IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Optimizing Machine Learning Models: A Comprehensive Evaluation For Superior Predictive Performance

[1]Sidharth Sivakumar, [2] Tamilvendan V, [3]Sushiiel A, [4*]Parthasarathy G

[123]UG Student, [3]Assistant Professor

School of Computer Science and Engineering
Vellore Institute of Technology, Vellore, India

Abstract: A key factor in the major disease prediction process is machine learning. Accuracy still needs to be improved in this area. The goal of this research article is to maximize prediction performance in diabetes diagnosis and treatment by utilizing machine learning. Examining the advantages of several algorithms, the study credits Random Forest's performance to its deft handling of non-linearity, feature importance, and efficient handling of missing data. Critical elements for improving model performance are also covered, including feature selection strategies and hyperparameter optimization. The results emphasize the need of thorough model evaluation, with Random Forest showing up as a reliable option for precise diabetes prediction. The knowledge gathered from this research has applications for data scientists and healthcare professionals. This process will help to make better decisions when developing prediction models for diabetes and other illnesses, which will ultimately lead to better patient care and disease management.

Index Terms **- Machine Learning, Prediction, Parameter tuning, Feature Selection.**

I. INTRODUCTION

The use of state-of-the-art technologies has become essential for bettering patient care and managing diseases in the constantly changing healthcare landscape [1]. Machine Learning (ML) is one such technical advancement that has completely changed the medical industry. Precision medicine has entered a new era thanks to the application of ML approaches, which have given medical professionals strong tools for risk assessment, early diagnosis, and individualized treatment[2]. Within this framework, this research sets out to investigate the importance of machine learning (ML), its underlying ideas, classification algorithms, and the particular methodology we use to maximize predictive accuracy using a diabetic dataset.

The increasing use of machine learning in healthcare is not coincidental; rather, it is a result of its demonstrated capacity to use data-driven insights to inform choices. Machine learning is particularly good at finding complex patterns in large datasets, which makes it possible to find hidden connections and trends that would otherwise remain hidden. These capabilities have a revolutionary effect on prognostication and early disease identification, which are critical components of patient outcomes. Fundamentally, machine learning (ML) is the process of teaching computers to understand and use data to forecast or make judgments. ML algorithms can be viewed as "intelligent" instruments in the healthcare industry, able to identify minute patterns and connections in patient data[3]. Predictive accuracy is used as the performance measure in this work, and the experience is derived from a diabetes dataset. The task involves diabetes prediction.

One of the most important ML tasks for medical diagnosis is classification. Sorting input data into preset classes or labels is the goal of classification[4]. We look into several classification methods in this context, including Random Forest, k-nearest neighbours, decision trees, logistic regression and support vector machines. Random Forest has been shown to be effective in classification tasks by Breiman (2001), and our work attempts to further explore its potential applications in diabetes prediction [5].

This work provides an extensive analysis of machine learning algorithms for the diagnosis and treatment of diabetes in this research. The following is our methodology: Choosing a model and figuring out the ideal data split ratio come first. Then, with this ratio, we determine which model fits the dataset the best. In order to attain the maximum predicted accuracy, this work utilizes Grid Search CV, which is grid search with cross-validation, for hyperparameter tweaking. Finding the model that works well with the diabetes dataset is our main goal in order to improve prediction accuracy. We laid the groundwork for a more thorough examination of machine learning's efficacy in diabetes prediction and management by examining the field's function in healthcare, comprehending its core ideas, looking at classification algorithms, and outlining our study approach. This paper's later sections will give a thorough explanation of literature survey , proposed system ,experiment and results and conclusion.

literature review

This review article sheds light on diabetes's prevalence and difficulties in underdeveloped nations. It provides insightful background information for appreciating the importance of diabetes detection and treatment [6]. A popular ensemble technique, the Random Forests algorithm, for classification and regression applications, was first presented in this groundbreaking study. Because of their stability and capacity to handle high-dimensional data, random forests are useful for diagnosing complicated illnesses like diabetes [7].

Gradient Boosting Machines, as described in this paper, have become a powerful tool in predictive modeling. They are used for classification tasks and have been applied in healthcare for disease diagnosis, including diabetes [8]. While this paper is primarily focused on document recognition, the concept of gradient-based learning and neural networks, as discussed by Yann LeCun, is foundational for many machines learning applications, including medical diagnosis and diabetes prediction [9].

