




ARTICLE

Diabetes diagnosis based on hard and soft voting classifiers combining statistical learning models

 Gustavo Peixoto de Oliveira,*  Anderson Fonsêca,* and  Paulo Canas Rodrigues*

Department of Statistics, Federal University of Bahia, Salvador, Brazil

*Corresponding author. Email: gugapeixoto@live.com; andersonof@ufba.br; paulocanas@gmail.com

(Received: June 1,2022; Revised: December 14,2022; Accepted: December 14,2022; Published: December 31,2022)

Abstract

Diabetes mellitus is one of the deadliest incurable diseases globally, and its cases continue upward. The identification of the disease in an early way helps fight it; however, blood tests can be considered invasive, discouraging its accomplishment. In this vein, this work aims to build a model as an alternative to traditional exams to identify the disease. Statistical learning algorithms such as logistic regression, K-nearest neighbors, decision trees, random forest, and support vector machines were used for diabetes classification. These models were considered separately and combined via hard and soft voting classifiers. The methods were applied to a widely known dataset of 768 individuals and nine variables, compared using several accuracy metrics based on the confusion matrix, and used to estimate the probability of diabetes for a given profile.

Keywords: Diabetes mellitus; Statistical learning; Random forest; Support vector machines; K-nearest neighbors; Hard and soft voting classifiers.

1. Introduction

As defined by the World Health Organization, diabetes mellitus, commonly known as diabetes, is a chronic metabolic disease characterized by high blood glucose levels, which over time, can lead to severe damage to the heart, veins, eyes, kidneys, and nerves. Diabetes is divided into Type 1, Type 2, and gestational. There is a great deal of attention around the globe regarding Type 2 diabetes, as it comprises 90% of cases, and it has been increasing in the last three decades in countries of all income levels and is linked to unhealthy habits, as reported by the World Health Organization and by the International Diabetes Federation (International Diabetes Federation, 2019; World Health Organization, 2020).

Among the incurable diseases, diabetes is one of the deadliest, with 1.5 million deaths attributed to the disease every year, making it the ninth leading cause of death in 2019 (World Health Organization, 2020). The number of cases has been growing consistently in recent decades; however, there is a global agreement to reverse this trend by 2025, and technology can play a key role in this mission (World Health Organization, 2021).

The American Diabetes Association recommends that any person with disease symptoms, like weight loss, frequent urination, or excessive thirst, should be tested for the disease. People older than 45 should also be tested once every three years even if they don't have any symptoms as referenced in endocrineweb.com. With cases on the rise, the global diabetes diagnostics market is expected to reach 42.4 billion dollars by 2026. The demand for diabetes diagnostics is being driven, among many other things, by developing non-invasive techniques like the one we present in this paper and others such as non-invasive wearable glucose testing devices as referenced in prnewswire.com. Shang *et al.*, 2022 compared some non-invasive testing techniques with invasive monitors and noticed that the non-invasive techniques were generally less accurate but less painful and produced less biological waste. *ibid.* acknowledged the barriers but predicted success for non-invasive techniques shortly.

Many computer models can be an alternative to help identify people with diabetes based on specific covariates. Aiming at proposing and/or applying models to identify and classify individuals with/without diabetes, several studies can be found in the literature, with some of them applied to the data that we considered in this paper. Kumari *et al.*, 2021 used a soft voting classifier algorithm but only achieved an accuracy of 79.1%, while Hina *et al.*, 2017 used more complex algorithms, such as the multi-layer perception achieving an accuracy of 81.8%, being this the only metric they used, which can be problematic, especially in unbalanced data, as is the case. Sisodia & Sisodia, 2018 also proposed to classify diabetes cases as positive or not using different machine learning models, and their best-performing model was the Naive Bayes, achieving an accuracy of 76.3%. Ara *et al.*, 2018 used Naive Bayes and Logistic Regression to predict the outcome variable, obtaining accuracy of 75% and 77%, respectively. Using a different diabetes dataset, Bressan *et al.*, 2020 managed to classify 73.7% of the patients in the intervention group correctly, as well as 69.2% of the patients in the control group.

Several papers use neural network algorithms to predict diabetes, including Jeatrakul *et al.*, 2010, which achieved an accuracy of 76.6% using an artificial neural network. Ayon & Islam, 2019 obtained a surprisingly high accuracy of 98% as well as a sensitivity of 98.8% and specificity of 96.6%; however, we could not locate enough details to allow for the reproducibility of the results.

