

## Stacking ensemble approach to diagnosing the disease of diabetes

Alfredo Daza <sup>a,\*</sup>, Carlos Fidel Ponce Sánchez <sup>b</sup>, Gonzalo Apaza-Perez <sup>c</sup>, Juan Pinto <sup>d</sup>, Karoline Zavaleta Ramos <sup>e</sup>

<sup>a</sup> Faculty of Engineering and Architecture, School of Systems Engineering, Universidad César Vallejo, Lima, Peru

<sup>b</sup> Faculty of Industrial and Systems Engineering, School of Industrial Engineering, Universidad Nacional de Ingeniería, Lima, Peru

<sup>c</sup> Graduate School, Professional School of Systems Engineering, Universidad Nacional del Altiplano, UNAP, Puno, Peru

<sup>d</sup> Faculty of Systems Engineering, Professional School of System Engineering, Universidad Andina Néstor Cáceres Velasquez, Puno, Peru

<sup>e</sup> Faculty of Business Sciences, School of Management, Universidad César Vallejo, Trujillo, Peru



### ARTICLE INFO

#### Keywords:

Machine learning  
Prediction  
Diabetes  
Oversampling  
Hyperparameters  
Stacking

### ABSTRACT

**Background:** Diabetes is a very common disease today and has acquired a worrying focus in the field of public health globally, in fact, it is estimated that the number of people with diabetes worldwide has reached 415 million.

**Objective:** Propose a method and 4 combined models based on Stacking ensemble to diagnose Diabetes. In addition, a web interface was developed with the best model proposed in this study.

**Methods:** The dataset collected from the Diabetes Dataset composed of 768 patient records was used. The data was then pre-processed using the Python programming language. To balance the data, it was divided into 4 values and an oversampling method was applied to distribute the data proportionally. Then, divisions were made on the balanced data using the cross-validation method for data training, and the models were calibrated. Regarding the development of base algorithms, 7 independent algorithms were used, and 4 combined algorithms based on Stacking were proposed, and finally obtain the evaluation of the model with their respective metrics.

**Results:** Stacking 1A (Logistic regression) with Oversampling reached the best value of Accuracy = 91.5 %, Sensitivity = 91.6 %, F1-Score = 91.49 % and Precision = 91.5 %, while with respect to the metric ROC Curve, Stacking 1A (Logistic regression) with Oversampling, Stacking 2A (Random Forest) with oversampling, and Random Forest (Independent) reached the best percentage, this being 97 %.

**Conclusions:** Implementing 4 stacking models using the oversampling method, helps to make an adequate diagnosis of diabetes. Therefore, by using the combined method, an improvement in diabetes prediction was observed, surpassing the performance of the independent algorithms used.

### 1. Introduction

Diabetes is one of the most common pathologies today and has become a constant problem in public health worldwide [1]. In fact, it has been estimated that diabetics have risen to 415 million worldwide and, in 2040, this percentage would rise to 642 million [2,3]; According to a report by the International Diabetes Federation (IDF), it points out that this disease is prone to middle-aged people, that is, between 40 and 59 years old, which has serious economic and social consequences, however particularly cases have also been seen in people from 55 to 59 years old, being respectively 5 % and 20 % [4]. Thus, according to statistics in Iran, it is observed that 8.7 % of Iranians aged 15 to 64 have diabetes,

and of this 4.1 % are patients who were recently diagnosed [5] as well as in India, whose inhabitants suffering from this disease has increased to 72.9 million in 2017 [6]; There are several causes that lead to diabetes (Type I, Type II and Gestational), among which are problems: Genetic, environmental, metabolic, overweight, family history, previous pregnancy, advanced age, etc., as well as physical inactivity and smoking [7, 8].

Most diabetes is diagnosed in two ways: Traditional (manually performed by health personnel) or by technological means, each having advantages and disadvantages [9]. Therefore, it is essential to detect diabetes in the early stages, this with the aim of providing more effective treatments, which help identify different stressors and responsibilities

**Abbreviations:** SVM, Support Vector Machine; KNN, K-Nearest Neighbors; ROC, Curve Receiver operating characteristic curve.

\* Corresponding author.

E-mail address: [\(A. Daza\).](mailto:adaza@ucv.edu.pe)

<https://doi.org/10.1016/j imu.2023.101427>

Received 5 September 2023; Received in revised form 5 December 2023; Accepted 6 December 2023

Available online 12 December 2023

2352-9148/© 2023 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

caused by the disease, while helping to reduce the death rate [10–12].

Therefore, the use of technology in the initial stage of the disease and symptoms of diabetes is helpful for identification through algorithms that, unlike traditional tests, serve to eliminate errors that occur if performed manually as well as complications [13]., thus, the techniques based on machine learning (ML) and Deep learning (DL), have been applied in different sectors, these being: education [14–19], finance [20–22], transport [23–26], however, studies have also been done in health [27–31]. In fact, if it comes to predicting early risks in people with diabetes as well as complications it is done automatically based on information provided [32–34]. In fact [35], they used K-Nearest Neighbors as a machine learning algorithm, which achieved an accuracy of 85.06 %, also [36] proposed a system to diagnose diabetes based on 3 machine learning techniques, these being: Random Forest, Naive Bayes and Decision Tree, whose accuracies were respectively 73.91 %, 75.65 % and 79.13 %, also, similarly [37] they designed an automatic system to predict diabetes, combining SVM and RBF Kernel, which obtained an accuracy of 83.2 %.

Among the studies regarding the prediction of diabetes using machine learning techniques, which are:

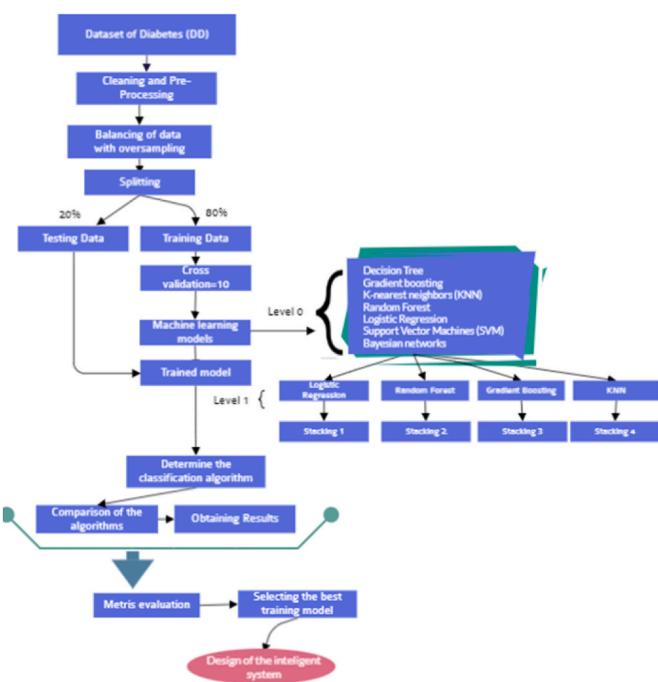
Singh y Singh [38], developed a system based on Stacking called "NSGA-II-Stacking" to predict the onset of type 2 diabetes mellitus in a span of 5 years, to achieve this, a public dataset on diabetes of the Pima indigenous population was used, being the independent algorithms used: Linear SVM (L-SVM), Radial basis function SVM (RBF-SVM), Polynomial SVM (Poly-SVM) and Decision Tree, as well as the KNN algorithm as a meta-classifier to combine the predictions of the base models. Regarding performance metrics, the proposed system achieved an accuracy of 83.8 %, sensitivity of 96.1 %, specificity 79.9 %, F1-Score of 88.5 % and ROC curve of 85.9 %.

Likewise, Kumari et al. [39], investigated the impact of using a classifier to predict diabetes using an ensemble model, in which the proposal consisted of combining three binary classifiers, which included: Random Forest, Logistic regression and Naive Bayes (NB), for which the Ensemble model achieved the best accuracy values of 79.04 %, precision of 73.48 %, recall of 71.45 % and F1 Score of 80.6 %.

In further research, Rajendra and Latifi [40], merged the PIMA diabetes dataset with the Vanderbilt dataset to develop predictive models of diabetes. The main models were built using Logistic Regression and two ensemble techniques: max voting and stacking, where additions were then added to the ensemble models such as: SVM, Decision Tree, KNN and Naive bayes. In both datasets, the ensemble model was observed to outperform the Logistic regression model with accuracy of 77.83 % and 93.41 %, respectively.

Xiong et al[41], applied an ensemble-based technique to assess the risk of type 2 diabetes mellitus in the Chinese urban population, which used a set of data from patients from Nanjing, including 3,845 confirmed cases of DM and 8,000 non-diabetic patients, and used five machine learning algorithms: Multilayer perceptron (MLP), Adaboost, Random Forest, SVM and Gradient Tree Boosting (GTB). The empirical results showed that the combination of ensemble-based classifiers achieved better results, with 91 % accuracy, 95 % specificity, 83 % sensitivity, 97 % AUC and 88 % precision.

Ahmad et al[42], studied the impact of health-related attributes on the prediction of type 2 diabetes mellitus using machine learning approaches, the dataset was composed of 3000 patient records from various Saudi hospitals, and had 16 attributes. Subsequently, the algorithms of Logistic Regression, Random Forest, Decision Tree, Ensemble Majority and SVM were applied for modeling, these models were evaluated twice, using Cross validation with 10 repetitions, one with 9 characteristics and another with eight characteristics. The results of the first set revealed that SVM outperformed the others by achieving 82.1 % accuracy with both nine attributes and eight. On the other hand, the results of the second dataset showed that Random Forest obtained the highest accuracy, reaching 88.27 % with 9 attributes and 87.65 % with eight attributes.



**Fig. 1.** Sequential chart of proposed model.

Various research endeavors involving independent algorithms within the realm of machine learning have been undertaken. Consequently, it is desired to combine these models in order to attain improved outcomes. Because many people around the world have diabetes, but most are unaware of their condition, as they find out when they have major health problems. The study seeks to propose a method that helps correctly identify the probability of having diabetes, which will allow early treatment and prevent complications, such as: kidney failure, amputation of diseases, loss of sight and neuropathy.

Therefore, this article aims to propose a method and 4 combined models based on Stacking in order to predict diabetes. In addition, a web interface was developed with the best model proposed in this study.

Thus, the significance of this study is that using stacking ensemble, i.e. combining multiple models, can provide more accurate predictions by leveraging the strengths of each individual model, leading to more accurate diagnosis of diabetes, as well as improving the stability and consistency of the predictions, and thus can contribute to the clinical reliability of diabetes diagnosis, which is essential for informed medical decision making.

This paper presents three fundamental contributions: First, an approach that describes the process of developing the study for future research is proposed. Second, four models based on independent algorithms are combined. Third, a web application that facilitates interaction with the end user is presented.

This article is divided as follows: Section 1 presents the Introduction. In section 2 the materials and methods are explained, in section 3 the respective analysis of the results and discussion based on the proposed method is exposed, and finally in section 4 the conclusions and future studies of the study are shown.

## 2. Material and methods

In section 2.1. Reference is made to the source of data related to diabetes. Section 2.2. It details the process of the methodology used. A visual representation of it is shown in Fig. 1. First, cleaning and pre-processing was performed. The data was then balanced, where the data was equalized equally. Then in the Splitting, distributing the data 80%-20 %. To proceed with Cross Validation and data training and

**Table 1**  
Dataset collection.

Nº	Property name	Property description	Type of Data	Data range	Missing value
1	Pregnancy	Number of female pregnancies	Integer	0–17	No
2	BMI	BMI (Kg/m <sup>2</sup> )	Float	0–67.1	Yes
3	Insulin	2 h serum insulin	Integer	0–846	Yes
4	Age	Year	Integer	21–81	No
5	Blood Pressure	Diastolic blood pressure (mmHg)	Integer	0–122	Yes
6	Skin thickness	Triceps skinfold thickness (mm)	Integer	0–99	Yes
7	Glucose	2 h blood glucose (mg/dl)	Integer	0–199	Yes
8	Diabetes Pedigree Function	Diabetes spectrum function	Float	0.078–2.42	No
9	Outcome	Diabetes population marker	Integer	0.1	No

testing. Finally, the modeling and evaluation of the proposed Stacking approaches is carried out, for which it is executed based on 7 independent algorithms: K-Nearest Neighbors (KNN), Bayesian networks, Decision Tree, Gradient Boost, Support Vector Machines (SVM), Logistic regression and Random Forest, to then improve the results, 4 Stacking models were implemented: Stacking 1A: Logistic Regression, Stacking 2A: Random Forest, Stacking 3A: Gradient Boosting and Stacking 4A: KNN. The section concludes with a brief description of the assessment parameters used to estimate accuracy in section 2.9. All models were built using the Python programming language.

## 2.1. Data set collection

The dataset consists of 768 instances obtained from the National Institute of Diabetes and Digestive and Kidney Diseases (<https://www.kaggle.com/datasets/mathchi/diabetes-data-set>). [43]. Attributes present in the database include Body Mass Index (BMI), age, glucose, blood pressure, skin thickness, hereditary factor, age, etc.

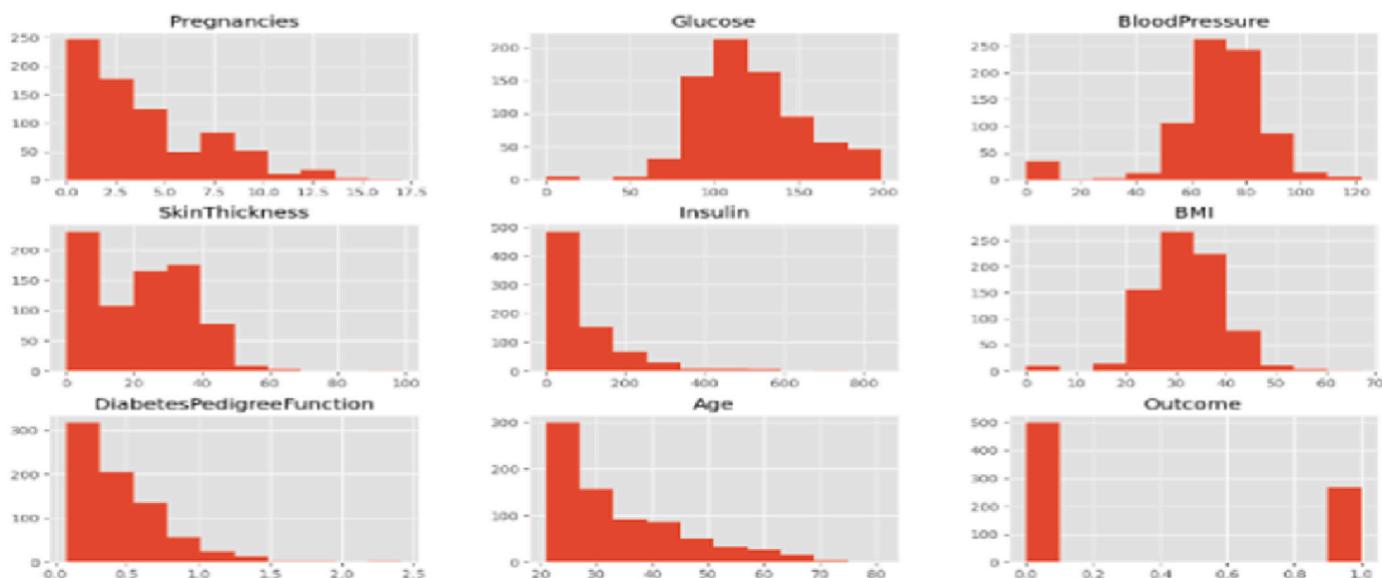


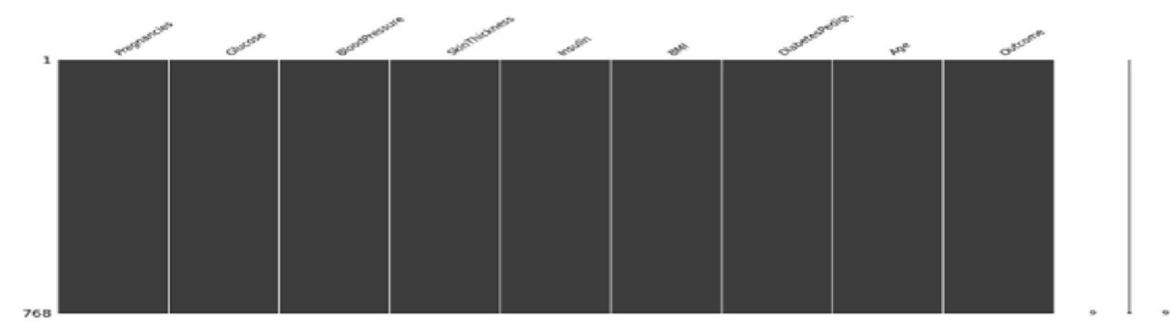
Fig. 2. Frequency diagram of items (1–9) of the diabetes datasets.