This paper presents an approach that uses the Naive Bayes classifier for the diagnosis of Type-2 diabetes. It demonstrates the application of machine learning in the healthcare domain for diabetes diagnosis [10]. Christopher M. Bishop's book provides a comprehensive introduction to pattern recognition and machine learning. It serves as a valuable resource for understanding the fundamental principles underlying machine learning techniques applied in healthcare and diabetes diagnosis [11]. AdaBoost is an ensemble learning method. While the paper focuses on explaining AdaBoost, the technique has been used in various medical applications, including diabetes prediction [12].

This paper discusses the application of optimization models to improve the Naive Bayes classifier. Naive Bayes is a popular technique for medical diagnosis, and this work explores enhancements [13]. The paper focuses on accurate risk stratification for diabetes using machine learning. It highlights the importance of accurate risk assessment in diabetes management[14]. This paper discusses the use of various classification algorithms for diabetes prediction, emphasizing the role of machine learning in healthcare [15]. This paper compares different classifiers for diabetes risk prediction, providing insights into the performance of various machine learning models in the context of diabetes [16]. The paper focuses on the classification of diabetes mellitus using various machine learning techniques, demonstrating the role of machine learning in healthcare [17]. This paper explores the use of genetic programming for the design of classifiers to detect diabetes mellitus, illustrating innovative approaches in diabetes diagnosis [18].

III METHODOLOGY

The proposed framework consists of (i) Preprocessing, (ii) Model Selection, (iii) Model Training, (iv) Parameter Tuning, and (iv) prediction. The Figure 1 shows the block diagram of our proposed work.

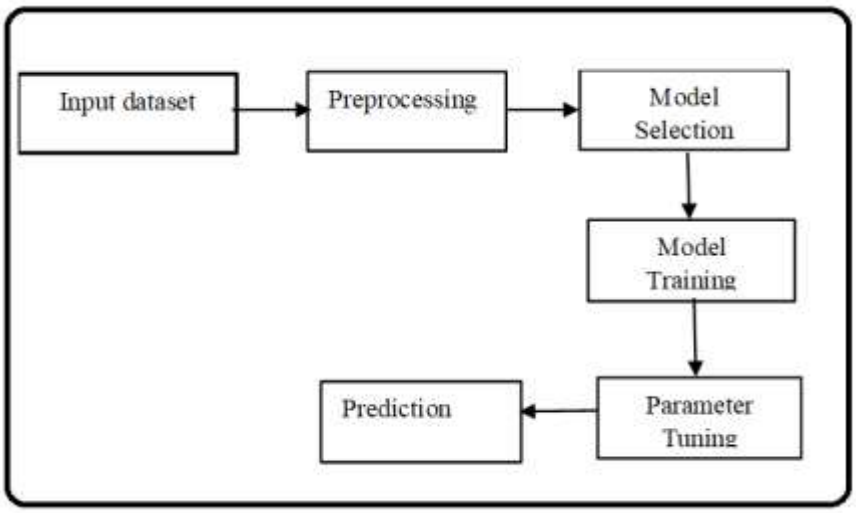

Figure 1. Proposed System

3.1 Preprocessing

Address any missing values present in the dataset. Implement appropriate strategies to handle missing data, ensuring the dataset's completeness. Additionally, perform feature scaling to normalize the range of features and encode categorical variables as required. These preprocessing steps are crucial to enhance the quality and usability of the data for subsequent analyses. Before applying machine learning classifiers, meticulous dataset preprocessing is essential. This phase involves several key steps,(i)Handling Missing Data Techniques such as imputation are employed to address missing values in the dataset,(ii) Normalizing features ensures that all variables contribute equally to model training,(iii) Techniques like oversampling or under sampling are used to mitigate class imbalance issues in the dataset.

3.2 Model Selection

There exists an optimal model denoted as Optimal_Model, which is defined as the combination of the best-performing machine learning model M_best and its corresponding optimal hyperparameter configuration H_optimal. In simpler terms, Optimal_Model represents the ideal pairing of a model and its hyperparameters that collectively maximize performance based on the specified criteria.