In this paper, we will use statistical and machine learning models to classify the presence/absence of diabetes. In particular, we will use the Logistic Regression (Cox, 1958), an algorithm that works based on the logistic function; the K-nearest neighbors (Silverman & Jones, 1989), which classifies new observations based on the nearest neighbors; the Decision tree (Breiman *et al.*, 1984), an algorithm with the style of a flowchart that can classify new observations by splitting the "nodes" based on conditions; the Random Forest (Breiman, 2001), a collection of decision trees that uses random samples from the dataset to obtain a lower correlation between the trees; and the support vector machines (Cortes & Vapnik, 1995), that uses mapping functions or hyperplanes to classify new observations. Moreover, we will also consider the hard and soft methods of a voting classifier to combine the results of the considered models and use them to estimate the probability of diabetes for a given profile. A more in-depth description of these models will be made, and all of them will be evaluated through the k-fold cross-validation procedure, which consists of dividing the dataset into K parts, and then, iteratively, using some of them to learn the model, while the others are explored in the evaluation of its performance (Anguita *et al.*, 2012). All models were implemented in python using the Scikit-learn (Pedregosa *et al.*, 2011) library and applied to a widely known dataset of 768 individuals and nine variables (Smith *et al.*, 1988).

To evaluate the model performance, several accuracy metrics will be used: accuracy, specificity,

sensitivity, positive predictive value (precision), negative predictive value, and F1 score. The use of multiple metrics has the advantage of avoiding overfitting when the model performs well only on training data. In contrast to other studies, other metrics will be used in addition to the accuracy because, in the field of disease prediction/classification, a balance is always sought in predicting false positives and false negatives, measured through sensitivity and specificity, respectively. In addition, as it is a database in which 65% of patients do not have diabetes, therefore unbalanced, metrics such as precision and F1 score may stand out in terms of accuracy. In this way, the results will provide a broad, robust approach with superior performance compared to other articles on the same field mentioned above (Hina *et al.*, 2017; Jeatrakul *et al.*, 2010; Kumari *et al.*, 2021; Sisodia & Sisodia, 2018).

This paper is organized as follows. Section 2 describes the data, the methods, and the accuracy measures. Section 3 provides the results, including the initial variable selection, a descriptive analysis, the model performance, and the discussion of the results. The paper ends with some concluding remarks in Section 4.

2. Materials and methods

This section will describe the data set and explain the methodology used in this paper. Figure 1 provides a summary of the methodology, which consists of (i) selecting the data; (ii) pre-processing by, e.g., removing inconsistent observations, standardizing the variables; (iii) selecting the most important variables for the models; (iv) modeling, i.e., applying the selected models with the adjusted parameters to the training data; (v) combining the models using the hard and soft voting classifiers and apply them to the training data; and (vi) use the validation set to assess the performance of the models in (iv) and (v), throughout the accuracy measures.

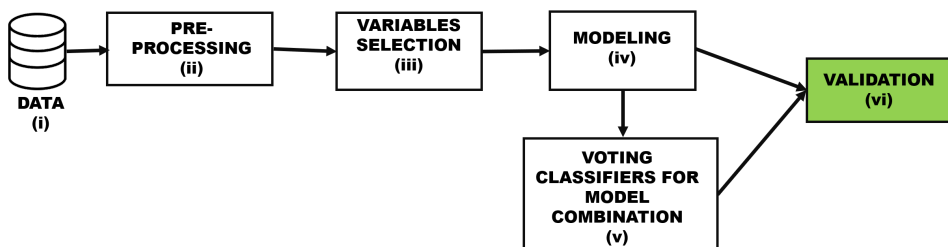


Figure 1. Flowchart describing the methodology used in this paper.

2.1 Data set

The original database obtained from the National Institute of Diabetes and Digestive and Kidney Diseases survey has 768 observations and nine variables with information related to the health of North American indigenous women of the Pima ethnicity (Smith *et al.*, 1988). During data pre-processing, the 11 observations with a zero body mass index (BMI) were treated as some kind of writing/measurement error and removed. In this way, we consider a data set with 757 individuals and nine variables: Age: years; blood pressure: diastolic blood pressure (mm Hg), BMI: body mass index (weight in $kg/(height\ in\ m)^2$); diabetes pedigree: diabetes pedigree function, glucose: plasma glucose concentration 2 hours after an oral glucose tolerance test (mg/dL); Insulin: 2-hour serum insulin ($\mu U/ml$); pregnancies: number of pregnancies; skin thickness: triceps skin fold thickness (mm); and outcome: 0 if no diabetes and 1 if diabetic. Among the women in the study, 491 (64.86%) do not have diabetes, while 266 (35.14%) have the disease. Hence, it is considered

that a prediction model with an accuracy lower than 64.86% is highly unsatisfactory, as this would be the accuracy achieved if it pointed out that no observed woman has diabetes. In Figure 2 and 3, the main descriptive measures of the eight covariates are presented, where it can be observed that the greatest variability is that of insulin while the smallest is the BMI. Insulin has the greatest discrepancy between the median and the mean, in which the latter is more than double the former.

