1 to q and N;
            end
            if n.x[f] - step >= min_value then
               //backward action
               x_r ← copy(n.x);
```

```
                 xr[f] ← n.x[f]+step;
                 nr ← node(xr,n,n.level+1);
                 nr.label ← B(xr);
                 nr.d ← distance(xr,x0);
                 add nr to q and N;
              end
           end
        end
     end
  end
```

Algorithm 3 shows the generation of closest neighbours of an instance x_o. The search tree starts from this instance and takes the available action from one of the features by taking its value and then adding or substracting the step value, in case of a numerical feature, or by randomly choosing a different value in case of a categorical feature. This method of changing x_o means that the neighbours in the first level of the tree will only be one feature different than x_o, and that the number of neighbours on this first level will be equal to the number of categorical features plus two times the number of numerical features, considering that the numerical features fall within their respective ranges.

If the outcome of the black-box model changes in one of the nodes, then this node becomes a leaf of the tree and will no longer be expanded further. Otherwise, nodes will be expanded by increasing order from x_o, until the desired number of neighbours in the set N has been reached, or all the nodes that should be expanded are further than the maximum level of exploration L.

4.1.1. Output

The output for Algorithm 3 is the set of neighbours N, in the form of a csv file, which is one of the input formats accepted by RuleStudio.

In Figure 10 we can see the first 25 neighbours generated with Patient 1, while Figure 11 shows the first 25 neighbours generated with the furthest opposite instance found within Patient's 1 neighbours, both pertaining to the same file.

CDKEPI	MA	BMI	Hb1AC	HTAR	TTM	edat	evol	sex	RD
97	30	36.81	11	0	0	57	5	0	0
97	0	35	11	0	0	57	5	0	0
97	0	36.81	8	0	0	57	8	0	0
97	0	36.81	9	0	0	57	5	0	0
86	30	36.81	11	0	0	57	5	0	0
86	0	35	11	0	0	57	5	0	0
86	0	36.81	8	0	0	57	8	0	0
86	0	36.81	9	0	0	57	5	0	0
97	45	36.81	11	0	0	57	5	0	0
97	45	36.81	11	0	0	57	8	1	0
97	30	36	11	0	0	57	5	0	0
97	30	37	11	0	0	57	5	0	0
97	30	36.81	10	0	0	57	5	0	0
97	30	36.81	11	0	0	53	5	0	0
97	30	36.81	11	0	0	57	5	1	0
97	15	35	11	0	0	57	5	0	0
97	15	36.81	8	0	0	57	8	0	0
97	15	36.81	9	0	0	57	5	0	0
97	0	34	11	0	0	57	5	0	0
97	0	35	10	0	0	57	5	0	0
97	0	35	11	0	0	53	5	0	0
97	0	35	11	0	0	57	5	1	0
97	0	36	8	0	0	57	8	0	0
97	0	36	9	0	0	57	5	0	0
97	0	37	8	0	0	57	8	0	0

Figure 10: Example of 25 first neighbours generated from Patient 1

97	0	36.81	11	0	1	57	11	0	1
108.93	15	36.81	11	0	1	57	11	0	1
108.93	0	36	11	0	1	57	11	0	1
108.93	0	37	11	0	1	57	11	0	1
108.93	0	36.81	10	0	1	57	11	0	1
108.93	0	36.81	11	0	2	57	11	0	1
108.93	0	36.81	11	0	1	53	11	0	1
108.93	0	36.81	11	0	1	61	11	0	1
108.93	0	36.81	11	0	1	57	8	0	1
108.93	0	36.81	11	0	1	57	14	0	1
108.93	0	36.81	11	0	1	57	11	1	1
86	0	36.81	11	0	1	57	11	0	1
97	15	36.81	11	0	1	57	11	0	1
97	0	36	11	0	1	57	11	0	1
97	0	37	11	0	1	57	11	0	1
97	0	36.81	10	0	1	57	11	0	1
97	0	36.81	11	0	0	57	11	0	1
97	0	36.81	11	0	1	53	11	0	1
97	0	36.81	11	0	1	61	11	0	1
97	0	36.81	11	0	1	57	8	0	1
97	0	36.81	11	0	1	57	14	0	1
97	0	36.81	11	0	1	57	11	1	1
108.93	30	36.81	11	0	1	57	11	0	1
108.93	15	36	11	0	1	57	11	0	1
108.93	15	37	11	0	1	57	11	0	1
108.93	15	36.81	10	0	1	57	11	0	1

Figure 11: Example of first 25 neighbours with opposite prediction from Patient 1

4.1.2. Objects

The algorithm uses the following objects:

- **Patient:** the class «Patient» has all the attributes present in Figure 1. They all have been implemented as *doubles*, since this is the data type the Retiprogram works with.
- **Node:** the class «Node» has an attribute «Patient» and «Distance», which saves how many levels is from the root of the tree.
- **Feature:** the knowledge base for the algorithm is implemented as an array of this class. It is defined as the following attributes:
  - Name
  - Type: if it is categorical or numerical.
  - Minimum value for the feature
  - Maximum value for the feature
  - Step: the value to be added or removed to generate a new instance. Only used in numerical features.

4.1.3. Opposite instances

The implementationfor the opposite instance selection algorithm was decided to be implemented according to the distance of each of these instances to the root. Three approaches were taken to see if one yielded consistently better results than the others:

1. MIN: taking the nearest neighbour from the root.
2. MAX: taking the furthest neighbour from the root.
3. MIX: adding one half of each of the previosuly generated list.

4.2. Fuzzification algorithm

The fuzzifier is the algorithm used to transform the existing csv files with the virtual instances (half sharing the prediction for the patient and half having the opposite one) from a numerical to a fuzzy version.

```
Algorithm 4: Fuzzifier
input: Set of neighbours N, knowledge base KB, and membership funtions M.
output: The set of fuzzy neighbours, FN.
   FN ← [];
   foreach instance n in N do
      foreach feature f in KB do
         if f is Numerical then
             label ← find the fuzzy set with highest M[f](n);
             FN ← add label as substitute of the numerical value;
      end
   end
```

Algorithm 4 takes a set of neighbours in a csv format and fuzzifies them according to the fuzzy categories defined in a knowledge-base and with their corresponding membership functions, in order to generate another csv file with the new set of neighbours.

Membership functions are trapezoidal functions as the ones shown in Figure 9. Therefore, four points are needed to define a fuzzy membership function: (a, b, c, d), Moreover, the fuzzy sets define a fuzzy partition, which means that for any point the sum of the membersip values is equal to 1. Consequently, the points (a+b)/2 and (c+d)/2 are the boundaries from one category to the other. Using these boundaries it is straightforward to find which category (i.e. label) corresponds to an input numerical value.

Once we found the fuzzy category, we substitute the numerical value with the name of the appropriate fuzzy category. When this is done for all features of an instance x , we go to the next, until all instances have been fuzzified.

4.2.1. Output

The output for Algorithm 4 is the fuzzified set of neighbours that entered as input. If we took Figure 10 as the file that served for input for the fuzzifier, we would get a csv file with an output as Figure 12.

CDKEPI	MA	BMI	Hb1AC	HTAR	TTM	age	evol	sex	RD
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	7to8	goodControl	Diet	Fifties	5to10	Man	Class0
VeryHigh	Correct	Obese-High	8to9	goodControl	Diet	Fifties	Less5	Man	Class0
High	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
High	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
High	Correct	Obese-High	7to8	goodControl	Diet	Fifties	5to10	Man	Class0
High	Correct	Obese-High	8to9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	High	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	High	Obese-High	More9	goodControl	Diet	Fifties	5to10	Woman	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Woman	Class0
VeryHigh	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	7to8	goodControl	Diet	Fifties	5to10	Man	Class0
VeryHigh	Correct	Obese-High	8to9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Woman	Class0
VeryHigh	Correct	Obese-High	7to8	goodControl	Diet	Fifties	5to10	Man	Class0
VeryHigh	Correct	Obese-High	8to9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	7to8	goodControl	Diet	Fifties	5to10	Man	Class0

Figure 12: Example of 25 first fuzzified neighbours generated from Patient 1

4.2.2. Objects

To transform from the numerical to the fuzzy version in order to compare the two, the fuzzifier makes use of the following classes:

- **Feature:** reused from the neighbour generation algorithm, since it needs to import the same knowledge base.
- **Fuzzy class:** information necessary to determine into which of the different fuzzy categories from a feature a numerical value falls into.
    - Name
    - Value: due to the limits of the fuzzy categories, with this value we can find the point where two different fuzzy categories overlap.

4.3. Rule generation metadata

RuleStudio needs the metadata described in section 3.5.1 «Metadata configuration» as a *JSON* file. A metadata file had to be generated for each of the version that had to be tested. That is, a numerical metadata file and a fuzzy one. This files have the information explained for each of the features in Table 1, with the exception of the different domains for each feature, some of which can be found in Table 2.

Table 2: Domain for each of the fuzzy attributes

Attribute	Domain
CKDEPI	"VeryLow", "Low", "Normal", "High", "VeryHigh"
MA	"Correct", "High"
BMI	"Underweight", "Normal-Low", "Normal-High", "Overweight-Low", ...
...	...
Sex	"Man", "Woman"
DR	"Class0", "Class1"

The first two attributes of Table 2 can be seen in Figure 13 in the way they are implemented in the fuzzy version of the *JSON* file.

```
[
  {
    "name": "CDKEPI",
    "active": true,
    "preferenceType": "none",
    "type": "condition",
    "valueType": "enumeration",
    "domain": [
        "VeryLow",
        "Low",
        "Normal",
        "High",
        "VeryHigh"
        ]
  },
  {
    "name": "MA",
    "active": true,
    "preferenceType": "gain",
    "type": "condition",
    "valueType": "enumeration",
    "domain": [
        "Correct",
        "High"
        ]
  },
```

Figure 13: JSON implementation of fuzzy attributes metadata

4.3.1. Numerical version

In Figures 14 we can see all the attributes this project works with loaded in the program, together with some virtual instances. In this screenshot we can observe how RuleStudio marks the attribute with a preference type of «cost» in green and of «none» in blue.

Figure 15 shows part of the same attributes after the attribute transformation mentioned in Section 3.6.2. The attributes of the «none» preference type get divided into two categories, one with the preference type «gain» and another one with «loss», marked by red. This way, RuleStudio will try to find if the data is more likely to generate better rules with one direction or the other.