3.3 Model Training

In the context of model selection, there exists a best-performing model M_best, where M_best is an element of the set of machine learning models M. This selection is based on the criterion that, for every model M_i within the set M, the performance metric P evaluated on the training set T and validation set V, considering the hyperparameters H, is greater or equal for M_best compared to M_i. In simpler terms, M_best is chosen from the available models because it consistently demonstrates superior performance across the specified training and validation sets under the given hyperparameters. This process ensures the identification of the most effective model within the considered set for the given task.

3.4 Parameter Tuning

In the context of hyperparameter tuning, for every hyperparameter H_i within the set of hyperparameters H, there exists an optimal hyperparameter configuration H_optimal. This optimal configuration is an element of the candidate hyperparameter set C, and it is determined based on the criterion that the performance metric P, evaluated for the best-performing model M_best on the training set T and validation set V, is greater or equal when using H_optimal compared to H_i. In other words, this statement asserts that there exists an optimal combination of hyperparameters within the candidate set, ensuring that the selected hyperparameter configuration consistently yields equal or superior performance for the given model across the specified training and validation sets, compared to individual hyperparameters within the original set.

Consider the diabetes dataset denoted as X, with Y representing the target variable indicating diabetes status. Let M be the set of machine learning models, where M = {Logistic Regression, Decision Trees, Support Vector Machines, K-Nearest Neighbors, Random Forest}. Additionally, let S be the data split ratio, a real number such that $0 < S < 1$, defining the proportion of data allocated for training and validation. P represents the performance metric, such as accuracy. H is the set of hyperparameters for a machine learning model, denoted as $H = \{H_1, H_2, ..., H_n\}$. The training set is represented as T, and V is the validation set. The set of candidate hyperparameters for the model is denoted as $C = \{C_1, C_2, ..., C_m\}$. These elements collectively form the framework for exploring and evaluating machine learning models on the diabetes dataset. The approach can be expressed using quantifiers and logical symbols as follows:

Algorithm 1: Optimal Model Selection

Data Splitting:
$\forall X, S, \exists T, V (T, V = \text{Split}(X, S))$
Model Selection:
$\exists M\_best (M\_best \in M \wedge \forall M\_i \in M, P(M\_best, T, V, H) \geq P(M\_i, T, V, H))$
Hyperparameter Tuning:
$\forall H\_i \in H, \exists H\_optimal (H\_optimal \in C \wedge P(M\_best, T, V, H\_optimal) \geq P(M\_best, T, V, H\_i))$
Optimal Model:
$\exists \text{Optimal\_Model} (\text{Optimal\_Model} = (M\_best, H\_optimal))$

3.5 Prediction

Prediction is defined as the output of an machine learning algorithm after it has been trained on a training dataset and applied to test data when predicting the value of a particular outcome. The accuracy of the prediction is calculated by the equation 1.

$$\text{Accuracy} = (\text{Number of Correct Predictions}) / (\text{Total Number of Predictions}) \quad (1)$$

IV.EXPERIMENTS AND RESULTS

The test bed is prepared using python under the windows environment. The diabetes data set[19] is used for our experiment and results are recorded in the following manner. The table 1 provides insights into the impact of different data split ratios on the accuracy of the Gaussian Naive Bayes (GNB) classifier. The three ratios considered are 60-40, 70-30, and 80-20. Among these, the 80-20 data split ratio stands out as the most efficient configuration, resulting in the highest accuracy of 79.22% for the GNB classifier.

Table 1 Identifying Optimal Training and Testing Ratio in GNB Classifier

	Training/Test Ratio	Accuracy
GNB	60/40	75.0000
	70/30	76.1904
	80/20	79.2207

The GNB classifier serves as a sample model in this context, illustrating the influence of data split ratios on model accuracy. The 80-20 split ratio offers the GNB classifier a larger training dataset, which contributes to its improved predictive performance. This observation emphasizes the importance of selecting an appropriate data split ratio when training machine learning models for accurate predictions.