Fig. 3. Correlation matrix.

```
In [8]: import missingno as msno
msno.matrix(data)

Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x183fa1ee988>
```

**Fig. 4.** Instruction for displaying missing data.**Fig. 5.** Scanning missing data.

```
In [6]: #numero de missing por columnas
data.isnull().sum()
```

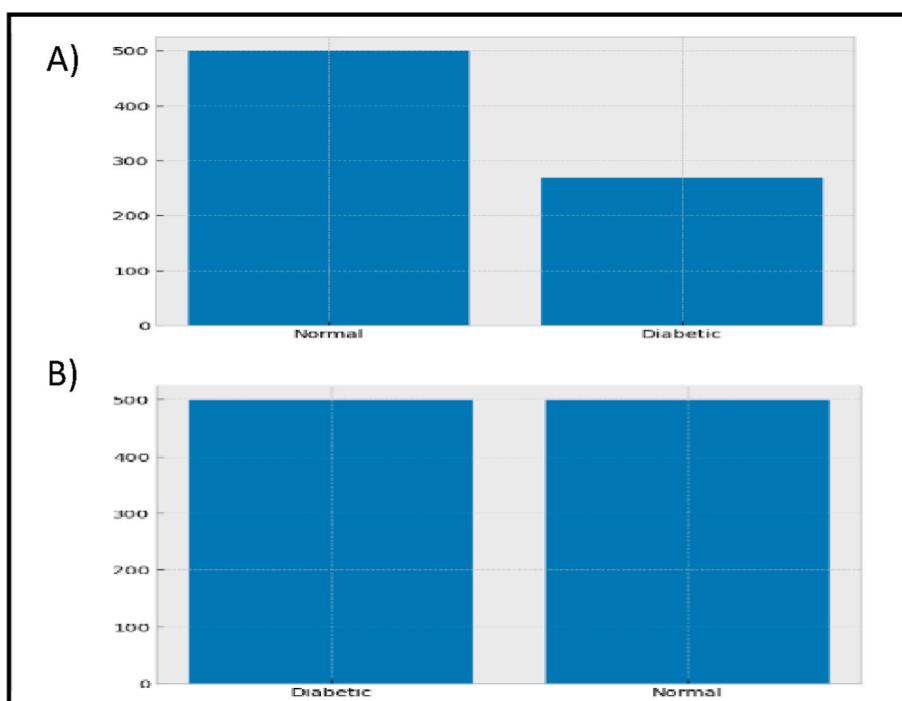
```
Out[6]: Pregnancies      0
Glucose          0
BloodPressure    0
SkinThickness    0
Insulin          0
BMI              0
DiabetesPedigreeFunction 0
Age              0
Outcome          0
dtype: int64
```

**Fig. 6.** Instruction for displaying the amount of missing data.

According to the answers to the items mentioned above, the patient's answers in relation to the questions are assigned ranges. The value ranges, data type and description of all attributes are found in [Table 1](#)

and [Fig. 2](#).

After carrying out a preliminary analysis of the data, a considerable relationship between the variables that predict and the variable that is

**Fig. 7.** Data a) Without balancing b) With balancing.

**Table 2**  
Splitting.

Dataset	Data con Oversampling		
	Training data (80 %)	Validation (20 %)	Total
768 records	614	154	768

predicted is observed in Fig. 3, because its value is greater than 0.7.

## 2.2. Data pre-processing

Prior to the data modeling process, data cleaning can help the model to be able to more effectively extract the real characteristics of the set. In the case of the diabetes dataset, it was not necessary to transform the data, since they did not have missing attribute columns.

To verify that there is no missing data, a code instruction was first used, as shown in Fig. 4, in which it was obtained that there are no empty data as shown in Fig. 5, where it can be seen that in each of the variables of the graph there is no missing data. For example, in the age variable, it can be seen that the bar in its entirety is gray and there are no white lines that would indicate that data is missing. To verify this, we used the instruction as shown in Fig. 6, which allows us to see the amount of missing data for each variable.

Fig. 6 shows that the following variables: pregnancies, glucose, blood pressure, skin thickness, insulin, BMI, diabetes pedigree function, age and outcome, do not have missing data, because the value in all of them is zero.

## 2.3. Balancing of data

At the beginning of the study, the dataset consisted of 768 patient records, with the following values: Normal = 500 and Diabetic = 268.

Subsequently, an oversampling method was applied, in which the data were balanced in relation to the diabetes condition: Normal = 500 and Diabetic = 500. This indicates that the data has been balanced taking into account the majority class, as shown in Fig. 7.

There are 2 methods to balance the data: Undersampling and oversampling. The Methods Oversampling, is a sampling approach to modify the dataset prior to learning, where appends data to original data and extends the size of the minority class [44]. For the study, the oversampling method was considered, because it allowed us to balance the data with as much data as possible (500 data), so that, by generating more data, the algorithms work better and this benefits when training the models.

## 2.4. Splitting

In the experiment conducted, the balanced data was divided into two sets: 80 % was intended for training and 20 % was used for validation. In the case of the experiment, involved data balancing by oversampling, 800 samples were used for training and 200 for validation, as presented in Table 2.

## 2.5. Optimization of models

The hyperparameters of a learning algorithm are those parameters that adjust the learning procedure and define the final values of the models [45], whose objective of hyperparameters optimization is to find the most appropriate configuration that allows optimal results to be obtained from the data efficiently [46]. Fig. 8 displays the various parameters that were allocated to the algorithms during their calibration process, aimed at determining the optimal parameter for each individual algorithm. After conducting the experiments involving different parameter settings for each algorithm, the optimal parameters for both

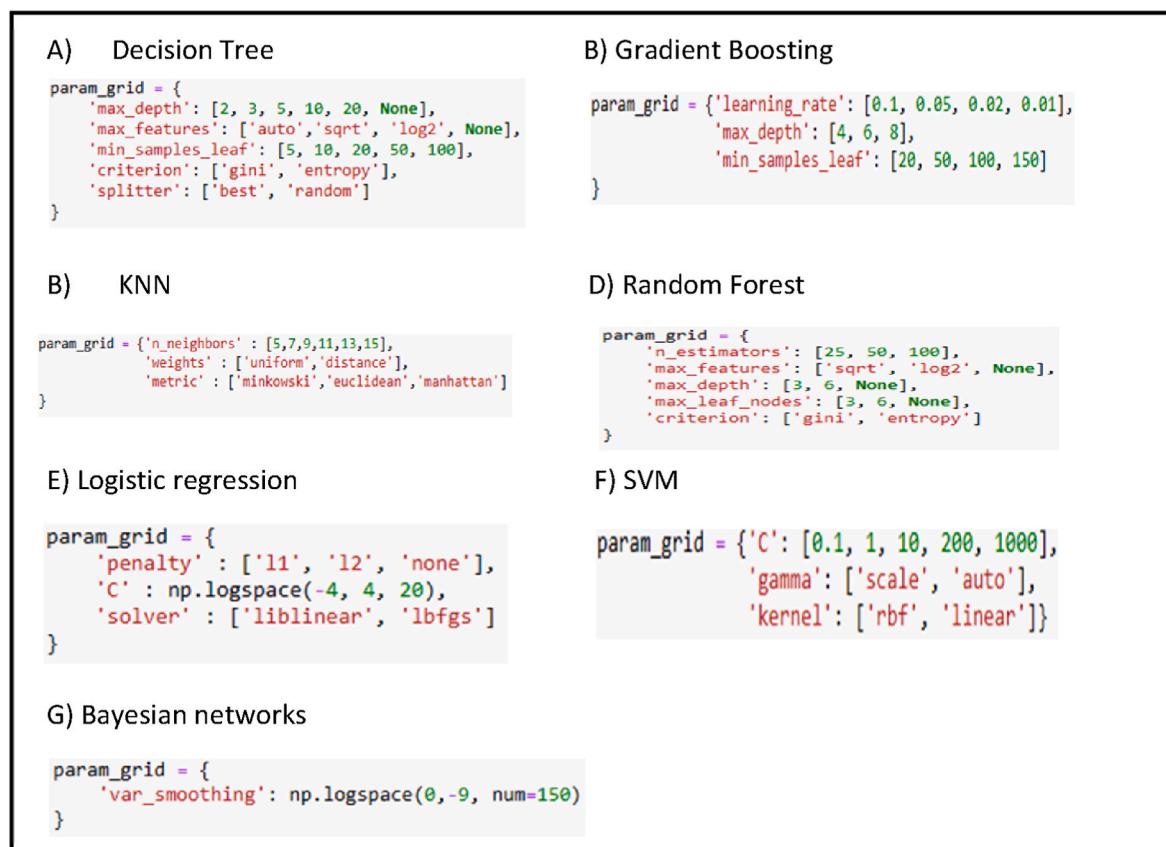


Fig. 8. Algorithm parameters.

**Table 3**  
Algorithm parameters.

Cod.	ALGORITHM	ALGORITHM PARAMETER SETTINGS
ML1	Decision Tree	{'criterion': 'entropy', 'max_depth': None, 'max_features': 'log2', 'min_samples_leaf': 20, 'splitter': 'best'}
ML2	Gradient Boosting	{'learning_rate': 0.02, 'max_depth': 4, 'min_samples_leaf': 20}
ML3	KNN	{'metric': 'manhattan', 'n_neighbors': 13, 'weights': 'uniform'}
ML4	Random Forest	{'criterion': 'entropy', 'max_depth': 3, 'max_features': 'log2', 'max_leaf_nodes': None, 'n_estimators': 25}
ML5	Logistic regression	{'C': 0.615848211066026, 'penalty': 'l2', 'solver': 'lbfgs'}
ML6	SVM	{'C': 10, 'gamma': 'scale', 'kernel': 'rbf'}
ML7	Bayesian networks	{'var_smoothing': 0.0066911899017156355}
ML8	Stacking 1 (Logistic regression)	StackingClassifier(cv = 5, estimators = [('rf', RandomForestClassifier(criterion = 'entropy', max_features = 'sqrt'), ('knn', KNeighborsClassifier(metric = 'manhattan', n_neighbors = 13, weights = 'distance')), ('gradien', GradientBoostingClassifier(max_depth = 8, min_samples_leaf = 20)), ('bayes', GaussianNB(var_smoothing = 1.2805087496773326e-05))], final_estimator = LogisticRegression(C = 0.615848211066026, max_iter = 500, solver = 'liblinear'))
ML9	Stacking 2 (Random forest)	StackingClassifier(cv = 5, estimators = [('knn', KNeighborsClassifier(metric = 'manhattan', n_neighbors = 13, weights = 'distance')), ('tree', DecisionTreeClassifier(max_depth = 20, max_features = 'log2', min_samples_leaf = 5)), ('gradien', GradientBoostingClassifier(max_depth = 8, min_samples_leaf = 20)), ('rl', LogisticRegression(C = 0.615848211066026, max_iter = 500, solver = 'liblinear'))], final_estimator = RandomForestClassifier(criterion = 'entropy', max_features = 'sqrt'))
ML10	Stacking 3 (Gradient Boosting)	StackingClassifier(cv = 5, estimators = [('rf', RandomForestClassifier(criterion = 'entropy', max_features = 'sqrt'), ('knn', KNeighborsClassifier(metric = 'manhattan', n_neighbors = 13, weights = 'distance')), ('tree', DecisionTreeClassifier(max_depth = 20, max_features = 'log2', min_samples_leaf = 5)), ('rl', LogisticRegression(C = 0.615848211066026, max_iter = 500, solver = 'liblinear'))], final_estimator = GradientBoostingClassifier(max_depth = 8, min_samples_leaf = 20))
ML11	Stacking 4 (KNN)	StackingClassifier(cv = 5, estimators = [('tree', DecisionTreeClassifier(max_depth = 20, max_features = 'log2', min_samples_leaf = 5)), ('gradien', GradientBoostingClassifier(max_depth = 8, min_samples_leaf = 20)), ('bayes', GaussianNB(var_smoothing = 1.2805087496773326e-05)), ('rl', LogisticRegression(C = 0.615848211066026, max_iter = 500, solver = 'liblinear'))], final_estimator = KNeighborsClassifier(metric = 'manhattan', n_neighbors = 13, weights = 'distance'))

individual models and the stacking approach were acquired, as demonstrated in Table 3.

Cross-validation is a method for evaluating the performance of a machine learning algorithm and estimating its performance with new data from the same distribution [47]. In this paper, cross-validation with 5 repetitions was used during the training process, with the aim of obtaining an optimal model.

To optimize the algorithms, the GridSearchCV command was used as shown in Fig. 9, for which the following parameters were assigned: param\_grid, refit and verbose, using a Cross validation CV = 5, and after

performing the training, the optimal parameters of the algorithms were obtained as shown in Table 3 with ML1 (Decision Tree) codes. ML2 (Gradient Boosting), ML3 (KNN), ML4 (Random Forest), ML5 (Logistic regression), ML6 (SVM) and ML7 (Bayesian networks).

## 2.6. Base Learner's development

Base models use different machine learning models that make different assumptions about predictions. These base models are then used as training sets for the metamodels, which generate final predictions.

### 2.6.1. K-Nearest network

The K-NN algorithm is one of the fundamental and widely used approaches in machine learning, based on the supervised learning methodology, K-NN applies both in regression tasks and mainly in classification [48]. This algorithm considers the similarity between a new case or dataset and existing data. The new case is then assigned to the category that is most similar to the available options [49]. To calculate the Euclidean distance, you can use this equation:

$$d(j, k) = \sqrt{\sum_{i=1}^n (k_i - j_i)^2} \quad (1)$$

Where:  $j_i$  and  $k_i$  indicates the points in Cartesian coordinates, whereas  $n$  indicates the Euclidean space.

### 2.6.2. Bayesian Networks

It refers to a probabilistic clustering and classification algorithm that is based on the autonomous relationship between different indicators, where method uses a data set as a source of information, carries out an analysis and predicts the classification of the class using Bayes' theorem, to determine the probability of classifying input data and helps predict the class of a test in information, by using the following equation [50]:

$$P\left(\frac{a}{z}\right) = \frac{\left(P\left(\frac{z}{a}\right) * P(a)\right)}{P(z)} \quad (2)$$

$$P\left(\frac{a}{z}\right) = \left(P\left(\frac{z1}{a}\right) * P\left(\frac{z2}{a}\right)\right) * \dots * P\left(\frac{zn}{a}\right) * P(c) \quad (3)$$

Where:  $P(a/z)$  indicates the probability that the target class will regress given the predictor provided, The previous probability of the target class is  $P(a)$ , the probability or  $P(a/c)$  is the measure of the probability that a predictor will be assigned to a class, and the previous probability of the predictor is represented as  $P(z)$ .

### 2.6.3. Decision tree

It is a classifier that has a tree-like structure and is used for both classification and regression, which helps in decision making and offers superior performance compared to other machine learning algorithms used [51], which is established with the following formula:

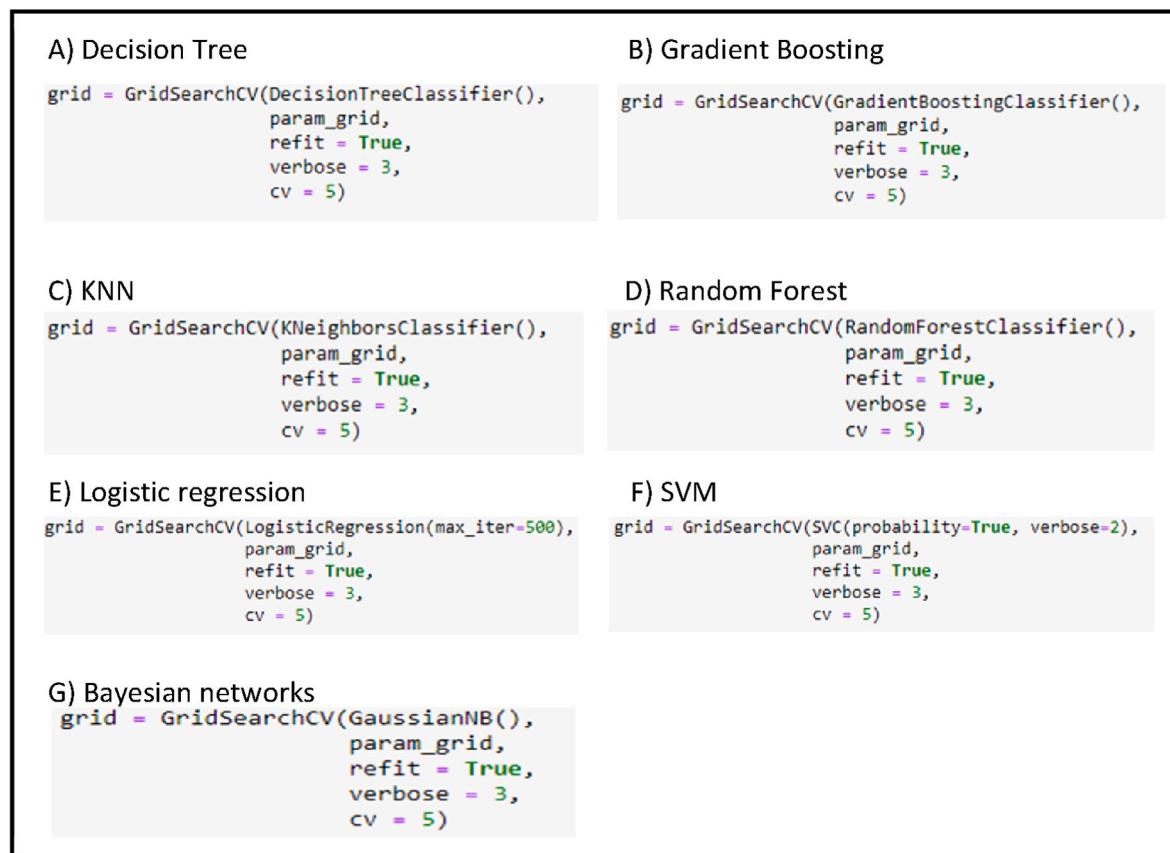
$$\text{Entropy}(S) = \sum_{i=1}^c -P_i \log_2 P_i \quad (4)$$

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S_v) \quad (5)$$

In the given Eq. (4) and Eq. (5),  $c$  denotes the classes present in the dataset, and  $P_i$  represents the instance (i) that has the highest probability. Moreover,  $S$  represents the sample set,  $S_v$  compromises the elements holding  $v$  in attribute  $A$  and  $V(A)$  includes attributes  $A$  values.

