Journal of Computational and Applied Mathematics 427 (2023) 115115

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Comparing and tuning machine learning algorithms to predict type 2 diabetes mellitus

Gabriel Aguilera-Venegas [a], Amador López-Molina [b], Gemma Rojo-Martínez [c], José Luis Galán-García [a,*]

[a] Depto. de Matemática Aplicada, Universidad de Málaga, Spain
[b] Escuela de Ingenierías Industriales, Universidad de Málaga, Spain
[c] UGC Endocrinología y Nutrición, Hospital Regional Universitario de Málaga, CIBERDEM, IBIMA-Plataforma BIONAND, Málaga, Spain

ARTICLE INFO

Article history:
Received 28 October 2022
Received in revised form 19 January 2023

Keywords:
Type 2 diabetes mellitus
Machine learning
Decision Trees
Random Forest
kNN
Neural Networks

ABSTRACT

The main goals of this work are to study and compare machine learning algorithms to predict the development of type 2 diabetes mellitus.

Four classification algorithms have been considered, studying and comparing the accuracy of each one to predict the incidence of type 2 diabetes mellitus seven and a half years in advance. Specifically, the techniques studied are: Decision Tree, Random Forest, kNN (k-Nearest Neighbours) and Neural Networks. The study not only involves the comparison among these techniques, but also, the tuning of the hyperparameters of each algorithm.

The algorithms have been implemented using the language R. The data base used has been obtained from the nation-wide cohort di@bet.es study.

This work includes the accuracy of each algorithm and therefore the best technique for this problem. The best hyperparameters for each algorithm will be also provided.

1. Introduction

In the world 537 million adults (20–79 years) are living with diabetes (10%). This number is predicted to rise to 643 million by 2030 and 783 million by 2045. Diabetes is responsible for 6.7 million deaths in 2021 [1].

In Spain around 13% of adults have diabetes and more than 500,000 new cases are diagnosed each year [2]. The first national study in Spain of prevalence of diabetes and impaired glucosa was the Di@bet.es Study carried out with 5072 participants [3].

The prediction models for T2DM (type 2 diabetes mellitus) are very important in order to prevent suffering from T2DM. The prevention in this field can improve the quality of life of human beings and even save lives.

1.1. This work

The main goal of this work is to study and compare various machine learning algorithms to predict the development of type 2 diabetes mellitus.

* Corresponding author.
E-mail addresses: gaguilera@uma.es (G. Aguilera-Venegas), ama2lm@outlook.com (A. López-Molina), gemma.rojo.m@gmail.com (G. Rojo-Martínez), jlgalan@uma.es (J.L. Galán-García).

https://doi.org/10.1016/j.cam.2023.115115

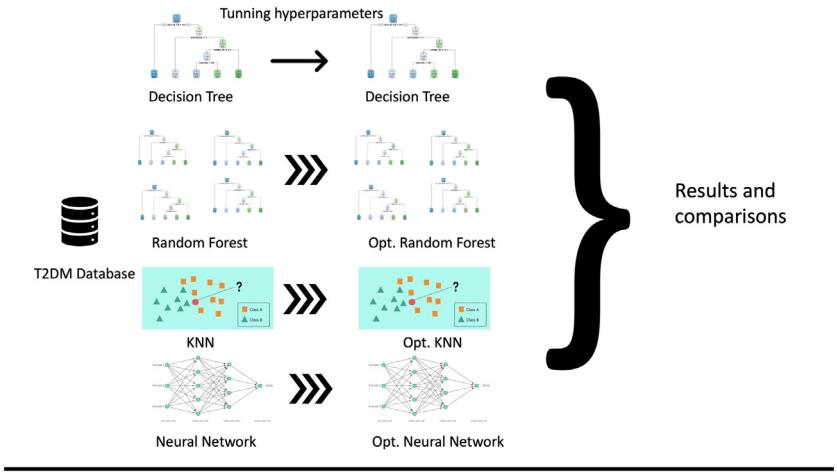

Fig. 1. Scheme of the work.

We have used a database obtained from the nation-wide cohort di@bet.es study [2] with thousands of individuals, some of them were diagnosed with type 2 diabetes 7.5 years after measuring the rest of the variables.

In the literature, several studies can be found in which machine learning algorithms are applied to problems in Medicine. For example, in [4], the authors apply machine learning technics to predict metabolic syndrome. The conclusion of this study is: "On average, machine learning models outperformed conventional statistical approaches for patient classification".

The present study involves the comparison, for the prediction of suffering T2DM in 7.5 years, among four machine learning techniques. The tuning of the hyperparameters of each algorithm are also detailed.

The algorithms have been implemented using the language R. "R is a potent and free software for graphical and statistical analysis" [5]. A brief history of the language R and some reasons of its wide use can be found in [6].

