Machine Learning Model to Classify Patients with Complicated and Uncomplicated Type 2 Diabetes Mellitus in the New Civil Hospital of Guadalajara “Juan I. Menchaca”

Víctor Manuel Medina-Pérez[1], Isaac Zúñiga-Mondragón[1], José Alfonso Cruz-Ramos[2,3], Kevin Javier Arellano-Arteaga[2,4], Iryna Rusanova[5], Gerardo García-Gil[1], and Gabriela del Carmen López-Armas[1](✉)

[1] Centro de Enseñanza Técnica Industrial, Subdirección de Investigación y Extensión, Laboratorio de Biomédica/Mecatrónica, C. Nueva Escocia 1885, Guadalajara, Jal, Mexico
glopez@ceti.mx

[2] Departamento de Clínicas Médicas, Universidad de Guadalajara, C. Sierra Mojada 950, Guadalajara, Jal, Mexico

[3] Instituto Jalisciense de Cancerología, Coronel Calderón 715, Guadalajara, Jal, Mexico

[4] Servicio de Medicina Interna, Nuevo Hospital Civil de Guadalajara “Dr. Juan I. Menchaca” Salvador Quevedo y Zubieta 750, Independencia Oriente, 44340 Guadalajara, Jal, Mexico

[5] Departamento de Bioquímica y Biología Molecular, Facultad de Ciencias, Universidad de Granada, 18019 Granada, Spain

Abstract. Diabetes Mellitus (DM) is a chronic disease worldwide. By 2030 are estimated to be 643 million people with DM, and by 2045 is projected to be 783 million, according to International Diabetes Federation. Machine Learning (ML) can be used as a smart preventive medicine tool for clinical records in hospitals, clinical laboratories, medical personnel, and patients. Implementing ML in current healthcare systems could translate into early diagnosis of DM. This work aimed to implement a classification algorithm for complicated Type 2 DM(T2DM), uncomplicated T2DM, and healthy Mexican participants. For this work, we enrolled 82 subjects from New Hospital Civil Juan I. Menchaca of Guadalajara, divided into 26 complicated T2DM, 26 uncomplicated T2DM, and 30 healthy subjects. ML algorithms used were decision tree (DT), Random Forest, AdaBoost, Bagging Classifier, and Support Vector Machine (SVM). The models use a dataset of 24 different clinical, biological, and molecular variables to discriminate between the 3 groups. The average accuracy was 78% from the C4.5 DT classifier, and we performed an AUC-ROC curve with value = 088. ML models can serve for early diagnosis of T2DM in healthcare systems, implementing this in preventive medicine clinics, developing an APP for smart mobile for personal care, and improving pharmaceutical approaches for treating T2DM.

Keywords: Diabetes Mellitus · Machine Learning · smart preventive medicine · decision tree

J. de J. A. Flores Cuautle et al. (Eds.): CNIB 2023, IFMBE Proceedings 96, pp. 67–83, 2024.
https://doi.org/10.1007/978-3-031-46933-6_8

1 Introduction

1.1 Diabetes Mellitus

Diabetes Mellitus (DM) is an old chronic disease that remains a significant challenge for the public health system. The main feature of diabetes is lifting blood glucose levels over 126mg/dl, caused by defective insulin secretion accompanied by several biological adjustments [1]. The most affected systems are well described and can lead to retinopathy, neuropathy, cardiovascular diseases, and several stages of kidney damage [2]. Until today DM is considered in three categories: type 1 (T1DM), type 2 (T2DM), and gestational diabetes; nevertheless, DM occurs in young adults, in people between the fourth and sixth years of life, and minus degree in the elderly and children; mainly T2DM is often associated with the development of metabolic syndrome, micro and macrovascular complications of DM besides sedentarism and improper eating habits [3]. The external biological factor contributing to this disease is living in a country with low income and middle-income, where 80% of people with T2DM; in this sense, México is considered a middle-income country [4]. This dilemma is increasing without a strategy effective until today. Therefore, exploring T2DM early diagnosis by Machine Learning (ML) tools is nowadays a reality implemented in medicine, primarily due to access to hospital database repositories in developed countries that contain biochemical and anthropometric variables that diagnose DM, such as glycosylated hemoglobin (HbA1c) fasting plasma glucose (FPG), oral glucose tolerance tests (OGTT), random blood glucose levels, Body Mass Index (BMI), waist circumference, insulin among others [5]. The early diagnosis of T2DM makes it easier to control the natural course of the disease and delay micro and macrovascular complications; this way, ML can help healthcare personnel make a preliminary diagnosis of T2DM based on their monthly physical examination data as well as improve their pharmacological management [6, 7].

1.2 Machine Learning in Diabetes Mellitus

Artificial intelligence has emerged as an area of help in medical diagnoses as well as in the most appropriate pharmacological approach for different pathologies; in this case, ML is a technique that has gained confidence in many professional guilds and especially in the medical area and is based on this there are several works with ML and diabetes mellitus this because it is a health emergency. It is necessary to implement more effective technological resources for early diagnosis and extend the micro and macrovascular complications as much as possible or alert people predisposed to developing DM to more significant health care [8]. In this sense, ML tools are affordable to be used even in a mobile application or the creation of a computational aid system (CAS) in more vulnerable populations. Today, in Berlin, exists a CAS since 1974, which is powered with data from 55,000 patients with T2DM and gives them prevalence, incidence, pharmacological treatments, and duration of DM; this is achieved by recording blood and urinary glucose for metabolic criteria [9].