CDKEPI (condition,active)	MA (condition,active)	BMI (condition,active)	HbA1c (condition,active)	HTAR (condition,active)	TTM (condition,active)	Age (condition,active)	EVOL (condition,active)	Sex (condition,active)	RD (decision,active)
e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20
97.0	30.0	36.81	11.0	0.0	0.0	57.0	5.0	0.0	0.0
97.0	0.0	35.0	11.0	0.0	0.0	57.0	5.0	0.0	0.0
97.0	0.0	36.81	8.0	0.0	0.0	57.0	8.0	0.0	0.0
97.0	0.0	36.81	9.0	0.0	0.0	57.0	5.0	0.0	0.0
86.0	30.0	36.81	11.0	0.0	0.0	57.0	5.0	0.0	0.0
86.0	0.0	35.0	11.0	0.0	0.0	57.0	5.0	0.0	0.0
86.0	0.0	36.81	8.0	0.0	0.0	57.0	8.0	0.0	0.0

Figure 14: Numerical RuleStudio metadata before transformation

CDKEPI_g (condition,active)	CDKEPI_c (condition,active)	MA (condition,active)	BMI_g (condition,active)	BMI_c (condition,active)	HbA1c (condition,active)
e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20
97.0	97.0	30.0	36.81	36.81	11.0
97.0	97.0	0.0	35.0	35.0	11.0
97.0	97.0	0.0	36.81	36.81	8.0
97.0	97.0	0.0	36.81	36.81	9.0
86.0	86.0	30.0	36.81	36.81	11.0
86.0	86.0	0.0	35.0	35.0	11.0
86.0	86.0	0.0	36.81	36.81	8.0
86.0	86.0	0.0	36.81	36.81	9.0
97.0	97.0	45.0	36.81	36.81	11.0
97.0	97.0	45.0	36.81	36.81	11.0
97.0	97.0	30.0	36.0	36.0	11.0
97.0	97.0	30.0	37.0	37.0	11.0
97.0	97.0	30.0	36.81	36.81	10.0
97.0	97.0	30.0	36.81	36.81	11.0
97.0	97.0	30.0	36.81	36.81	11.0

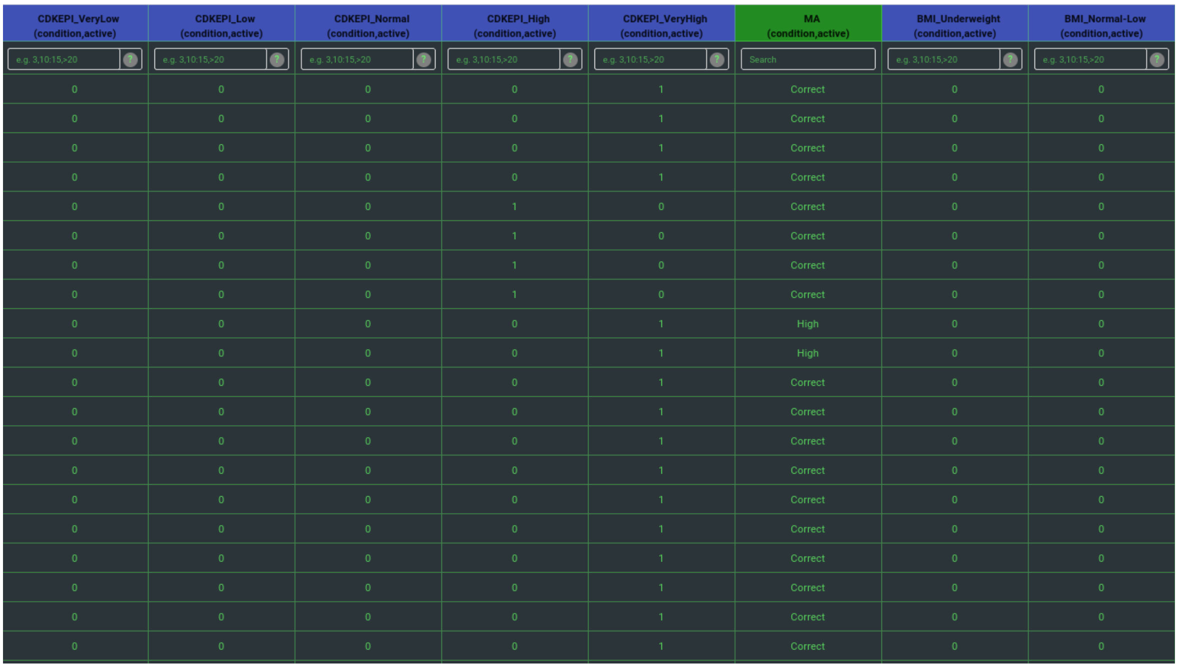
Figure 15: Numerical RuleStudio metadata after transformation

4.3.2. Fuzzy version

Figures 16 and 17 show the same situation than Figures 14 and 15, but for the fuzzy version. The main difference appears in Figure 17. Since the data is no longer a real number, after the transformation, RuleStudio multiplies this category into as many columns as elements make up the enumeration for it.

CDKEPI (condition,active)	MA (condition,active)	BMI (condition,active)	HbA1c (condition,active)	HTAR (condition,active)	TTM (condition,active)	Age (condition,active)	EVOL (condition,active)	Sex (condition,active)	RD (decision,active)
Search	Search	Search	Search	Search	Search	Search	Search	Search	Search
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	7to8	goodControl	Diet	Fifties	5to10	Man	Class0
VeryHigh	Correct	Obese-High	8to9	goodControl	Diet	Fifties	Less5	Man	Class0
High	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
High	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
High	Correct	Obese-High	7to8	goodControl	Diet	Fifties	5to10	Man	Class0
High	Correct	Obese-High	8to9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	High	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	High	Obese-High	More9	goodControl	Diet	Fifties	5to10	Woman	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	More9	goodControl	Diet	Fifties	Less5	Woman	Class0
VeryHigh	Correct	Obese-Low	More9	goodControl	Diet	Fifties	Less5	Man	Class0
VeryHigh	Correct	Obese-High	7to8	goodControl	Diet	Fifties	5to10	Man	Class0
VeryHigh	Correct	Obese-High	8to9	goodControl	Diet	Fifties	Less5	Man	Class0

Figure 16: Fuzzy RuleStudio metadata before transformation



CDKEPI_VeryLow (condition,active)	CDKEPI_Low (condition,active)	CDKEPI_Normal (condition,active)	CDKEPI_High (condition,active)	CDKEPI_VeryHigh (condition,active)	MA (condition,active)	BMI_Underweight (condition,active)	BMI_Normal-Low (condition,active)
e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	e.g. 3,10:15,>20	Search	e.g. 3,10:15,>20	e.g. 3,10:15,>20
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	1	0	Correct	0	0
0	0	0	1	0	Correct	0	0
0	0	0	1	0	Correct	0	0
0	0	0	1	0	Correct	0	0
0	0	0	0	1	High	0	0
0	0	0	0	1	High	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0
0	0	0	0	1	Correct	0	0

Figure 17: Fuzzy RuleStudio metadata after transformation

5. Evaluation

In this chapter we will go through the process of evaluating the data and the algorithms described in Chapters 3 and 4. We will start with the training set of the ML model Retiprogram, to then move to the instances generated by the neighbour generation algorithm. Finally, we will talk about the rules generated with RuleStudio and compare them to the ones that can be generated with the initial algorithm [18]

5.1. Testing data

I was given a dataset of 25 patients with the corresponding data, as shown in Table 3. These was the set to be used for testing. For each patient, we have an id, the values of the 9 risk factors (in their corresponding numerical scales) and finally the class label assigned by the doctors, indicating the presence (1) or abscence (0) of DR.

Table 3: Retiprogram training set

Id	Age	Sex	EVOL	TTM	HbA1c	CDKEPI	MA	BMI	HTAR	DR
1	57	0	11	1	11	108,93	0	36,81	0	1
2	72	1	15	2	8	69,77	33	30,68	1	1
3	54	0	5	1	8,3	63,17	0	28,47	0	1
4	88	0	12	2	8,2	49,75	246	26,21	1	1
5	39	0	7	2	9	120,01	0	25,87	1	1
6	70	1	13	2	11	75,03	0	32,26	1	1
7	48	0	10	2	10,7	78,06	38	32,74	1	1
8	76	1	7	1	9	83,01	0	46,31	1	1
9	71	1	18	1	11	89,61	0	24,1	1	1
10	61	0	13	2	14	46,44	259	36,09	1	1
11	65	1	15	1	6,5	64,04	0	28,61	0	0
12	75	0	22	1	9,1	76,71	0	30,81	0	0
13	50	0	7	1	6,2	73,05	0	29,76	0	0
14	72	1	5	1	6,6	90,86	0	32,04	1	0
15	64	1	3	0	7	69,8	0	35,49	0	0
16	79	1	9	0	8,7	26,44	240	26,22	1	0
17	63	0	5	2	8,1	86,65	0	44,37	1	0
18	38	1	8	2	5	101,5	0	29,07	0	0
19	76	1	1	2	14,4	91,4	0	29,49	1	0
20	65	0	10	1	6,2	96,23	0	34,06	0	0
21	68	1	2	0	7,5	51,36	0	42,54	1	0
22	55	1	11	2	8	100,78	0	25,12	0	0
23	70	0	8	1	5,9	95,98	0	38,53	1	0
24	67	1	4	1	6,2	94,62	0	25,47	0	0
25	62	1	7	2	6,3	93,16	0	33,2	0	0

In order to see if all patients were useful for the testing, we checked two conditions:

1. Whether or not the Retiprogram had a prediction for the patient (as the program sometimes give an “undefined” output).
2. Whether or not it could generate an acceptable number of opposit instances,

Results of these indicators are displayed in Table 4.

Table 4: List of patients with predicted output and number of opposite instances

Patient	DR	Retiprogram	Number of opposite instances (Number of neighbours generated)				
			100	200	300	500	1500
1	1	1	3	6	18	29	
2	1	1	2	8	12	40	
3	**1**	**undefined**					
4	1	1	9	25	50	82	
5	1	1	3	5	26	39	
6	**1**	**1**	**0**	**0**	**0**	**0**	**4**
7	1	1	1	2	2	11	
8	1	1	33	60	102	178	
9	**1**	**1**	**0**	**0**	**0**	**0**	**3**
10	**1**	**1**	**0**	**0**	**0**	**0**	**4**
11	0	0	5	13	19	39	
12	0	1	9	15	24	52	
13	0	0	0	2	1	9	30
14	0	0	2	2	3	18	
15	**0**	**0**	**0**	**0**	**0**	**0**	**1**
16	**0**	**undefined**					
17	0	1	37	89	145	257	
18	0	0	0	2	2	8	45
19	0	1	19	51	71	136	
20	0	0	0	2	9	12	53
21	0	0	9	16	22	47	
22	0	1	42	100	167	308	
23	0	0	1	8	12	31	
24	**0**	**0**	**0**	**0**	**0**	**0**	**0**
25	0	0	2	4	8	14	

After evaluating the list, patients 3 and 16, in yellow, were removed due to being «undefined» by the Retiprogram. Without a prediction by the ML algorithm we are unable to generate rules that differentiate the virtual patients generated. Patients 6, 9, 10, 15 and 24, in blue, were eliminated, because they did not have neighbours near them that were put in their opposite category. In this stage of the project, after counting how many opposite instances were generated per number of neighbours, we decided that the optimal number of neighbour generation was 300. After choosing not to work with the mentioned patients, the number of patients whose rules are going to be generated is 18.

One interesting insight from Table 4 is that the majority of patients with a big number of opposite instances are misclassified by the Retiprogram, as seen in patients 9, 17, 19 and 22. Initially we thought that this could be used to detect and filter out misclassified instances, but patient 8 is an exception to that hypothesis. Most likely, this can be explained as the Retiprogram misclassifying patients when they are closer to the limit of the two outcomes. In this scenario, the changes the algorithm in this project makes to the different features of the patient would be more likely to result in opposite instances.

5.2. Evaluation of the neighbours generated

Once the algorithm could find the nearest and furthest opposite instances from the original patient (minimum and maximum), they were evaluated to see which of the features were being changed the most (in red), which can be observed in Table 5.

Table 5: Evaluation of changes in opposite instances

Id	Age	Sex	EVOL	TTM	HbA1c	CDKEPI	MA	BMI	HTAR	DR
1	57	0	11	1	11	108,93	0	36,81	0	1
min	57	0	8	0	11	108,93	0	36,81	0	0
max	53	0	5	1	11	108,93	0	36,81	0	0
4	88	0	12	2	8,2	49,75	246	26,21	1	1
min	88	0	9	0	8,2	49,75	246	26,21	0	0
max	80	0	12	0	8,2	49,75	246	26,21	1	0
25	62	1	7	2	6,3	93,16	0	33,2	0	0
min	66	1	13	2	6,3	93,16	0	33,2	0	1
max	58	1	13	2	6,3	93,16	0	33,2	0	1
20	65	0	10	1	6,2	96,23	0	34,06	0	0
min	69	0	13	2	6,2	96,23	0	34,06	0	1
max	61	0	13	2	6,2	96,23	0	34,06	0	1

We realized that the algorithm was generating instances that were different from the patient in ways that the patient or the medical staff had little control over, for example the age of the patient or how long they have been undergoing DR treatment. This was due to the order of the features in the knowledge base, which the algorithm was prioritizing, and the desired number of neighbours was being reached before changing the significant features. After consluting with doctors, the feature order was changed to: CDKEPI, MA, BMI, HbA1c, HTAR, TTM, Age, EVOL, Sex and DR; and the neighbours generation algorithm was executed again, as it can be seen in Table 6.

Table 6: Evaluation of opposite instances after changing the feature order

Id	Age	Sex	EVOL	TTM	HbA1c	CDKEPI	MA	BMI	HTAR	DR
1	57	0	11	1	11	108,93	0	36,81	0	1
min	57	0	5	1	11	108,93	0	36,81	0	0
max	53	0	8	0	11	97	0	36.81	0	0
4	88	0	12	2	8,2	49,75	246	26,21	1	1
min	88	0	12	0	8,2	49,75	231	26,21	1	0
max	88	0	12	0	8,2	49	246	26,21	1	0
25	62	1	7	2	6,3	93,16	0	33,2	0	0
min	62	1	13	2	6,3	93,16	15	33,2	0	1
max	62	1	13	2	6,3	82	0	33,2	0	1
20	65	0	10	1	6,2	96,23	0	34,06	0	0
min	65	0	13	2	6,2	96,23	15	34,06	0	1
max	65	0	13	2	6,2	85	0	34,06	0	1

After the changes, although some of the differences still happen in categories in which people have no control, it still shifts the difference into the new top priority features: CDKEPI and MA, that could be modified with appropriate medical treatment, as they are indicators of bad performance of some organs.