Four classification algorithms have been considered, studying and comparing the accuracy of each one to predict the incidence of type 2 diabetes mellitus 7.5 years in advance. Specifically, the techniques studied for this problem are:

- Decision Tree (R command: rpart)
- Random Forest (R command: randomForest)
- kNN (k-Nearest Neighbours) (R command: knn)
- Neural Networks (R command: nnet)

This work includes the accuracy of each algorithm and therefore the best technique for this problem. The best hyperparameters for each algorithm will be also provided.

A scheme of the work is showed in Fig. 1.

1.2. T2DM database

The T2DM database has been obtained as a result of the nation-wide cohort di@bet.es study [2].

5072 people older than 18 years were randomly selected from all over Spain.

A total of 2408 subjects participated in the follow-up. In total, 154 people developed diabetes in 7.5 years of follow-up.

18 variables have been considered including socio-demographic and clinical data.

One of the variables contains information about if the individual has been diagnosed of T2DM in the follow-up. This variable is the objective variable for the classification algorithms studied and the rest of the variables are used as independent variables.

2. Tuning the hyperparameters

Machine learning technics have been using for the diagnosis of type 2 diabetes mellitus in recent years, for example in [7]. In the present work we have studied the accuracy of four technics, changing the hyperparameters in order to obtain the best values for them for the aforementioned problem. The metric used for evaluating the model is accuracy $= (TP + TN)/(TF + FP + TN + FN)$, that is, the sum of the elements of the main diagonal of the confusion matrix divided by the sum of all the elements of the confusion matrix.

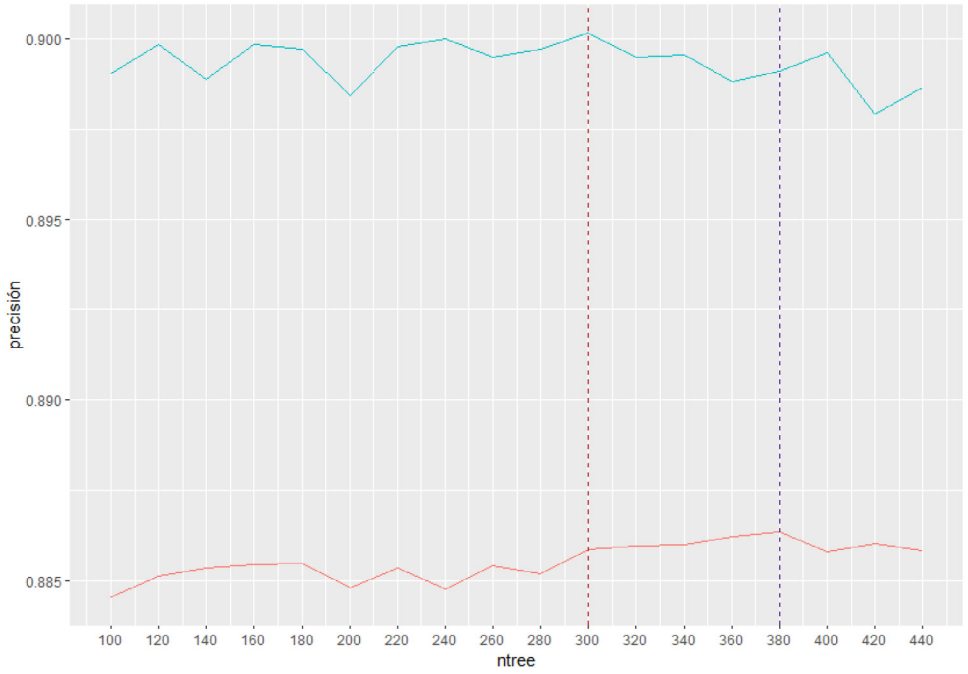

Fig. 2. The parameter ntree of the command randomForest.

2.1. Decision tree

Decision tree is a machine learning technic that has previously used for diagnosing type 2 diabetes, for example in [8].

Decision tree algorithm has been included only for comparison with the other methods, since it can be considered as a particular case of the Random Forest method, in which the forest has one unique tree.

We have used the R command: rpart of the R package rpart (in CRAN). A description of this command can be found in [9].

The hyperparameters have been not tuned in this case since:

- Just one decision tree have very high instability in the measurement of its accuracy.
- The method is included in the Random Forest algorithm.

2.2. Random forest

Random Forest algorithm has been previously used for estimating the risk of suffering type 2 diabetes in some occasions, for example in [10].

We have used the R command randomForest of the R package randomForest (in CRAN). A description of this command can be found in [11]

We have considered four parameters of the command randomForest for tuning: ntree, mtry, nodesize and maxnodes.

2.2.1. ntree

The parameter ntree fixes the number of trees of the forest. The graphic in Fig. 2 shows the accuracy (red line) measured with a set of individuals different to the individuals used to train the forest and 1- OOB Error (1 - Out Of Bag error) (green line). The red vertical dotted line marks the maximum value of the accuracy and the violet vertical line marks the maximum value of 1-OOB.

From the values showed in the previous graphic, a grid from 290 to 310 has been selected in order to obtain the best value in conjunction with the other parameters.