Algorithms classifiers like multilayer perceptron (MLP) and long short-term memory (LSTM) for DM were used by Butt et al. [10], obtained an accuracy of 86.08% and 87.26%, respectively, using PIMA Indian Diabetes database consisting in 8 variables, and 1was considered like target variable, is important to mention that they only compared the state-of-the-art with previously work reported in the same database. On the other hand, Phongying et al. [11] worked on the dataset from the Department of medical services in Bangkok from 2019-2021 with 20,227 samples with 10 attributes related to developing DM; they applied ML tools like DT, random forests, SVM, and K-nearest neighbors and obtained after tuning the hyperparameters, and the interaction terms that the random forest tree had the best performance classification, with 97.5% accuracy, 97.4% precision, 96.6% recall, and a 97% F1-score.Also, Habibollah et al. [12] worked with DT and random forest tools, and they explored the (MASHAD) Study program from Iran; they obtained 9258 records and 18 attributes, considering 17 like predictors and 1 like target; they only identified T2DM or no event of T2DM. In conclusion, they observed the best results with the random forest model and reported 71.1% accuracy, 71.3% sensitivity, and 69.9% specificity, and the AUC of the ROC curve measured 77.3% correspondingly. It should be noted that they used several attributes biochemical and anthropometric, like our work, but not cellular or molecular attributes. In this sense, there is no report of ML with this kind of attributes for diabetes mellitus disease; mainly, the rest of the reported manuscript focus on only classier o predict DM. Recently, Schallmoser et al. [13] worked with ML models, logistic regression (LR), and gradient-boosted decision trees (GBDTs) for the prediction of the risk of developing micro or macrovascular complications in 13,904 participants with 105 predictors in prediabetes or DM participants from dataset EHRs (Israeli health provider); they included 3 microvascular complications (retinopathy, nephropathy, and neuropathy), and 3 macrovascular complications: peripheral vascular, cerebrovascular and cardiovascular diseases. Their results for prediabetic individuals were plotted in a AUC-ROC of the LR, and GBDTs correspondingly, 0.657 and 0.681 for retinopathy, 0.807 and 0.815 for nephropathy, 0.727 and 0.706 for neuropathy, 0.730 and 0.727 for peripherical vascular disease, 0.687 and 0.693 for cerebral vascular disease, 0.707 and 0.705 for the cardiovascular condition; on the other hand DM participants obtained AUC-ROC in the same ML models LR and GBDTs, respectively, 0.673 and 0.726 for retinopathy, 0.763 and 0.775 for nephropathy, 0.745 and 0.771 for neuropathy, 0.698 and 0.715 for peripherical vascular disease, 0.651 and 0.646 for cerebral vascular disease, and 0.686 and 0.680 for the cardiovascular condition. They conclude that ML models can predict prediabetes and DM patients to develop micro and macrovascular complications in this population. Hence, ML methods are widely used in diabetes prediction and obtain favorable results; in this work, a DT model was implemented to classify T2DM patients into uncomplicated, complicated patients and healthy participants.

2 Methods

2.1 Database

For this study, we enrolled 82 Mexican participants from New Hospital Civil of Guadalajara obtained through 2020–2021 previously reported [14]. The study was conducted following the Helsinki Declaration and the General Health Law on research; the hospital's Ethical Committee accepted this study with the number approbation 17 CI 14 039

116 COFEPRIS, and by the no: 940/CEIH/2019 of the Ethics Committee of the University of Granada, Spain. Before biological samples were collected, all participants were informed about the study and signed informed consent forms. All participants were adults from 40–65 years of age and divided into three groups of study: 1. Healthy individuals without a family history of first-degree diabetes, besides non-disease of kind inflammatory, endocrine, vascular, or autoimmune, were not taking any pharmaceutical treatment. For patients with T2DM who had at least 5 years with a diagnosis of DM according to ADA criterion [5] (fasting plasma glucose $\geq$ 126 mg/dL, oral glucose tolerance test $\geq$ 200 mg/dL, or glycosylated hemoglobin (HbA1c) $\geq$ 6.5%/) and were divided as T2DM uncomplicated and T2DM complicated. The uncomplicated patients were defined with reasonable glucose control and free of micro (diabetic neuropathy, diabetic nephropathy, and diabetic retinopathy) or macrovascular complications of DM (atherosclerosis, myocardial infarction, stroke, and cerebrovascular disease) [15]. Patients with DM complications were those with any micro or macro complications of DM before mentioned. We obtained a complete clinical history of each patient and measured anthropometric parameters, Body Mass Index (BMI) and waist circumference, and biochemical and molecular variables.

2.2 Data Preprocessing

The initial dataset comprised 56 clinical, biological, and molecular variables from 82 subjects. Aiming to create a more concise dataset, variables with greater significance were selected to be kept in the final dataset. Also, variables representing 2TDM complications in patients were removed (retinopathy, nephropathy, neuropathy, stroke, pharmacological treatment). Variables with a small percentage of missing data were imputed with the mean value of the corresponding group [16]. For variables such as sex, a value of 1 for females and 2 for males was given, as well as for the variables of alcohol consumption, smoking, and physical activity, 1 represents an affirmation and 0 a negation. The final dataset comprises the following 24 variables (see Table 1).

The 24 variables listed in the table above correspond to the more significant attributes that helped the DT to discriminate between healthy subjects, patients with uncomplicated T2DM, and patients with complicated T2DM. The primary reason for analyzing the groups in a split manner corresponds precisely to differentiating clinical profiles; dividing patients into groups according to the presence or absence of complications helps to understand the unique characteristics of each group better and to identify specific risk factors associated with complications; similarly by analyzing and comparing these three groups, unique patterns, biomarkers, and associations can be discovered that might not be evident if all diabetes patients were considered as a homogeneous group. This may lead to new insights into the underlying mechanisms of complications and diabetes in general.