These neighbours were deemed appropiate, and as such the files for the 18 patients that made the cut were generated, with their new Id, which is the one that will be referenced from now on.

5.3. Evaluation of the DRSA rules

With the all the neighbour files for all patients, rules could be generated. The evaluation of these rules can be found in Tables 7 and 8. To evaluate these rules we will make use of the number of rules, the strength[5] of the rules, and whether or not the rules generated are the same for the three models of opposite instances, max, min and mix. In the Tables below, if the «Same» row is green it means that the rules are the same for at least another of the models.

5 In the context of the characteristics that define a rule, strength is defined as the ratio of the number of objects supporting the rule to the number of objects included in the learning set (decision table)

	Patient	Class	# Rules	# Rules 0	# Rules 1	Min strength	Max strength	Same	Comments
			Numerical values						
Max	1	1	5	2	3	0.165	0.487		
Min	1	1	6	3	3	0.1	0.487		
Mix	1	1	6	3	3	0.173	0.423		
Max	2	1	3	1	2	0.407	0.5		
Min	2	1	3	1	2	0.407	0.5		
Mix	2	1	3	1	2	0.407	0.5		
Max	3	1	4	2	2	0.26	0.475		
Min	3	1	5	2	3	0.305	0.452		
Mix	3	1	4	2	2	0.28	0.465		
Max	4	1	4	2	2	0.12	0.498		
Min	4	1	7	4	3	0.17	0.435		
Mix	4	1	9	5	4	0.08	0.362		
Max	5	1	5	2	3	0.142	0.458		
Min	5	1	5	2	3	0.34	0.458		
Mix	5	1	5	2	3	0.332	0.45		
Max	6	1	5	2	3	0.083	0.49		
Min	6	1	5	3	2	0.207	0.467		
Mix	6	1	5	3	2	0.123	0.467		
Max	7	0	2	1	1	0.5	0.5		Only TTM
Min	7	0	2	1	1	0.5	0.5		
Mix	7	0	2	1	1	0.5	0.5		
Max	8	1	9	4	5	0.073	0.467		
Min	8	1	11	5	6	0.053	0.467		
Mix	8	1	11	7	4	0.053	0.467		
Max	9	0	2	1	1	0.5	0.5		Only TTM
Min	9	0	2	1	1	0.5	0.5		
Mix	9	0	2	1	1	0.5	0.5		
Max	10	0	4	2	2	0.233	0.487		
Min	10	0	4	2	2	0.317	0.487		
Mix	10	0	4	2	2	0.317	0.487		
Max	11	1	2	1	1	0.5	0.5		Only TTM
Min	11	1	2	1	1	0.5	0.5		
Mix	11	1	2	1	1	0.5	0.5		
Max	12	0	5	3	2	0.2	0.457		
Min	12	0	4	2	2	0.133	0.495		
Mix	12	0	5	3	2	0.2	0.455		
Max	13	1	4	2	2	0.23	0.497		

	Patient	Class	# Rules	# Rules 0	# Rules 1	Min strength	Max strength	Same	Comments
			Fuzzy values						
Max	1	1	4	2	2	0.165	0.487		
Min	1	1	6	3	3	0.1	0.487		
Mix	1	1	6	3	3	0.1	0.487		
Max	2	1	3	1	2	0.407	0.5		Hb1AC and TTM
Min	2	1	3	1	2	0.407	0.5		
Mix	2	1	3	1	2	0.407	0.5		
Max	3	1	4	2	2	0.26	0.475		TTM and/or HbA1C
Min	3	1	5	2	3	0.305	0.452		
Mix	3	1	5	2	3	0.305	0.452		+BMI
Max	4	1	7	3	4	0.075	0.427		
Min	4	1	7	4	3	0.177	0.435		
Mix	4	1	6	2	4	0.015	0.413		
Max	5	1	6	2	4	0.015	0.435		
Min	5	1	6	2	4	0.02	0.435		
Mix	5	1	6	2	4	0.02	0.435		
Max	6	1	4	2	2	0.238	0.49		Hb1AC and TTM
Min	6	1	5	3	2	0.207	0.467		Hb1AC and TTM
Mix	6	1	5	3	2	0.207	0.467		Hb1AC and TTM
Max	7	0	2	1	1	0.5	0.5		Only TTM
Min	7	0	2	1	1	0.5	0.5		
Mix	7	0	2	1	1	0.5	0.5		
Max	8	1	5	3	2	0.073	0.467		
Min	8	1	6	4	2	0.073	0.467		Same except 1
Mix	8	1	6	4	2	0.073	0.467		
Max	9	0	2	1	1	0.5	0.5		Only TTM
Min	9	0	2	1	1	0.5	0.5		
Mix	9	0	2	1	1	0.5	0.5		
Max	10	0	6	1	5	0.013	0.487		
Min	10	0	4	1	3	0.055	0.487		
Mix	10	0	4	1	3	0.055	0.487		
Max	11	1	2	1	1	0.5	0.5		Only TTM
Min	11	1	2	1	1	0.5	0.5		
Mix	11	1	2	1	1	0.5	0.5		
Max	12	0	5	3	2	0.2	0.457		
Min	12	0	5	3	2	0.137	0.496		
Mix	12	0	5	3	2	0.137	0.496		
Max	13	1	4	2	2	0.3	0.497		
Min	13	1	2	1	1	0.5	0.5		Only TTM
Mix	13	1	2	1	1	0.5	0.5		
Max	14	0	2	1	1	0.5	0.5		Only TTM
Min	14	0	2	1	1	0.5	0.5		
Mix	14	0	2	1	1	0.5	0.5		
Max	15	0	4	2	2	0.1	0.498		Same + HbA1c I CDKEPI
Min	15	0	3	1	2	0.035	0.493		TTM + Hb1Ac or EVOL
Mix	15	0	3	1	2	0.035	0.493		
Max	16	1	4	2	2	0.18	0.477		
Min	16	1	4	2	2	0.023	0.497		TTM + EVOL (for two)
Mix	16	1	4	2	2	0.023	0.497		
Max	17	0	2	1	1	0.5	0.5		Only TTM
Min	17	0	2	1	1	0.5	0.5		
Mix	17	0	2	1	1	0.5	0.5		
Max	18	0	2	1	1	0.5	0.5		Only EVOL
Min	18	0	2	1	1	0.5	0.5		
Mix	18	0	2	1	1	0.5	0.5		

Table 8: Evaluation of fuzzy rules

After reviewing Tables 7 and 8 we can see that both models yield a generally similar number of rules, which is a maximum of 11 for the numerical version and 7 for the fuzzy, although the average is closer to 4 or 5 rules per patient on average. As we can see, sometimes the rules are the same independently of the opposite instance used as seed when generating the neighbours, especially in the fuzzy version. This makes sense, since a small numerical difference can disappear when fuzzifying the neighbour filesk, and as such the rules would end up being the same.

The last inshight that we can observe from Tables 7 and 8 is that patients 7, 9, 11, 14, 17 and 18 in the numerical version, plus pacient 16 in the fuzzy one, depend only on variables that the patient has no control over: the type of treatment (TTM) and the duration of the treatment they are undergoing (EVOL).

5.4. Comparison with Guided-LORE

In this section we will compare the rules generated through the DRSA approach taken in this project with the best rules obtained with a Guided-LORE (GLORE) approach, such as the one in the initial algorithm [18].

The tables show on the left column, the rules generated by GLORE. In this method, we first have a set of rules that represent a decision tree. From these rules, the algorithm selects the one as explanation of the classification made by the Retiprogram. Then, it generates some counterfactual conditions that lead to the opposite class. In the right column we have the results of DRSA method, first using fuzzy labels and below with the numerical values.

5.4.1. Patient 4

RETIPROGRAM *predicts* class 1

Age	Sex	EVOL	TTM	HbA1c	CDKEPI	MA	BMI	HTAR	RD
39	**0**	**7**	**2**	**9**	**120,01**	**0**	**25,87**	**1**	**1**

Table 9: Rule comparison for Patient 4

	GLORE rules	DRSA rules
min	{"RD": 1}, {"TTM": "2"} {"RD": 0}, {"TTM": "0", "HbA1c": "<=8.000000"} {"RD": 0}, {"TTM": "1", "HbA1c": "<=7.000000"} {"RD": 1}, {"TTM": "1", "HbA1c": ">7.000000"} {"RD": 0}, {"TTM": "0", "HbA1c": ">8.000000", "EVOL": "<=4"} {"RD": 1}, {"TTM": "0", "HbA1c": ">8.000000", "EVOL": ">4"}	(TTM >= Insuline) & (EVOL >= 5to10) => (RD = 1) (TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD = 1) (HbA1c >= 8to9) & (EVOL >= 5to10) => (RD = 1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <= 0) (HbA1c <= 6to7) & (EVOL <= Less5) => (RD <= 0) (TTM <= Diet) & (HbA1c <= 7to8) => (RD <= 0) (TTM <= Diet) & (EVOL <= Less5) => (RD <= 0)
	r = {'TTM': '2'} → {'RD': 1} delta {'TTM': '0', 'HbA1c': '<=8.0'} delta {'TTM': '1', 'HbA1c': '<=7.0'}	(TTM >= 2.0) & (EVOL >= 7.0) => (RD >= 1.0) (HbA1c >= 9.0) & (EVOL >= 7.0) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (HbA1c <= 7.0) & (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 7.0) & (EVOL <= 4.0) => (RD <= 0.0) (TTM <= 0.0) & (HbA1c <= 8.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 4.0) => (RD <= 0.0)
max	{"RD": 1}, {"HbA1c": ">8.000000"}, {"RD": 1}, {"HbA1c": "<=8.000000", "CDKE-PI": ">109.000000"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKE-PI": "<=109.000000", "TTM": "0"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKE-PI": "<=109.000000", "TTM": "1"}, {"RD": 1}, {"HbA1c": "<=8.000000", "CDKE-PI": "<=109.000000", "TTM": "2"},	(HbA1c >= 8to9) => (RD = 1) (TTM >= Insuline) & (HbA1c >= 7to8) => (RD = 1) (TTM >= Insuline) & (EVOL >= 5to10) => (RD = 1) (HbA1c >= 7to8) & (TTM >= OralAntidiab) & (Age >= Forties) => (RD = 1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <= 0) (HbA1c <= 6to7) & (EVOL <= Less5) => (RD <= 0) (TTM <= Diet) & (HbA1c <= 7to8) => (RD <= 0)
	r = {'HbA1c': '>8.000000'} → {'RD': 1} delta {'HbA1c': '<=8.0', 'CDKEPI': '<=109.0', 'TTM': '0'} delta {'HbA1c': '<=8.0', 'CDKEPI': '<=109.0', 'TTM': '1'}	(HbA1c >= 9.0) => (RD >= 1.0) (CDKEPI_g >= 120.01) => (RD >= 1.0) (TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (HbA1c <= 6.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 1.0) => (RD <= 0.0)
mix	{"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "0"} {"RD": 1}, {"HbA1c": "<=8.000000", "TTM": "2"} {"RD": 0}, {"HbA1c": ">8.000000", "MA": ">30"}	(TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD >= Class1) (HbA1c >= 8to9) & (EVOL >= 5to10) => (RD >= Class1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <=

	{"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "1", "CDKEPI": "<=109.000000"} {"RD": 1}, {"HbA1c": ">8.000000", "MA": "<=30", "EVOL": ">4"} {"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "1", "CDKEPI": ">109.000000", "EVOL": "<=4"} {"RD": 1}, {"HbA1c": "<=8.000000", "TTM": "1", "CDKEPI": ">109.000000", "EVOL": ">4"} {"RD": 0}, {"HbA1c": ">8.000000", "MA": "<=30", "EVOL": "<=4", "TTM": "0"} {"RD": 1}, {"HbA1c": ">8.000000", "MA": "<=30", "EVOL": "<=4", "TTM": "1"} {"RD": 1}, {"HbA1c": ">8.000000", "MA": "<=30", "EVOL": "<=4", "TTM": "2"}	Class0) (HbA1c <= 6to7) & (EVOL <= Less5) => (RD <= Class0) (TTM <= Diet) & (HbA1c <= 7to8) => (RD <= Class0) (TTM <= Diet) & (EVOL <= Less5) => (RD <= Class0)
	r = {'HbA1c': '>8.000000', 'MA': '<=30', 'EVOL': '>4'} → {'RD': 1} delta {'MA': '>30.0'}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (HbA1c >= 9.0) & (TTM >= 1.0) => (RD >= 1.0) (HbA1c >= 9.0) & (EVOL >= 7.0) => (RD >= 1.0) (CDKEPI_g >= 120.01) & (TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (HbA1c <= 6.0) => (RD <= 0.0) (HbA1c <= 7.0) & (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 4.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 1.0) & (CDKEPI_g <= 109.0) => (RD <= 0.0)

5.4.2. Patient 8

RETIPROGRAM predicts class 1

Age	**Sex**	**EVOL**	**TTM**	**HbA1c**	**CDKEPI**	**MA**	**BMI**	**HTAR**	**RD**
75	**0**	**22**	**1**	**9,1**	**76,71**	**0**	**30,81**	**0**	**0**

Table 10: Rule comparison for Patient 6

	GLORE rules	**DRSA rules**
min	{"RD": 1}, {"TTM": "2"} {"RD": 0}, {"TTM": "0", "HbA1c": "<=8.000000"} {"RD": 1}, {"TTM": "1", "Sex": "0"} {"RD": 0}, {"TTM": "0", "HbA1c": ">8.000000", "EVOL": "<=4"} {"RD": 1}, {"TTM": "0", "HbA1c": ">8.000000", "EVOL": ">4"} {"RD": 1}, {"TTM": "1", "Sex": "1", "Age": "<=72.000000"} {"RD": 0}, {"TTM": "1", "Sex": "1", "Age": ">72.000000", "HbA1c": "<=8.000000"} {"RD": 1}, {"TTM": "1", "Sex": "1", "Age": ">72.000000", "HbA1c": ">8.000000"}	(TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD >= Class1) (HbA1c >= 8to9) & (EVOL >= 5to10) => (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0) (TTM <= Diet) & (EVOL <= Less5) => (RD <= Class0)