2.2.2. mtry

The parameter mtry fixes the maximum number of variables used in each tree of the forest. As in the previous graphic, Fig. 3 shows the accuracy (red line) and 1- OOB Error (green line). The red vertical line corresponds with the maximum value of the accuracy and the violet vertical line with the maximum value of 1-OOB.

From the values showed in the previous graphic, a grid from 2 to 4 has been selected in order to obtain the best value in conjunction with the other parameters.

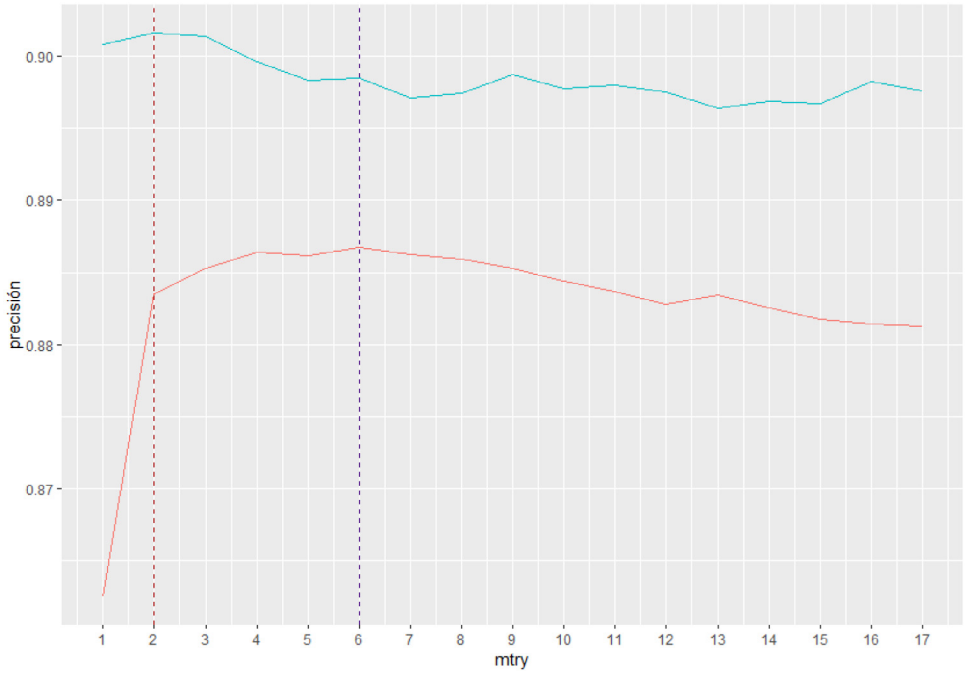

Fig. 3. The parameter mtry of the command randomForest.

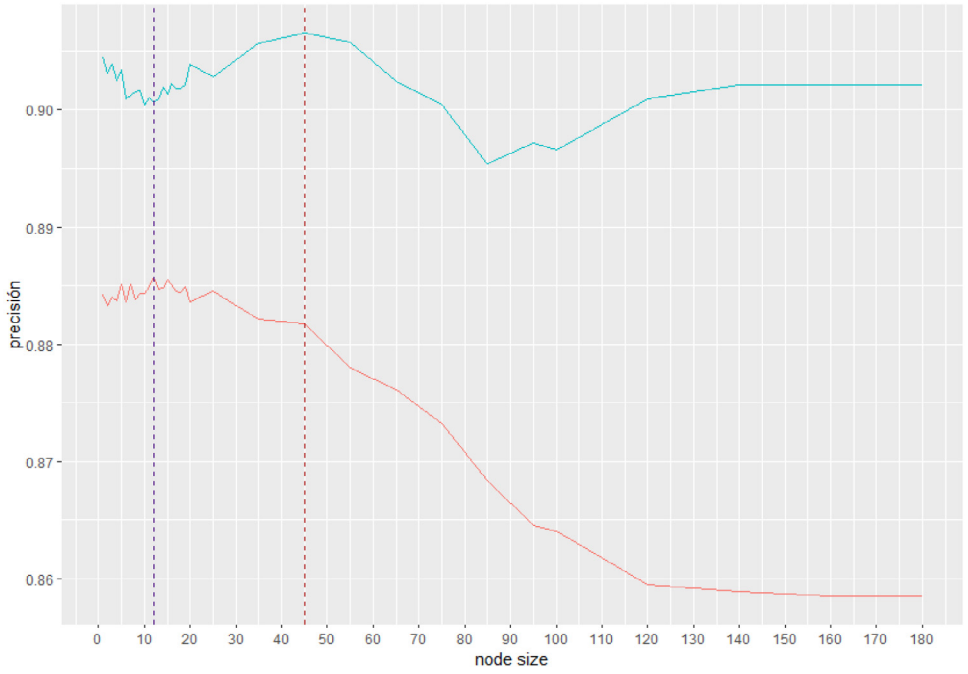

Fig. 4. The parameter nodesize of the command randomForest.