2.3 Implementation and Training of Classifier Models

Different classifier models were trained and validated with the corresponding datasets, and all classifier models were implemented using the scikit-learn (version 1.0.2) and Python library existing models (version 3.9.13). The dataset was divided into train and validation subsets using the scikit-learn train/test splitter; 70% (*n* = *57*) of the records were used for the model training, while the remaining 30% (*n* = *25*) were used for the model validation and we performance a five-fold cross-validation procedure. The split was performed using stratification; train and validation subsets were tested and had a significant representation of the 3 groups; the training subset was composed of 21 healthy subjects, 18 uncomplicated, and 18 complicated T2DM patients, and the validation subset with 9 healthy subjects, 8 uncomplicated and 8 complicated T2DM patients, are shown in Fig. 1.

The dataset split train-validation process, training, and model validation were executed 200 times for the classifier models. The optimization of hyperparameters was carried out exhaustively through the modification (tuning) in its maximum depth, the proportion or number of cases of the training and test set, the minimum number of subjects per sheet, and all pertinent biological markers were selected according to the physiology of Type 2 Diabetes mellitus, that is, the characteristics were designated in such a way that those variables that were not in the definition of complicated diabetes were included. The number of iterations for optimizing and adjusting the model in a total of 200 executions was also modified and adjusted. For this work, it was the performance of those classifier models.

2.3.1 Support Vector Machine

This algorithm was built with the default parameters provided by the Scikit library), is a supervised learning algorithm used for classification and regression. It searches for a hyperplane that best separates data classes in a multidimensional space. SVM maximizes the margin between classes, resulting in good generalization to new data. It can handle both linearly separable and non-linearly separable data using kernel functions.

2.3.2 Random Forest

This algorithm was built using 350 estimators, while the remaining parameters were set to the default option; this ML tool creates multiple decision trees and combines their predictions to obtain a result. Each tree is trained on a random subsample of the data and uses bootstrapping to build different data sets. The predictions of the individual trees are then averaged or voted to make a final decision.

2.3.3 ADAboost

This classifier was built with 100 estimators and a 0.01 learning rate; all other parameters were set as the default option; this ensemble algorithm combines several weak learning models (e.g., weak decision trees) to form a stronger model. Adaboost gives more weight to instances misclassified in the previous iteration at each iteration, allowing the model to focus on hard-to-classify cases. The weak models are then weighted according to accuracy and combined to make joint decisions.

Table 1. Summary of clinical, biochemical, and molecular characteristics of T2DM subjects. ‡Relative expression was calculated using the $2 - \Delta\Delta CT$ method (ΔCt target miR-ΔCt control gene)

N°	Variable	Type	Value
0	Age (Years)	Numeric	[1–100]
1	Sex	Nominal	[1–2]
2	Waist circumference (cm)	Numeric	[70–150]
3	BMI index (kg/mt^2)	Numeric	[17–47]
4	Systolic Blood pressure (mm/Hg)	Numeric	[80–205]
5	Diastolic Blood pressure mm/Hg)	Numeric	[60–115]
6	HOMA IR (fasting glucosa mg/dL * fasting insuline IU/L/405)	Numeric	[0.25–20]
7	HbA1c (%)	Numeric	[4–13]
8	TG (mg/dL)	Numeric	[40–360]
9	Cholesterol(mg/dL)	Numeric	[100–320]
10	Urea (mg/dL)	Numeric	[13–135]
11	Creatinine (mg/dL)	Numeric	[0.1–2.6]
12	miR_21 (relative expression)‡	Numeric	[30–350]
13	miR_126 (relative expression) ‡	Numeric	[10–195]
14	IL_6 (pg/mL)	Numeric	[0.01–5.50]
15	IL_10 (pg/mL)	Numeric	[0.20–13.5]
16	IL_18 (pg/mL)	Numeric	[10–426]
17	TNF_alfa (pg/mL)	Numeric	[6–127]
18	Diabetic Family	Nominal	[0–6]
19	Alcohol intake	Nominal	[0–1]
20	Smoke	Nominal	[0–1]
21	Physical activity	Nominal	[0–1]
22	Glucose (mg/dL)	Numeric	[70–380]
23	Years of diabetic (years)	Numeric	[0–35]

2.3.4 Bagging Classifier

This model was built using 200 estimators, with the remaining parameters set to the default value; this algorithm is an ensemble technique used to improve the accuracy and stability of ML models. Using bootstrapping, the Bagging Classifier builds multiple base models (e.g., decision trees) on random subsets of the training data. Then, the predictions of the base models are combined by voting to determine the final class of an instance, and finally, we implemented the C4.5 DT (the class to measure the quality of a split was set to "gini," splitter to "best," no max depth, and the rest of the parameters as the default option). Since this last ML model was the one that gave us the best overall performance, we will focus on describing it in detail. Finally, the precision, recall, F1-score, and accuracy results of every execution were saved and averaged to review the results further.