### 2.6.4. Gradient Boosting

It is a decision tree set method used for both classification and regression, which combines the predictions of several trees to obtain the



**Fig. 9.** Algorithm Optimization Command (a) Decision Tree, (b) Gradient Boosting, (c) KNN, (d) Random Forest, (e) Logistic regression, (f) SVM, (g) Bayesian networks.

final prediction model; So it is a robust approach to building predictive models, as it helps to improve the accuracy of each individual tree, where you try to correct the deficiencies of the previous model by creating a new model [52]. So the following formula is used:

$$g_t(x) = E_y \left[ \frac{\partial \Psi(y, f(x))}{\partial f(x)} | x \right] f(x) = \hat{f}^{t-1}(x) \quad (6)$$

Where: where  $x = (x_1, \dots, x_d)$  refers to the explanatory input variables and  $y$  to the corresponding labels of the response variable,  $\Psi(y, f)$  is the loss-function,  $f(x)$  is the true functional dependence.

#### 2.6.5. Support Vector Machine (SVM)

It is a widely recognized and effective supervised learning method for the classification, regression and detection of outliers [53]; which uses extreme points or vectors to construct hyperplanes that allow us to assign new data points to the corresponding category by separating the feature space into  $n$  dimensions, whose size of the hyperplane is determined by the number of features present in the dataset [45]. Using the following equation:

$$\text{If } Y_i = +1; w \cdot x_i + b \geq 0 \quad (7) \quad \text{If } Y_i = -1; w \cdot x_i + b \leq 0 \quad (8) \quad \text{For all } i; Y_i (< w, x_i > + b) \geq 0 \quad (9)$$

In the given formula,  $x$  represents a vector point and  $w$  is the associated weight. The data must meet the following criteria: (1) must be greater than zero, (2) must be less than zero, and (3) the separation hyperplane must satisfy the constraints set forth by the corresponding example.

#### 2.6.6. Logistic regression

It is a statistical method used in classification tasks, whose prediction is based on a sigmoid function, which is a shaped curve that produces

output values in the range [0,1]. If the value of the sigmoid function is greater than or equal to 0.5, a positive result is issued; Otherwise, a negative result is issued, it should be noted that the input to the sigmoid function is obtained by calculating the linear regression function, taking into account parameters such as weight and bias [54], this being the following:

$$\log \frac{p}{1 - (p-1)} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k \quad (10)$$

Where:  $\beta_0, \beta_k$  scalar, weight vector,  $x$  a data sample and  $p$  is the number of independent variables.

#### 2.6.7. Random forest

It is a machine learning algorithm as a whole that is based on the bagging technique, which consists of using different random samples to train multiple decision trees and using a voting method to obtain the final result of the classification, this algorithm being considered the best unlike decision tree, this is because it is less likely to overfitting and has better performance [55]. Using the following mathematical equation, taking into account the Gini criterion:

$$Gini(A_i) = 1 - \sum_{i=1}^n p_i^2 \quad (11)$$

In que Eq. (11),  $p_i$  refers to the "probability" of the number of classes in their corresponding position and  $n$  represents the number of classes.

#### 2.7. Stacking ensemble model

It is a machine learning algorithm as a whole, used to achieve greater accuracy through different low-level learners and then combine them

**Table 4**  
Description of the evaluation metrics.

Metric	Description	Formula
Accuracy	It is the total number of predictions correctly classified on the total of records evaluated, where this evidences the effectiveness of the algorithm.	$Accuracy = (TP + TN)/(TP + TN + FP + FN)$
Sensitivity	Represented by the proportion of true positives correctly classified as diabetic patients over the total number of positive records.	$Sensitivity = TP/(TP + FN)$
F1-Score	It is the harmonic mean of recall (sensitivity) and accuracy	$F1Score = 2 * (precision * recall)/(precision + recall)$
Precision	Represented by the number of true positives who have been correctly classified as having diabetes over the total positive prognoses.	$Precision = TP/(TP + FP)$
ROC Curve	It is a graphical representation that shows the ability of a classifier to differentiate between 2 classes (diabetic and non-diabetic patient), which includes TPR (True Positive Rate) on the "y" axis as FPR (False Positive Rate) on the "x" axis, where the higher the value of the ROC Curve, it means that the better the performance of the algorithm.	$1 - Specificity = FP/(TN + FP)$

using a higher-level base meta-learner algorithm, where it should be noted that the meta-learner combines all the predictions of low-level multiple learners to create a general one [56]. Several independent models were used during the modeling process, which were detailed in section 2.7. In addition, four combined models were proposed using stacking approaches with different levels.

In relation to the Ensemble models, the following configurations

were defined: For the Stacking 1 model, a level 0 was established that included K-Nearest Neighbors (KNN), Bayesian networks, Decision Tree, Gradient Boost, Support Vector Machines (SVM), Logistic regression and Random Forest, and a level 1 that consisted of Logistic Regression. On the other hand, the Stacking 2 model had a level 0 composed of K-Nearest Neighbors (KNN), Bayesian Networks, Decision Tree, Gradient Boost, Support Vector Machines, Logistic regression and Random Forest (SVM), and a level 1 composed of Random Forest.

Similarly, the Stacking 3 model was implemented, in which K-Nearest Neighbors (KNN), Bayesian networks, decision Tree, Gradient Boost, Support Vector Machines (SVM), Logistic regression and Random Forest were used at level 0, while at level 1 it was represented by Gradient Boosting. Finally, the Stacking 4 model was applied with a level 0 composed of K-Nearest neighbors (KNN), Bayesian networks, Decision Tree, Gradient Boost, Support Vector Machine (SVM), Logistic regression and Random Forest, and level 1 was represented by K-Nearest Neighbors (KNN).

These model combination strategies were carried out with the purpose of increasing the accuracy and robustness of the prediction process.

## 2.8. Model evaluation

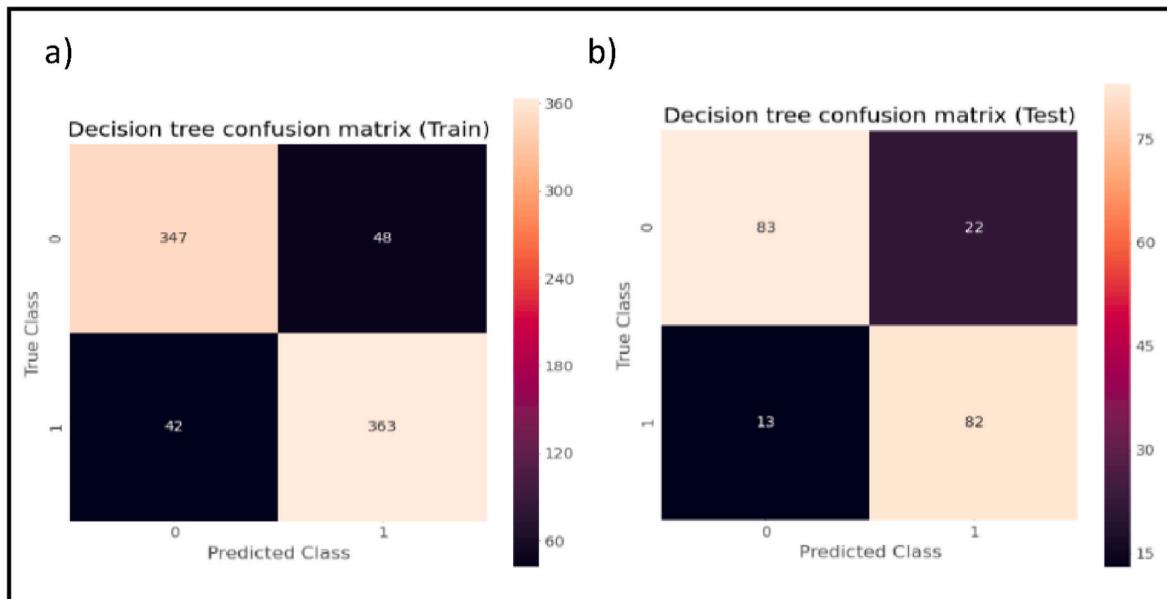
The efficiency of the model was evaluated using various metrics, such as: Accuracy, Sensitivity, F1-Score, Precision [57] and ROC Curve [58], calculated with the formulas presented in Table 4.

The true positive (PT) indicates the number of cases of people who have diabetes that the model correctly predicts. False positives (FPs) represent the number of cases that the model predicts a person has diabetes, but in reality it is not. True negative (NT) refers to the number of cases without diabetes that the model correctly predicts. On the other hand, false negatives (FN) represent the number of cases that the model predicts as not diabetes, but in reality it is.

## 3. Results of base and stacked models

### 3.1. Algorithm confusion matrix

A tabular layout is presented that compares the predicted class label with the actual class label for each class in all data instances. This design allows the following elements to be identified: False positive (FP), False negative (FN), True negative (TN) and true positive (TP) [49].



**Fig. 10.** Confusion matrix (a) decision tree (train), (b) decision tree (test).

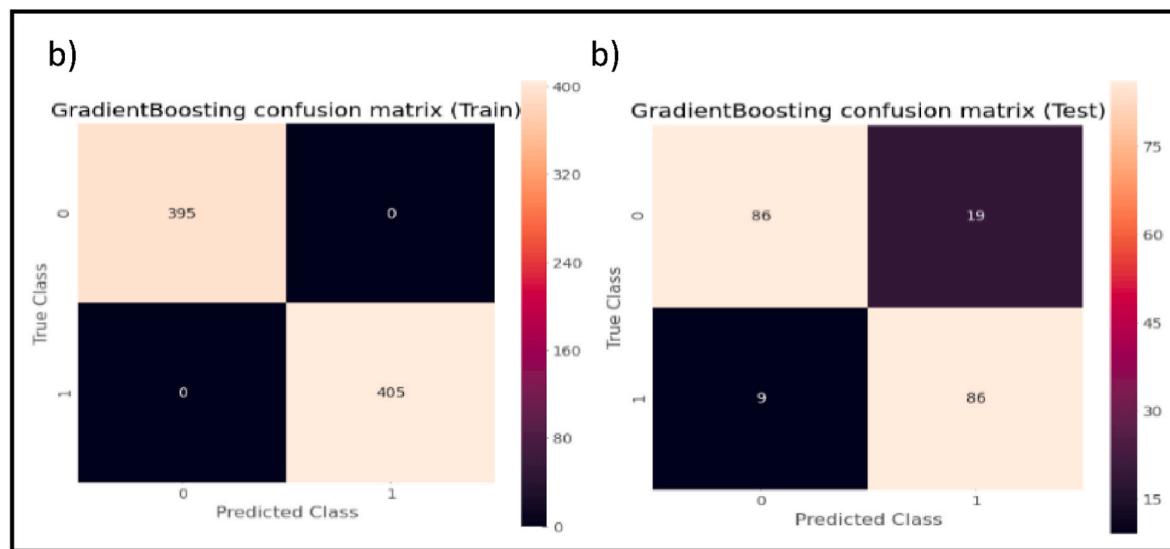


Fig. 11. Confusion matrix (a) gradient boosting (train), (b) gradient boosting (test).

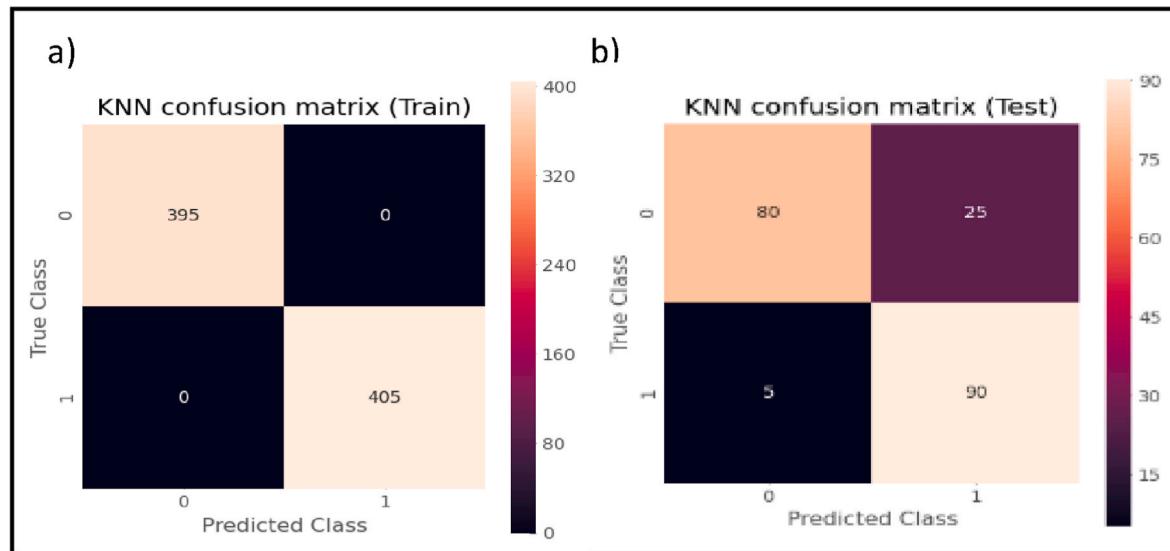


Fig. 12. Confusion matrix (a) KNN (train), (b) KNN (test).

Fig. 10(a) depicts the Decision Tree confusion matrix from the data training. It shows that out of a total of 395 individuals without diabetes, 87.85 % (347) were correctly predicted, while 12.15 % (48) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 405, 89.63 % (363) were correctly predicted, and 10.37 % (42) were predicted incorrectly. Also, Fig. 10(b) depicts the Decision Tree confusion matrix from the data testing. It shows that out of a total of 105 individuals without diabetes, 79.05 % (83) were correctly predicted, while 20.95 % (22) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 95, 86.32 % (82) were correctly predicted, and 13.68 % (13) were predicted incorrectly.

Fig. 11(a) depicts the Gradient Boosting confusion matrix from the data training. It shows that out of a total of 395 individuals without diabetes, 100 % (395) were correctly predicted, while 0 % (0) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 405, 100 % (405) were correctly predicted, and 0 % (0) were predicted incorrectly. Also, Fig. 11(b) depicts the Gradient Boosting confusion matrix from the data testing. It shows that out of a total of 105 individuals without diabetes, 81.9 % (86) were correctly predicted,

while 18.1 % (19) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 95, 90.53 % (86) were correctly predicted, and 9.47 % (9) were predicted incorrectly.

Fig. 12(a) depicts the KNN confusion matrix from the data training. It shows that out of a total of 395 individuals without diabetes, 100 % (395) were correctly predicted, while 0 % (0) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 405, 100 % (405) were correctly predicted, and 0 % (0) were predicted incorrectly. Also, Fig. 12(b) depicts the KNN confusion matrix from the data testing. It shows that out of a total of 105 individuals without diabetes, 76.19 % (80) were correctly predicted, while 23.81 % (25) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 95, 94.74 % (90) were correctly predicted, and 5.26 % (5) were predicted incorrectly.