2.2.3. Nodesize

The parameter nodesize fixes the minimum size of terminal nodes. The graphic in Fig. 4 shows the accuracy (red line) and 1- OOB Error (1 - Out Of Bag error) (green line). The red vertical dotted line determines the maximum value of the accuracy and the violet vertical line the maximum value of 1-OOB.

From the values showed in the previous graphic, a grid from 40 to 50 has been selected in order to obtain the best value in conjunction with the other parameters.

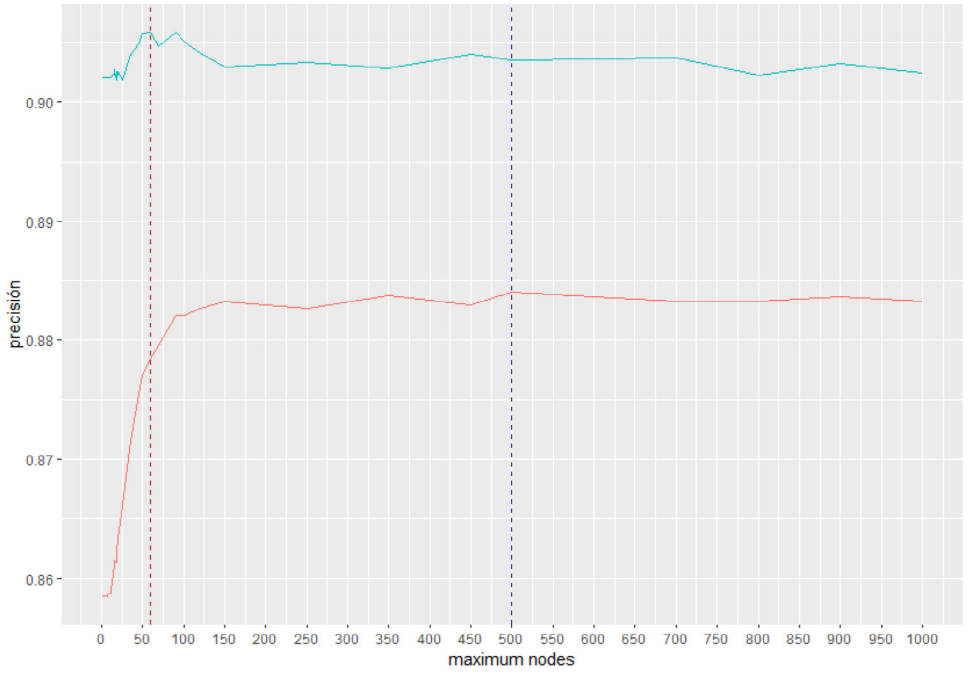

Fig. 5. The parameter maxnodes of the command randomForest.

2.2.4. Maxnodes

The parameter maxnodes fixes the maximum number of terminal nodes. The graphic in Fig. 5 shows the accuracy (red line) measured with an external to the forest set of individuals and 1- OOB Error (1 - Out Of Bag error) (green line). The red vertical dotted line marks the maximum value of the accuracy and the violet vertical line marks the maximum value of 1-OOB.

From the values showed in the previous graphic, a grid from 50 to 90 has been selected in order to obtain the best value in conjunction with the other parameters.

2.2.5. Optimal values of the parameters

Combining the previously obtained grids the optimal values for the parameters in order to maximise the accuracy are:

- ntree: The optimal value is 305.
- mtry: The optimal value is 4.
- nodesize: The optimal value is 50.
- maxnodes: The optimal value is 50.

2.3. kNN (k-Nearest Neighbours)

kNN algorithm has been applied to predict diabetes mellitus in several occasions, for example in [12].

We have used the R command knn of the R package class. A description of this command can be found in [13]

2.3.1. k

The parameter k in the command knn fixes the number of neighbours considered.

The graphic in Fig. 6 shows the points analysed (axis X corresponds with k and axis Y corresponds with the accuracy). The vertical red line marks the value of k with maximum value of the accuracy.

The grid selected has been from 15 to 25.

2.3.2. l

Parameter l in the command knn fixes the minimum vote for definite decision.

The graphic in Fig. 7 shows the points analysed (axis X is k and axis Y is accuracy). The vertical red line marks the maximum value of the accuracy.

The grid selected has been form 0 to 9. Please, note that for $l > 9$ the precision is higher but for these values the algorithm doubts in many cases. In the following two paragraphs an explanation about this fact is included:

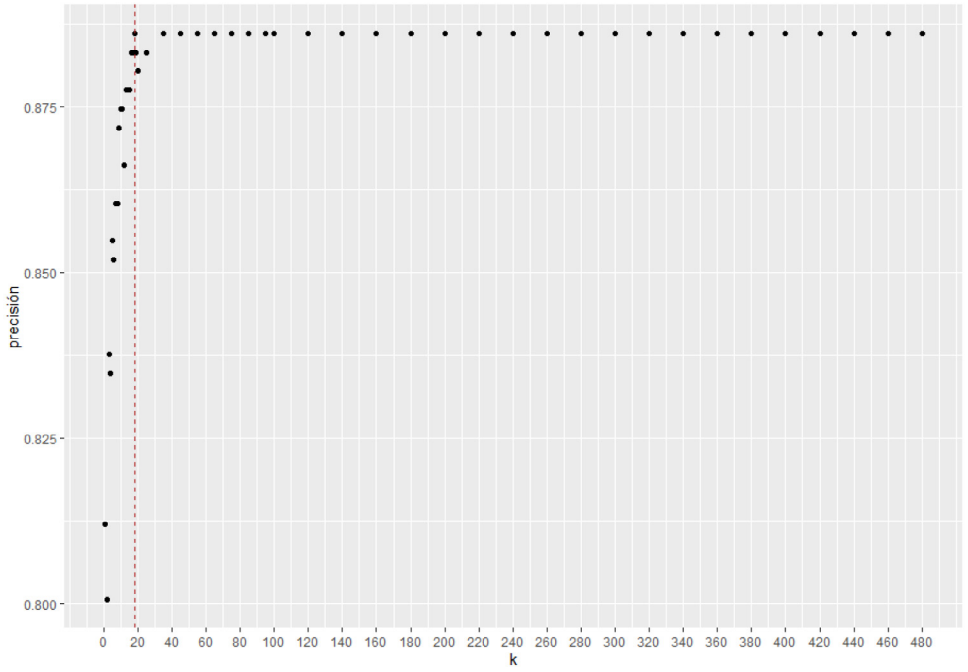

Fig. 6. The parameter k of the command knn.

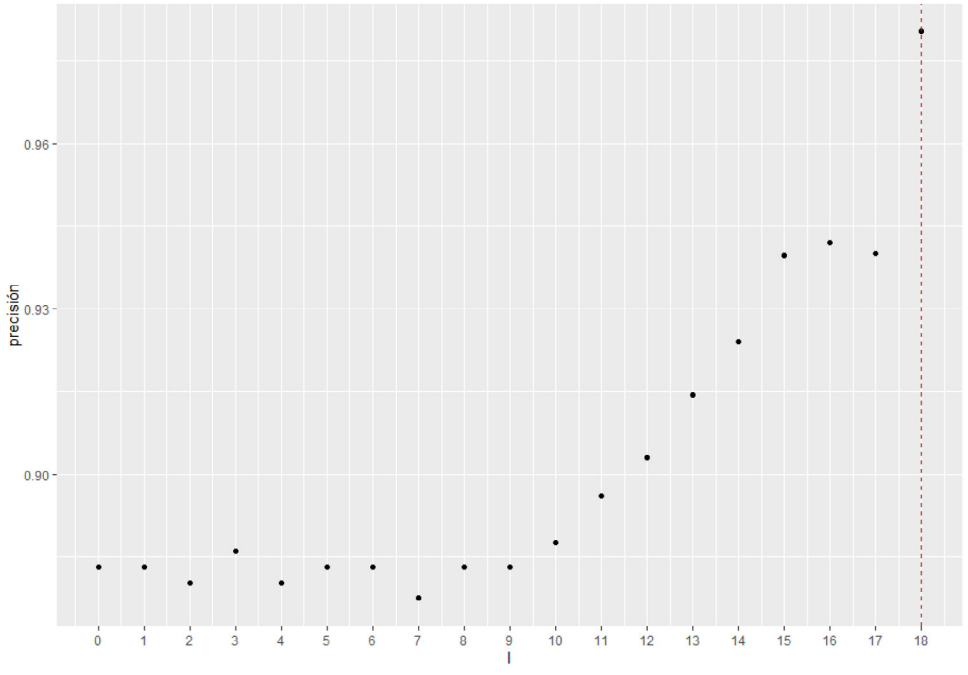

Fig. 7. The parameter l of the command knn.

The parameter l indicates the variation of the minimum number of votes to classify an observation. This hyperparameter depends on the parameter k, since less than k-l dissenting votes are allowed (it could be less if there are not enough minimum number of votes), otherwise the algorithm doubts how to classify the observation and produces error. The values that have been specified for l are the following: sequence from 0 up to the value of k.

Not all the values are valid. Specifically, from $l = 10$ some observations produce errors because the algorithm does not have enough dissenting votes and does not know how to classify the individuals. Since the algorithm does not provide the