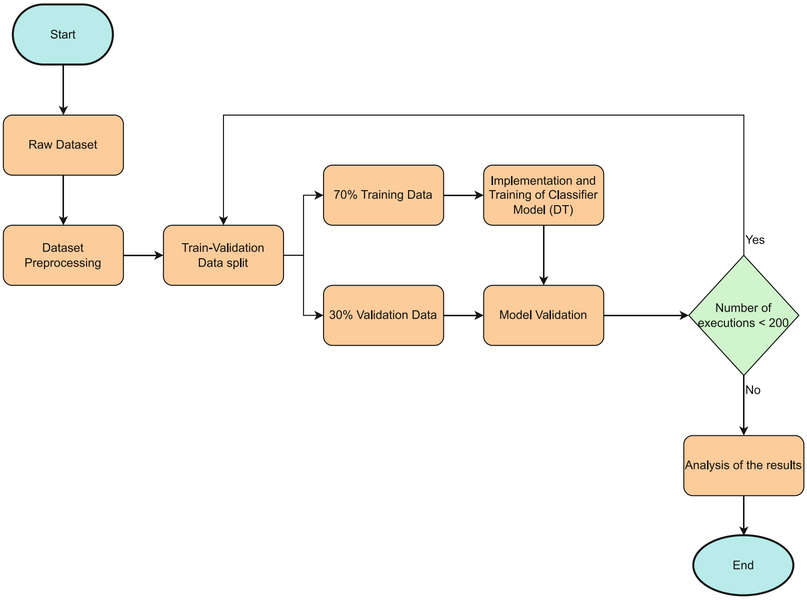

Fig. 1. Flow diagram for data processing and functioning sequences of the proposed diabetes classification model.

2.4 Model Decision Tree

The DT methodology was used, C4.5, as a classifier. It was implemented in Python, a high-level programming language whose function emphasizes the readability of its code [17]. Due to its free software characteristics, it is used to develop academic, scientific, and technological applications. It is a programming language that allows the coexistence of different paradigms, such as oriented object programming, structured programming, imperative programming, reactive programming, and functional programming searching to improve the production of the development of the projects. The DT model is a primary and regression method with a tree structure describing classifying instances based on features [18]. It is considered a set of "if-then" rules, which also function as conditional probability distributions defined in feature and class space.

The DT utilizes a tree structure, starting with a single node representing the training samples [19]. If the samples are located inside the same class, the node is transformed into a leaf and labeled with the same class. Otherwise, the algorithm chooses the discriminative attributes as the actual node of the DT. According to the value of the current attribute of the decision node, the training samples are divided into different subsets, each in different forms, values, and branches. The anterior steps are repeated for each subset or for each branch obtained, recursively forming a DT on each partitioned sample [20]; see the pseudocode in Table 2. Typical DT algorithms are ID3, C4.5, and CART, among others.

This study used the DT C4.5 and some parameters like entropy and GINI index that work in DT with the information that was input from the dataset used to classify each subset test. The dataset generated has nominal attributes for the classification tasks with non-missing values. In general, if a probability distribution *P = (p1, p2, p3, …, pn)* and an example S is given, then the information carried by this distribution is known as the entropy P, and is calculated by: Eq. (1):

$$\text{Entropy} \quad \text{P} = -\sum_{i=1}^{n} p_i \cdot \log(p_i) \tag{1}$$

In this work, when the entropy level is 0, it will be the level of order, 1. On the other hand, the information gain allows us to measure the degree of impurity of the classes for all the examples and, therefore, any position of the tree under construction. It must have a new function that allows one to select the attribute that should label the current node. This defines the gain Eq. (2) for a test T and a position p:

$$\text{Gain(p, T)} = \text{Entropy P} - \sum_{j=1}^{n} (\text{pj} \cdot \text{Entropy})(\text{pj})) \tag{2}$$

Where the values (p_j) is the set of all possible values for attribute T. This measure can then be used to determine which attribute is best and build the DT where at each node is the attribute with the highest information gain of all the attributes not yet considered in the path from the root node [21].

Information gain of attribute A Eq. (3):

$$\text{Gain(A)} = \text{Info(D)} - \text{InfoA(D)} \tag{3}$$

Pre-segmentation of information entropy Eq. (4):

$$\text{Info(D)} = \text{Entropy(D)} = -\Sigma \text{p(j|D)} \log_2 \text{p(j|D)} \tag{4}$$

Entropy of distributed information Eq. (5):

$$\text{InfoA(D)} = \Sigma(\text{ni/n})\text{Info(Di)} \tag{5}$$

The impurity GINI is one of the possible measures for generating the DT. It provides more information about the data distribution per node than the classification used in the accuracy reporting. The impurity of the model nodes is calculated using each count of each objective category divided by every record obtained by one node. The total GINI impurity is calculated as a sum of squares of the count of proportion divided by all the objective categories per node from which one is subtracted. The result is multiplied by the quantity of records Eq. (6). For example, when splitting a tree node, the algorithm searches for a field with the highest improvement in total impurity, calculated as the total impurity among all potential child nodes subtracted from the total impurity of the parent node. The goal is for the gini index to be as small as possible.

$$\text{Gini}(\text{Xq}) = 1 - \sum_{k=1}^{k} \left(p_{k,q}\right)^2 \tag{6}$$

2.5 Algorithm C4.5

C4.5 is an algorithm used to generate a DT developed by Ross Quinlan [18], and this is an extension of the ID3 algorithm. DT generated by C4.5 can be used for classification; therefore, C4.5 is almost always referred to as a statistical classifier. C4.5 builds DT from a training data set as ID3 does, using the concept of information entropy. In a DT, the training data refers to a collection of examples used to train the tree. Each example is denoted as Si and represents a data point with associated information. This collection of examples forms the dataset the DT algorithm uses to learn and build the tree structure. Each example Si is represented as a vector of attributes or features, such as X1, X2, etc. These features describe the characteristics or properties of the data point Si.