Fig. 13(a) depicts the Random Forest confusion matrix from the data training. It shows that out of a total of 395 individuals without diabetes, 100 % (395) were correctly predicted, while 0 % (0) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 405, 100 % (405) were correctly predicted, and 0 % (0) were predicted

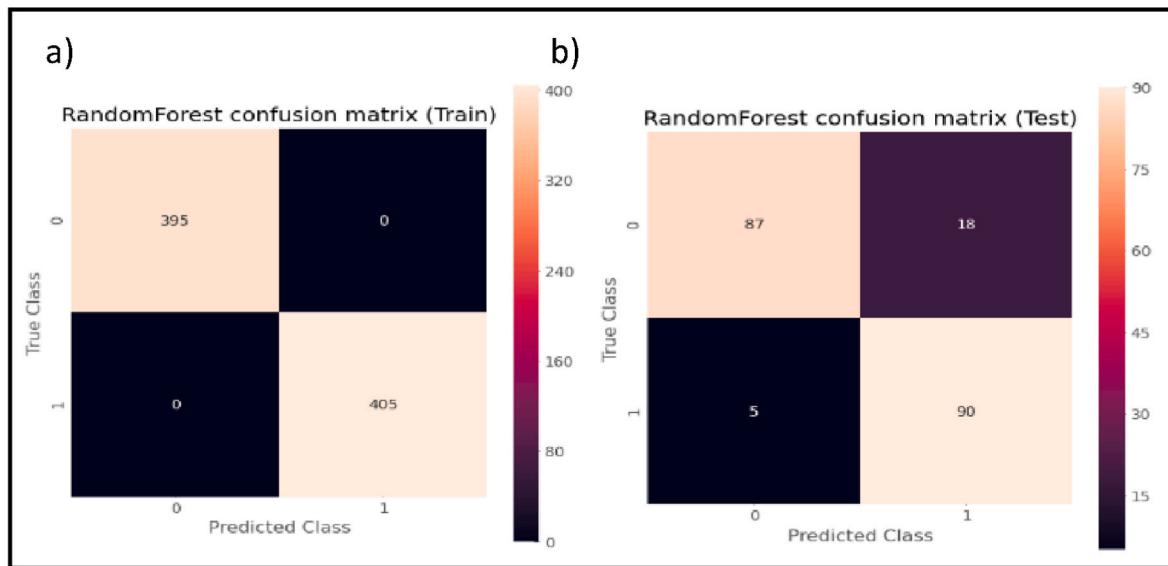


Fig. 13. Confusion matrix (a) random Forest(Train), (b) random forest (test).

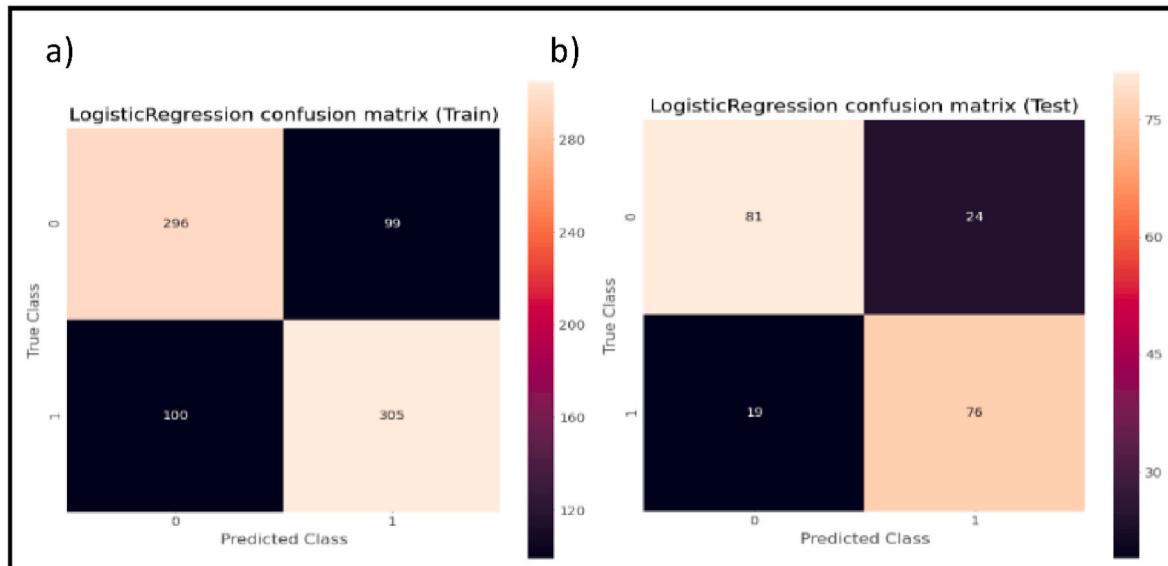


Fig. 14. Confusion matrix (a) logistic regression (train), (b) logistic regression (test).

incorrectly. Also, Fig. 13(b) depicts the Random Forest confusion matrix from the data testing. It shows that out of a total of 105 individuals without diabetes, 82.86 % (87) were correctly predicted, while 17.14 % (18) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 95, 94.74 % (90) were correctly predicted, and 5.26 % (5) were predicted incorrectly.

Fig. 14(a) depicts the Logistic Regression confusion matrix from the data training. It shows that out of a total of 395 individuals without diabetes, 74.94 % (296) were correctly predicted, while 25.06 % (99) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 405, 75.31 % (305) were correctly predicted, and 24.69 % (100) were predicted incorrectly. Also, Fig. 14(b) depicts the Logistic Regression confusion matrix from the data testing. It shows that out of a total of 105 individuals without diabetes, 77.14 % (81) were correctly predicted, while 22.86 % (24) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 95, 80 % (76) were correctly predicted, and 20 % (19) were predicted incorrectly.

Fig. 15(a) depicts the SVM confusion matrix from the data training. It

shows that out of a total of 395 individuals without diabetes, 92.91 % (367) were correctly predicted, while 7.09 % (28) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 405, 95.31 % (386) were correctly predicted, and 4.69 % (19) were predicted incorrectly. Also, Fig. 15(b) depicts the SVM confusion matrix from the data testing. It shows that out of a total of 105 individuals without diabetes, 79.05 % (83) were correctly predicted, while 20.95 % (22) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 95, 76.84 % (73) were correctly predicted, and 23.16 % (22) were predicted incorrectly.

Fig. 16(a) depicts the Bayesian network confusion matrix from the data training. It shows that out of a total of 395 individuals without diabetes, 80.76 % (319) were correctly predicted, while 19.24 % (76) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 405, 66.91 % (271) were correctly predicted, and 33.09 % (134) were predicted incorrectly. Also, Fig. 16(b) depicts the Bayesian network confusion matrix from the data testing. It shows that out of a total of 105 individuals without diabetes, 82.86 % (87) were correctly

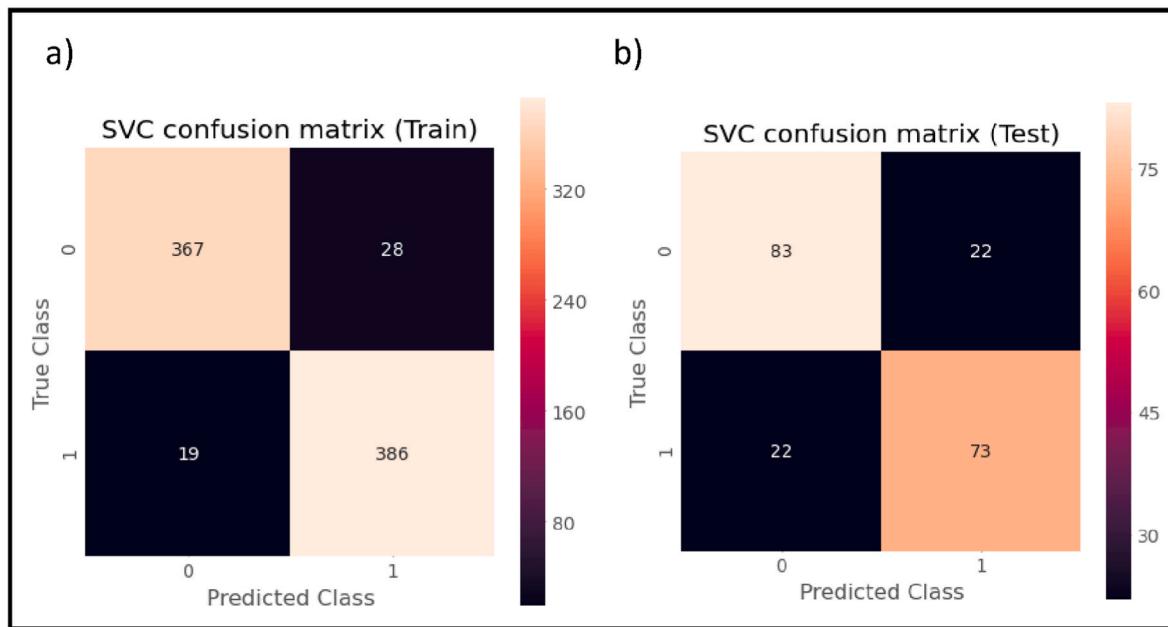


Fig. 15. Confusion matrix (a) SVM (train), (b) SVM(Test).

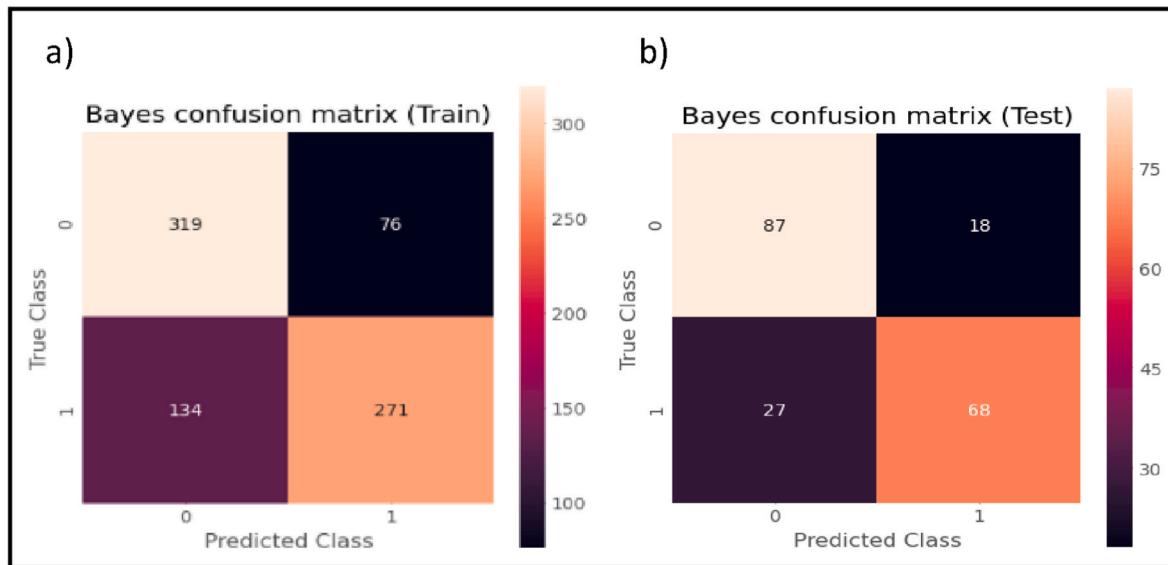


Fig. 16. Confusion Matrix (a) Bayesian network (Train), (b) Bayesian network (Test).

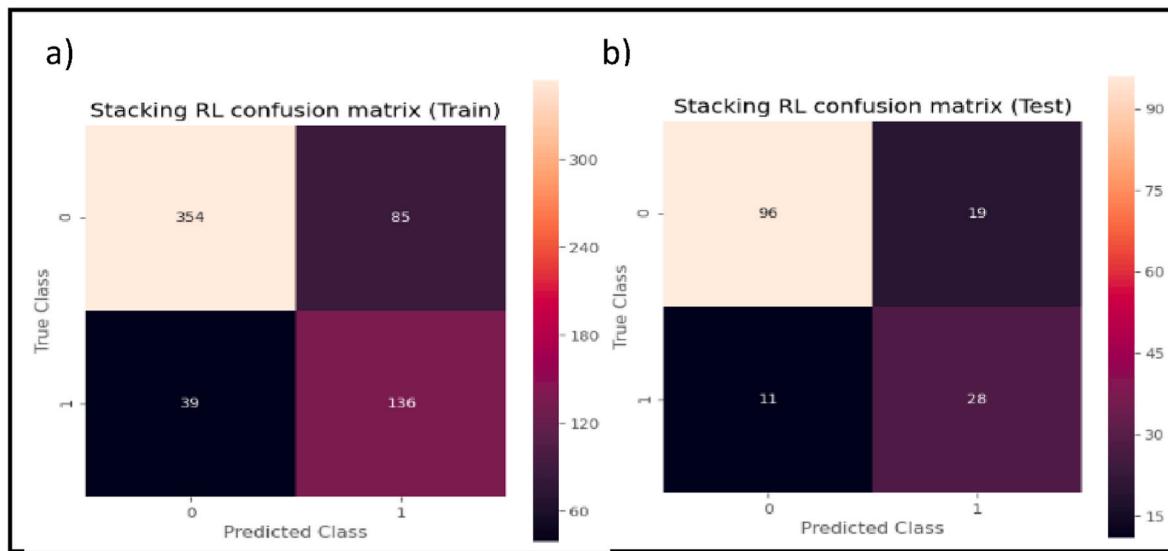
predicted, while 17.14 % (18) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 95, 71.58 % (68) were correctly predicted, and 28.42 % (27) were predicted incorrectly.

Fig. 17(a) depicts the Stacking 1 Logistic Regression from the data training. It shows that out of a total of 439 individuals without diabetes, 80.64 % (354) were correctly predicted, while 19.36 % (85) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 175, 77.71 % (136) were correctly predicted, and 22.29 % (39) were predicted incorrectly. Also, Fig. 17(b) depicts the Stacking 1 Logistic Regression from the data testing. It shows that out of a total of 115 individuals without diabetes, 83.48 % (96) were correctly predicted, while 16.52 % (19) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 39, 71.79 % (28) were correctly predicted, and 28.21 % (11) were predicted incorrectly.

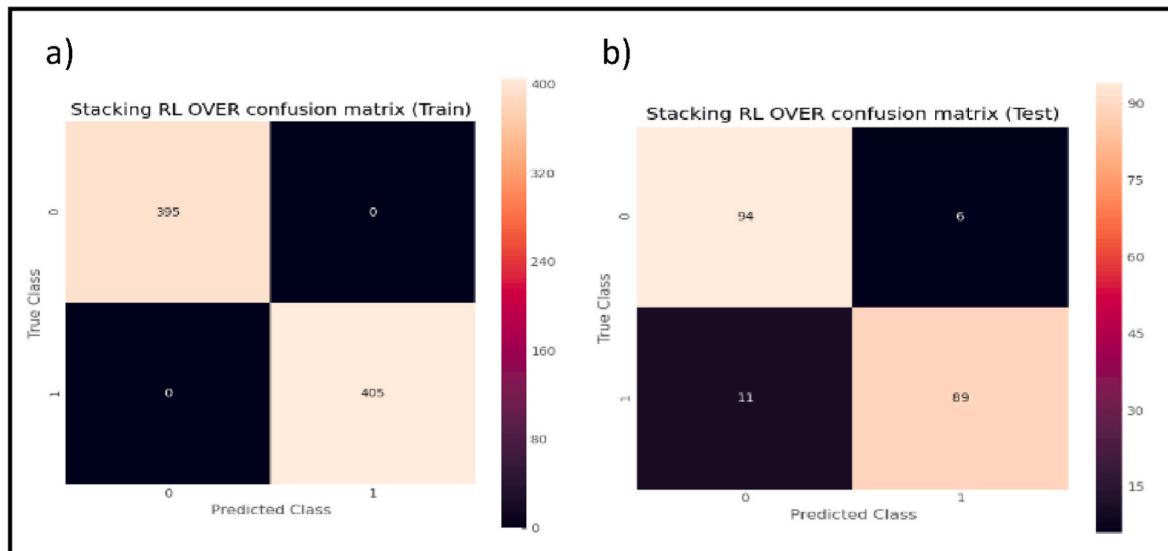
Fig. 18 (a) depicts the Stacking 1 Logistic Regression with oversampling from the data training. It shows that out of a total of 395

individuals without diabetes, 100 % (395) were correctly predicted, while 0 % (0) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 405, 100 % (405) were correctly predicted, and 0 % (0) were predicted incorrectly. Also, Fig. 18(b) depicts the Stacking 1 Logistic Regression with oversampling from the data testing. It shows that out of a total of 100 individuals without diabetes, 94 % (94) were correctly predicted, while 6 % (6) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 100, 89 % (89) were correctly predicted, and 11 % (11) were predicted incorrectly.