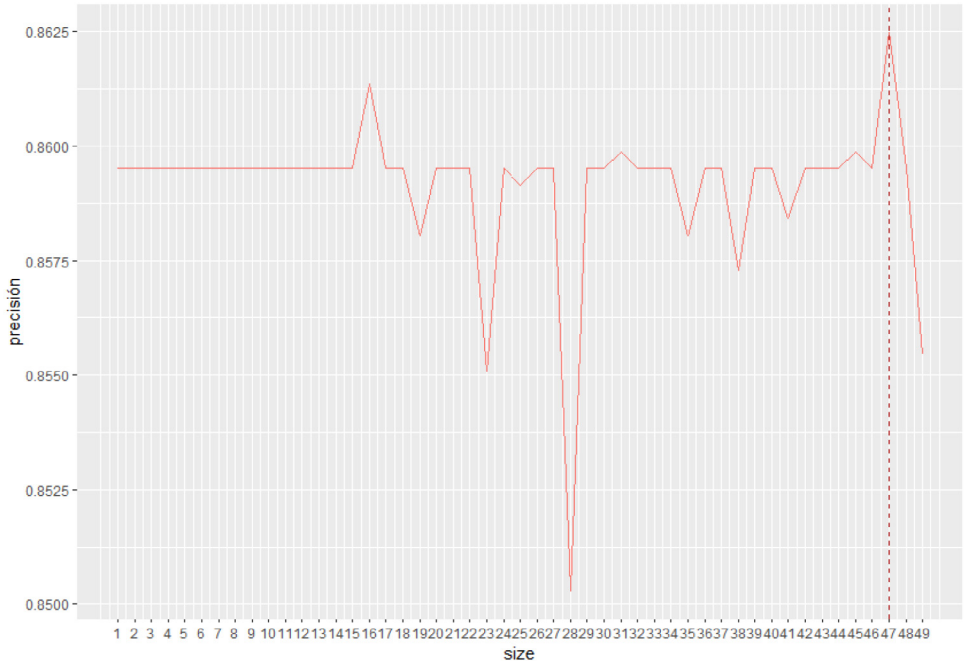

Fig. 8. Parameter size of command nnet.

number of neighbours when classifying an observation, it is not possible to know the number of dissenting neighbours. Therefore, when the value of l approaches the value of k, the algorithm is not able to classify the observation because it does not have l dissenting votes. In the range of l from 10 to 18 not all the observations can be classified because there are not enough votes to classify an observation.

Combining the previously obtained grids, the optimal values for the parameters in order to maximise the accuracy are:

- The optimal value of k is 23.
- The optimal value of l is 9.

2.4. Neural network

An example of use of artificial neural networks for prediction of diabetes can be found in [14]

Single-hidden-layer neural network is implemented in package nnet (shipped with base R). [15]

2.4.1. Size

Parameter size in command nnet fixes the number of units in the hidden layer.

The graphic in Fig. 8 shows the accuracy in function of the parameter size. The vertical red line marks the maximum value of the accuracy.

The grid selected has been from 46 to 48.

2.4.2. Decay

Parameter decay of command nnet fixes weight decay.

The graphic in Fig. 9 shows the accuracy in function of the parameter decay. The vertical red line marks the value of decay with maximum accuracy.

The grid selected has been from 1.5 to 2.

2.4.3. Maxit

Parameter maxit of command nnet fixes maximum number of iterations.

The graphic in Fig. 10 shows the accuracy in function of the parameter maxit. The vertical red line marks the maximum value of the accuracy.

The grid selected has been from 2500 to 2650.

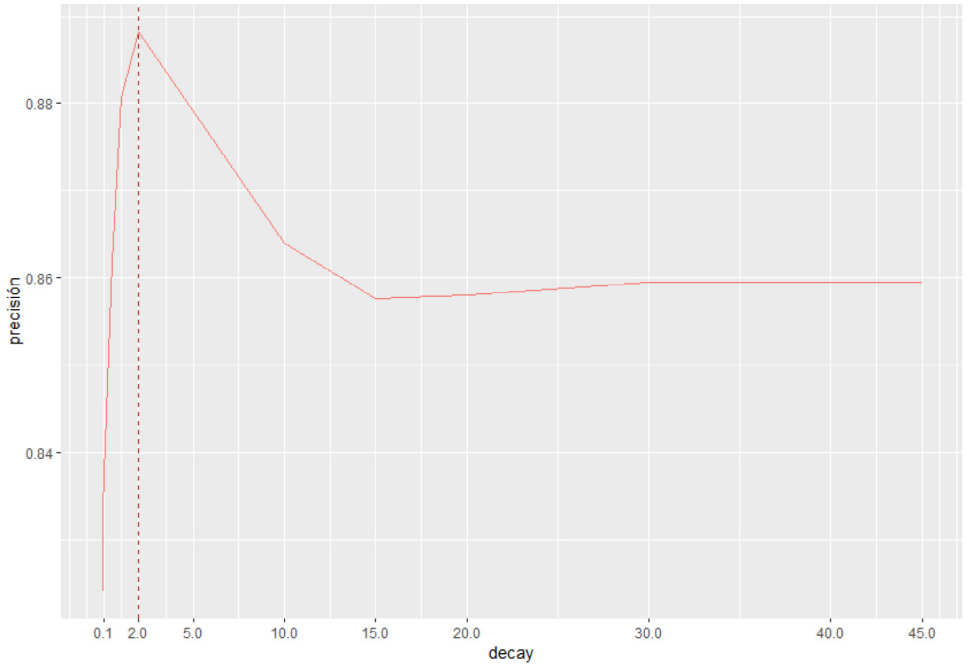

Fig. 9. Parameter decay of command nnet.