The training data are augmented with a vector $C = C_1$, C_2, where C_1, C_2, represent the class to which each sample belongs. Each example Si is associated with a specific class or category, and a class label represents this class. The vector C contains these class labels for each example Si. These labels indicate the predefined categories that each data point belongs to.

In every tree node, the DT C4.5 selects the variable that better divides the sample into different data subsets with a predominant class; the criteria are normalized for information gain resulting in the selection of the variable that better divides the data. The decision parameter will be the variable that achieves the best-normalized information gain [17]. See pseudocode Table 2.

This algorithm has 3 main features; firstly, the DT C4.5 creates a leaf node when all samples are classified in the same class. In the second instance, when none of the variables show a significative information gain, the DT C 4.5 creates a decision node to continue with the tree; finally, when an unseen class is found, the DT also creates a decision node to continue with the tree C4.5.

2.6 Metrics to Evaluate the Performance of Classifiers on Unbalanced Datasets

It is essential to mention that the performance of a classification method on data sets with two classes is to use a confusion matrix. This contains information about the predictions made by the classifier using the number of true positives, true negatives, false positives, and false negatives. Table 3 shows an example of the confusion matrix.

The confusion matrix should only be used for classification accuracy to determine the performance of a classifier; it is not suitable when dealing with unbalanced data sets. This is because high classification accuracy values can be obtained in these types of applications by simply ignoring the minority class; however, the latter is essential in cases such as web mining detection, direct marketing, and medical diagnostics [22].

Table 2. Algorithm C4.5 for the construction of DT

C4.5 Gnal Pseudocode

```
FUNCTION CreateDecisionTree(data, attributes)
    IF all data belongs to a single class THEN
        RETURN a leaf node with that class
    IF no attributes left THEN
        RETURN a leaf node with the majority class in the data
    ELSE
        chosenAttribute = SelectWinningAttribute(data, attributes)
        NEW NODE decisionNode WITH chosenAttribute

        FOR each value in chosenAttribute.VALUES DO
            subsetData = FilterData(data, chosenAttribute, value)
            IF subsetData is empty THEN
                RETURN a leaf node with the majority class in the data
            ELSE
                NEW NODE child WITH CreateDecisionTree(subsetData, attributes - {chosenAttribute})
                ADD child TO decisionNode WITH label value
            END IF
        END FOR

        RETURN decisionNode
    END IF
END FUNCTION

FUNCTION SelectWinningAttribute(data, attributes)
    bestAttribute = NULL
    bestGain = 0

    FOR each attribute IN attributes DO
        gain = CalculateGain(data, attribute)
        IF gain > bestGain THEN
            bestGain = gain
            bestAttribute = attribute
        END IF
    END FOR

    RETURN bestAttribute
END FUNCTION
```

(*continued*)

Table 2. (*continued*)

```
FUNCTION CalculateGain(data, attribute)
    dataEntropy = CalculateEntropy(data)
    conditionalEntropy = 0

    FOR each value IN attribute.VALUES DO
        subsetData = FilterData(data, attribute, value)
        weight = size of subsetData / size of data
        conditionalEntropy = conditionalEntropy + (weight * CalculateEntropy(subsetData))
    END FOR

    gain = dataEntropy - conditionalEntropy
    RETURN gain
END FUNCTION

FUNCTION CalculateEntropy(data)
    countClass(class) counts the number of examples in data with the given class label

    entropy = 0
    FOR each class IN classes DO
        probability = countClass(class) / size of data
        IF probability > 0 THEN
            entropy = entropy - (probability * log2(probability))
        END IF
    END FOR

    RETURN entropy
END FUNCTION

FUNCTION FilterData(data, attribute, value)
    NEW SET subset
    FOR each example IN data DO
        IF example.attribute = value THEN
            ADD example TO subset
        END IF
    END FOR
    RETURN subset
END FUNCTION
```

According to the Confusion Matrix, True Positive (TP) is a Positive Value that is Correctly Classified as Positive, then False Positive (FP) is a Negative Value that is Incorrectly Classified as Positive; also, We Have False Negative (FN) that is a Positive Value Classified as Negative and Finally We Have True Negative (TN) Which Indicates a

Table 3. Confusion matrix

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

Negative Value Predicted Classified Negative (See Table 3). in This Work, the Confusion Matrix Was Calculated in a Python Program and Obtained the Subsequent Metrics: $Accuracy = [(TP + TN) / (TP + TN + FP + FN)] \times 100$, Sensitivity or $Recall = [(TP/(TP + FN)]$, $Specificity = [TN/(TN + FP)]$, $Precision = [TP/(TP + FP)]$, and $F1 - score = (2 \times Precision \times Recall)/(Precision + Recall)$ See Fig. 3. Besides, We Performed in the Phyton Program an Area Under the Curve (AUC) with a Cutoff of 0.5, and Values Were Plotted in the AUC- Receiver Operation Curve (ROC) of DT's Best and Worst Performance. Fig. 4.

3 Results and Discussion

The design objective of this DT C4.5 was programmed to identify 3 groups of patients belonging to the following categories: 1. Healthy subjects, 2. Patients with uncomplicated T2DM, and 3. Patients with T2DM with complications; the results are shown in Fig. 2.