Fig. 19(a) depicts the Stacking 2 Random Forest from the data training. It shows that out of a total of 433 individuals without diabetes, 80.6 % (349) were correctly predicted, while 19.4 % (84) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 181, 75.69 % (137) were correctly predicted, and 24.31 % (44) were predicted incorrectly. Also, Fig. 19(b) depicts the Stacking 2 Random Forest from the data testing. It shows that out of a total of 112 individuals



**Fig. 17.** Confusion matrix (a) stacking 1 logistic regression (train), (b) stacking 1 logistic regression (test).



**Fig. 18.** Confusion Matrix (a) Stacking 1 Logistic Regression with oversampling (Train), (b) Stacking 1 Logistic Regression with oversampling (Test).

without diabetes, 83.93 % (94) were correctly predicted, while 16.07 % (18) were predicted incorrectly. Similarly, for the patients with diabetes, out of a total of 42, 69.05 % (29) were correctly predicted, and 30.95 % (13) were predicted incorrectly.

Fig. 20(a) depicts the Stacking 2 Random forest with oversampling from the data training. It shows that out of a total of 394 individuals without diabetes, 100 % (394) were correctly predicted, while 0 % (0) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 406, 99.75 % (405) were correctly predicted, and 0.25 % (1) were predicted incorrectly. Also,

Fig. 20(b) depicts the Stacking 2 Random forest with oversampling from the data testing. It shows that out of a total of 102 individuals without diabetes, 90.2 % (92) were correctly predicted, while 9.8 % (10) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 98, 86.73 % (85) were correctly predicted, and 13.27 % (13) were predicted incorrectly.

Fig. 21(a) depicts the Stacking 3 Gradient Boosting from the data training. It shows that out of a total of 457 individuals without diabetes, 79.43 % (363) were correctly predicted, while 20.57 % (94) were

predicted incorrectly. Similarly, for the patients with diabetes, out a total of 157, 80.89 % (127) were correctly predicted, and 19.11 % (30) were predicted incorrectly. Also, Fig. 21(b) depicts the Stacking 3 Gradient Boosting from the data testing. It shows that out of a total of 116 individuals without diabetes, 83.62 % (97) were correctly predicted, while 16.38 % (19) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 38, 73.68 % (28) were correctly predicted, and 26.32 % (10) were predicted incorrectly.

Fig. 22(a) depicts the Stacking 3 Gradient Boosting with oversampling from the data training. It shows that out of a total of 392 individuals without diabetes, 100 % (392) were correctly predicted, while 0 % (0) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 408, 99.51 % (405) were correctly predicted, and 0.49 % (3) were predicted incorrectly. Also, Fig. 22(b) depicts the Stacking 3 Gradient Boosting with oversampling from the data testing. It shows that out of a total of 102 individuals without diabetes, 89.22 % (91) were correctly predicted, while 10.78 % (11) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 98, 85.71 % (84) were correctly predicted, and 14.29 % (14) were predicted

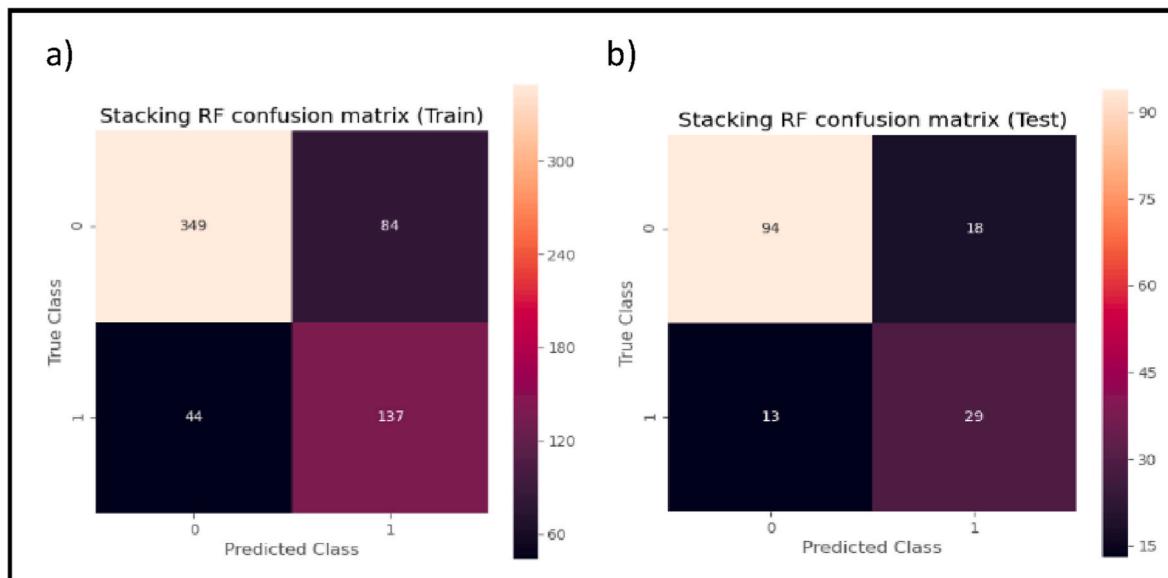


Fig. 19. Confusion matrix (a) stacking 2 random forest (train), (b) stacking 2 random forest (test).

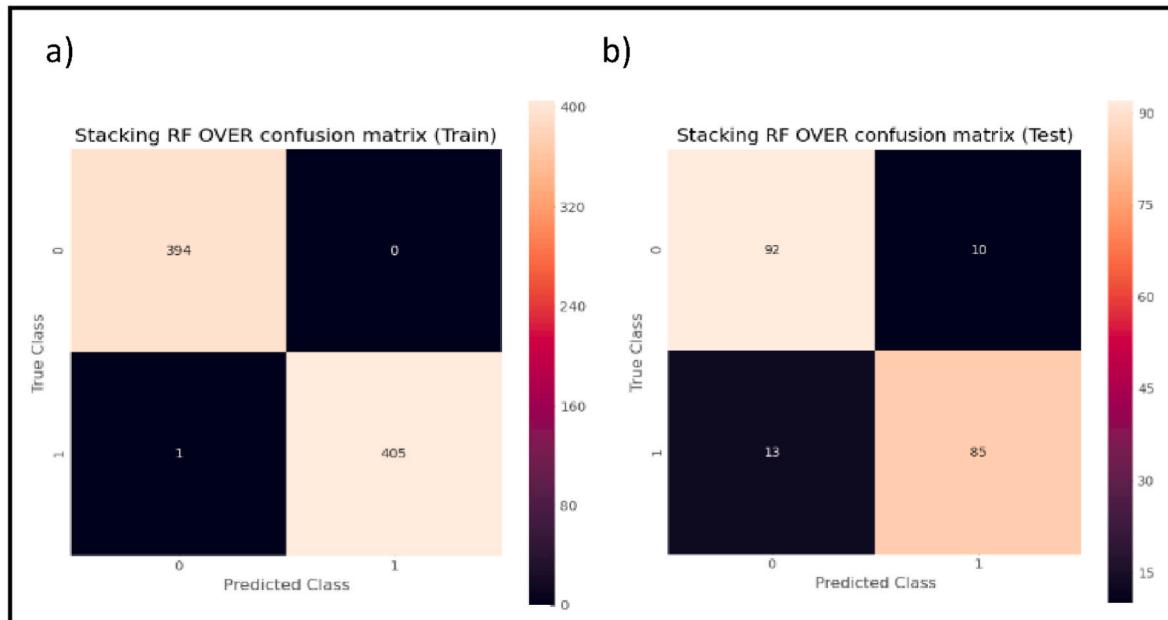


Fig. 20. Confusion Matrix (a) Stacking 2 Random Forest with oversampling (Train), (b) Stacking 2 Random Forest with oversampling (Test).

incorrectly.

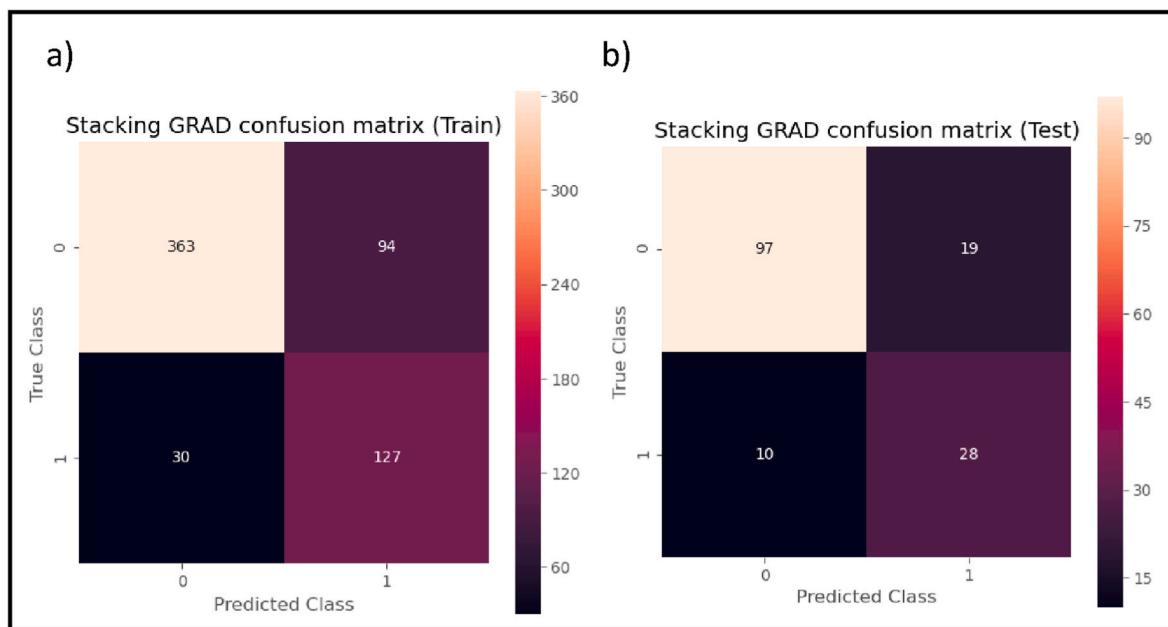
Fig. 23(a) depicts the Stacking 4 KNN from the data training. It shows that out of a total of 420 individuals without diabetes, 81.9 % (344) were correctly predicted, while 18.1 % (76) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 194, 74.74 % (145) were correctly predicted, and 25.26 % (49) were predicted incorrectly. Also, Fig. 23(b) depicts the Stacking 4 KNN from the data testing. It shows that out of a total of 116 individuals without diabetes, 81.9 % (95) were correctly predicted, while 18.1 % (21) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 38, 68.42 % (26) were correctly predicted, and 31.58 % (12) were predicted incorrectly.

Fig. 24(a) depicts the Stacking 4 KNN with oversampling from the data training. It shows that out of a total of 379 individuals without diabetes, 99.74 % (378) were correctly predicted, while 0.26 % (1) were predicted incorrectly. Similarly, for the patients with diabetes, out a

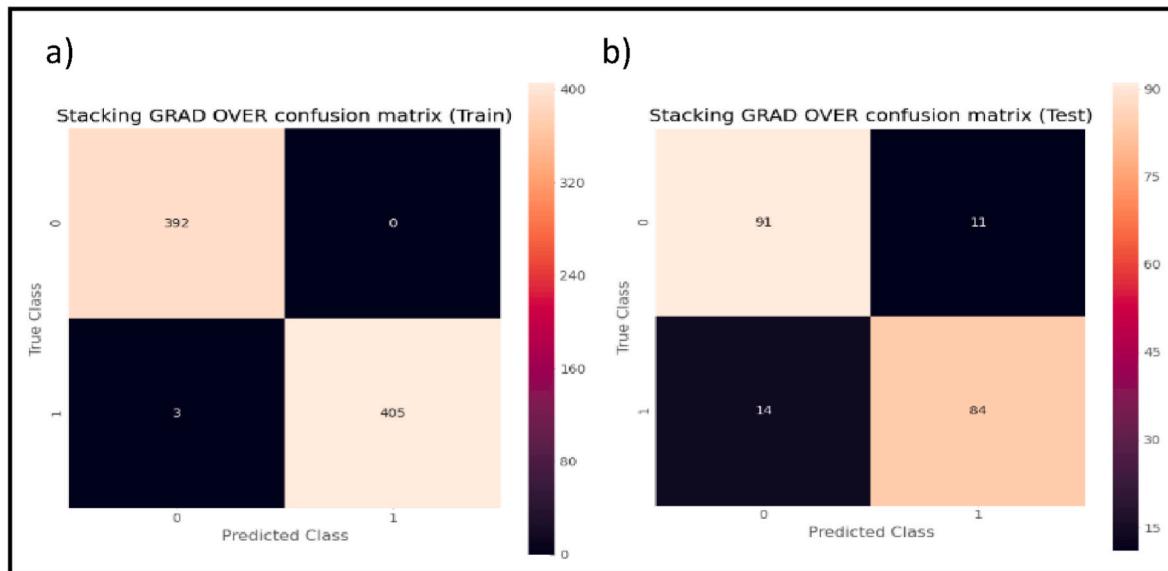
total of 421, 95.96 % (404) were correctly predicted, and 4.04 % (17) were predicted incorrectly. Also, Fig. 24(b) depicts the Stacking 4 KNN with oversampling from the data testing. It shows that out of a total of 93 individuals without diabetes, 93.55 % (87) were correctly predicted, while 6.45 % (6) were predicted incorrectly. Similarly, for the patients with diabetes, out a total of 107, 83.18 % (89) were correctly predicted, and 16.82 % (18) were predicted incorrectly.

### 3.2. Evaluation metrics

Table 5 and Fig. 25 present a comparison of the performance of the proposed combined algorithms, evaluated in terms of the performance measures mentioned above, for experiments. According to the results of the Stacking 1A using Oversampling (Test) obtained the best values of the mentioned metrics, these being: Accuracy = 91.5 %, Sensitivity = 91.6 %, F1 Score = 91.49 % and Precision = 91.5 %. Likewise, with



**Fig. 21.** Confusion matrix (a) stacking 3 gradient boosting (train), (b) stacking 3 gradient boosting (test).



**Fig. 22.** Confusion Matrix (a) Stacking 3 Gradient Boosting with oversampling (Train), (b) Stacking 3 Gradient Boosting with oversampling (Test).

respect to the ROC Curve metric, as shown in Table 5 and Appendix A and Appendix B, the models that obtained the highest values were: Random Forest (Independent), Stacking 1A (Logistic regression) with Oversampling and Stacking 2A (Random Forest) with Oversampling, all of them with 97 % using testing data. Table 6 shows a comparative analysis of the evaluation metrics: accuracy, precision, sensitivity, F1-Score and ROC Curve, obtained by different authors, as well as by the model proposed in the study.

After examining the data presented in Table 6, we can see that the algorithms that yielded the most outstanding results in the test were: Stacking 1A with oversampling. Based on these findings, we proceeded to develop the interface for intelligent system, as illustrated in Appendix C, that would facilitate the direct interaction of doctors and patients.

In Appendix C, Experiments were conducted with recent data, it is therefore the Stacking 1A algorithm demonstrated an identical prediction: DIABETIC, in (a) detail the Intelligent system interface with

Stacking 1A for a single data and (b) detail the Intelligent system interface with Stacking 1A for many data, indicating in both a good accuracy of 100 %.

### 3.3. Discussion

The study proposed a method and 4 combined models based on Stacking to predict diabetes, which consisted of 768 patient records and 9 attributes. At the beginning of the study, with unbalanced data and then applying the oversampling method, equalizing the data to 500 instances.

For the attribute, Glucose, as detailed in Table 1, the value is 2 h blood glucose (mg/dl), within the range 0–99.