	r = {'TTM': '1', 'Sex': '1', 'Age': '>72.000000', 'HbA1c': '>8.000000'} → {'RD': 1} delta {'HbA1c': '<=8.0'}	(TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (HbA1c >= 9.0) & (EVOL >= 7.0) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 4.0) => (RD <= 0.0)
max	{"RD": 1}, {"HbA1c": ">8.000000"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKEPI": "<=61.000000"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKEPI": "61.0< CDKEPI <=83.0"}, {"RD": 1}, {"HbA1c": "<=8.000000", "CDKEPI": ">83.0"},	(HbA1c >= 8to9) => (RD >= Class1) (HbA1c >= 7to8) & (TTM >= OralAntidiab) => (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0)
	r = {'HbA1c': '>8.000000'} → {'RD': 1} delta {'HbA1c': '<=8.0', 'CDKEPI': '<=61.0'}	(HbA1c >= 9.0) => (RD >= 1.0) (CDKEPI_g >= 83.01) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0)
mix	{"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "0"} {"RD": 1}, {"HbA1c": "<=8.000000", "TTM": "2"} {"RD": 1}, {"HbA1c": ">8.000000", "EVOL": ">4"} {"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "1", "EVOL": "<=7"}	(TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD >= Class1) (HbA1c >= 8to9) & (EVOL >= 5to10) => (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0) (TTM <= Diet) & (EVOL <= Less5) => (RD <= Class0)

	{"RD": 0}, {"HbA1c": ">8.000000", "EVOL": "<=4", "TTM": "0"} {"RD": 1}, {"HbA1c": ">8.000000", "EVOL": "<=4", "TTM": "1"} {"RD": 1}, {"HbA1c": ">8.000000", "EVOL": "<=4", "TTM": "2"} {"RD": 0}, {"HbA1c": "<=7.0", "TTM": "1", "EVOL": ">7"} {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "TTM": "1", "EVOL": ">7"}	
	r = {'HbA1c': '>8.000000', 'EVOL': '>4'} → {'RD': 1} delta {'HbA1c': '<=8.0', 'TTM': '0'} delta {'EVOL': '<=4.0', 'TTM': '0'}	(HbA1c >= 9.0) & (EVOL >= 7.0) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 4.0) => (RD <= 0.0)

5.4.3. Patient 10

RETIPROGRAM predicts class 0

Age	Sex	EVOL	TTM	HbA1c	CDKEPI	MA	BMI	HTAR	RD
72	**1**	**5**	**1**	**6,6**	**90,86**	**0**	**32,04**	**1**	**0**

Table 11: Rule comparison for Patient 10

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2", "EVOL": ">5"} {"RD": 0}, {"TTM": "2", "EVOL": "<=5", "HbA1c": "<=6.600000"} {"RD": 1}, {"TTM": "2", "EVOL": "<=5", "HbA1c": ">6.600000"}	(TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM >= Insuline) & (BMI = Overweight-High) => (RD >= Class1) (TTM >= Insuline) & (Sex = Woman) & (CDKEPI = High) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
	r = {'TTM': '1'} → {'RD': 0} delta {'TTM': '2', 'EVOL': '>5.0'}	(TTM >= 2.0) & (EVOL >= 8.0) => (RD >= 1.0) (TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (EVOL <= 5.0) & (HbA1c <= 6.6) => (RD <= 0.0)
max	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2", "Age": "<=68.000000"} {"RD": 0}, {"TTM": "2", "Age": ">68.000000", "HbA1c": "<=6.600000"} {"RD": 1}, {"TTM": "2", "Age": ">68.000000", "HbA1c": ">6.600000"}	(MA >= High) & (TTM >= Insuline) => (RD >= Class1) (CDKEPI = Normal) & (TTM >= Insuline) => (RD >= Class1) (TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM >= Insuline) & (BMI = Overweight-High) => (RD >= Class1) (TTM >= Insuline) & (Sex = Woman) & (CDKEPI = High) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
	r = {'TTM': '1'} → {'RD': 0} delta {'TTM': '2', 'Age': '<=68.0'}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (TTM >= 2.0) & (EVOL >= 8.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 6.6) & (EVOL <= 5.0) => (RD <= 0.0)

mix	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2", "HbA1c": ">6.600000"} {"RD": 1}, {"TTM": "2", "HbA1c": "<=6.600000", "Age": "<=68.000000"} {"RD": 0}, {"TTM": "2", "HbA1c": "<=6.600000", "Age": ">68.000000"}	(TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM >= Insuline) & (BMI = Overweight-High) => (RD >= Class1) (TTM >= Insuline) & (Sex = Woman) & (CDKEPI = High) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
	r = {'TTM': '1'} → {'RD': 0} delta {'TTM': '2', 'HbA1c': '>6.6'}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (TTM >= 2.0) & (EVOL >= 8.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 6.6) & (EVOL <= 5.0) => (RD <= 0.0)

5.4.4. Patient 13

RETIPROGRAM predicts class 1

Age	Sex	EVOL	TTM	HbA1c	CDKEPI	MA	BMI	HTAR	RD
76	1	1	2	14,4	91,4	0	29,49	1	0

Table 12: Rule comparison for Patient 13

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 1}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= OralAntidiab) => (RD = 1) (TTM <= Diet) => (RD <= 0)
	r = {'TTM': '2'} → {'RD': 1} delta {'TTM': '0'}	(TTM >= 1.0) => (RD >= 1.0) (TTM <= 0.0) => (RD <= 0.0)
max	{"RD": 0}, {"TTM": "0"}, {"RD": 1}, {"TTM": "2"}, {"RD": 0}, {"TTM": "1", "CDKEPI": "<=69.000000"} {"RD": 1}, {"TTM": "1", "CDKEPI": ">69.000000"}	(TTM >= Insuline) => (RD = 1) (CDKEPI = VeryHigh) => (RD = 1) (TTM <= Diet) => (RD <= 0) (TTM <= OralAntidiab) & (CDKEPI = Normal) => (RD <= 0)
	r = {'TTM': '2'} → {'RD': 1} delta {'TTM': '0'}	(TTM >= 2.0) => (RD >= 1.0) (CDKEPI_g >= 91.4) => (RD >= 1.0) (TTM <= 0.0) => (RD <= 0.0) (TTM <= 1.0) & (CDKEPI_g <= 58.0) => (RD <= 0.0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 1}, {"TTM": "2"} {"RD": 0}, {"TTM": "1", "CDKEPI": "<=69.000000"} {"RD": 1}, {"TTM": "1", "CDKEPI": ">69.000000"}	(TTM >= OralAntidiab) => (RD >= Class1) (TTM <= Diet) => (RD <= Class0)
	r = {'TTM': '2'} → {'RD': 1} delta {'TTM': '0'}	(TTM >= 2.0) => (RD >= 1.0) (TTM >= 1.0) & (CDKEPI_g >= 91.4) => (RD >= 1.0) TTM <= 0.0) => (RD <= 0.0) (TTM <= 1.0) & (CDKEPI_g <= 58.0) => (RD <= 0.0)

5.4.5. Comments

The 4 patients whose rules are being compared in this section have been chosen in order to review different situations this system can encounter, specifically when the Retiprogram predicts a correctly an instance with class 1, Table 9, when it predicts correctly an instance with class 0, Table 11, and when it predicts incorrectly, Tables 10 and 12. These last 2 Patients are both instances where the Retiprogram predicts class 1 when in reality the patient doesn't suffer from DR. The opposite case would have been reviewed, an instance with DR where the Retiprogram doesn't predict it, but there was no such instance in the training data.

From Tables 9 to 12 we can observe some insights, such as the best rule for GLORE also apprearing in the DRSA-based approach most of the time, gives validation to the method developed in this project. Both methods usually pick the same attributes for their rules. The rules generated with both methods are short, most of the time being the combination of 2 attributes, which makes them easy to comprehend to doctors and patients alike, so the rules tend to be quite explainable. Although the rules are usually explainable, the fuzzy version of the DRSA approach is the most explainable of the three, due to the labels of its categories. Thanks to that, the exact number for each rule cannot be a source of confusion, and these rules do not require anything more in order to be readable.

If we take a look at the most used attributes for rule generation in Tables 9 to 12, we can observe that TTM is the most popular one in both GLORE and DRSA approaches, followed by HbA1c and CDKEPI. Although TTM taking the first spot is not ideal because changing the treatment type of a patient is not possible without dangerous complications, and will actually be detrimental to the development of DR, the appearence of HbA1c and CDKEPI is positive. GLORE rules mention other attributes, like Age, more often than DRSA, although none rely too much on them. Other attributes that do not appear often are HTAR and Sex, probably because of the change in the order that we implemented after evaluating the neighbours.

An ophtalmologist doctor, expert in Diabetic Retinopathy, has analysed the explanatory rule and counterfactuals of GLORE and the fuzzy explanatory rules of DRSA and his conclusions are the following:

- The infomation provided by GLORE is too simple. Most of the times the explanatory rule contains a single attribute, which is not giving many clues to the doctors about the reasons for the class obtained.
- GLORE explanations mainly concentrate on TTM (treatment type) variable, which is certainly a factor indicating the conditions of a diabetic person, much related to appearance of DR, but it is a factor that cannot be modified. So, it is not useful to try to improve the patient's State.
- DRSA fuzzy rules also use TTM but include other factors like EVOL, HBA1 and CKDEPI. The two last are relevant attributes that can be modified with proper treatment. The evolution time is not possible to decrease, but it is certainly also a variable that influences the appearance of DR (longer diabetis induces more complications).
- The explanatory Black rules about the class predicted by Retiprogram are totally confirmed by the doctors.

- However, the Blue rules indicating how to modify the patients' conditions to change the class should not include variables that cannot be controled, like age, sex, evolution time, or treatment type.
- In patient 8 the Retiprogram predicts class 1 but the diagnosis made by the doctors is class 0. The doctor, thanks to the DRSA fuzzy rules, has seen that in this case the main reason for the output is the high HbA1 value and its medication with insuline (since it is an old patient with diabetis for a long time). However, this high value in HbA1 could be just temporal due to some short period of bad control in medication, since the MA and other indicators is correct. So, it seems that HbA1 is an anomalous value taking into account the rest of values of this patient. The rules indicating that the patient should decrease HbA1c and keep CDKEPI in normal conditions are correct in the max version (as min version includes more rules with sex).