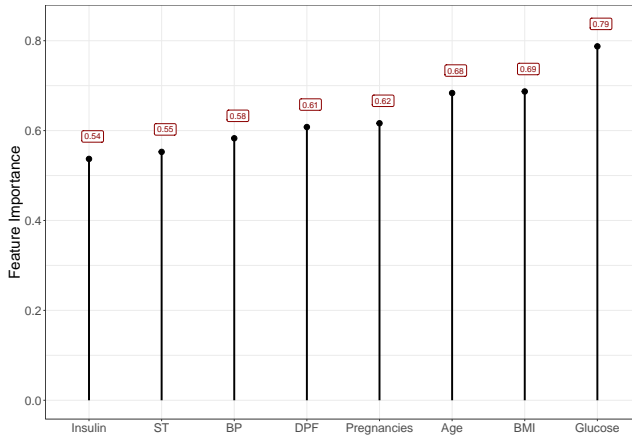


Figure 2. Feature importance.

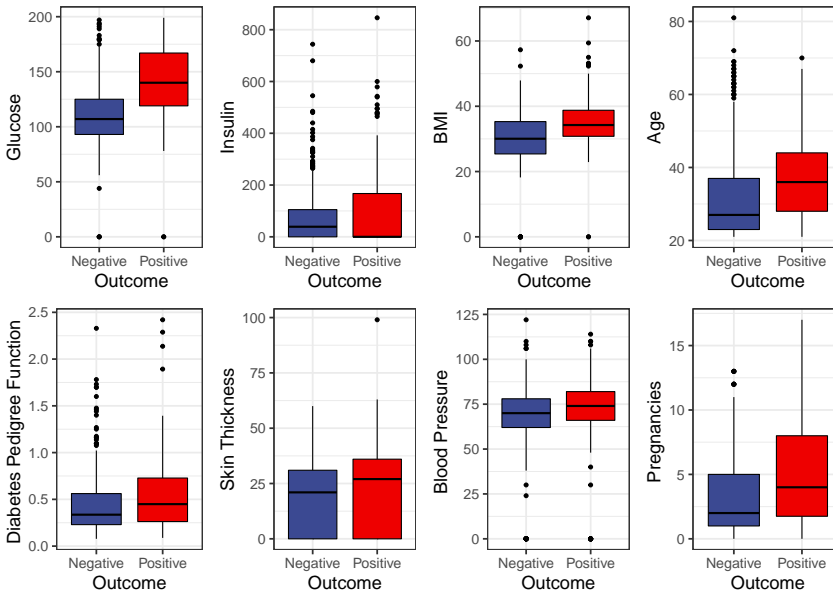


Figure 3. Boxplots of covariates.

2.2 Models

In this subsection, we will present a brief description of the statistical and machine learning models used in this paper.

2.2.1 Logistic regression

Logistic regression is a model that works based on the logistic function, which transforms any value into a number in the range between zero and one. As our variable of interest is categorical and binary, it allows the use of a regression model to calculate the probability of a specific event. Unlike simple linear regression, in binary logistic regression, the output variable is on the nominal scale, and the error has a Bernoulli distribution with zero mean and variance $\pi(x)(1 - \pi(x))$, with $\pi(x)$ the probability of success. In addition, this model does not accept multicollinearity; that is, the independent variables cannot have a strong correlation with each other (Cox, 1958). The posterior probabilities of the K classes ($K = 2$ for binary response variables) in the logistic regression models are obtained via linear functions of x , while ensuring that they sum one, and, for $K = 2$, have the form:

$$\log \left(\frac{\pi(x)}{1 - \pi(x)} \right) = \beta_0 + \beta_1^T x_1 \quad (1)$$

where β_0 is the intercept from the linear regression equation and $\beta_1^T x_1$ is the transposed vector of regression coefficients multiplied by the value of the predictor.