As detailed in Table 5, when comparing the results with the models Stacking 1A (Logistic regression) with Oversampling, Stacking 2A (Random Forest) with Oversampling and with the Random Forest

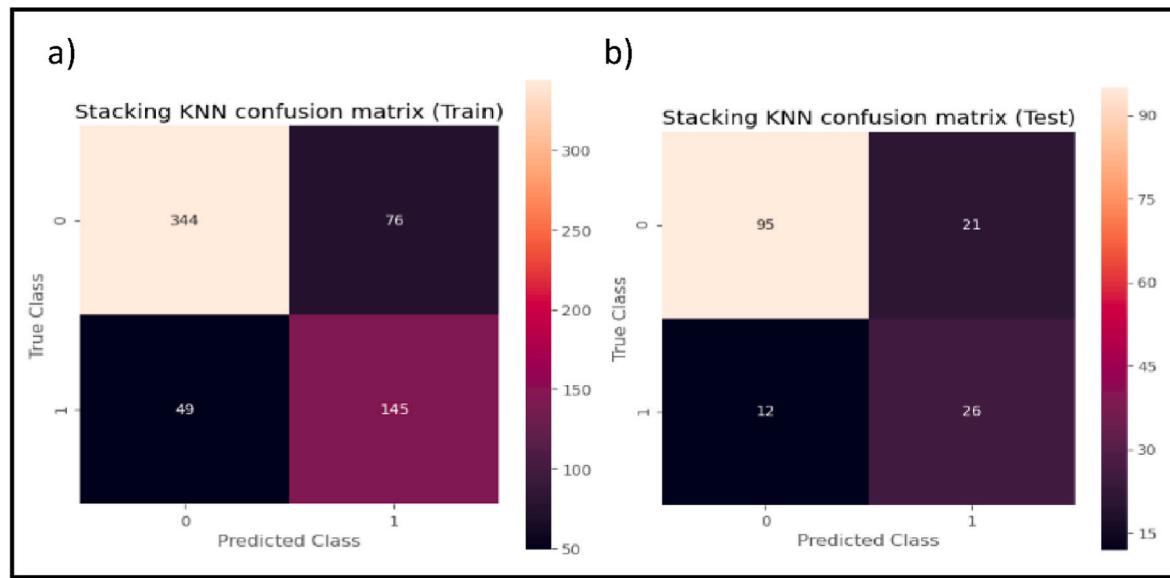


Fig. 23. Confusion matrix (a) stacking 4 KNN (train), (b) stacking 4 KNN (test).

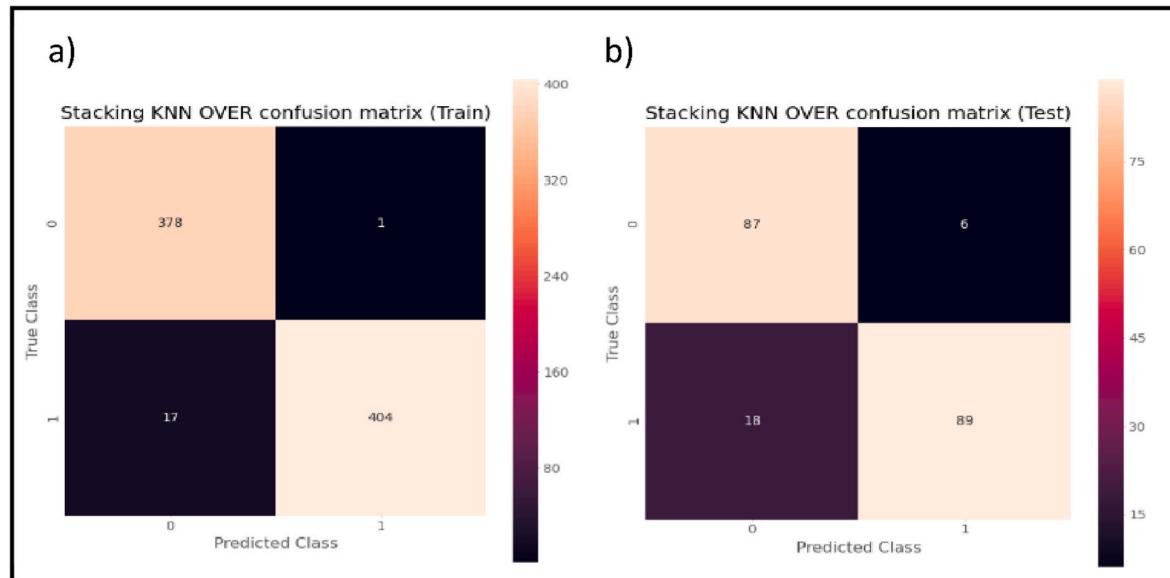


Fig. 24. Confusion Matrix (a) Stacking 4 KNN with oversampling (Train), (b) Stacking 4 KNN with oversampling (Test).

algorithm (Independent). The highest Accuracy, Sensitivity, F1-Score and Precision scores of 91.5 %, 91.6 %, 91.49 % and 91.5 % were obtained with Stacking 1A (Logistic regression) with Oversampling. However, with respect to the ROC Curve, it is visualized that both Stacking 1A, Stacking 2A and Random Forest (Independent) achieved a value greater than 97 %. The scores of the metrics achieved with Stacking 1A, Stacking 2A and Random Forest are higher than those obtained with the other independent machine learning algorithms: Decision Tree, Gradient Boosting, KNN, Logistic Regression, SVM and Bayesian networks.

Taking into account the results obtained in Table 5, Stacking 1 using Oversampling achieved the best values of all metrics. Roy et al. [36], compared 3 machine learning methods, which are: Random Forest, Naive Bayes and Decision Tree, and obtained an accuracy of 73.91 %, 75.65 % and 79.13 %; also Rajendra and Latifi [40], took into account the information provided by the Pima Diabetes Dataset which obtained accuracy values of 77.83 % (experiment 1), however 93.41 % (experiment 2), using ensemble stacking. Therefore, these results are confirmed

with our algorithm, where it is evident that logistic regression using the Stacking 1A technique with Oversampling, which achieved the highest value of accuracy = 91.5 %.

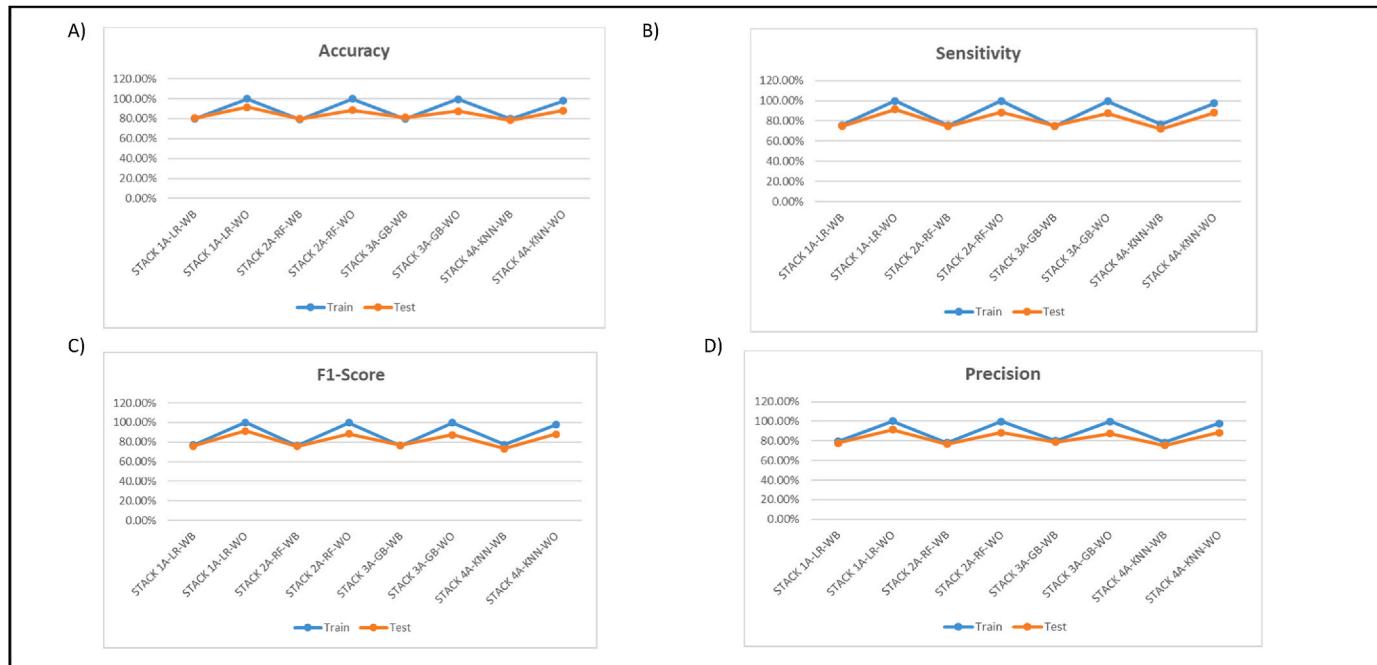
El-Sappagh et al. [61], proposed a set framework for diagnosing diabetes by optimally employing multiple classifiers based on Bagging techniques and random subspaces, for which they took into account a dataset of 60 patients and 13 attributes using 5 machine learning algorithms: Random Forest, Bagging, Voting, Adaboost and ensemble model, where it can be seen that the ensemble model achieved the best sensitivity of 90.2 %. However, it is contrasted with the study by Singh and Singh [38], used data from the Pima indigenous population of 5 consecutive years, which achieved the best values of sensitivity of 96.1 %, this indicates that the model has a low rate of false negatives, which implies that it does not let many positive cases pass undetected. Important in situations where missing positive cases is dangerous, such as in medical applications, where it is critical not to miss the detection of serious illness.

These results are surpassed by what was obtained in our study, since

**Table 5**

Performance metric scores of experimental methods using combined algorithms.

	Algorithm	Acronym	Accuracy		Sensitivity		F1-Score		Precision		ROC Curve	
			Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Independent (Without data balancing)	Decision Tree	DT	88.75 %	82.5 %	88.74 %	82.68 %	88.75 %	82.5 %	88.76 %	82.65 %	96 %	91 %
	Gradient Boosting	GB	100 %	86 %	100 %	86.22 %	100 %	86 %	100 %	86.22 %	100 %	93 %
	K-nearest neighbors (KNN)	KNN	100 %	85 %	100 %	85.46 %	100 %	84.96 %	100 %	86.19 %	100 %	95 %
	Random Forest	RF	100 %	88.5 %	100 %	88.8 %	100 %	88.5 %	100 %	88.95 %	100 %	97 %
	Logistic Regression	LR	75.13 %	78.5 %	75.12 %	78.57 %	75.12 %	78.49 %	75.12 %	78.5 %	82 %	88 %
	Support Vector Machines (SVM)	SVM	94.13 %	78 %	94.11 %	77.94 %	94.12 %	77.94 %	94.16 %	77.94 %	98 %	84 %
	Bayesian networks	BN	73.75 %	77.5 %	73.84 %	77.22 %	73.66 %	77.3 %	74.26 %	77.69 %	82 %	86 %
Second level	Stacking 1A (Logistic regression)- Without balancing	STACK 1A-LR-WB	79.8 %	80.52 %	75.81 %	74.65 %	76.89 %	75.8 %	79.18 %	77.64 %	89 %	85 %
	Stacking 1A (Logistic regression)- With Oversampling	STACK 1A-LR-WO	100 %	91.5 %	100 %	91.6 %	100 %	91.49 %	100 %	91.5 %	100 %	97 %
	Stacking 2A (Random Forest)- Without balancing	STACK 2A-RF-WB	79.15 %	79.87 %	75.4 %	74.78 %	76.33 %	75.51 %	78.15 %	76.49 %	88 %	86 %
	Stacking 2A (Random Forest)- With Oversampling	STACK 2A-RF-WO	99.88 %	88.5 %	99.87 %	88.55 %	99.87 %	88.49 %	99.88 %	88.47 %	100 %	97 %
	Stacking 3A (Gradient Boosting)- Without balancing	STACK 3A-GB-WB	79.8 %	81.17 %	74.92 %	75.11 %	76.3 %	76.44 %	80.16 %	78.65 %	86 %	89 %
	Stacking 3A (Gradient Boosting)- With Oversampling	STACK 3A-GB-WO	99.62 %	87.5 %	99.62 %	87.54 %	99.62 %	87.48 %	99.63 %	87.46 %	100 %	96 %
	Stacking 4A (KNN)- Without balancing	STACK 4A-KNN-WB	79.64 %	78.57 %	76.57 %	72.05 %	77.25 %	73.19 %	78.32 %	75.16 %	86 %	82 %
	Stacking 4A (KNN)- With Oversampling	STACK 4A-KNN-WO	97.75 %	88 %	97.72 %	88.27 %	97.75 %	88 %	97.85 %	88.36 %	100 %	93 %

**Fig. 25.** Evaluation metric (a) Accuracy (Stacking models), (b) Sensitivity (Stacking models), (c) F1-Score (Stacking models), (d) Precision (Stacking models).

it was evidenced that Logistic Regression using the Ensemble Stacking 1 technique was the best technique, which was 91.6 %.

Morgan et al. [62], in their research developed a Max Voting ensemble approach with the purpose of categorizing patients suffering

from diabetes, for this 3 machine learning algorithms were used, these being: Generalized Linear Regression (GLM), Support vector machines (SVM) and artificial neural networks (ANN), where it could be observed that Max Voting Ensemble reached the best F1-Score value of 90.22 %.

**Table 6**

Metrics to detect diabetes disease with different machine learning techniques.

Authors	Algorithm	Accuracy	Sensitivity	F1-Score	Precision	ROC Curve
Singh and Singh [38]	Linear SVM (L-SVM)	76.2 %	88.4 %	82.9 %	—	83.6 %
	Radial basis function SVM (RBF-SVM)	69.1 %	92.5 %	79.6 %	—	72.8 %
	Polynomial SVM (Poly-SVM)	76.3 %	88.2 %	83.2 %	—	82.1 %
	Decision Tree	73.7 %	82.4 %	80.2 %	—	75.1 %
	NSGA-II-Stacking	83.8 %	96.1 %	88.5 %	—	85.9 %
	Random Forest	77.48 %	58.75 %	64.38 %	71.21 %	78.1 %
Kumari et al. [39]	Logistic regression	74.89 %	61.25 %	62.82 %	64.47 %	80.1 %
	Naive Bayes (NB)	74.12 %	65 %	63.41 %	61.9 %	79.01 %
	Ensemble model	79.04 %	71.45 %	80.6 %	73.48 %	80.98 %
	SVM	Experiment: 74.89 %, Experiment 2: 88.03 %	—	—	—	—
Rajendra and Latifi [40]	Decision Tree	Experiment 1: 72.29 %, Experiment 2: 88.03 %	—	—	—	—
	KNN	Experiment 1: 67.97 %, Experiment 2: 88.03 %	—	—	—	—
	Naive bayes	Experiment 1: 74.03 %, Experiment 2: 88.89 %	—	—	—	—
	Logistic regression	Experiment 1: 75.32 %, Experiment 2: 88.89 %	—	—	—	—
	Ensemble model	Experiment 1: 77.83 %, Experiment 2: 93.41 %	—	—	—	—
	Multilayer perceptron (MLP)	87 %	—	—	—	—
Xiong et al. [41]	Adaboost	86 %	—	—	—	—
	Random Forest	86 %	—	—	—	—
	SVM	86 %	—	—	—	—
	Gradient Tree Boosting (GTB)	86 %	—	—	—	—
	Ensemble-based classifiers	91 %	83 %	—	88 %	97 %
Ahmad et al. [42]	Logistic Regression	Experiment 1: 80.86 %, Experiment 2: 83.33 %	Experiment 1: 80.86 %, Experiment 2: 83.33 %	Experiment 1: 80.83 %, Experiment 2: 83.04 %	Experiment 1: 80.95 %, Experiment 2: 83.31 %	—
	Random Forest	Experiment 1: 81.48 %, Experiment 2: 88.27 %	Experiment 1: 81.48 %, Experiment 2: 88.27 %	Experiment 1: 81.38 %, Experiment 2: 88.29 %	Experiment 1: 81.91 %, Experiment 2: 88.31 %	—
	Decision Tree	Experiment 1: 74.07 %, Experiment 2: 80.86 %	Experiment 1: 74.07 %, Experiment 2: 80.86 %	Experiment 1: 74.06 %, Experiment 2: 81.03 %	Experiment 1: 74.07 %, Experiment 2: 81.5 %	—
	Ensemble Majority	Experiment 1: 77.78 %, Experiment 2: 83.95 %	Experiment 1: 77.78 %, Experiment 2: 83.95 %	Experiment 1: 77.66 %, Experiment 2: 83.84 %	Experiment 1: 78.14 %, Experiment 2: 83.34 %	—
	SVM	Experiment 1: 82.1 %, Experiment 2: 84.57 %	Experiment 1: 82.1 %, Experiment 2: 84.57 %	Experiment 1: 82.05 %, Experiment 2: 84.22 %	Experiment 1: 82.3 %, Experiment 2: 84.74 %	—
	Linear-SVM	77.47 %	77.5 %	76.7 %	77 %	72.7 %
Singh and Singh [59]	Polynomial	55.34 %	55.3 %	56.3 %	58.5 %	54.4 %
	Radial basis function	65.1 %	65.1 %	51.3 %	42.4 %	50 %
	Sigmoid	65.1 %	65.1 %	51.3 %	42.4 %	50 %
	Stacking (SMO as meta-learner)	79 %	78.9 %	77.8 %	78.5 %	73.2 %
Liu et al. [60]	Majority voting	—	63.7 %	—	—	69.9 %
	Weighted voting	—	63.7 %	—	—	70.1 %
	Majority voting with model selection	—	66.2 %	—	—	80.2 %
	Weighted voting with model	—	66.2 %	—	—	80.2 %
	Stacking: Logistic regression	—	62.7 %	—	—	69.8 %
	Stacking: LASSO	—	62.7 %	—	—	69.9 %
El-Sappagh et al. [61]	Stacking: SCAD	—	62.7 %	—	—	69.9 %
	Stacking: MCP	—	62.7 %	—	—	69.9 %
	Stacking: Step wise regression	—	62.7 %	—	—	69.9 %
	Random forest	50 %	40.6 %	46.4 %	54.2 %	—
	Bagging	53.3 %	53.1 %	54.8 %	56.7 %	—
	Voting	55 %	40.6 %	49.1 %	61.9 %	—
Morgan et al. [62]	AdaBoost	60 %	50 %	57.1 %	66.7 %	—
	Ensemble model	90 %	90.2 %	92.5 %	94.9 %	—
	Support vector machine	89.82 %	87.5 %	90.02 %	92.69 %	92.08 %
	Artificial neural network	88.46 %	85.59 %	88.79 %	92.24 %	90.5 %
Kibria et al. [63]	Generalized linear regression (GLM)	88.01 %	84.87 %	88.4 %	92.24 %	90.5 %
	Max voting Ensemble	90.05 %	87.88 %	90.22 %	92.69 %	90.5 %
	Artificial neural network	79 %	78 %	78 %	77 %	87 %
	Support vector machine	79 %	81 %	80 %	79 %	88 %
	Logistic regression	78 %	79 %	78 %	78 %	86 %
	Random forest	88 %	88 %	87 %	87 %	94 %
	XGBoost	88 %	89 %	88 %	88 %	92 %
	Adaboost	83 %	85 %	83 %	82 %	95 %
The proposed model	Voting Classifier (XGB + RF)	90 %	89 %	89 %	88 %	95 %
	Random Forest (Without data balancing)	88.5 %	88.8 %	88.5 %	88.95 %	97 %