- In patient 13, the Retiprogram again predicts class 1 while the diagnosis is class 0. GLORE does not provide many information, but the DRSA fuzzy rules highlight the high value in the CDKEPI atribuí as cause of the DR, together with the insuline medication. According to the doctor, this old person has been diagnosed recently (1 year of evolution) for this reason is taking insuline and has anormal values in HbA1c and CKDEPI, because they are still not controlled in an appropriate way. However, DR may not be present in this case as it is a new diabetic person and DR cannot appear so early. Therefore, the information given by DRSA rules is correct and helps to the interpretation of this case. With appropriate treatment the patient can leave insuline and just take oral antidiabetic medication soon (as the system also proposes in the max version).

6. Conclusion

Considering the developed methodologies and the obtained results, it can be concluded that:

- A DRSA-based approach for rule generation adds a layer of explainability to FRF, as this was the main objective of the project. The rules it generates are short, usually involving no more than 2 attributes, and especially in the fuzzy version are readable and understandable from its generation. This provides the level of explainability the project was looking for, that can help both doctors and patients.
- After comparing the DRSA-based approach with an established and tested method such as GLORE, the results shown rules similar enough to validate the method used in this project, but still with some differences that make this approach worth exploring, due to the inclusion of relevant factors in the rules that GLORE sometimes misses, as reviewed by an ophtalmologist specialised in DR.
- Of the three approaches taken to generate the neighbour files, it appears that taking the furthest opposite instance, or max in this project, is the one that results in more rules and explainability, and usually involves more relevant attributes, while depending less on attributes that fall outside of the control of the doctors and patients. Having said that, further research should be made before electing one as supperior.

6.1. Future work

As further work, several research lines are proposed:

- Modify some of the features, so the ones that cannot be altered are not used for neighbour generation, and others can only be changed in one direction, i.e. the type of treatment.
- Change the pipeline of the whole process so it uses the library RuleLearn instead of the GUI of RuleStudio, so obtaining the rules for a patient is an automatic process.
- Further analyse the changes between the 3 models for choosing the opposite instance.

6.2. Personal opinion

Personally, I have enjoyed the development of this project. Not only has it allowed me to work and learn more in ML, which is already an exciting field, but also in a project that involves public health trying to help with real issues. This project works with a quality of screening data that is not usually available, so being able to work with it and contributing to this research group was an oportunity I liked.

Most of the original research for this project was made during an Erasmus mobility grant in Poznań, Poland, in the Poznań University of technology (PUT). PUT is the university where RuleLearn and RuleStudio were developed, and as such this project is a combined effort between URV and PUT. I would like to thank specifically Marcin Szeląg for his help in the parts of this project that involved RuleStudio.

7. References

[1] Y. Zheng, M. He, and N. Congdon, “The worldwide epidemic of diabetic retinopathy,” *Indian J. Ophthalmol.*, vol. 60, no. 5, pp. 428–431, Sep. 2012, doi: 10.4103/0301-4738.100542.

[2] S. Jones and RT. Edwards, “Diabetic retinopathy screening: a systematic review of the economic evidence,” *Diabet. Med.*, vol. 27, no. 3, pp. 249–256, 2010, doi: 10.1111/J.1464-5491.2009.02870.X.

[3] T. T. Nguyen, J.J. Wang, A.R. Sharret, F.M. Islam, R. Klein, B.E.K. Klein, M.F. Cotch and T.Y. Wong, “Relationship of Retinal Vascular Caliber With Diabetes and Retinopathy” *Diabetes Care*, vol. 31, no. 3, pp. 544–549, Mar. 2008, doi: 10.2337/DC07-1528.

[4] J. Lechner, O. O’Leary, and AW Stitt, “The pathology associated with diabetic retinopathy,” *Vision Res.*, vol. 139, pp. 7–14, Oct. 2017, doi: 10.1016/J.VISRES.2017.04.003.

[5] M. M. Khansari, J. Wanek, M. Tan, C.E. Joslin, J. J. Kresovich, N.Camardo, N.P.-Blair, M.Shahidi , “Assessment of Conjunctival Microvascular Hemodynamics in Stages of Diabetic Microvasculopathy,” *Sci. Rep.*, vol. 7, Apr. 2017, doi: 10.1038/SREP45916.

[6] S. Vujosevic, S.- Aldington, P. Silva, C. Hernández, P. Scanlon, T. Peto, and R. Simó, “Screening for diabetic retinopathy: new perspectives and challenges,” *lancet. Diabetes Endocrinol.*, vol. 8, no. 4, pp. 337–347, Apr. 2020, doi: 10.1016/S2213-8587(19)30411-5.

[7] “What is Machine Learning? | IBM.” https://www.ibm.com/cloud/learn/machine-learning (accessed Aug. 02, 2021).

[8] M. I. Jordan and C. M. Bishop, “Neural Networks,” Mar. 1996, Accessed: Aug. 03, 2021. [Online]. Available: https://dspace.mit.edu/handle/1721.1/7186.

[9] C. Molnar, *Interpretable Machine Learning*. 2019.

[10] D. W. Apley and J. Zhu, “Visualizing the effects of predictor variables in black box supervised learning models,” *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 82, no. 4, pp. 1059–1086, Sep. 2020, doi: 10.1111/RSSB.12377.

[11] P. Romero-Aroca, A. Valls, A. Moreno, R. Sagarra-Alamo, J. Basora-Gallisa, E. Saleh, M. Baget.Bernaliz, D. Puig, “A Clinical Decision Support System for Diabetic Retinopathy Screening: Creating a Clinical Support Application,” *Telemed. J. E. Health.*, vol. 25, no. 1, pp. 31–40, Jan. 2019, doi: 10.1089/TMJ.2017.0282.

[12] E. Saleh J. Blaszcynski, A. Moreno, A. Valls, P. Romero-Aroca, S. Riva-Fernandez, R. Slowinski , "Learning ensemble classifiers for diabetic retinopathy assessment" *Artif. Intell. Med.*, vol. 85, pp. 50–63, Apr. 2018, doi: 10.1016/J.ARTMED.2017.09.006.

[13] P. Romero-Aroca, M. Baget-Bernaliz, R. Navarro-Gil, A. Moreno, A. Valls, R. Sagarra-Alamo, j. Barrot and X. Mundet, "Glomerular filtration rate and/or ratio of urine albumin to creatinine as markers for diabetic retinopathy: A ten-year follow-up study," *J. Diabetes Res.*, vol. 2018, 2018, doi: 10.1155/2018/5637130.

[14] P. Romero-Aroca, S. Riva-Fernandez, A. Valls, R. Sagarra-Alamo, A. Moreno, and N. Soler, "Changes observed in diabetic retinopathy: eight-year follow-up of a Spanish population," *Br. J. Ophthalmol.*, vol. 100, no. 10, pp. 1366–1371, Oct. 2016, doi: 10.1136/BJOPHTHALMOL-2015-307689.

[15] P. Romero-Aroca, S. Riva-Fernandez, A. Valls, R. Sagarra-Alamo, A. Moreno, N. Soler and D. Puig, "Cost of diabetic retinopathy and macular oedema in a population, an eight year follow up," *BMC Ophthalmol.*, vol. 16, no. 1, Aug. 2016, doi: 10.1186/S12886-016-0318-X.

[16] E. Vittinghoff and C. E. McCulloch, "Relaxing the Rule of Ten Events per Variable in Logistic and Cox Regression," *Am. J. Epidemiol.*, vol. 165, no. 6, pp. 710–718, Mar. 2007, doi: 10.1093/AJE/KWK052.

[17] Şebnem Yılmaz Balaman, "Decision-Making for Biomass-Based Production Chains," *Decis. Biomass-Based Prod. Chain.*, 2019, doi: 10.1016/C2017-0-00910-2.

[18] N. Maaroof, A. Moreno, A. Valls, and M. Jabreel, "Guided-LORE: Improving LORE with a Focused Search of Neighbours," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12641 LNAI, pp. 49–62, Sep. 2020, doi: 10.1007/978-3-030-73959-1_4.

[19] J. Błaszczyński, R. Słowiński, and M. Szelag, "Sequential covering rule induction algorithm for variable consistency rough set approaches," *Inf. Sci. (Ny).*, vol. 181, no. 5, pp. 987–1002, Mar. 2011, doi: 10.1016/J.INS.2010.10.030.

[20] R. Słowiński, S. Greco, and B. Matarazzo, "Rough Set Methodology for Decision Aiding," *Springer Handb. Comput. Intell.*, pp. 349–370, 2015, doi: 10.1007/978-3-662-43505-2_22.

ANNEX

7.1. Comparison of rules

Patient 1

	GLORE rules	DRSA rules
min	"RD": 1 "EVOL": ">5", "HbA1c": ">9.000000" "RD": 1 "EVOL": ">5", "HbA1c": "8.0< HbA1c <=9.0" "RD": 0 "EVOL": ">5", "HbA1c": "<=8.0" "RD": 0 "EVOL": "<=5"	(EVOL >= 11.0) => (RD >= 1.0) (TTM >= 2.0) & (HbA1c >= 10.0) => (RD >= 1.0) (EVOL >= 8.0) & (HbA1c >= 9.0) & (TTM >= 1.0) => (RD >= 1.0) (HbA1c <= 8.0) => (RD <= 0.0) (EVOL <= 5.0) & (TTM <= 1.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 8.0) => (RD <= 0.0)
	{'EVOL': '>5', 'HbA1c': '>9.000000'} --> {'RD': 1} delta {'EVOL': '<=5.0'} delta {'HbA1c': '<=8.0'} {'EVOL': '<=5.0'} → RD:0 {'HbA1c': '<=8.0'} → RD:0	(EVOL >= 10to15) => (RD >= Class1) (TTM >= Insuline) & (HbA1c >= More9) => (RD >= Class1) (EVOL >= 5to10) & (HbA1c >= 8to9) & (TTM >= OralAntidiab) => (RD >= Class1) (HbA1c <= 7to8) => (RD <= Class0) (EVOL <= Less5) & (TTM <= OralAntidiab) => (RD <= Class0) (TTM <= Diet) & (EVOL <= 5to10) => (RD <= Class0)
max	"RD": 1 "EVOL": "<=8", "CDKEPI": ">97.000000" "RD": 1 "EVOL": ">8", "HbA1c":">9.000000" "RD": 1 "EVOL": "<=8", "CDKEPI": "<=97.000000", "TTM": "2" "RD": 1 "EVOL": "<=8", "CDKEPI": "<=97.000000", "TTM": "1", "HbA1c": ">9.000000" "RD": 1 "EVOL": ">8", "HbA1c": "8.0< HbA1c <=9.0"	(CDKEPI >= 108.93) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 9.0) => (RD >= 1.0) (EVOL >= 11.0) & (HbA1c >= 10.0) => (RD >= 1.0) (HbA1c <= 8.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 8.0) => (RD <= 0.0)