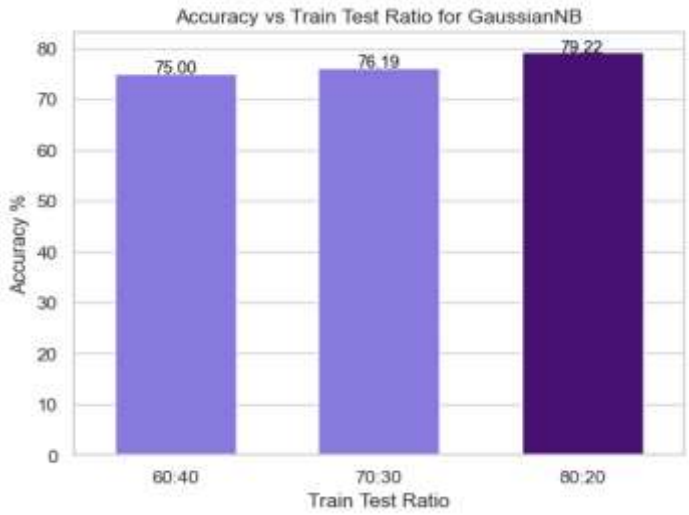

Figure 3. Accuracy measure of GNB Classifier

In summary, this table 1 demonstrates that the 80-20 data split ratio is particularly effective for the GNB classifier, resulting in the highest accuracy among the evaluated scenarios. The choice of data split ratio plays a crucial role in model performance and should be considered when building predictive models.

Upon analyzing the results, we observed that the 80-20 data split ratio consistently produced the highest accuracy across multiple models. This ratio allowed our models to benefit from a larger training dataset, resulting in improved predictive performance. The table above presents the best accuracy obtained for each model, emphasizing the importance of data split ratios in model training.

The accuracy of the Random Forest model both before and after applying grid search. It is important to note that we found Random Forest to be a robust model with impressive accuracy. Before Grid Search: The Random Forest model achieved an accuracy of 81.17%. This high level of accuracy indicates that Random Forest is a promising choice for diabetes prediction even without hyperparameter tuning. After hyperparameter tuning using grid search, the accuracy of the Random Forest model improved to 81.82%.

This demonstrates the power of grid search in fine-tuning model hyperparameters, leading to a slight but notable enhancement in accuracy. It is worth mentioning that the Random Forest model performed exceptionally well with the 80-20 data split, highlighting the efficiency of this split ratio. The 80-20 split allowed the model to benefit from a larger training dataset, contributing to its improved predictive performance.

Table 3 Test run of models with various training and test rations

MODEL	RATIO	test 1	test 2	test 3	test 4	test 5	Average
GNB	60	75	75	75	75	75	75
	70	76.1904	76.1904	76.1904	76.1904	76.1904	76.1904
	80	79.2207	79.2207	79.2207	79.2207	79.2207	79.2207
SVM	60	75.6493	75.6493	75.6493	75.6493	75.6493	75.6493
	70	75.3246	75.3246	75.3246	75.3246	75.3246	75.3246
	80	79.2207	79.2207	79.2207	79.2207	79.2207	79.2207
Decision Tree	60	69.4805	68.5064	69.1558	70.1298	71.4285	69.7402
	70	75.3246	73.5930	74.4588	74.4588	74.0259	74.3722
	80	75.9740	76.6233	79.2207	80.5194	78.5714	78.1818
KNN	60	74.3506	74.3506	74.3506	74.3506	74.3506	74.3506
	70	74.8917	74.8917	74.8917	74.8917	74.8917	74.8917
	80	75.3246	75.3246	75.3246	75.3246	75.3246	75.3246
SGD	60	50.3246	0.76623	67.5324	66.8831	67.8571	50.6727
	70	54.1125	0.79220	70.9956	43.7229	46.7532	43.2753
	80	70.1298	0.79220	53.2467	72.7272	57.7922	50.9376
Random Forest	60	75.0000	75.0000	75.0000	75.0000	76.6233	75.3246
	70	76.6233	77.4891	77.4891	76.6233	77.0562	77.0562
	80	81.8181	81.8181	79.2207	80.5194	82.4675	81.1688
Bagging-Meta	60	75.9740	74.6753	72.7272	74.3506	72.7272	74.0909
	70	77.9220	75.3246	75.7575	78.7878	76.1904	76.7965
	80	80.5194	79.8701	77.2727	81.1688	79.8701	79.7402
Ada Boost	60	77.2727	66.5584	66.5584	66.5584	66.5584	68.7013
	70	75.3246	67.9653	67.9653	67.9653	67.9653	69.4372
	80	77.9220	70.7792	70.7792	70.7792	70.7792	72.2077
Gradient Boost	60	75.6493	75.6493	75.6493	75.6493	75.6493	75.6493
	70	78.3549	78.7878	78.3549	77.9220	78.7878	78.4415
	80	80.5194	81.1688	80.5194	81.1688	81.1688	80.9090
Meta-Bagging	60	75.6493	75.6493	73.7013	75.6493	73.3766	74.8051
	70	77.9220	74.8917	75.3246	76.1904	73.5930	75.5844
	80	81.8181	85.0649	76.6233	78.5714	79.2207	80.2597