These models can be fitted by maximum likelihood, using the conditional probability of G given X , with its log-likelihood for N observations given by:

$$l(\theta) = \sum_{i=1}^N \log p_{g_i}(x_i; \theta) \quad (2)$$

where $p_k(x_i; \theta) = Pr(G = k | X = x_i; \theta)$ (Hastie *et al.*, 2009). When t independent covariates $X_1 \dots X_t$, are available to model the response variable, the multiple logistic regression model can be written as:

$$\log \left(\frac{\pi(x)}{1 - \pi(x)} \right) = \beta_0 + \beta_1^T x_1 + \dots + \beta_t x_t. \quad (3)$$

where $\beta_t x_t$ is the vector of the t -th regression coefficients multiplied by the value of the t -th predictor.

2.2.2 *k*-nearest neighbors (KNN)

The K -Nearest Neighbors is an algorithm that is based on the idea that observations close to new data that we seek to classify provide important information. Classification by this method usually involves dividing the sample into test and training categories, and it is based on the Euclidean distance between a new observation and specified training samples. During the training process, the real class of each sample is used, while in the test process, it is sought to predict the real classes of each sample. In this model, each value of k chosen by the researcher produces a "neighborhood" and a different final result (Silverman & Jones, 1989). In practice, given an observation in the test data, x_0 , we find the k training observations closest, in Euclidean distance, to x_0 and classify x_0 using the majority vote among its k neighbors.

2.2.3 Decision tree

A decision tree is a flowchart-style algorithm that works from the binary slicing of attributes. The division starts at the so-called "root node", which represents the entire dataset; over time, it is divided into two or more homogeneous sets to find the best attribute until reaching the "leaf node," where it is no longer possible to split the dataset. This is a very commonly used algorithm due to the greater ease of interpretation and understanding when compared to other classification algorithms (Breiman *et al.*, 1984).

The classification of observations is done as follows. In a node m representing the region R_m with N_m observations, we have that:

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k) \quad (4)$$

is the proportion of observations of class k at node m . If the majority class at node m is k , all observations in the node are classified as belonging to that class (Hastie *et al.*, 2009).

2.2.4 *Random Forest*

Random forest (Breiman, 2001) is a collection of classification and regression trees trained on datasets of the same size as the training sets so that each tree depends on the value of an independent and identically distributed random vector. The algorithm, based on bagging (i.e., bootstrap aggregation), starts by selecting random samples from the dataset and then building a decision tree for each sample. As seen before, the different decision trees return the best attributes, after which the attribute with the most repeated appearances is chosen. The generalization error of a random forest depends on the strength of each individual tree and the correlation between them (Ho, 1995).

To classify new observations, the Random Forest model obtains a class vote for each tree, as explained in the previous subsection, and then classifies the new observation using the majority vote. So if $\hat{C}_b(x)$ is the class prediction of the b -th tree, $b = 1, \dots, B$, B the number of bootstrap samples, of the random forest, we have that the class obtained by the random forest algorithm $\hat{C}_{rf}^B(x)$ is given by majority vote among all B regression trees $\hat{C}_b(x)$, $b = 1, \dots, B$ (Hastie *et al.*, 2009).

2.2.5 *Support vector machines*

Support Vector Machines (SVM) is a classification methodology proposed by (Cortes & Vapnik, 1995), that seeks to build a hyperplane as a decision surface, in such a way that the separation between the classes is maximum. This is considering linearly separable patterns. For non-linearly separable patterns, we look for an appropriate mapping function to make the mapped set linearly separable. SVM stands out for at least two characteristics: it has a solid theoretical foundation and can achieve high performance in practical applications. Learning theory can precisely identify the factors that must be considered for successful learning and build models that are quite complex. More details can be found elsewhere, e.g., (Hastie *et al.*, 2009).

2.3 *Hard and soft voting classifier*

The combination of the results from two or more classification models can be done through a voting classifier. Here two voting classifiers are considered: the hard voting classifier, and the soft voting classifier. In the hard voting classifier method, the predicted class is the one that represents the majority according to the individual classification of each model. For example, if in a scenario with three models, two are classified as "A," and one is classified as "B", the predicted class will be "A" because it had the voting majority of the classifications.

In the soft voting classifier method, the averages of the probabilities of each class assigned by each model are used, returning the one with the highest average of the predicted probabilities. For example, in the situation in the previous paragraph, if the three models return the following probabilities for classes "A" and "B", respectively, (0.55, 0.52 and 0.4) and (0.45, 0.48 and 0.6), the soft voting classifier will classify the new observation in class B because it presented a higher average probability (0.51 against 0.49).