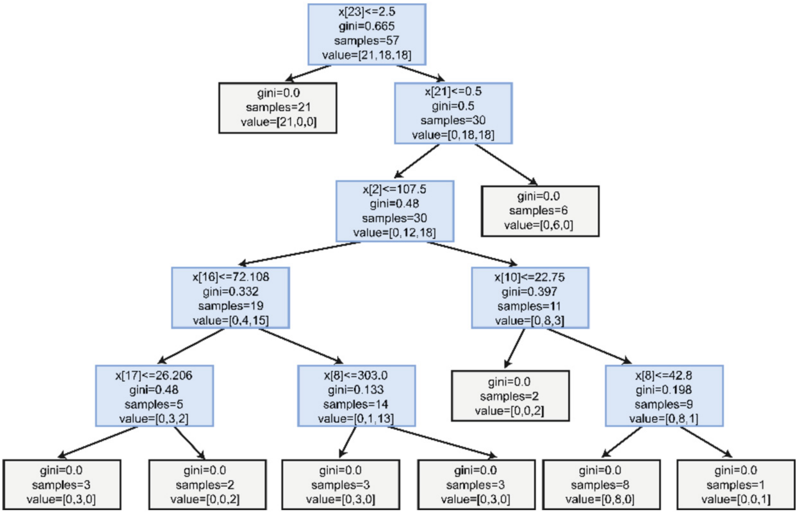

Fig. 2. DT C4.5: this tree shows in the leaves (white boxes) where "x" represents the variable taken into consideration to make a classification; "gini" expresses the degree of impurity; the "samples" are subjects. Finally, the "value" represents the vector where the first index belongs to healthy, the second is T2DM uncomplicated, and the last is T2DM complicated subjects.

The results of the ML classifiers are shown in Table 4, precision, recall, F1 score, and classification accuracy corresponding to the mean ± SD of the dataset obtained from the New Hospital Civil Juan I. Menchaca, considering 24 attributes of 82 patients (Table 1), subdivided in healthy, uncomplicated T2DM and complicated T2DM patients. We obtain from DT C 4.5 the best results of all classifiers, showing for precision of 0.77 ± 0.04, recall of 0.76 ± 0.03, F1 score of 0.76 ± 0.02 and accuracy of 78% ± 0.02. The rest of the classifier are described in the Table 4.

Table 4. Performance metrics of ML model

Classifier	Precision	Recall F1 Score	Accuracy
C4.5 (DT)	0.77 ± 0.04	0.76 ± 0.03 0.76 ± 0.02	78% ± 0.02
Adaboost	0.75 ± 0.0	0.75 ± 0.0 0.76 ± 0.0	76% ± 0.0
Bagging	0.73 ± 0.04	0.72 ± 0.03 0.70 ± 0.04	73% ± 0.03
RF	0.74 ± 0.04	0.72 ± 0.03 0.69 ± 0.04	73% ± 0.03
SVM	0.66 ± 0.0	0.66 ± 0.0 0.64 ± 0.0	68% ± 0.0

We also performed a confusion matrix multiclass to determine the classification problem of 3 groups: healthy subjects, 2TDM uncomplicated, and 2TDM complicated the results are the average of 200 executions of the DT C4.5.

Finally, we plotted the results of DT C4.5 in T2DM complicated and was analyzed by ROC curve showing accuracy in the best performance with an area under the curve *(AUC)* = *0.88* and the worst performance with *AUC* = *0.76*. Fig. 4.

Several ML works have been developed to manage databases of patients with diabetes mellitus to establish a prediction in the course of the disease; some focus on complicated T2DM, such as the work of Ljubic et al. [19], who worked with a dataset of more than 1 million people diagnosed with DM over 9 years, they focused on predicting micro and macrovascular complications of DM. They considered 10 of these and the number of hospitalizations on average 1 to 4 and employing the Recurrent neural network (RNN) with a gated recurrent unit (GRU), they obtained an accuracy of 73% (myocardial infarction) and 83% (chronic ischemic heart disease). On the other hand, Agliata et al. [23] worked with 3 different datasets to diagnose DM2 and included characteristics such as insulin sensitivity, age, sex, BMI, and glycosylated hemoglobin, among other parameters; to train their data; they chose a binary classifier trained in scratch which gave them an area under the receiver operating characteristic curve (AUROC) of 0. 934; however, these authors mention the importance of the characteristics to be used in a database; since there are too many variables that may not be considered, they decided to work with the simplest to obtain from almost any medical history. Pan et al. [24].