This table3 provides a comprehensive overview of the performance of various machine learning models under different data split ratios, including 60-40, 70-30, and 80-20. The models evaluated include Gaussian Naive Bayes (GNB), Support Vector Machines (SVM), Decision Trees, K-Nearest Neighbors (KNN), Stochastic Gradient Descent (SGD), Random Forest, Bagging-Meta, Ada Boost, Gradient Boost, and Meta-Bagging.

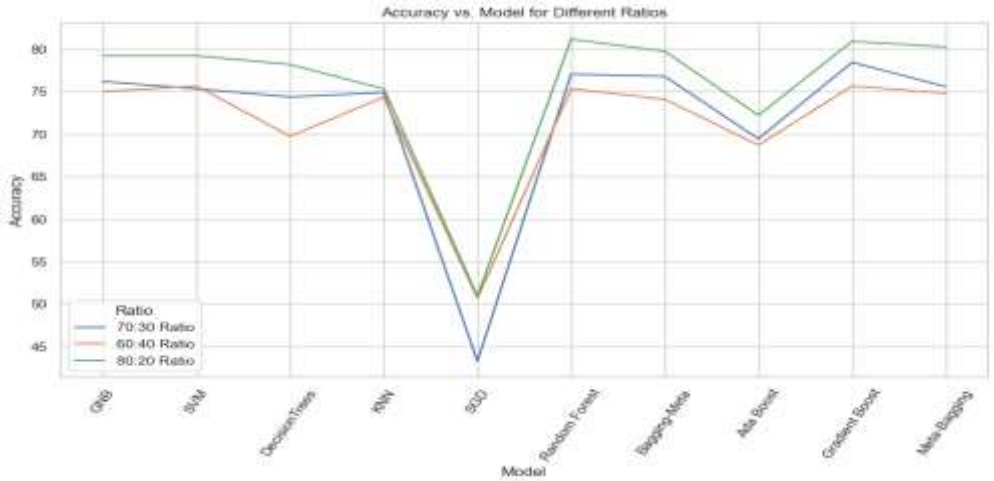

Figure 4. Accuracy measure of different classifiers

The accuracy of each model is assessed through five rounds of testing (Test 1 to Test 5), and the average accuracy is computed to provide a consolidated measure of model performance. The table 3offers a valuable insight into how different data split ratios influence model accuracy. The 80-20 data split ratio appears to be particularly efficient, resulting in high accuracies across various models, such as GNB, SVM, Random Forest, and others. It is notable that without the use of grid search, the models' performance showcases how effective an 80-20 data split can be.

Understanding the most suitable data split ratio is critical for optimizing model performance in practice. In this case, the 80-20 split consistently yields impressive results. The absence of grid search suggests that these results are achieved without extensive hyperparameter tuning. The provided data can guide model selection and data preparation in real-world machine learning applications, allowing practitioners to make informed decisions about data split ratios and model choices for their specific tasks.

5. CONCLUSION

This study systematically assessed the performance of various machine learning models under different data split ratios, both with and without grid search optimization. The findings offer valuable insights for practitioners in the field of machine learning. The choice of data split ratio significantly influences model accuracy. Among the tested ratios (60-40, 70-30, and 80-20), the 80-20 data split consistently yielded the highest accuracy across multiple models, demonstrating its effectiveness in enhancing predictive performance. Without the use of grid search, several models excelled in accuracy. Notably, Random Forest, Bagging-Meta, and Gradient Boost achieved the highest accuracies at 80-20 data splits, highlighting their potential for practical applications.