For both cases, it is possible to establish weights, that is, to increase the importance of one classification method to the detriment of the others. In the majoritarian approach, it would count n times, and in the probabilistic approach, a weighted average is calculated (Wolpert, 1992).

It is worth mentioning that this strategy of combining models increases the computational cost and, in situations where a large number of models is considered, and a large volume of data is used, this strategy can become time-consuming.

2.4 Accuracy measures

To compare and evaluate the classification accuracy of each method, several accuracy measures are used. To understand what each metric represents, we first introduce the concept of a confusion matrix. Table 1 shows a confusion matrix, in which we have true negative (TN) and true positive (TP) when the observation is correctly classified, and false negative (FN) and false positive (FP) when the wrong classification occurs.

Table 1. Confusion Matrix

Predicted Value	True Value	
	Y = 0	Y = 1
Y = 0	True Negative (TN)	False Negative (FN)
Y = 1	False Positive (FP)	True Positive (TP)

Based on the confusion matrix in Table 1, we can define several metrics that will be used to measure the classification accuracy of the considered models.

- Accuracy (A): is the number of correct predictions divided by the total number of predictions, that is,

$$A = \frac{TN + TP}{TN + TP + FN + FP} \tag{5}$$

informs the percentage of correctly classified diagnoses of diabetes;

- Specificity (E): is the rate of correctly identified negative cases, that is,

$$E = \frac{TN}{TN + FP} \tag{6}$$

informs the percentage of correctly classified negative diagnoses of diabetes;

- Sensitivity (S): is the rate of correctly identified positive cases, that is,

$$S = \frac{TP}{TP + FN} \tag{7}$$

informs the percentage of correctly classified positive diagnoses of diabetes;

- Positive predictive value (PPV): it is the rate of true positives in relation to all positive predictions, that is,

$$PPV = \frac{TP}{TP + FP} \tag{8}$$

informs the percentage of positive diagnoses of diabetes among all that the model classified as positive;

- Negative predictive value (NPV): it is the rate of true negatives in relation to all negative predictions, that is,

$$NPV = \frac{TN}{TN + FN} \quad (9)$$

informs the percentage of negative diagnoses of diabetes among all that the model classified as negative;

- F1 score (F1): is the harmonic mean between sensitivity and the positive predictive value, that is,

$$F1 = \frac{2}{\frac{1}{S} + \frac{1}{PPV}} \quad (10)$$

More details about these metrics can be found elsewhere, e.g., Izbicki & dos Santos, 2020.

3. Results and discussion

3.1 Variables selection

After verifying that there were no strong correlations between the independent variables, a selection of variables was performed using the method known as recursive elimination of variables, which is essentially a backward selection of the predictors. This technique begins by building a model on the entire set of predictors and computing an importance score for each predictor. The least important predictors are then removed, the model is rebuilt, and importance scores are computed again. The subset size that maximizes the accuracy is used to select the predictors based on the importance rankings (Kuhn & Johnson, 2019). This analysis was carried out in the software R, version 4.1.2, (R Core Team, 2022) using the package `Caret` (Kuhn *et al.*, 2021) and consists of evaluating the performance of the Random Forest model for all possible subsets of independent variables. We chose to pair the recursive elimination of variables with the Random Forest because it tends not to exclude variables from the prediction equation, and it has a well-known internal method for measuring feature importance (Kuhn & Johnson, 2019). From this analysis, it was verified that the maximum accuracy was obtained in the presence of four variables, which according to the `predictors` function of R are age (*age*), the number of pregnancies (*pregnancies*), BMI (*BMI*) and glucose level (*glucose*), which are the most important variables as shown in the last column of Table ?? . During the analysis, we verified that the performance of all algorithms improved by using these four features.

The features' importance in the last column of Table ?? were obtained with the use of `varImp` function from `Caret` package in R (Kuhn *et al.*, 2021). It was estimated by a Learning Vector Quantization Model, which is a prototype-based supervised classification algorithm proposed by Kohonen, 1995. This method aims to divide the data space into distinct regions and define a vector prototype for each region.