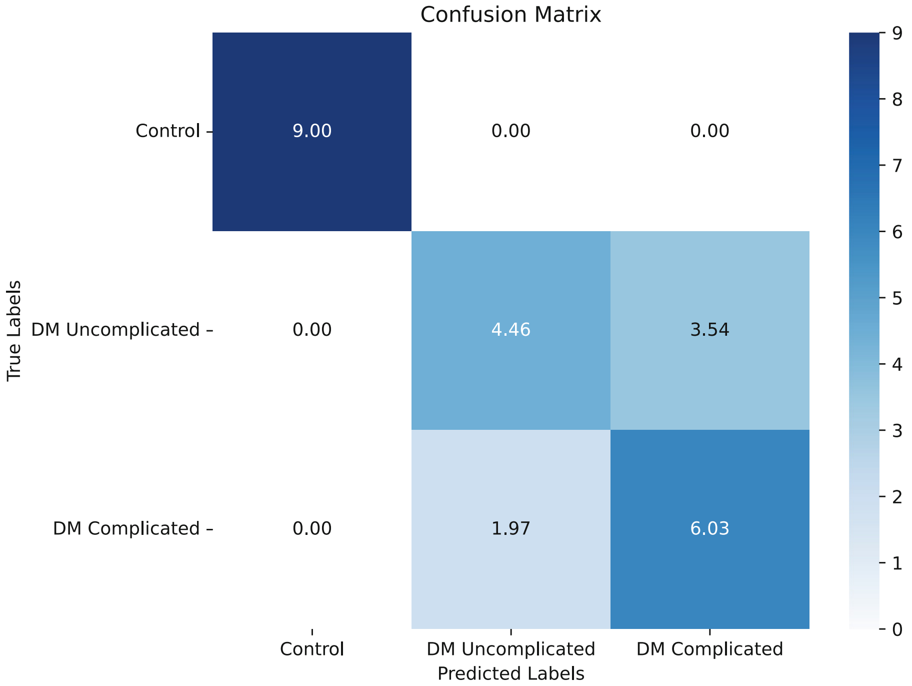

Fig. 3. Confusion matrix for multiclass DT classifier.

They focused on building a multivariate logistic regression model to predict the risk of diabetic retinopathy in a Chinese population of 2385 patients already diagnosed with DM, where they built model I with classical predictors such as glycosylated hemoglobin, systolic blood pressure, ongoing disease, postprandial blood glucose, and the urinary albumin/creatinine ratio, obtaining an AUROC of 0.703 with an accuracy of 0.79. In this sense, our database is of Mexican patients, and with variables beyond only clinical, anthropometric, and biochemical data, but also included cellular (interleukins) and molecular (microRNAs) variables that, to date, there are no similar reports in Mexican patients where DM2 is a severe health problem, our ML model consisted of a DT C4. 5 which gave us an accuracy of 78%, which for the number of patients in a small sample shows that it is very efficient at the time of classifying healthy patients with uncomplicated T2DM and with T2DM with complications.

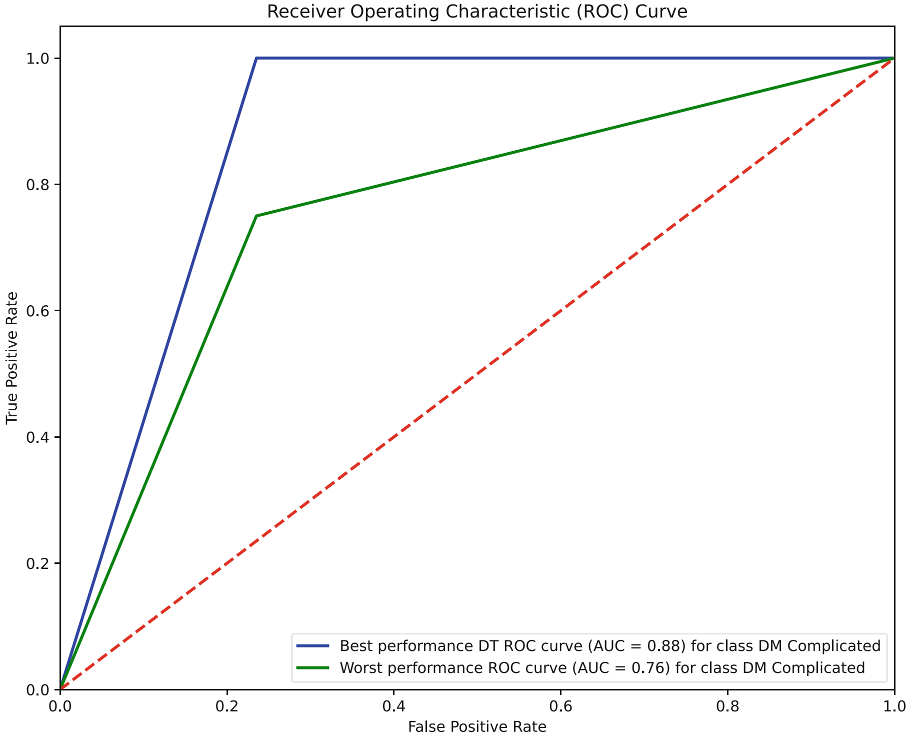

Fig. 4. ROC curve analysis for DT C4.5 classifier during the use of the best performance with a value of $AUC = 0.88$ and worst performance with a value of $AUC = 0.76$ for T2DM complicated class.

4 Conclusion

This work was carried out in Mexican patients with T2DM and is the first to our knowledge in the classification by ML of the complications of the disease itself, as well as the inclusion of novel cellular and molecular parameters already studied T2DM, such as proinflammatory and anti-inflammatory interleukins (Il-6, TNF-α, IL-10, and IL-18) and microRNAs (Mirs-21 and 126) that can regulate gene expression and thus, may become a therapeutic target in the future; our primary intention is to encourage the existence of more robust databases with markers that can contribute to a better early diagnosis in order to avoid the disabling complications of DM2 since it is a worldwide public health problem. Our future work will be focus on making this database bigger, since this was only the implementation of the machine learning algorithms as an exploratory classification method as well as developing deep learning to discover possible best results with a larger database.