	"RD": 0 "EVOL": ">8", "HbA1c":"<=8.0" "RD": 0 "EVOL": "<=8", "CDKEPI": "<=97.000000", "TTM": "1", "HbA1c": "<=9.000000" "RD": 0 "EVOL": "<=8", "CDKEPI": "<=97.000000", "TTM": "0"	
	{'EVOL': '>8', 'HbA1c': '>9.000000'} --> {'RD': 1} delta {'HbA1c': '<=8.0'} {'HbA1c': '<=8.0'} → RD:0	(TTM >= OralAntidiab) & (HbA1c >= 8to9) => (RD >= Class1) (EVOL >= 10to15) & (HbA1c >= 8to9) => (RD >= Class1) (HbA1c <= 7to8) => (RD <= Class0) (TTM <= Diet) & (EVOL <= 5to10) => (RD <= Class0)
mix	"RD": 1 "EVOL": "<=8", "TTM": "2" "RD": 1"EVOL": ">8", "HbA1c": ">9.000000" "RD": 1 "EVOL": "5.0< EVOL <=8.0", "TTM": "1" "RD": 1 "EVOL": ">8", "HbA1c": "8.0< HbA1c <=9.0" "RD": 0 "EVOL": "<=8", "TTM": "0" "RD": 0 "EVOL": "<=5.0", "TTM": "1"}, "RD": 0 "EVOL": ">8", "HbA1c": "<=8.0"	(EVOL >= 11.0) & (HbA1c >= 9.0) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 10.0) => (RD >= 1.0) (TTM >= 1.0) & (EVOL >= 8.0) & (HbA1c >= 9.0) => (RD >= 1.0) (HbA1c <= 8.0) => (RD <= 0.0) (EVOL <= 5.0) & (TTM <= 1.0) => (RD <= 0.0) (EVOL <= 8.0) & (TTM <= 0.0) => (RD <= 0.0)
	{'EVOL': '>8', 'HbA1c': '>9.000000'} --> {'RD': 1} delta {'HbA1c': '<=8.0'} {'HbA1c': '<=8.0'} → RD:0	(EVOL >= 10to15) => (RD >= Class1) (TTM >= Insuline) & (HbA1c >= More9) => (RD >= Class1) (EVOL >= 5to10) & (HbA1c >= 8to9) & (TTM >= OralAntidiab) => (RD >= Class1) (HbA1c <= 7to8) => (RD <= Class0) (EVOL <= Less5) & (TTM <= OralAntidiab) => (RD <= Class0) (TTM <= Diet) & (EVOL <= 5to10) => (RD <= Class0) [

Patient 2

	GLORE rules	DRSA rules
min	{"RD": 1}, {"HbA1c": ">7.000000"} {"RD": 0}, {"HbA1c": "<=7.000000", "TTM": "0"} {"RD": 0}, {"HbA1c": "<=7.000000", "TTM": "1"} {"RD": 1}, {"HbA1c": "<=7.000000", "TTM": "2"}	(HbA1c >= 8.0) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (HbA1c <= 7.0) & (TTM <= 1.0) => (RD <= 0.0)
	r = {'HbA1c': '>7.000000'} --> {'RD': 1} delta {'HbA1c': '<=7.0', 'TTM': '0'} delta {'HbA1c': '<=7.0', 'TTM': '1'}	(HbA1c >= 7to8) => (RD >= Class1) TTM >= Insuline) => (RD >= Class1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <= Class0)
max	{"RD": 1}, {"HbA1c": ">7.000000"} {"RD": 0}, {"HbA1c": "<=7.000000", "TTM": "0"}, {"RD": 0}, {"HbA1c": "<=7.000000", "TTM": "1"}, {"RD": 1}, {"HbA1c": "<=7.000000", "TTM": "2"},	(HbA1c >= 8.0) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (HbA1c <= 7.0) & (TTM <= 1.0) => (RD <= 0.0)
	r = {'HbA1c': '>7.000000'} --> {'RD': 1} delta {'HbA1c': '<=7.0', 'TTM': '0'} delta {'HbA1c': '<=7.0', 'TTM': '1'}	(HbA1c >= 7to8) => (RD >= Class1) TTM >= Insuline) => (RD >= Class1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <= Class0)
mix	{"RD": 1}, {"HbA1c": ">7.000000"} {"RD": 0}, {"HbA1c": "<=7.000000", "TTM": "0"} {"RD": 0}, {"HbA1c": "<=7.000000", "TTM": "1"} {"RD": 1}, {"HbA1c": "<=7.000000", "TTM": "2"}	(HbA1c >= 8.0) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (HbA1c <= 7.0) & (TTM <= 1.0) => (RD <= 0.0)
	r = {'HbA1c': '>7.000000'} --> {'RD': 1} delta {'HbA1c': '<=7.0', 'TTM': '0'} delta {'HbA1c': '<=7.0', 'TTM': '1'}	(HbA1c >= 7to8) => (RD >= Class1) TTM >= Insuline) => (RD >= Class1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <= Class0)

Patient 3

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 1}, {"TTM": "2"} {"RD": 0}, {"TTM": "1", "HbA1c": "<=7.000000"} {"RD": 1}, {"TTM": "1", "HbA1c": ">7.000000"}	(TTM >= 2.0) => (RD >= 1.0) (BMI_g >= 26.0) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 8.2) => (RD >= 1.0) (TTM <= 0.0) => (RD <= 0.0) (TTM <= 1.0) & (HbA1c <= 7.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'}	(TTM >= Insuline) => (RD >= Class1) (BMI = Overweight-Low) => (RD >= Class1) (TTM >= OralAntidiab) & (HbA1c >= 8to9) => (RD >= Class1) (TTM <= Diet) => (RD <= Class0) (TTM <= OralAntidiab) & (HbA1c <= 6to7) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"}, {"RD": 1}, {"TTM": "2"}, {"RD": 0}, {"TTM": "1", "HbA1c": "<=7.000000"}, {"RD": 1}, {"TTM": "1", "HbA1c": ">7.000000"},	(TTM >= 2.0) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (TTM <= 0.0) => (RD <= 0.0) (TTM <= 1.0) & (HbA1c <= 7.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'}	(TTM >= Insuline) => (RD >= Class1) (TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD >= Class1) (TTM <= Diet) => (RD <= Class0) (TTM <= OralAntidiab) & (HbA1c <= 6to7) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 1}, {"TTM": "2"} {"RD": 0}, {"TTM": "1", "HbA1c": "<=7.000000"} {"RD": 1}, {"TTM": "1", "HbA1c": ">7.000000"}	(TTM >= 2.0) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (TTM <= 0.0) => (RD <= 0.0) (TTM <= 1.0) & (HbA1c <= 7.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'}	(TTM >= Insuline) => (RD >= Class1) (BMI = Overweight-Low) => (RD >= Class1) (TTM >= OralAntidiab) & (HbA1c >= 8to9)

		=> (RD >= Class1) (TTM <= Diet) => (RD <= Class0) (TTM <= OralAntidiab) & (HbA1c <= 6to7) => (RD <= Class0)

Patient 4

	GLORE rules	DRSA rules
min	{"RD": 1}, {"TTM": "2"} {"RD": 0}, {"TTM": "0", "HbA1c": "<=8.000000"} {"RD": 0}, {"TTM": "1", "HbA1c": "<=7.000000"} {"RD": 1}, {"TTM": "1", "HbA1c": ">7.000000"} {"RD": 0}, {"TTM": "0", "HbA1c": ">8.000000", "EVOL": "<=4"} {"RD": 1}, {"TTM": "0", "HbA1c": ">8.000000", "EVOL": ">4"}	(TTM >= 2.0) & (EVOL >= 7.0) => (RD >= 1.0) (HbA1c >= 9.0) & (EVOL >= 7.0) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (HbA1c <= 7.0) & (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 7.0) & (EVOL <= 4.0) => (RD <= 0.0) (TTM <= 0.0) & (HbA1c <= 8.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 4.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0', 'HbA1c': '<=8.0'} delta {'TTM': '1', 'HbA1c': '<=7.0'}	(TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD >= Class1) (HbA1c >= 8to9) & (EVOL >= 5to10) => (RD >= Class1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <= Class0) (HbA1c <= 6to7) & (EVOL <= Less5) => (RD <= Class0) (TTM <= Diet) & (HbA1c <= 7to8) => (RD <= Class0) (TTM <= Diet) & (EVOL <= Less5) => (RD <= Class0)
max	{"RD": 1}, {"HbA1c": ">8.000000"}, {"RD": 1}, {"HbA1c": "<=8.000000", "CDKEPI": ">109.000000"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKEPI": "<=109.000000", "TTM": "0"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKEPI": "<=109.000000", "TTM": "1"}, {"RD": 1}, {"HbA1c":	(HbA1c >= 9.0) => (RD >= 1.0) (CDKEPI_g >= 120.01) => (RD >= 1.0) (TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (HbA1c <= 6.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 1.0) => (RD <= 0.0)

	"<=8.000000", "CDKEPI": "<=109.000000", "TTM": "2"},	
	r = {'HbA1c': '>8.000000'} --> {'RD': 1} delta {'HbA1c': '<=8.0', 'CDKEPI': '<=109.0', 'TTM': '0'} delta {'HbA1c': '<=8.0', 'CDKEPI': '<=109.0', 'TTM': '1'}	(HbA1c >= 8to9) => (RD >= Class1) (TTM >= Insuline) & (HbA1c >= 7to8) => (RD >= Class1) (TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (HbA1c >= 7to8) & (TTM >= OralAntidiab) & (Age >= Forties) => (RD >= Class1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <= Class0) (HbA1c <= 6to7) & (EVOL <= Less5) => (RD <= Class0) (TTM <= Diet) & (HbA1c <= 7to8) => (RD <= Class0)
mix	{"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "0"} {"RD": 1}, {"HbA1c": "<=8.000000", "TTM": "2"} {"RD": 0}, {"HbA1c": ">8.000000", "MA": ">30"} {"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "1", "CDKEPI": "<=109.000000"} {"RD": 1}, {"HbA1c": ">8.000000", "MA": "<=30", "EVOL": ">4"} {"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "1", "CDKEPI": ">109.000000", "EVOL": "<=4"} {"RD": 1}, {"HbA1c": "<=8.000000", "TTM": "1", "CDKEPI": ">109.000000", "EVOL": ">4"} {"RD": 0}, {"HbA1c": ">8.000000", "MA": "<=30", "EVOL": "<=4", "TTM": "0"} {"RD": 1}, {"HbA1c": ">8.000000", "MA": "<=30", "EVOL": "<=4", "TTM": "1"} {"RD": 1}, {"HbA1c": ">8.000000", "MA": "<=30", "EVOL": "<=4", "TTM": "2"}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (HbA1c >= 9.0) & (TTM >= 1.0) => (RD >= 1.0) (HbA1c >= 9.0) & (EVOL >= 7.0) => (RD >= 1.0) (CDKEPI_g >= 120.01) & (TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (HbA1c <= 6.0) => (RD <= 0.0) (HbA1c <= 7.0) & (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 4.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 1.0) & (CDKEPI_g <= 109.0) => (RD <= 0.0)
	r = {'HbA1c': '>8.000000', 'MA':	(TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1)

	'<=30', 'EVOL': '>4'} --> {'RD': 1} delta {'MA': '>30.0'}	(TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD >= Class1) (HbA1c >= 8to9) & (EVOL >= 5to10) => (RD >= Class1) (HbA1c <= 6to7) & (TTM <= OralAntidiab) => (RD <= Class0) (HbA1c <= 6to7) & (EVOL <= Less5) => (RD <= Class0) (TTM <= Diet) & (HbA1c <= 7to8) => (RD <= Class0) (TTM <= Diet) & (EVOL <= Less5) => (RD <= Class0)

Patient 5

	GLORE rules	DRSA rules
min	{"RD": 1}, {"EVOL": ">7"} {"RD": 0}, {"EVOL": "<=7", "TTM": "0"} {"RD": 0}, {"EVOL": "<=7", "TTM": "1"} {"RD": 1}, {"EVOL": "<=7", "TTM": "2"}	(EVOL >= 10.0) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (TTM >= 1.0) & (EVOL >= 7.0) => (RD >= 1.0) (EVOL <= 4.0) & (TTM <= 1.0) => (RD <= 0.0) (EVOL <= 7.0) & (TTM <= 0.0) => (RD <= 0.0)
	{r = {'EVOL': '>7'} --> {'RD': 1} delta {'EVOL': '<=7.0', 'TTM': '0'} delta {'EVOL': '<=7.0', 'TTM': '1'}	(TTM >= Insuline) => (RD >= Class1) (EVOL >= 10to15) => (RD >= Class1) (TTM >= OralAntidiab) & (EVOL >= 5to10) => (RD >= Class1) (Sex = Woman) & (EVOL >= 5to10) => (RD >= Class1) (EVOL <= Less5) & (TTM <= OralAntidiab) => (RD <= Class0) (HbA1c <= 8to9) & (TTM <= Diet) => (RD <= Class0)