After data manipulation and organization, the mean and median of all eight features present in the dataset grouped into "diabetic" and "non-diabetic" were obtained (Table 2). Table 2 also features the values for the Mann-Whitney test statistics and p-values, whose alternative hypothesis is that the medians of the groups are different, being the test statistic the sum of the rankings of the two groups (Mann & Whitney, 1947; Wilcoxon, 1945). A non-parametric test was used because none of the features' observations followed a normal distribution. From the values of mean and median presented in the table, one can infer that there seems to be a significant difference between the two groups for most features. After conducting the hypotheses tests, we rejected the null hypothesis based on the p-values for the following features: Glucose, BMI, Age, Pregnancies, Blood Pressure, Skin Thickness, and Diabetes Pedigree Function, meaning that the median of the two groups is

different. For the feature Insulin, we failed to reject the null hypothesis if we consider a significance level of 0.05, meaning that the two groups have the same median. However, for a significance level above 0.07, the null hypothesis of equality between medians is rejected.

Table 2. Mean, median, Mann-Whitney test statistic and p-value of the features' observations grouped/compared into "diabetic" and "non-diabetic"

Features	Median		Statistic	P-value
	Non-diabetic	Diabetic		
Glucose	107.0	140.0	27762	0.00
BMI	30.1	34.3	40875	0.00
Age	27.0	36.0	41325	0.00
Pregnancies	2.0	4.0	50112	0.00
Blood Pressure	70.0	74.0	54477	0.00
Skin Thickness	21.0	27.0	58430	0.02
Insulin	42.0	0.0	60472	0.07
DPF	0.34	0.45	51207	0.00

3.2 Models Performance

After data preparation, the models described in the previous section were fitted to the data. For this, the database was divided into two parts, 80% for cross-validation (k-fold) and 20% for validation. After that, the value of 50 was chosen for k-fold; that is, the dataset was divided into 50 parts, using 49 (k - 1) to train the model and the remaining part to test it, repeating this process 50 times. As a result, the models gain greater generalization capacity and greater stability, avoiding overfit (Stone, 1974).

The models were adjusted, through optimization methods, so that they were as accurate as possible in the training set. In the sequence, they were selected for validation. The best parameters for this case are 29 neighbors for the KNN, a maximum depth of two for the decision tree, and a maximum depth of five for the random forest.

Table 3 shows the values of the metrics for the selected models calculated with the validation set, being the values between parentheses the best hyperparameters of the models, obtained via optimization. The random forest (5) yielded the best performance in accuracy (0.8224), sensitivity (0.6596), NPV (0.8545), and F1 score (0.6966); the SVM obtained the best performance in the specificity (0.9048) and PPV (0.7436), and the second best accuracy (0.8158). The lowest accuracy was obtained by the decision tree with a value of 0.7697.

Table 3. Evaluation metrics (accuracy, specificity, sensitivity, positive predictive value (PPV), negative predictive value (NPV), and F1 score, respectively) for the five models under consideration (random forest, support vector machines (SVM), k-nearest neighbor (KNN), logistic regression, and decision tree). The values between parentheses give the best hyperparameters of the models.

	Accuracy	Sensitivity	Specificity	PPV	NPV	F1 score
Random forest (5)	0.8224	0.6596	0.8952	0.7381	0.8545	0.6966
SVM	0.8158	0.6170	0.9048	0.7436	0.8407	0.6744
KNN (29)	0.8092	0.6383	0.8857	0.7143	0.8455	0.6742
Logistic regression	0.8026	0.6170	0.8857	0.7073	0.8378	0.6591
Decision tree (2)	0.7697	0.5319	0.8762	0.6579	0.8070	0.5882

The strategy adopted in this paper to improve the evaluation metrics was using a combination of models for classification, that is, combining them to obtain a classification result. The first two rows of Table 4 shows the performance using the hard voting (majority) and soft voting (probability) methods combining all models. Subsequently, an analysis was made to search for the best combination of models, and it was found that Random Forest, SVM, KNN together via soft voting with weights 2, 1, and 1, respectively, was the best alternative, being the results added to Table 4. Figure 4 shows a summary of this process, in which post-processing and variable selection is applied to models via soft voting with their weights. It is observed that this combination was the holder of the best result in all six metrics. It obtained 83.55% of the diagnoses right, with a rate of 65.96% for the positive cases, and 91.43% for the negative ones, a PPV of 77.50%, an NPV of 85.71 % (Table 4). The average between sensitivity and PPV is 0.7126 (F1 score). For this reason, it was determined as the chosen classifier. The confusion matrix for this classifier is presented in Table 5.