(continued on next page)

**Table 6 (continued)**

Authors	Algorithm	Accuracy	Sensitivity	F1-Score	Precision	ROC Curve
	Stacking 1A (Logistic regression)- With Oversampling	91.5 %	91.6 %	91.49 %	91.5 %	97 %
	Stacking 2A (Random Forest)- With Oversampling	88.5 %	88.55 %	88.49 %	88.47 %	97 %

Kibria et al. [63], proposed a voting classifier approach to detect diabetes in 768 instances using the following machine learning algorithms: Artificial neural network, Support vector machine, Logistic regression, Random forest, XGBoost and Adaboost, which Voting Classifier (XGB + RF) achieved the best F1-Score value of 89 %.

Our algorithm outperforms the others in this case, as it achieved a higher F1-Score of 91.49 % by employing Logistic Regression-Stacking 1A with oversampling.

Likewise, Xiong et al. [41], used a dataset of the urban population of China, this being 3,445 data records, which achieved the following results of precision = 88 %; where it should be noted that all the results obtained were achieved using the Ensemble Stacking method. Singh and Singh [59], made use of individual learners (Linear-SVM, Polynomial, Radial basis function, Sigmoid) and Stacking to predict the diabetes of 768 people, which dataset was collected from the Pima Indian Diabetes (PID), where it is evident that Stacking (SMO as meta-learner) was the algorithm that reached the best Precision of 78.5 %.

In the case of our algorithm, this is superior, where it is evident that the Logistic Regression using the Stacking 1A technique with Oversampling achieved the best precision value of 91.5 %.

Liu et al. [60], proposed the use of machine learning algorithms in order to increase the accuracy of diabetes predictions in 5,481 people through non-intrusive risk assessment systems, for which they made use of combined methods such as Majority voting and Stacking (Logistic regression, Lasso, SCAD, MCP and Step wise regression). where Majority voting with model selection and Weighted voting with model reached the best values of ROC Curve = 80.2 %, while Xiong et al. [41], in their study reached an area under the ROC curve of 97 %, both using stacking models. In our study, the best value was 97 % using Stacking 1A, stacking 2A and Random Forest, which confirms the usefulness of using combined methods.

Finally, this article is the first to propose a method, and perform experiments with a greater number of base algorithms (7 algorithms), which allows to obtain the best models to have an optimal calculation of precision, which is also the first article that makes a system that can work on the web page or mobile devices that in no research has been done, being indispensable for the early identification of patients at risk of having diabetes.

#### 3.4. Limitations

Overall, our research demonstrates the effectiveness of using stacking methods to predict diabetes by making use of people's characteristics. However, due to the complexity of the disease, our article had some limitations: First, the size of the dataset was one of the deficiencies considered, due to the few existing sources about the disease, also the small number of attributes collected from the Kaggle database, so if they took into account a greater amount of data the accuracy could increase. On the other hand, while it is true that the ensemble Stacking 1A method using oversampling was the one that obtained the best values of the metrics and improved the accuracy of the prediction, other feasible algorithms could also be chosen to optimize the performance of the models, so it is difficult to say that the most robust Stacking framework was established.

#### 4. Conclusions and future studies

Diabetes is one of the most common, chronic and most complicated

diseases, which present a large part of the population worldwide. The test to rule out whether a person has diabetes is expensive, in addition to not being as efficient, due to the alterations that can have (when the patient consumes medications or is using insulin before performing the test). So, predicting diabetes can help people know and improve their health.

In this article he proposed a method and 4 combined models based on Stacking to predict diabetes. In addition, a web interface was developed with the best model proposed in this study. Through the training of 4 Stacking based on 7 independent machine learning algorithms, for which the oversampling method was used, where we were able to achieve good results.

To measure the performance of the models, 5 evaluation metrics (Accuracy, Sensitivity, FQ1-Score, Precision and ROC Curve) were used, where Stacking 1A (Logistic regression) with Oversampling achieved the best value for all metrics, while Stacking 2A (Random Forest) with Oversampling and Random Forest (Independent) also achieved the best ROC Curve values. It should be noted that the Cross validation method was used with 10 repetitions to evaluate the robustness of the methods and the Python programming language (in obtaining the statistics of the metrics).

For future work, it is recommended to train the stacking models using other independent algorithms with different dataset sizes, to deepen the study to detect diabetes. 4 Stacking models have been developed that can be used in future work. At the same time, the proposed intelligent system can be used, serving as a basis with another amount of data and other types of diseases. Finally, it is suggested to test other set-related approaches.

#### Data available

All the data and the python coding produced through this investigation can be made available upon request to the journal.

#### CRediT authorship contribution statement

**Alfredo Daza:** Writing – review & editing, Writing – original draft, Methodology, Conceptualization. **Carlos Fidel Ponce Sánchez:** Writing – original draft. **Gonzalo Apaza-Perez:** Writing – review & editing, Methodology, Investigation. **Juan Pinto:** Formal analysis. **Karoline Zavaleta Ramos:** Writing – review & editing, Formal analysis.

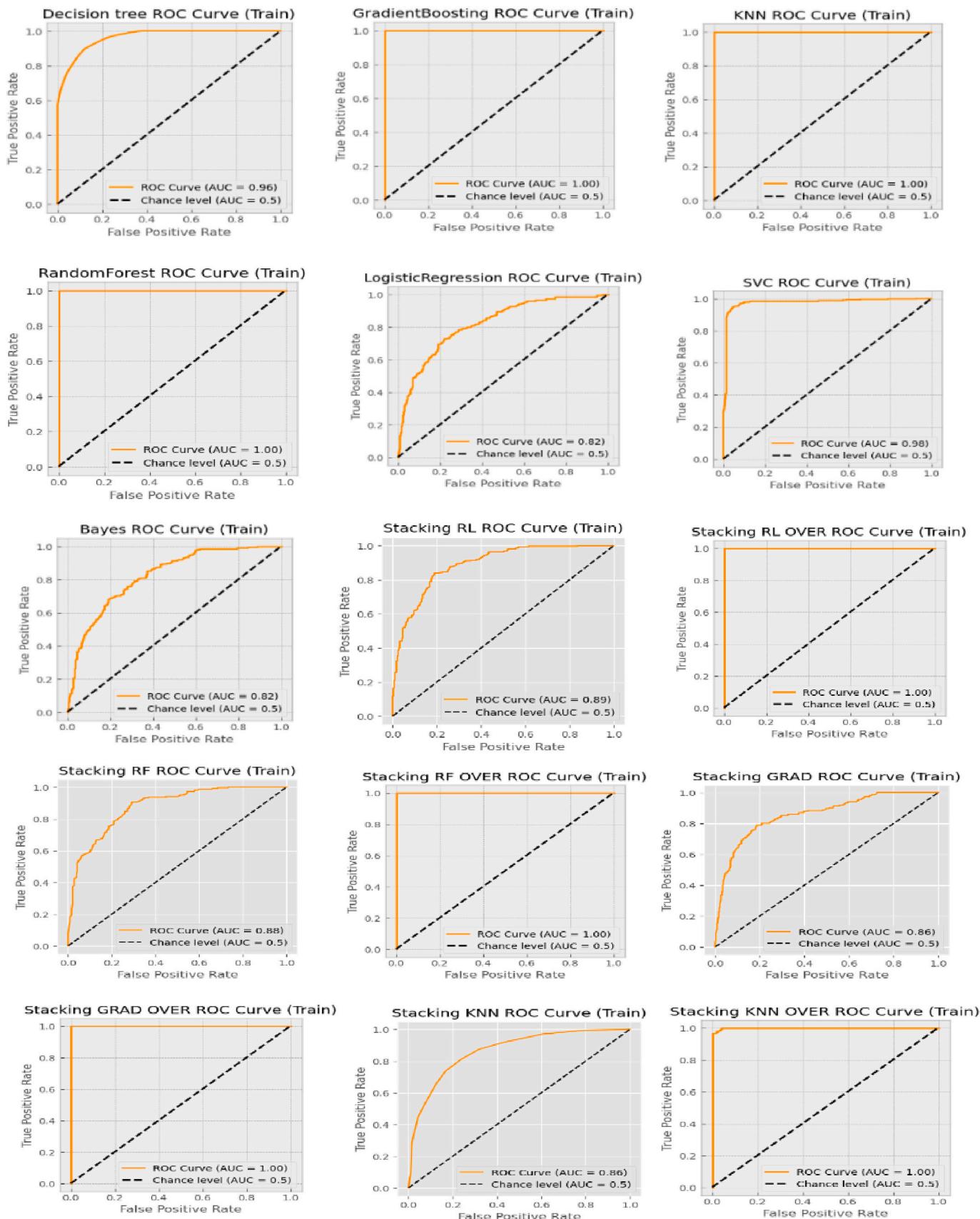
#### Declaration of competing interest

The authors declare that they have no conflicts of interest to report regarding the present study.

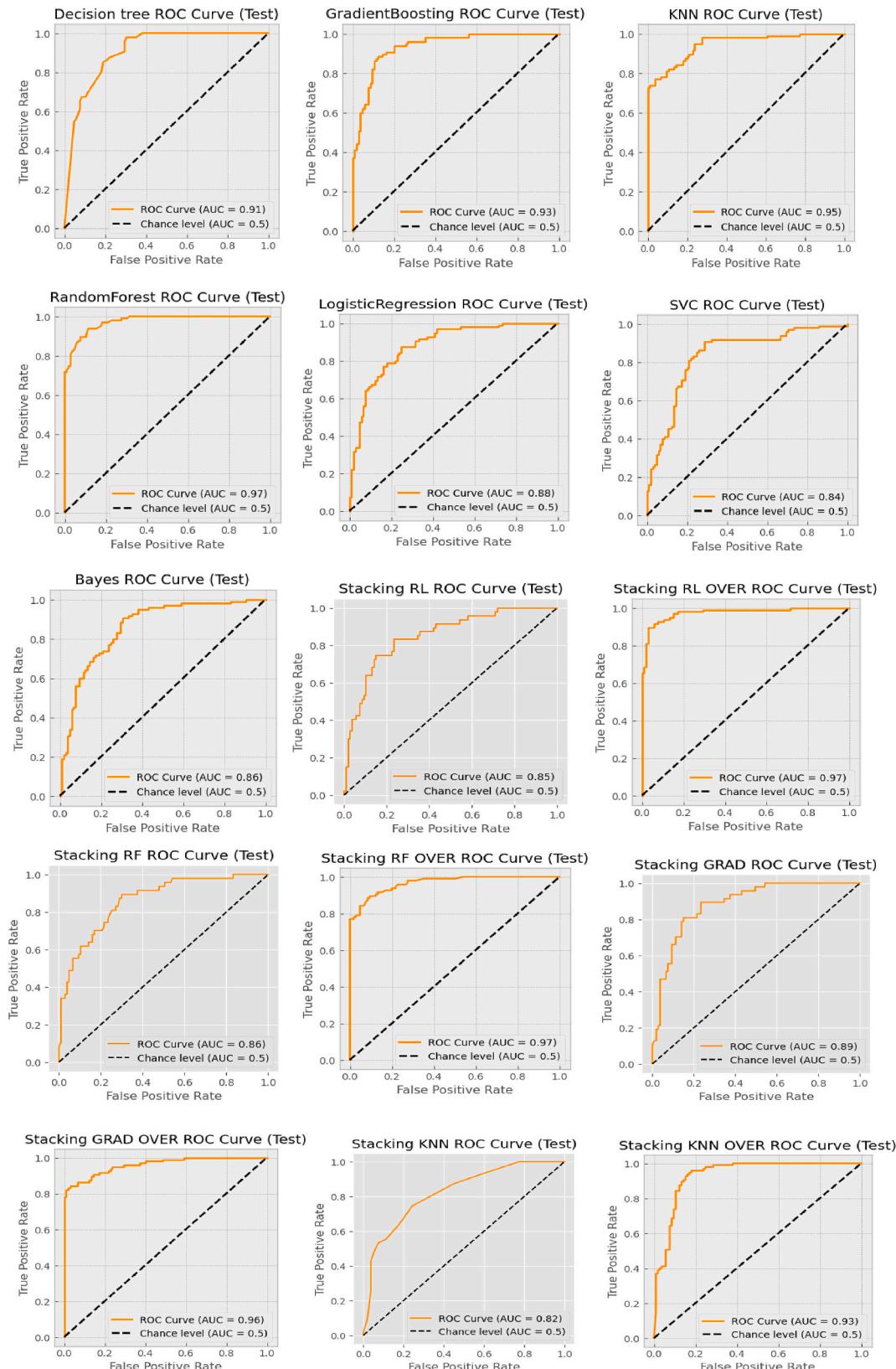
#### Acknowledgments

The authors gratitude to the Universidad César Vallejo for providing the opportunity to conduct this study and for their financial support towards this paper.