max	{"RD": 1}, {"EVOL": ">7"}, [{"RD": 0}, {"EVOL": "<=7", "TTM": "0"}, {"RD": 0}, {"EVOL": "<=7", "TTM": "1"}, {"RD": 1}, {"EVOL": "<=7", "TTM": "2"},	(EVOL >= 10.0) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (CDKEPI_g >= 78.06) & (CDKEPI_c <= 78.06) => (RD >= 1.0) (EVOL <= 4.0) & (TTM <= 1.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 7.0) => (RD <= 0.0)
	r = {'EVOL': '>7'} --> {'RD': 1} delta {'EVOL': '<=7.0', 'TTM': '0'} delta {'EVOL': '<=7.0', 'TTM': '1'}	(TTM >= Insuline) => (RD >= Class1) (EVOL >= 10to15) => (RD >= Class1) (TTM >= OralAntidiab) & (EVOL >= 5to10) => (RD >= Class1) (Sex = Woman) & (EVOL >= 5to10) => (RD >= Class1) (EVOL <= Less5) & (TTM <= OralAntidiab) => (RD <= Class0) (HbA1c <= 8to9) & (TTM <= Diet) => (RD <= Class0)
mix	{"RD": 1}, {"EVOL": ">7"}	(EVOL >= 10.0) => (RD >= 1.0)

	{"RD": 0}, {"EVOL": "<=7", "TTM": "0"} {"RD": 0}, {"EVOL": "<=7", "TTM": "1"} {"RD": 1}, {"EVOL": "<=7", "TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM >= 1.0) & (EVOL >= 7.0) => (RD >= 1.0) (EVOL <= 4.0) & (TTM <= 1.0) => (RD <= 0.0) (EVOL <= 7.0) & (TTM <= 0.0) => (RD <= 0.0)
	r = {'EVOL': '>7'} --> {'RD': 1} delta {'EVOL': '<=7.0', 'TTM': '0'} delta {'EVOL': '<=7.0', 'TTM': '1'}	(TTM >= Insuline) => (RD >= Class1) (EVOL >= 10to15) => (RD >= Class1) (TTM >= OralAntidiab) & (EVOL >= 5to10) => (RD >= Class1) (Sex = Woman) & (EVOL >= 5to10) => (RD >= Class1) (EVOL <= Less5) & (TTM <= OralAntidiab) => (RD <= Class0) (HbA1c <= 8to9) & (TTM <= Diet) => (RD <= Class0)

Patient 6

	GLORE rules	DRSA rules
min	{"RD": 1}, {"TTM": "2"} {"RD": 0}, {"TTM": "0", "HbA1c": "<=8.000000"} {"RD": 1}, {"TTM": "1", "Sex": "0"} {"RD": 0}, {"TTM": "0", "HbA1c": ">8.000000", "EVOL": "<=4"} {"RD": 1}, {"TTM": "0", "HbA1c": ">8.000000", "EVOL": ">4"} {"RD": 1}, {"TTM": "1", "Sex": "1", "Age": "<=72.000000"} {"RD": 0}, {"TTM": "1", "Sex": "1", "Age": ">72.000000", "HbA1c": "<=8.000000"} {"RD": 1}, {"TTM": "1", "Sex": "1", "Age": ">72.000000", "HbA1c": ">8.000000"}	(TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (HbA1c >= 9.0) & (EVOL >= 7.0) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 4.0) => (RD <= 0.0)
	r = {'TTM': '1', 'Sex': '1', 'Age': '>72.000000', 'HbA1c': '>8.000000'} --> {'RD': 1} delta {'HbA1c': '<=8.0'}	(TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD >= Class1) (HbA1c >= 8to9) & (EVOL >= 5to10) => (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0) (TTM <= Diet) & (EVOL <= Less5) => (RD <= Class0)
max	{"RD": 1}, {"HbA1c": ">8.000000"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKEPI": "<=61.000000"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKEPI": "61.0< CDKEPI <=83.0"}, {"RD": 1}, {"HbA1c": "<=8.000000", "CDKEPI": ">83.0"},	(HbA1c >= 9.0) => (RD >= 1.0) (CDKEPI_g >= 83.01) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0)
	r = {'HbA1c': '>8.000000'} --> {'RD': 1} delta {'HbA1c': '<=8.0', 'CDKEPI':	(HbA1c >= 8to9) => (RD >= Class1) (HbA1c >= 7to8) & (TTM >= OralAntidiab)

	'<=61.0'}	=> (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0)
mix	{"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "0"} {"RD": 1}, {"HbA1c": "<=8.000000", "TTM": "2"} {"RD": 1}, {"HbA1c": ">8.000000", "EVOL": ">4"} {"RD": 0}, {"HbA1c": "<=8.000000", "TTM": "1", "EVOL": "<=7"} {"RD": 0}, {"HbA1c": ">8.000000", "EVOL": "<=4", "TTM": "0"} {"RD": 1}, {"HbA1c": ">8.000000", "EVOL": "<=4", "TTM": "1"} {"RD": 1}, {"HbA1c": ">8.000000", "EVOL": "<=4", "TTM": "2"} {"RD": 0}, {"HbA1c": "<=7.0", "TTM": "1", "EVOL": ">7"} {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "TTM": "1", "EVOL": ">7"}	(HbA1c >= 9.0) & (EVOL >= 7.0) => (RD >= 1.0) (TTM >= 1.0) & (HbA1c >= 8.0) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (TTM <= 0.0) & (EVOL <= 4.0) => (RD <= 0.0)
	r = {'HbA1c': '>8.000000', 'EVOL': '>4'} --> {'RD': 1} delta {'HbA1c': '<=8.0', 'TTM': '0'} delta {'EVOL': '<=4.0', 'TTM': '0'}	(TTM >= OralAntidiab) & (HbA1c >= 7to8) => (RD >= Class1) (HbA1c >= 8to9) & (EVOL >= 5to10) => (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0) (TTM <= Diet) & (EVOL <= Less5) => (RD <= Class0)

Patient 7

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"}, {"RD": 0}, {"TTM": "1"}, {"RD": 1}, {"TTM": "2"},	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)

patient 8

	GLORE rules	DRSA rules
min	{"RD": 1}, {"HbA1c": ">8.000000"} {"RD": 0}, {"HbA1c": "<=7.0"} {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": "<=19"} {"RD": 0}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": ">19", "TTM": "0"} {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": ">19", "TTM": "2"} {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": ">19", "TTM": "1", "MA": "<=15"} {"RD": 0}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": ">19", "TTM": "1", "MA": ">15"}	(HbA1c >= 9.0) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (CDKEPI_c <= 65.0) & (HbA1c >= 8.0) & (TTM >= 1.0) => (RD >= 1.0) (CDKEPI_c <= 76.71) & (CDKEPI_g >= 76.71) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0) (CDKEPI_c >= 98.0) & (HbA1c <= 8.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (HbA1c <= 8.0) & (Sex_c >= 1.0) => (RD <= 0.0) (HbA1c <= 8.0) & (BMI_g <= 30.0) & (CDKEPI_c >= 87.0) => (RD <= 0.0) (HbA1c <= 8.0) & (BMI_c >= 31.0) & (CDKEPI_c >= 87.0) => (RD <= 0.0) (HbA1c <= 8.0) & (CDKEPI_g <= 76.0) & (CDKEPI_c >= 76.0) => (RD <= 0.0)
	r = {'HbA1c': '>8.000000'} --> {'RD': 1} delta {'HbA1c': '<=7.0'}	(HbA1c >= 8to9) => (RD >= Class1) (TTM >= Insuline) => (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0) (CDKEPI = VeryHigh) & (HbA1c <= 7to8) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0) (HbA1c <= 7to8) & (Sex = Woman) => (RD <= Class0)
max	{"RD": 1}, {"HbA1c": ">8.000000"}, {"RD": 1}, {"HbA1c": "<=8.000000", "CDKEPI": ">76.000000"}, {"RD": 0}, {"HbA1c": "<=8.000000", "CDKEPI": "<=76.000000", "TTM": "0"}, {"RD": 1}, {"HbA1c": "<=8.000000", "CDKEPI": "<=76.000000", "TTM": "2"},	(HbA1c >= 9.0) => (RD >= 1.0) TTM >= 2.0) => (RD >= 1.0) (CDKEPI_g >= 76.71) & (CDKEPI_c <= 76.71) => (RD >= 1.0) (CDKEPI_g >= 87.0) & (HbA1c >= 8.0) & (TTM >= 1.0) => (RD >= 1.0) (Age >= 79.0) & (HbA1c >= 8.0) & (CDKEPI_g >= 65.0) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0)

	{"RD": 0}, {"HbA1c": "<=7.0", "CDKEPI": "<=76.000000", "TTM": "1"}, {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "CDKEPI": "<=76.000000", "TTM": "1", "MA": "<=15"}, {"RD": 0}, {"HbA1c": "7.0< HbA1c <=8.0", "CDKEPI": "<=76.000000", "TTM": "1", "MA": ">15"},	(CDKEPI_g <= 54.0) & (HbA1c <= 8.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (HbA1c <= 8.0) & (BMI_g <= 30.0) & (CDKEPI_g <= 65.0) => (RD <= 0.0)
	r = {'HbA1c': '>8.000000'} --> {'RD': 1} delta {'HbA1c': '<=8.0', 'CDKEPI': '<=76.0', 'TTM': '0'} delta {'HbA1c': '<=7.0', 'CDKEPI': '<=76.0', 'TTM': '1'}	(HbA1c >= 8to9) => (RD >= Class1) (TTM >= Insuline) => (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0) (CDKEPI = Normal) & (HbA1c <= 7to8) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0)
mix	{"RD": 1}, {"HbA1c": ">8.000000"} {"RD": 0}, {"HbA1c": "<=7.0"} {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": "<=19"} {"RD": 0}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": ">19", "TTM": "0"} {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": ">19", "TTM": "2"} {"RD": 1}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": ">19", "TTM": "1", "MA": "<=15"} {"RD": 0}, {"HbA1c": "7.0< HbA1c <=8.0", "EVOL": ">19", "TTM": "1", "MA": ">15"}	(HbA1c >= 9.0) => (RD >= 1.0) (TTM >= 2.0) => (RD >= 1.0) (CDKEPI_g >= 76.71) & (CDKEPI_c <= 76.71) => (RD >= 1.0) (HbA1c >= 8.0) & (TTM >= 1.0) & (CDKEPI_g >= 65.0) & (CDKEPI_c <= 65.0) & (BMI_g >= 30.81) => (RD >= 1.0) (HbA1c <= 7.0) => (RD <= 0.0) (CDKEPI_c >= 98.0) & (HbA1c <= 8.0) => (RD <= 0.0) (CDKEPI_g <= 54.0) & (HbA1c <= 8.0) => (RD <= 0.0) (HbA1c <= 8.0) & (TTM <= 0.0) => (RD <= 0.0) (HbA1c <= 8.0) & (Sex_c >= 1.0) => (RD <= 0.0) (HbA1c <= 8.0) & (BMI_g <= 30.0) & (CDKEPI_g <= 76.0) => (RD <= 0.0) (HbA1c <= 8.0) & (BMI_g <= 30.0) & (CDKEPI_c >= 87.0) => (RD <= 0.0)
	r = {'HbA1c': '>8.000000'} --> {'RD': 1} delta {'HbA1c': '<=7.0'}	(HbA1c >= 8to9) => (RD >= Class1) (TTM >= Insuline) => (RD >= Class1) (HbA1c <= 6to7) => (RD <= Class0)

		(CDKEPI = VeryHigh) & (HbA1c <= 7to8) => (RD <= Class0) (HbA1c <= 7to8) & (TTM <= Diet) => (RD <= Class0) (HbA1c <= 7to8) & (Sex = Woman) => (RD <= Class0)

patient 9

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)

Patient 10

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2", "EVOL": ">5"} {"RD": 0}, {"TTM": "2", "EVOL": "<=5", "HbA1c": "<=6.600000"} {"RD": 1}, {"TTM": "2", "EVOL": "<=5", "HbA1c": ">6.600000"}	(TTM >= 2.0) & (EVOL >= 8.0) => (RD >= 1.0) (TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (EVOL <= 5.0) & (HbA1c <= 6.6) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2', 'EVOL': '>5.0'}	(TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM >= Insuline) & (BMI = Overweight-High) => (RD >= Class1) (TTM >= Insuline) & (Sex = Woman) & (CDKEPI = High) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2", "Age": "<=68.000000"} {"RD": 0}, {"TTM": "2", "Age": ">68.000000", "HbA1c": "<=6.600000"} {"RD": 1}, {"TTM": "2", "Age": ">68.000000", "HbA1c": ">6.600000"}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (TTM >= 2.0) & (EVOL >= 8.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 6.6) & (EVOL <= 5.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2', 'Age': '<=68.0'}	(MA >= High) & (TTM >= Insuline) => (RD >= Class1) (CDKEPI = Normal) & (TTM >= Insuline) => (RD >= Class1) (TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM >= Insuline) & (BMI = Overweight-High) => (RD >= Class1) (TTM >= Insuline) & (Sex = Woman) & (CDKEPI = High) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0)