Furthermore, the study delved into the performance of the Random Forest model before and after grid search optimization. It was observed that grid search, an effective hyperparameter tuning technique, further improved the model's accuracy, emphasizing the importance of fine-tuning in model optimization.

In conclusion, this research provides valuable guidance for selecting data split ratios and models for machine learning tasks. The 80-20 data split emerges as an efficient choice, and grid search optimization can significantly enhance model accuracy. These insights support informed decision-making in real-world machine learning applications and underscore the importance of data preparation and model selection in achieving superior predictive performance.

6. FUTURE DIRECTIONS

Future research could expand on this study by considering a broader array of performance metrics to assess model quality, accounting for the complexities of healthcare data, and exploring the generalizability of the approach to other medical conditions. Additionally, incorporating domain-specific knowledge and additional features into the analysis may lead to more accurate and interpretable models. In conclusion, this study provides a structured methodology for optimizing machine learning models for diabetes prediction. The approach, as demonstrated through the evaluation of various models and hyperparameter tuning, holds promise for improving healthcare decision-making and patient outcomes. However, it is important to acknowledge the limitations and the need for further research to refine and extend these findings in the field of healthcare and medical data analysis.

REFERENCES

[1] Bhat, S. S., Banu, M., Ansari, G. A., & Selvam, V. (2023). A risk assessment and prediction framework for diabetes mellitus using machine learning algorithms.

[2] Chowdhury, M. M., Ayon, R. S., & Hossain, M. S. (2023). An investigation of machine learning algorithms and data augmentation techniques for diabetes diagnosis using class imbalanced BRFSS dataset.

[3] N.G., B. A. (2024). En-RfRsK: An ensemble machine learning technique for prognostication of diabetes mellitus.

[4] Dharmarathne, G., Jayasinghe, T. N., Bogahawaththa, M., Meddage, D. P. P., & Rathnayake, U.(2024). A novel machine learning approach for diagnosing diabetes with a self-explainable interface.

[5] Pal, T. (2021). A review on current advances in machine learning-based diabetes prediction. pp.1.

[6] Misra,A., Gopalan,H., Jayawardena,R., Hills,A.P., Soares,M., Reza-Albarrán,A.A., Ramaiya,K.L.(2019).Diabetes in developing countries

[7] Breiman, L. (2001). Random Forests. Machine learning. Springer. 45(1), 5-32.

[8] Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of statistics, 1189-1232.

[9] LeCun, Y. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[10] Sarwar, A., & Sharma, V., (2012). Intelligent Naive Bayes Approach to Diagnose Diabetes Type-2. Special Issue of International Journal of Computer Applications on Issues and Challenges in Networking Intelligence and Computing Technologies.

[11] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer

[12] R. E. Schapire, "Explaining AdaBoost" in Empirical Inference, Berlin, Germany: Springer, pp. 37-52, Oct. 2013.

[13] Taheri, S., & Mammadov, M., (2013). Learning the naive Bayes classifier with optimization models. Int. J. Appl. Math. Computer. Sci., vol. 23, no. 4, pp. 787-795.

[14] Maniruzzaman, M., Rahman, M. J., Al-MehediHasan, M., Suri, H. S., Abedin, M. M., El-Baz, A., (2018) Accurate diabetes risk stratification using machine learning: Role of missing value and outliers. J. Med. Syst., vol. 42, no. 5, pp. 92.

[15] Sisodia, D., & Sisodia, D. S., (2018). Prediction of diabetes using classification algorithms. Procedia Computer. Sci. vol. 132. pp. 1578-1585.

[16] Nai-arun, N., & Moungmai, R., (2015). Comparison of classifiers for the risk of diabetes prediction. Procedia Computer. Science., vol. 69, pp. 132-142.

[17] Dewangan, K., & Agrawal, P., (2015). Classification of diabetes mellitus using machine learning techniques. Int. J. Eng. Appl. Sci., vol. 2, no. 5, pp. 145-148.

[18] Pradhan, M., & Bamnote, G., R., (2015). Design of classifier for detection of diabetes mellitus using genetic programming. Proc. 3rd Int. Conf. Frontiers Intelligent. Computer. Theory Appl., pp. 763-770.

[19] Diabetes dataset: https://www.kaggle.com/datasets/mathchi/diabetes-data-set