## Appendix A

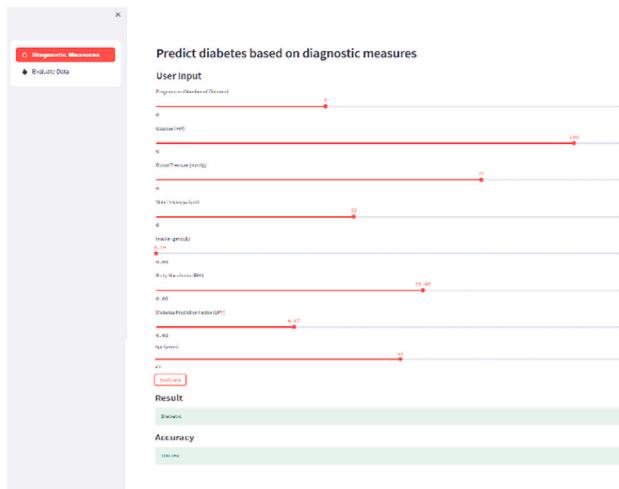


## Appendix B

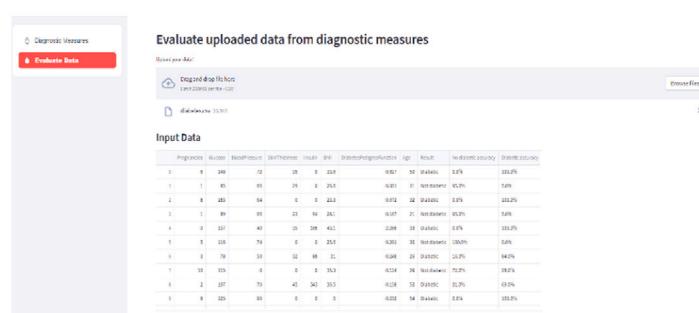


## Appendix C

A)



B)



## References

- [1] Joshi RD, Dhakal CK. Predicting type 2 diabetes using logistic regression and machine learning approaches. *Int J Environ Res Publ Health* 2021;18(14):7346. <https://doi.org/10.3390/ijerph18147346>.
- [2] Xu G, Liu B, Sun Y, Du Y, Snetselaar LG, Hu FB, et al. Prevalence of diagnosed type 1 and type 2 diabetes among US adults in 2016 and 2017: population based study. *BMJ* 2018;362:k1497. <https://doi.org/10.1136/bmj.k1497>.
- [3] Saeedi P, Petersohn I, Salpea P, Malanda B, Karuranga S, Unwin N, et al. Global and regional diabetes prevalence estimates for 2019 and projections for 2030 and 2045: results from the international diabetes federation diabetes atlas. *Diabetes Res Clin Pract* 2019;157:107843. <https://doi.org/10.1016/j.diabres.2019.107843>.
- [4] International Diabetes Federation. IDF diabetes atlas ninth edition. 2019. <https://diabetesatlas.org/atlas/ninth-edition/>. [Accessed 10 August 2023]. 2019.
- [5] Najafpour H, Farjam M, Sanjari M, Amirzadeh R, Shadkam M, Mirzazadeh A. Prevalence and incidence rate of diabetes, pre-diabetes, uncontrolled diabetes, and their predictors in the adult population in southeastern Iran: findings from KERCADR Study. *Front Public Health* 2021;9:611652. <https://doi.org/10.3389/fpubh.2021.611652>.
- [6] Sneha N, Gangil T. Analysis of diabetes mellitus for early prediction using optimal features selection. *J Big Data* 2019;6(1):1–19. <https://doi.org/10.1186/s40537-019-0175-6>.
- [7] World Health Organization. Global report on diabetes. 2016. [https://apps.who.int/iris/bitstream/handle/10665/204871/9789241565257\\_eng.pdf](https://apps.who.int/iris/bitstream/handle/10665/204871/9789241565257_eng.pdf). [Accessed 10 August 2023].
- [8] Mekashaw E, Kahissay MH, Workneh BD. Patients' perceptions, associations, and justifications for the causes of diabetes in North-East Ethiopia: a qualitative study. *Diabetes Metabol Syndr* 2022;16(5):102502. <https://doi.org/10.1016/j.dsx.2022.102502>.
- [9] Krishnamoorthi R, Joshi S, Almarzouki HZ, Shukla PK, Rizwan A, Kalpana C, et al. A novel diabetes healthcare disease prediction framework using machine learning techniques. *J Healthc Eng* 2022;1:1–11. <https://doi.org/10.1155/2022/1684017>.
- [10] Chatterjee S, Khunti K, Davies MJ. Type 2 diabetes. *Lancet* 2017;389(10085):2239–51. [https://doi.org/10.1016/S0140-6736\(17\)30058-2](https://doi.org/10.1016/S0140-6736(17)30058-2).
- [11] Kopitar L, Kocbek P, Cilar L, Sheikh A, Stiglic G. Early detection of type 2 diabetes mellitus using machine learning-based prediction models. *Sci Rep* 2020;10(1):11981. <https://doi.org/10.1038/s41598-020-68771-z>.
- [12] Sortsø C, Komkova A, Sandbæk A, Griffin SJ, Emneus M, Lauritzen T, et al. Effect of screening for type 2 diabetes on healthcare costs: a register-based study among 139,075 individuals diagnosed with diabetes in Denmark between 2001 and 2009. *Diabetologia* 2018;61(6):1306–14. <https://doi.org/10.1007/s00125-018-4594-2>.
- [13] Chaki J, Ganesh ST, Cidham SK, Theeran SA. Machine learning and artificial intelligence based diabetes mellitus detection and self-management: a systematic review. *J King Saud Univ Comput Inf Sci* 2020;34(6):3204–25. <https://doi.org/10.1016/j.jksuci.2020.06.013>.
- [14] Daza A, Guerra C, Cervera N, Burgos E. Predicting academic performance through data mining: a systematic literature. *TEM J* 2022;11(2):939–49. <https://doi.org/10.18421/TEM112-57>.
- [15] Daza A, Guerra C, Cervera N, Burgos E. Predicting academic performance using a multiclassification model: case study. *Int J Adv Comput Sci Appl* 2022;13(9):1–9. <https://doi.org/10.14569/IJACSA.2022.01309102>.
- [16] Vijayalakshmi V, Venkatachalapathy K. Comparison of predicting student's performance using machine learning algorithms. *Int J Intell Syst Appl* 2019;11(12):34. <https://doi.org/10.5815/ijisa.2019.12.04>.
- [17] Olabanjo OA, Wusu AS, Manuel M. A machine learning prediction of academic performance of secondary school students using radial basis function neural network. *Trends Neurosci Educ* 2022;100190. <https://doi.org/10.1016/j.tine.2022.100190>.
- [18] Waheed H, Hassan SU, Aljohani NR, Hardman J, Alelyani S, Nawaz R. Predicting academic performance of students from VLE big data using deep learning models. *Comput Hum Behav* 2020;104:106189. <https://doi.org/10.1016/j.chb.2019.106189>.
- [19] Aslam N, Khan I, Alamri L, Almuslim R. An improved early student's academic performance prediction using deep learning. *Int J Emerg Technol Learn* 2021;16(12):108–22. <https://doi.org/10.3991/ijet.v16i12.20699>.
- [20] Patel MM, Tanwar S, Gupta R, Kumar N. A deep learning-based cryptocurrency price prediction scheme for financial institutions. *J Inf Secur Appl* 2020;55:102583. <https://doi.org/10.1016/j.jisa.2020.102583>.
- [21] Ma X, Lv S. Financial credit risk prediction in internet finance driven by machine learning. *Neural Comput Appl* 2019;31:8359–67. <https://doi.org/10.1007/s00521-018-3963-6>.
- [22] Nikou M, Mansourfar G, Bagherzadeh J. Stock price prediction using DEEP learning algorithm and its comparison with machine learning algorithms. *Intell Syst Account Finance Manag* 2019;26(4):164–74. <https://doi.org/10.1002/isaf.1459>.
- [23] Kashyap AA, Raviraj S, Devarakonda A, Nayak K, Kv S, Bhat SJ. Traffic flow prediction models-A review of deep learning techniques. *Cogent Eng* 2022;9(1):2010510. <https://doi.org/10.1080/23311916.2021.2010510>.
- [24] Almeida RO, Munis RA, Camargo DA, Da Silva T, Sasso VA, Simões D. Prediction of road transport of wood in Uruguay: approach with machine learning. *Forests* 2022;13(10):1737. <https://doi.org/10.3390/f13101737>.
- [25] Servos N, Liu X, Teucke M, Freitag M. Travel time prediction in a multimodal freight transport relation using machine learning algorithms. *Logistics* 2019;4(1):1. <https://doi.org/10.3390/logistics4010001>.
- [26] Pamula T, Pamula D. Prediction of electric buses energy consumption from trip parameters using deep learning. *Energies* 2022;15(5):1747. <https://doi.org/10.3390/en15051747>.
- [27] Daza A, Bobadilla J, Apaza O, Pinto J. Stacking ensemble learning model for predict anxiety level in university students using balancing methods. *Inform Med Unlocked* 2023;101340. <https://doi.org/10.1016/j.imu.2023.101340>.
- [28] Hershey M, Burris HH, Cereceda D, Nataraj C. Predicting the risk of spontaneous premature births using clinical data and machine learning. *Inform Med Unlocked* 2022;32:101053. <https://doi.org/10.1016/j.imu.2022.101053>.
- [29] Andjelkovic J, Ljubic B, Hai AA, Stanojevic M, Pavlovski M, Diaz W, et al. Sequential machine learning in prediction of common cancers. *Inform Med Unlocked* 2022;30:100928. <https://doi.org/10.1016/j.imu.2022.100928>.

- [30] Elujide I, Fashoto SG, Fashoto B, Mbunge E, Folorunso SO, Olamijuwon JO. Application of deep and machine learning techniques for multi-label classification performance on psychotic disorder diseases. *Inform Med Unlocked* 2021;23:100545. <https://doi.org/10.1016/j imu.2021.100545>.
- [31] Daza A, Herrera JC, Bobadilla J, López AR, Ponce CF. Predicting the depression in university students using stacking ensemble techniques over oversampling method. *Inform Med Unlocked* 2023;101295. <https://doi.org/10.1016/j imu.2023.101295>.
- [32] Kumari SK, Mathana JM. Blood sugar level indication through chewing and swallowing from acoustic MEMS sensor and deep learning algorithm for diabetic management. *J Med Syst* 2019;43:1–9. <https://doi.org/10.1007/s10916-018-1115-2>.
- [33] Somasundaram SK, Alli P. A machine learning ensemble classifier for early prediction of diabetic retinopathy. *J Med Syst* 2017;41:1–12. <https://doi.org/10.1007/s10916-017-0853-x>.
- [34] Nicolucci A, Romeo L, Bernardini M, Vespaiani M, Rossi MC, Petrelli M, et al. Prediction of complications of type 2 Diabetes: a Machine learning approach. *Diabetes Res Clin Pract* 2022;190:110013. <https://doi.org/10.1016/j.diabres.2022.110013>.
- [35] Gupta SC, Goel N. Performance enhancement of diabetes prediction by finding optimum K for KNN classifier with feature selection method. In: Proceedings of the 3rd international conference on smart systems and inventive technology; 2020 aug. 20-22; India. Tirunelveli. Electrical and Electronics Engineers Inc; 2020.
- [36] Roy K, Ahmad M, Waqar K, Priyaaah K, Nebhen J, Alshamrani SS, et al. An enhanced machine learning framework for type 2 diabetes classification using imbalanced data with missing values. *Complexity* 2021;1:1–21. <https://doi.org/10.1155/2021/9953314>.
- [37] Ramesh J, Aburukba R, Sagahyoon A. A remote healthcare monitoring framework for diabetes prediction using machine learning. *Healthcare Technol Lett* 2021;8:45–57. <https://doi.org/10.1049/htl2.12010>.
- [38] Singh N, Singh P. Stacking-based multi-objective evolutionary ensemble framework for prediction of diabetes mellitus. *Biocybern Biomed Eng* 2020;40(1):1–22. <https://doi.org/10.1016/j.bbe.2019.10.001>.
- [39] Kumari S, Kumar D, Mittal M. An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier. *Int J Cogn Comput Eng* 2021;2:40–6. <https://doi.org/10.1016/j.jccep.2021.01.001>.
- [40] Rajendra P, Latifi S. Prediction of diabetes using logistic regression and ensemble techniques. *Comput Methods Programs Biomed Update* 2021;1:100032. <https://doi.org/10.1016/j.cmpbup.2021.100032>.
- [41] Xiong X, Zhang R, Bi Y, Zhou W, Yu Y, Zhu D. Machine learning models in type 2 diabetes risk prediction: results from a cross-sectional retrospective study in Chinese adults. *Curr Med Sci* 2019;39(4):582–8. <https://doi.org/10.1007/s11596-019-2077-4>.
- [42] Ahmad HF, Mukhtar H, Alaqaile H, Seliaman M, Alhumam A. Investigating health-related features and their impact on the prediction of diabetes using machine learning. *Appl Sci* 2021;11(3):1173. <https://doi.org/10.3390/app11031173>.
- [43] Kaggle. Diabetes. Dataset. <https://www.kaggle.com/datasets/mathchi/diabetes-data-set>. [Accessed 10 August 2023].
- [44] Gnip P, Vokorokos L, Drotár P. Selective oversampling approach for strongly imbalanced data. *PeerJ Comput Sci* 2021;7:e604. <https://doi.org/10.7717/peerj.cs.604>.
- [45] Rahim MA, Hossain MA, Hossain MN, Shin J, Yun KS. Stacked ensemble-based type-2 diabetes prediction using machine learning techniques. *Ann Emerg Technol Comput* 2021;7(1):30–9. <https://doi.org/10.33166/AETIC.2023.01.003>.
- [46] Li Y, Zhang Y, Cai Y. A new hyperparameter optimization method for power load forecast based on recurrent neural networks. *Algorithms* 2021;14:163. <https://doi.org/10.3390/a14060163>.
- [47] Maleki F, Muthukrishnan N, Ovens K, Reinhold C, Forghani R. Machine learning algorithm validation: from essentials to advanced applications and implications for regulatory certification and deployment. *Neuroimage Clin* 2020;30(4):433–45. <https://doi.org/10.1016/j.nic.2020.08.004>.
- [48] Brownlee J. K-nearest neighbors for machine learning. 2020. <https://machinelearningmastery.com/k-nearest-neighbors-for-machine-learning/>. [Accessed 10 August 2023].
- [49] Ahmed N, Ahammed R, Islam MM, Uddin MA, Akther A, Talukder MA, et al. Machine learning based diabetes prediction and development of smart web application. *Int J Cogn Comput Eng* 2021;2:229–41. <https://doi.org/10.1016/j.jccep.2021.12.001>.
- [50] Chang V, Ganatra MA, Hall K, Golightly L, Xu QA. An assessment of machine learning models and algorithms for early prediction and diagnosis of diabetes using health indicators. *Health Care Anal* 2022;2:100118. <https://doi.org/10.1016/j.jhealth.2022.100118>.
- [51] Gollapalli M, Alansari A, Alkhorasani H, Alsabai M, Sakloua R, Alzahrani R, et al. A novel stacking ensemble for detecting three types of diabetes mellitus using a Saudi Arabian dataset: pre-diabetes, T1DM, and T2DM. *Comput Biol Med* 2022;147:105757. <https://doi.org/10.1016/j.compbioemed.2022.105757>.
- [52] Afsanah E, Sharifidini A, Ghazzabi H, Ghobadi MZ. Recent applications of machine learning and deep learning models in the prediction, diagnosis, and management of diabetes: a comprehensive review. *Diabetol Metab Syndr* 2022;14(1):1–39. <https://doi.org/10.1186/s13098-022-00969-9>.
- [53] Yong S. Support vector machine classification. first ed. Switzerland: Springer; 2022.
- [54] Kalagotla SK, Gangashetty SV, Giridhar K. A novel stacking technique for prediction of diabetes. *Comput Biol Med* 2021;135:104554. <https://doi.org/10.1016/j.compbioemed.2021.104554>.
- [55] Liu R, Zhan Y, Liu X, Zhang Y, Gui L, Qu Y, et al. Stacking ensemble method for gestational diabetes mellitus prediction in Chinese pregnant women: a prospective cohort study. *J Healthc Eng* 2022;1–14. <https://doi.org/10.1155/2022/8948082>.
- [56] Wang G, Hao J, Ma J, Jiang H. A comparative assessment of ensemble learning for credit scoring. *Expert Syst Appl* 2011;38(1):223–30. <https://doi.org/10.1016/j.eswa.2010.06.048>.
- [57] Alshammari R, Atiyah N, Daghistani T, Alshammari A. Improving accuracy for diabetes mellitus prediction by using deepnet. *Online J Public Health Inform* 2020;12(1):1–12. 10.5210%2Fojphi.v12i1.10611.
- [58] Agrawal H, Jain P, Joshi AM. Machine learning models for non-invasive glucose measurement: towards diabetes management in smart healthcare. *Health Technol* 2022;12(5):955–70. <https://doi.org/10.1007/s12553-022-00690-7>.
- [59] Singh N, Singh P. A stacked generalization approach for diagnosis and prediction of type 2 diabetes mellitus. Computational intelligence in data mining: proceedings of the international conference on ICCIDM. India: Springer; 2020. Feb.- July 2018.
- [60] Liu Y, Ye S, Xiao X, Sun C, Wang G, Wang G, et al. Machine learning for tuning, selection, and ensemble of multiple risk scores for predicting type 2 diabetes. *Risk Manag Healthc Pol* 2019;189–98. <https://doi.org/10.2147/RMHP.S225762>.
- [61] El-Sappagh S, Elmogy M, Ali F, Abuhamd T, Islam SR, Kwak KS. A comprehensive medical decision-support framework based on a heterogeneous ensemble classifier for diabetes prediction. *Electronics* 2019;8(6):635. <https://doi.org/10.3390/electronics8060635>.
- [62] Morgan JA, Galván CE, Cruz M, Galván JI, Gamboa H, Arceo JG, et al. Hard voting ensemble approach for the detection of type 2 diabetes in mexican population with non-glucose related features. *Healthcare* 2022;10(8):1362. <https://doi.org/10.3390/healthcare10081362>.
- [63] Kibria HB, Nahiduzzaman M, Goni MO, Ahsan M, Haider J. An ensemble approach for the prediction of diabetes mellitus using a soft voting classifier with an explainable AI. *Sensors* 2022;22(19):7268. <https://doi.org/10.3390/s22197268>.