	{"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2", "HbA1c": ">6.600000"} {"RD": 1}, {"TTM": "2", "HbA1c": "<=6.600000", "Age": "<=68.000000"} {"RD": 0}, {"TTM": "2", "HbA1c": "<=6.600000", "Age": ">68.000000"}	(TTM >= 2.0) & (EVOL >= 8.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 6.6) & (EVOL <= 5.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2', 'HbA1c': '>6.6'}	(TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM >= Insuline) & (BMI = Overweight-High) => (RD >= Class1) (TTM >= Insuline) & (Sex = Woman) & (CDKEPI = High) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)

patient 11

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'} delta {'TTM': '1'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'} delta {'TTM': '1'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'} delta {'TTM': '1'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)

Patient 12

	GLORE rules	DRSA rules
min	{"RD": 1}, {"EVOL": ">14"} {"RD": 0}, {"EVOL": "<=14", "Age": "<=38.000000"} {"RD": 0}, {"EVOL": "<=11.0", "Age": ">38.000000"} {"RD": 1}, {"EVOL": "11.0< EVOL <=14.0", "Age": ">38.000000"}	(EVOL >= 17.0) => (RD >= 1.0) (EVOL >= 14.0) & (Sex_c <= 0.0) => (RD >= 1.0) (EVOL <= 11.0) => (RD <= 0.0) (EVOL <= 14.0) & (Age <= 38.0) => (RD <= 0.0)
	r = {'EVOL': '<=14', 'Age': '<=38.000000'} --> {'RD': 0} delta {}	(EVOL >= 15to20) => (RD >= Class1) (Age >= Forties) & (EVOL >= 10to15) & (TTM >= Insuline) => (RD >= Class1) (EVOL <= 5to10) => (RD <= Class0) (TTM <= OralAntidiab) => (RD <= Class0) (EVOL <= 10to15) & (Age <= Thirties) => (RD <= Class0)
max	{"RD": 0}, {"EVOL": "<=8"} {"RD": 0}, {"EVOL": "8.0< EVOL <=14.0", "Age": "<=38.000000"} {"RD": 1}, {"EVOL": ">14.0", "Age": "<=38.000000"} {"RD": 1}, {"EVOL": ">8", "Age": ">38.000000", "Sex": "0"} {"RD": 0}, {"EVOL": "8.0< EVOL <=11.0", "Age": ">38.000000", "Sex": "1"} {"RD": 1}, {"EVOL": ">11.0", "Age": ">38.000000", "Sex": "1"}	(EVOL >= 17.0) => (RD >= 1.0) (Age >= 42.0) & (EVOL >= 11.0) & (TTM >= 2.0) => (RD >= 1.0) (EVOL <= 8.0) => (RD <= 0.0) (TTM <= 1.0) => (RD <= 0.0) (Age <= 38.0) & (EVOL <= 14.0) => (RD <= 0.0)
	r = {'EVOL': '<=8'} --> {'RD': 0} delta {}	(EVOL >= 15to20) => (RD >= Class1) (Age >= Forties) & (EVOL >= 10to15) & (TTM >= Insuline) => (RD >= Class1) (EVOL <= 5to10) => (RD <= Class0) (TTM <= OralAntidiab) => (RD <= Class0) (Age <= Thirties) & (EVOL <= 10to15) => (RD <= Class0)
mix	{"RD": 0}, {"EVOL": "<=11", "Age": "<=38.000000"} {"RD": 1}, {"EVOL": ">14.0"} {"RD": 0}, {"EVOL": "<=11",	(EVOL >= 17.0) => (RD >= 1.0) (Age >= 42.0) & (EVOL >= 11.0) & (TTM >= 2.0) => (RD >= 1.0) (EVOL <= 8.0) => (RD <= 0.0)

	"Age": ">38.000000", "Sex": "1"} {"RD": 0}, {"EVOL": "11.0< EVOL <=14.0", "Age": "<=38.000000"} {"RD": 1}, {"EVOL": "11.0< EVOL <=14.0", "Age": ">38.000000"} {"RD": 1}, {"EVOL": "<=11", "Age": ">38.000000", "Sex": "0", "BMI": "<=29.000000"} {"RD": 0}, {"EVOL": "<=8.0", "Age": ">38.000000", "Sex": "0", "BMI": ">29.000000"} {"RD": 1}, {"EVOL": "8.0< EVOL <=11.0", "Age": ">38.000000", "Sex": "0", "BMI": ">29.000000"}	(TTM <= 1.0) => (RD <= 0.0) (EVOL <= 14.0) & (Age <= 38.0) => (RD <= 0.0)
	r = {'EVOL': '<=11', 'Age': '<=38.000000'} --> {'RD': 0} delta {}	(EVOL >= 15to20) => (RD >= Class1) (Age >= Forties) & (EVOL >= 10to15) & (TTM >= Insuline) => (RD >= Class1) (EVOL <= 5to10) => (RD <= Class0) (TTM <= OralAntidiab) => (RD <= Class0) (EVOL <= 10to15) & (Age <= Thirties) => (RD <= Class0)

Patient 13

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 1}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 1.0) => (RD >= 1.0) (TTM <= 0.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'}	(TTM >= OralAntidiab) => (RD >= Class1) (TTM <= Diet) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"}, {"RD": 1}, {"TTM": "2"}, {"RD": 0}, {"TTM": "1", "CDKEPI": "<=69.000000"} {"RD": 1}, {"TTM": "1", "CDKEPI": ">69.000000"}	(TTM >= 2.0) => (RD >= 1.0) (CDKEPI_g >= 91.4) => (RD >= 1.0) (TTM <= 0.0) => (RD <= 0.0) (TTM <= 1.0) & (CDKEPI_g <= 58.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'}	(TTM >= Insuline) => (RD >= Class1) (CDKEPI = VeryHigh) => (RD >= Class1) (TTM <= Diet) => (RD <= Class0) (TTM <= OralAntidiab) & (CDKEPI = Normal) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 1}, {"TTM": "2"} {"RD": 0}, {"TTM": "1", "CDKEPI": "<=69.000000"} {"RD": 1}, {"TTM": "1", "CDKEPI": ">69.000000"}	(TTM >= 2.0) => (RD >= 1.0) (TTM >= 1.0) & (CDKEPI_g >= 91.4) => (RD >= 1.0) TTM <= 0.0) => (RD <= 0.0) (TTM <= 1.0) & (CDKEPI_g <= 58.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {'TTM': '0'}	(TTM >= OralAntidiab) => (RD >= Class1) (TTM <= Diet) => (RD <= Class0)

Patient 14

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)

Patient 15

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (TTM >= 2.0) & (EVOL >= 8.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 6.0) & (EVOL <= 2.0) => (RD <= 0.0)
	r = {'TTM': '0'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) & (HbA1c >= 7to8) => (RD >= Class1) (TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (TTM >= 2.0) & (CDKEPI_c <= 29.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 6.0) & (CDKEPI_c >= 51.36) => (RD <= 0.0)
	r = {'TTM': '0'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) & (HbA1c >= 7to8) => (RD >= Class1) (TTM >= Insuline) & (CDKEPI = Low) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0) (HbA1c <= 6to7) & (CDKEPI = Normal) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) & (HbA1c >= 7.0) => (RD >= 1.0) (TTM >= 2.0) & (CDKEPI_c <= 29.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0) (HbA1c <= 6.0) & (CDKEPI_c >= 51.36) => (RD <= 0.0)
	r = {'TTM': '0'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) & (HbA1c >= 7to8) => (RD >= Class1) (TTM >= Insuline) & (EVOL >= 5to10) => (RD >= Class1)

		(TTM <= OralAntidiab) => (RD <= Class0)

patient 16

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (EVOL >= 17.0) & (TTM >= 1.0) => (RD >= 1.0) (TTM <= 0.0) => (RD <= 0.0) (TTM <= 1.0) & (EVOL <= 14.0) => (RD <= 0.0)
	r = {'TTM': '2'} --> {'RD': 1} delta {}	(TTM >= Insuline) => (RD >= Class1) (EVOL >= 15to20) & (TTM >= OralAntidiab) => (RD >= Class1) (TTM <= Diet) => (RD <= Class0) (TTM <= OralAntidiab) & (EVOL <= 10to15) => (RD <= Class0)
max	{"RD": 1}, {"HbA1c": ">7.000000"} {"RD": 0}, {"HbA1c": "<=7.000000", "CDKEPI": "<=78.000000"} {"RD": 1}, {"HbA1c": "<=7.000000", "CDKEPI": ">78.000000", "EVOL": ">11"} {"RD": 0}, {"HbA1c": "<=7.000000", "CDKEPI": "78.0< CDKEPI <=100.0", "EVOL": "<=11"} {"RD": 1}, {"HbA1c": "<=7.000000", "CDKEPI": ">100.0", "EVOL": "<=11"}	(HbA1c >= 8.0) => (RD >= 1.0) (CDKEPI_g >= 100.78) => (RD >= 1.0) (EVOL >= 14.0) & (TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) & (HbA1c <= 7.0) => (RD <= 0.0) (HbA1c <= 7.0) & (EVOL <= 8.0) => (RD <= 0.0)
	r = {'HbA1c': '>7.000000'} --> {'RD': 1} delta {}	(HbA1c >= 7to8) => (RD >= Class1) (TTM >= Insuline) & (EVOL >= 10to15) => (RD >= Class1) (TTM <= OralAntidiab) & (HbA1c <= 6to7) => (RD <= Class0) (HbA1c <= 6to7) & (EVOL <= 5to10) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2", "CDKEPI": ">67.000000"} {"RD": 0}, {"TTM": "2", "CDKEPI": "<=67.000000", "HbA1c": "<=7.000000"}	(EVOL >= 17.0) => (RD >= 1.0) (TTM >= 2.0) & (HbA1c >= 8.0) => (RD >= 1.0) (TTM >= 2.0) & (CDKEPI_g >= 89.0) => (RD >= 1.0) (TTM >= 2.0) & (EVOL >= 14.0) => (RD >= 1.0)

	{"RD": 1}, {"TTM": "2", "CDKEPI": "<=67.000000", "HbA1c": ">7.000000"}	(TTM <= 1.0) & (EVOL <= 14.0) => (RD <= 0.0) (HbA1c <= 7.0) & (EVOL <= 8.0) => (RD <= 0.0)
	r = {'TTM': '2', 'CDKEPI': '>67.000000'} --> {'RD': 1} delta {}	(TTM >= Insuline) => (RD >= Class1) (EVOL >= 15to20) & (TTM >= OralAntidiab) => (RD >= Class1) (TTM <= Diet) => (RD <= Class0) (TTM <= OralAntidiab) & (EVOL <= 10to15) => (RD <= Class0)

patinent 17

	GLORE rules	DRSA rules
min	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
max	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)
mix	{"RD": 0}, {"TTM": "0"} {"RD": 0}, {"TTM": "1"} {"RD": 1}, {"TTM": "2"}	(TTM >= 2.0) => (RD >= 1.0) (TTM <= 1.0) => (RD <= 0.0)
	r = {'TTM': '1'} --> {'RD': 0} delta {'TTM': '2'}	(TTM >= Insuline) => (RD >= Class1) (TTM <= OralAntidiab) => (RD <= Class0)

Patient 18

	GLORE rules	DRSA rules
min	{"RD": 0}, {"EVOL": "<=10"} {"RD": 1}, {"EVOL": ">10"}	EVOL >= 13.0) => (RD >= 1.0) (EVOL <= 10.0) => (RD <= 0.0)
	r = {'EVOL': '<=10'} --> {'RD': 0} delta {'EVOL': '>10.0'}	(EVOL >= 10to15) => (RD >= Class1) (EVOL <= 5to10) => (RD <= Class0)
max	{"RD": 0}, {"EVOL": "<=10"} {"RD": 1}, {"EVOL": ">10"}	(EVOL >= 13.0) => (RD >= 1.0) (EVOL <= 10.0) => (RD <= 0.0)
	r = {'EVOL': '<=10'} --> {'RD': 0} delta {'EVOL': '>10.0'}	(EVOL >= 10to15) => (RD >= Class1) (EVOL <= 5to10) => (RD <= Class0)
mix	{"RD": 0}, {"EVOL": "<=10"} {"RD": 1}, {"EVOL": ">10"}	(EVOL >= 13.0) => (RD >= 1.0) (EVOL <= 10.0) => (RD <= 0.0)
	r = {'EVOL': '<=10'} --> {'RD': 0} delta {'EVOL': '>10.0'}	(EVOL >= 10to15) => (RD >= Class1) (EVOL <= 5to10) => (RD <= Class0)