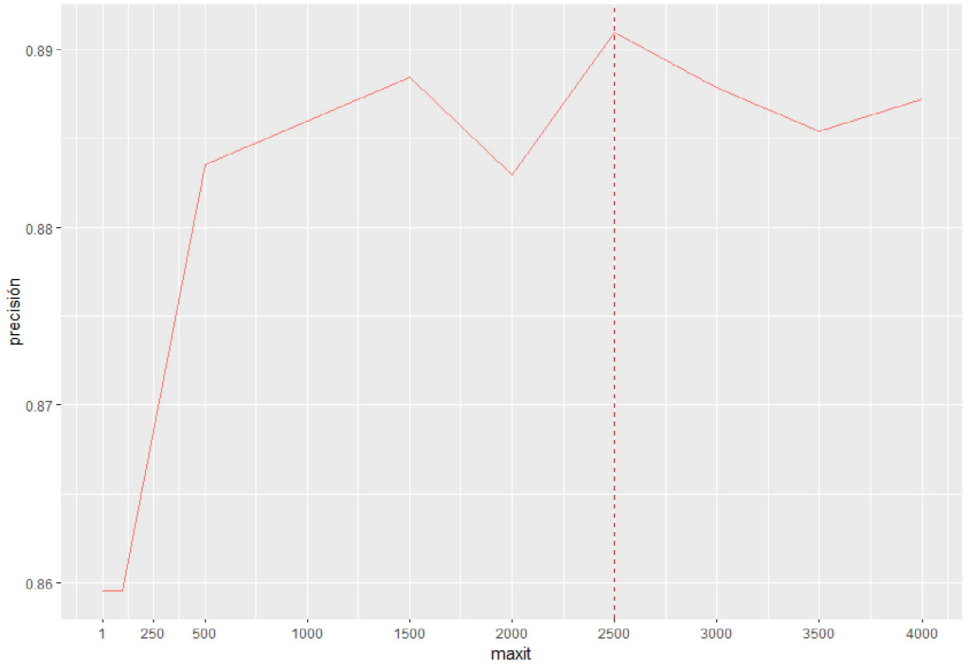

Fig. 10. Parameter maxit of command nnet.

2.4.4. Optimal values of the parameters

Combining the previously obtained grids, the optimal values for the parameters in order to maximise the accuracy are:

- The optimal value of size is 46.
- The optimal value of decay is 1.5.
- The optimal value of maxit is 2650.

Table 1
Confusion matrix for decision tree.

	DT No	DT Yes
Real No	304	7
Real Yes	29	11

Table 2
Confusion matrix for ramdom forest.

	RF No	RF Yes
Real No	350	4
Real Yes	24	17

Table 3
Confusion matrix for kNN.

	kNN No	kNN Yes
Real No	353	3
Real Yes	60	1

3. Results

In this section, the confusion matrix and the accuracy of each technique for the optimal parameters selected previously are shown.

To obtain the confusion matrix the dataset has been randomly split into a set of training and a set of testing. Approximately the 80% of the individuals have been assigned to the set of training and the rest to the set of testing. The procedure has been the following:

1. A sample of pseudo-random numbers from a continuous uniform distribution on the interval $[0, 1]$ has been generated (one for each individual in the dataset).
2. Individuals which random number is less or equal than 0.8 have been assigned to the set of training.
3. Individuals which random number is greater than 0.8 have been assigned to the set of testing.

The confusion matrix is obtained comparing the prediction of the technic with the real values in the set of test (these individuals have not been used for the training in the technic). The accuracy is measured dividing the sum of the elements in the main diagonal of the confusion matrix (right results) by the sum of all elements of the confusion matrix (right and wrong results).

3.1. Results of decision tree

The confusion matrix for Decision Tree is shown in Table 1.
Decision tree accuracy $= \frac{304+11}{304+7+29+11} = \frac{315}{351} = 0.8974359 \approx 89.74\%$

3.2. Results of random forest

The confusion matrix for Random Forest with the optimal hyperparameters is shown in Table 2.
Random-Forest accuracy $= \frac{350+17}{350+4+24+17} = \frac{354}{395} = 0.929139 \approx 92.91\%$

3.3. Results of kNN

The confusion matrix for kNN with the optimal hyperparameters is shown in Table 3.
kNN accuracy $= \frac{353+1}{353+3+60+1} = \frac{354}{417} = 0.8489209 \approx 84.89\%$

3.4. Results of neural network

The confusion matrix for Neural Network with the optimal hyperparameters is shown in Table 4.
Neural-network accuracy $= \frac{447+15}{447+18+61+15} = \frac{462}{541} = 0.8539741 \approx 85.40\%$

Table 4
Confusion matrix for Neural Network.

	NN No	NN Yes
Real No	447	18
Real Yes	61	15

4. Conclusions and future work

4.1. Conclusions

Four technics (decision trees, random forests, kNN and neural networks) have been applied to a database of a set of individuals randomly selected. A set of variables were measured for each individual. A new variable showing whether the individual was diagnosed of type 2 diabetes mellitus or not was added seven and a half years after the initial data was taken. This variable is considered the objective variable and the technics classifies with this variable using the rest of the variables. This way, the technics predict if the person is going to be diagnosed with type 2 diabetes mellitus 7.5 years in the future.