Table 4. Evaluation metrics (accuracy, specificity, sensitivity, positive predictive value (PPV), negative predictive value (NPV), and F1 score, respectively) for the hard voting (majority) and soft voting (probability) methods combining all models, and the soft voting considering random forest, SVM, KNN with weights 2, 1 and 1, respectively

	Accuracy	Sensitivity	Specificity	PPV	NPV	F1 score
Hard (all)	0.8224	0.6383	0.9048	0.7500	0.8482	0.6897
Soft (all)	0.8158	0.6170	0.9048	0.7436	0.8407	0.6744
Soft (RF, SVM, KNN)	0.8355	0.6596	0.9143	0.7750	0.8571	0.7126

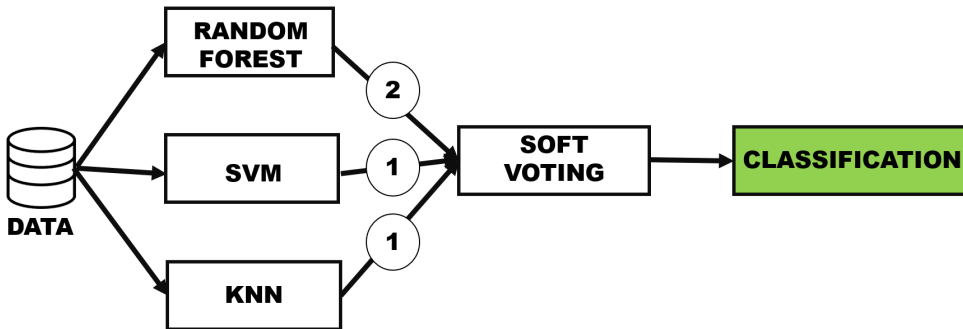


Figure 4. Algorithm of the best performing method for diabetes classification: soft voting combining random forest, SVM, and KNN, with weights 2, 1, and 1, respectively.

Table 5. Confusion Matrix for the best performing method for diabetes classification: soft voting combining random forest, SVM, and KNN, with weights 2, 1, and 1, respectively.

Predicted Value	True Value	
	Y = 0	Y = 1
Y = 0	96	9
Y = 1	16	31

The model resulting from the voting classifier, in addition to diabetes classification, computes the probability of such an event. In that way, it is possible to estimate the probability of given profiles. For example, a 33-year-old woman with a BMI of 32, a glucose of 121, and one pregnancy is 49.6% probability of being diabetic according to the model. A 24-year-old woman with a BMI of 27, a glucose of 91, and one pregnancy has a 4.5% probability of being diabetic. While a 41-year-old woman with a BMI of 36, a glucose of 141, and 6 times pregnant has a 64.3% probability.

The classification is based on these probabilities, if it is greater than 50%, it is considered diabetic. Having access to these probabilities, it is possible to modify the threshold to other values, such as 60%. This approach was applied, but no superior performance was obtained with it, so the default threshold (50%) was maintained.

4. Conclusions

The detection of diabetes is extremely important for humans because this disease causes many complications, and the sooner it is identified, the better it can be fought. Therefore, contributions to help improve the diagnosis are of great importance. As an alternative to traditional exams, computational algorithms can be used to help identify diabetes using a set of covariates.

In this paper, statistical learning algorithms such as logistic regression, K-nearest neighbors, decision trees, random forest, and support vector machines were used individually and combined using hard and soft voting classifiers, for diabetes classification. The results showed that in this case, using a soft voting classifier combining random forest, SVM, and KNN yielded the best performance in all metrics under consideration, indicating that this is the most suitable model for predicting diabetes in this database. Moreover, based on the chosen model, the probability of diabetes could be estimated for any given profile. The codes and data used in this work can be accessed at <https://github.com/Andersonof30/Diabetes-hard-soft-classifier>.

The methodology and strategies used in this paper are of great generality and can be applied and extended to other data sets in the same field of research, and also to other areas where the aim is to make a classification based on a set of covariates.