References

1. Ludwig, D.S.: The glycemic index: Physiological mechanisms relating to obesity, diabetes, and cardiovascular disease. JAMA **287**(18), 2414 (2002). https://doi.org/10.1001/jama.287.18.2414
2. Chawla, A., Chawla, R., Jaggi, S.: Microvasular and macrovascular complications in diabetes mellitus: Distinct or continuum? Indian J. Endocr. Metab. **20**(4), 546 (2016). https://doi.org/10.4103/2230-8210.183480
3. Ceriello, A., Ihnat, M.A., Thorpe, J.E.: The "metabolic memory": Is more than just tight glucose control necessary to prevent diabetic complications? J. Clin. Endocrinol. Metab. **94**(2), 410–415 (2009). https://doi.org/10.1210/jc.2008-1824
4. Chan, J.C.N., et al.: The Lancet commission on diabetes: Using data to transform diabetes care and patient lives. *Lancet* **396**(10267), 2019–2082 (2020). https://doi.org/10.1016/S0140-6736(20)32374-6
5. ElSayed, N.A., et al.: 10. Cardiovascular disease and risk management: Standards of care in Diabetes—2023. Diabetes Care **46**(Supplement_1), S158–S190 (2023). https://doi.org/10.2337/dc23-S010
6. Fujihara, K., Sone, H.: Machine learning approach to drug treatment strategy for diabetes care. Diabetes Metab. J. **47**(3), 325–332 (2023). https://doi.org/10.4093/dmj.2022.0349
7. Lee, B.J., Kim, J.Y.: Identification of type 2 diabetes risk factors using phenotypes consisting of anthropometry and triglycerides based on machine learning. IEEE J. Biomed. Health Inform. **20**(1), 39–46 (2016). https://doi.org/10.1109/JBHI.2015.2396520
8. Chaki, J., Thillai Ganesh, S., Cidham, S.K., Ananda Theertan, S.: Machine learning and artificial intelligence based Diabetes Mellitus detection and self-management: A systematic review. J. King Saud Univ. - Comput. Inform. Sci. **34**(6), 3204–3225 (2022). https://doi.org/10.1016/j.jksuci.2020.06.013
9. Thoelke, H., Meusel, K., Ratzmann, K.-P.: Computer-aided system for diabetes care in Berlin, G.D.R. Comput. Methods Programs Biomed. **32**(3–4), 339–343 (1990). https://doi.org/10.1016/0169-2607(90)90118-S
10. Butt, U.M., Letchmunan, S., Ali, M., Hassan, F.H., Baqir, A., Sherazi, H.H.R.: Machine learning based diabetes classification and prediction for healthcare applications. J. Healthcare Eng. **2021**, 1–17 (2021). https://doi.org/10.1155/2021/9930985
11. Phongying, M., Hiriote, S.: Diabetes classification using machine learning techniques. Computation **11**(5), 96 (2023). https://doi.org/10.3390/computation11050096
12. Esmaily, H., Tayefi, M., Doosti, H., Ghayour-Mobarhan, M., Nezami, H., Amirabadizadeh, A.: A comparison between decision tree and random forest in determining the risk factors associated with type 2 diabetes. J. Res. Health Sci. **18**(2), 1–7 (2018)
13. Schallmoser, S., Zueger, T., Kraus, M., Saar-Tsechansky, M., Stettler, C., Feuerriegel, S.: Machine learning for predicting micro- and macrovascular complications in individuals with prediabetes or diabetes: Retrospective cohort study. J. Med. Int. Res. **25**, e42181 (2023). https://doi.org/10.2196/42181
14. López-Armas, G.C., et al.: Role of c-miR-21, c-miR-126, redox status, and inflammatory conditions as potential predictors of vascular damage in T2DM patients. Antioxidants **11**(9), 1675 (2022). https://doi.org/10.3390/antiox11091675
15. Bin Rakhis, S.A., AlDuwayhis, N.M., Aleid, N., AlBarrak, A.N., Aloraini, A.A.: Glycemic control for type 2 diabetes mellitus patients: A systematic review. Cureus **14**, e26180 (2022). https://doi.org/10.7759/cureus.26180
16. Zhang, Z.: Missing data imputation: Focusing on single imputation. Ann. Transl. Med. **4**(1), 9 (2016). https://doi.org/10.3978/j.issn.2305-5839.2015.12.38

17. Quinlan, J.R.: Improved use of continuous attributes in C4.5. arXiv, 29 de febrero de (1996). Accedido: 30 de junio de 2023. [En línea]. Disponible en: http://arxiv.org/abs/cs/9603103
18. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986). https://doi.org/10.1007/BF00116251
19. Friedl, M.A., Brodley, C.E.: Decision tree classification of land cover from remotely sensed data. Remote Sens. Environ. **61**(3), 399–409 (1997). https://doi.org/10.1016/S0034-4257(97)00049-7
20. Habibi, S., Ahmadi, M., Alizadeh, S.: Type 2 diabetes mellitus screening and risk factors using decision tree: Results of data mining. Glob. J. Health Sci. **7**(5), p304 (2015). https://doi.org/10.5539/gjhs.v7n5p304
21. Hssina, B., Merbouha, A., Ezzikouri, H., Erritali, M.: A comparative study of decision tree ID3 and C4.5. Int. J. Adv. Comput. Sci. Appl. **4**(2), 13–19 (2014). https://doi.org/10.14569/SpecialIssue.2014.040203
22. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic minority over-sampling technique. Jair **16**, 321–357 (2002). https://doi.org/10.1613/jair.953
23. Agliata, A., Giordano, D., Bardozzo, F., Bottiglieri, S., Facchiano, A., Tagliaferri, R.: Machine learning as a support for the diagnosis of type 2 diabetes. IJMS **24**(7), 6775 (2023). https://doi.org/10.3390/ijms24076775
24. Pan, H., et al.: A risk prediction model for type 2 diabetes mellitus complicated with retinopathy based on machine learning and its application in health management. Front. Med. **10**, 1136653 (2023). https://doi.org/10.3389/fmed.2023.1136653