The technics have been applied using the R language. A study of the accuracy of each technic which some parameters has been carried out, changing the value of each parameter. A grid of best values has been selected for each parameter and for each technic. A study of all possible combinations of the values in the grids in all the parameters of each technic in order to obtain an optimal value for each parameter of each technic. Using these optimal values a confusion matriz for each technic has been obtained and therefore an accuracy for each technic has been obtained.

The data of the accuracy for each technic for this problem has been:

- Decision tree accuracy = 89.74%
- Random Forest accuracy = 92.91%
- kNN accuracy = 84.89%
- Artificial Neural Network accuracy = 85.40%

The best technic for this problem among the four analysed has been Random Forest, with an accuracy of 92.91%.

4.2. Future work

The authors are extending the study to other machine learning technics as Support Vector Machine and Naive Bayes. Also, they are working on increasing the number of hidden layers conducting the study towards deep learning. New metrics as *recall* are being studied in order to decrease the number of false negatives.

Data availability

Data will be made available on request.

Acknowledgements

This work was partially supported by the Ministerio de Sanidad, Servicios Sociales e Igualdad-ISCIII, Instituto de Salud Carlos III (PI20/01322), European Regional Development Fund (ERDF) "A way to build Europe". Funding for open access charge: Universidad de Málaga/CBUA. We thank the anonymous reviewers for their useful suggestions and corrections which have improved the quality of the paper.

References

[1] IDF Diabetes Atlas, tenth ed., International Diabetes Federation, Brussels, Belgium, 2021, Available online https://www.diabetesatlas.org (accessed 27 October 2022).
[2] G. Rojo-Martínez, et al., Incidence of diabetes mellitus in Spain as results of the nation-wide cohort di@bet.es study, Sci. Rep. 10 (1) (2020) 2765, http://dx.doi.org/10.1038/s41598-020-59643-7.
[3] F. Soriguer, et al., Prevalence of diabetes mellitus and impaired glucose regulation in Spain: the di@bet.es study, Diabetologia 55 (2012) 88–93, http://dx.doi.org/10.1007/s00125-011-2336-9.
[4] M. Akbarzadeh, N. Alipour, H.H. Moheimani, et al., Evaluating machine learning-powered classification algorithms which utilize variants in the GCKR gene to predict metabolic syndrome: Tehran cardio-metabolic genetics study, J. Trans. Med. 20 164 (2022) http://dx.doi.org/10.1186/s12967-022-03349-z.
[5] R.B. Dessau, C.B. Pipper, R–project for statistical computing, Ugeskr. Laeger. 170 (5) (2008) 328–330, PMID: 18252159.
[6] F.M. Giorgi, C. Ceraolo, D. Mercatelli, The r language: An engine for bioinformatics and data science, Life (Basel) 12 (5) (2022) 648, http://dx.doi.org/10.3390/life12050648, PMID: 35629316; PMCID: PMC9148156.
[7] P. Tumuluru, et al., DPMLT: Diabetes prediction using machine learning techniques, in: 2022 International Conference on Electronics and Renewable Systems, ICEARS, 2022, pp. 1127–1133.

[8] A.A. Al Jarullah, Decision tree discovery for the diagnosis of type II diabetes, in: 2011 International Conference on Innovations in Information Technology, 2011, pp. 303–307, http://dx.doi.org/10.1109/INNOVATIONS.2011.5893838.
[9] Rpart: Recursive partitioning and regression trees, 2022, https://cran.r-project.org/web/packages/rpart/index.html (accessed 27 October 2022).
[10] W. Xu, J. Zhang, Q. Zhang, X. Wei, Risk prediction of type II diabetes based on random forest model, in: 2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics, AEEICB, 2017, pp. 382–386, http://dx.doi.org/10.1109/AEEICB.2017.7972337.
[11] Randomforest: Breiman and cutler's random forests for classification and regression, 2022, https://cran.r-project.org/web/packages/r{and}omForest/index.html (accessed 27 October 2022).
[12] M. NirmalaDevi, S.A. alias Balamurugan, U.V. Swathi, An amalgam KNN to predict diabetes mellitus, in: 2013 IEEE International Conference on Emerging Trends in Computing, Communication and Nanotechnology, ICECCN, 2013, pp. 691–695, http://dx.doi.org/10.1109/ICE-CCN.2013.6528591.
[13] Knn: k-nearest neighbour classification, 2022, https://www.rdocumentation.org/packages/class/versions/7.3-20/topics/knn (accessed 27 October 2022).
[14] N.S. El_Jerjawi, S.S. Abu-Naser, Diabetes prediction using artificial neural network, Int. J. Adv. Sci. Technol. 121 (2018) 54–64.
[15] Nnet: Feed-forward neural networks and multinomial log-linear models, 2022, https://cran.r-project.org/web/packages/nnet/index.html (accessed 27 October 2022).