Acknowledgments

P.C. Rodrigues acknowledges financial support from the Brazilian National Council for Scientific and Technological (CNPq) grant “bolsa de produtividade PQ-2” 305852/2019-1.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Anguita, D., Ghelardoni, L., Ghio, A., Oneto, L. & Ridella, S. *The ‘K’in K-fold cross validation in 20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)* (2012), 441–446.
2. Ara, A., Louzada, F. & Milan, L. A. Classification binary models for biomedical data: simple probabilistic networks and logistic regression. *Brazilian Journal of Biometrics* **36**, 48–55 (2018).
3. Ayon, S. I. & Islam, M. M. Diabetes prediction: a deep learning approach. *International Journal of Information Engineering and Electronic Business* **12**, 21 (2019).
4. Breiman, L., Friedman, J., Stone, C. & Olshen, R. *Classification and Regression Trees: Taylor & Francis* 1984.
5. Breiman, L. Random forests. *Machine learning* **45**, 5–32 (2001).
6. Bressan, G. M., de Azevedo, B. C. F. & de Souza, R. M. Métodos de classificação automática para predição do perfil clínico de pacientes portadores do diabetes mellitus. *Brazilian Journal of Biometrics* **38**, 257–273 (2020).
7. Cortes, C. & Vapnik, V. Support-vector networks. *Machine learning* **20**, 273–297 (1995).

8. Cox, D. R. The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)* **20**, 215–232 (1958).
9. Hastie, T., Tibshirani, R., Friedman, J. H. & Friedman, J. H. *The elements of statistical learning: data mining, inference, and prediction* (Springer, 2009).
10. Hina, S., Shaikh, A. & Sattar, S. A. Analyzing diabetes datasets using data mining. *Journal of Basic and Applied Sciences* **13**, 466–471 (2017).
11. Ho, T. K. *Random decision forests in Proceedings of 3rd international conference on document analysis and recognition* **1** (1995), 278–282.
12. International Diabetes Federation. IDF diabetes atlas ninth. *Dunia: Idf* **9**, 5–9 (2019).
13. Izbicki, R. & dos Santos, T. M. *Aprendizado de máquina: uma abordagem estatística* (Rafael Izbicki, 2020).
14. Jeatrakul, P., Wong, K. W. & Fung, C. C. Data cleaning for classification using misclassification analysis. *Journal of Advanced Computational Intelligence and Intelligent Informatics* **14**, 297–302 (2010).
15. Kohonen, T. in *Self-organizing maps* 175–189 (Springer, 1995).
16. Kuhn, M. & Johnson, K. *Feature engineering and selection: A practical approach for predictive models* (CRC Press, 2019).
17. Kuhn, M., Wing, J., Weston, S., Williams, A., Keefer, C., Engelhardt, A., *et al.* *caret: classification and regression training*. 2020 2021.
18. Kumari, S., Kumar, D. & Mittal, M. An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier. *International Journal of Cognitive Computing in Engineering* **2**, 40–46 (2021).
19. Mann, H. B. & Whitney, D. R. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, 50–60 (1947).
20. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *Journal of machine learning research* **12**, 2825–2830 (2011).
21. R Core Team. *R: A Language and Environment for Statistical Computing* R Foundation for Statistical Computing (Vienna, Austria, 2022). <https://www.R-project.org/>.
22. Shang, T., Zhang, J. Y., Thomas, A., Arnold, M. A., Vetter, B. N., Heinemann, L. & Klonoff, D. C. Products for monitoring glucose levels in the human body with noninvasive optical, noninvasive fluid sampling, or minimally invasive technologies. *Journal of diabetes science and technology* **16**, 168–214 (2022).
23. Silverman, B. W. & Jones, M. C. E. fix and jl hodges (1951): An important contribution to nonparametric discriminant analysis and density estimation: Commentary on fix and hodges (1951). *International Statistical Review/Revue Internationale de Statistique*, 233–238 (1989).
24. Sisodia, D. & Sisodia, D. S. Prediction of diabetes using classification algorithms. *Procedia computer science* **132**, 1578–1585 (2018).
25. Smith, J. W., Everhart, J. E., Dickson, W., Knowler, W. C. & Johannes, R. S. Using the ADAP learning algorithm to forecast the onset of diabetes mellitus, 261 (1988).
26. Stone, M. Cross-validated choice and assessment of statistical predictions. *Journal of the royal statistical society: Series B (Methodological)* **36**, 111–133 (1974).
27. Wilcoxon, F. Some uses of statistics in plant pathology. *Biometrics Bulletin* **1**, 41–45 (1945).
28. Wolpert, D. H. Stacked generalization. *Neural networks* **5**, 241–259 (1992).

29. World Health Organization. *Diabetes* [Online; accessed 28-November-2021]. 2021. https://www.who.int/health-topics/diabetes#tab=tab_1.
30. World Health Organization. *The top 10 causes of death* [Online; accessed 28-November-2021]. 2020. <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death>.