SHM PUBLISHER
SURYA HIJAU MANFAAT

Journal of Soft Computing Exploration

Homepage: https://shmpublisher.com/index.php/joscex

JOSCEX

p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

Comparison of gridsearchcv and bayesian hyperparameter optimization in random forest algorithm for diabetes prediction

Rini Muzayanah[1*], Dwika Ananda Agustina Pertiwi[2], Muazam Ali[3], Much Aziz Muslim[4]
[1, 4]Department of Computer Science, Universitas Negeri Semarang, Indonesia
[2, 4]Faculty of Technology Management, Universiti Tun Hussein Onn Malaysia, Johor 86400, Malaysia
[3]Faculty of Management Science, HITEC University Taxila, Pakistan

Article Info

Article history:

Received November 25, 2023
Revised December 19, 2023
Accepted January 24, 2024

Keywords:

Function point analysis
Use case diagrams
Software effort estimation
Adjusted function points
Estimation accuracy

ABSTRACT

Diabetes Mellitus (DM) is a chronic disease whose complications have a significant impact on patients and the wider community. In its early stages, diabetes mellitus usually does not cause significant symptoms, but if it is detected too late and not handled properly, it can cause serious health problems. To overcome these problems, diabetes detection is one of the solutions used. In this research, diabetes detection was carried out using Random Forest with gridsearchcv and bayesian hyperparameter optimization. The research was carried out through the stages of study literature, model development using Kaggle Notebook, model testing, and results analysis. This study aims to compare GridSearchCV and Bayesian hyperparameter optimizations, then analyze the advantages and disadvantages of each optimization when applied to diabetes prediction using the Random Forest algorithm. From the research conducted, it was found that GridSearchCV and Bayesian hyperparameter optimization have their own advantages and disadvantages. The GridSearchCV hyperparameter excels in terms of accuracy of 0.74, although it takes longer for 338,416 seconds. On the other hand, Bayesian hyperparameter optimization has a lower accuracy rate than GridSearchCV optimization with a difference of 0.01, which is 0.73 and takes less time than GridSearchCV for 177,085 seconds.

Corresponding Author:

Rini Muzayanah,
Department of Computer Science,
Universitas Negeri Semarang,
Sekaran, Kota Semarang, Jawa Tengah, Indonesia
Email: rinimuzayanah0415@students.unnes.ac.id

https://doi.org/10.52465/joscex.v5i1.308

1. INTRODUCTION

Diabetes Mellitus (DM) is a chronic disease characterized by hyperglycemia due to impaired insulin secretion, impaired insulin action or both [1]. In Indonesia, 133 million people are also reported to be living with diabetes mellitus, and 87.5% of them suffer from uncontrolled glycemic [2]. Cumulative evidence suggests that long-term glycemic control is a major risk factor for the development of micro- and macrovascular complications in diabetic patients. Diabetes and its complications have a significant impact on patients and society at large. Diabetes has an impact on increasing health care costs in the health care system and reducing life expectancy and quality of life of the population.In its early stages, diabetes mellitus usually does not cause significant symptoms, but if it is detected too late and is not handled properly, it can cause serious health problems, such as: heart attack, blindness, kidney failure, limb amputation and even death. Early diagnosis of this disease can significantly improve the patient's quality of life [3].

Research on the prediction of diabetes has been carried out by [4] with the Random Forest method and optimization of hyperparameter tuning. The research produced accuracy, f1score, precision, recall and specificity of 88.61%, 75.68%, 100%, 60.87% and 100% respectively. In this experimental analysis, an accuracy rate of 88.61% was achieved when the 'n_estimator' value was 5 while the parameter range was tested between 1 and 50. The same experiment was carried out with the min_sample_leaf value".

One better approach to increase the outcome of any classifier is to tune the hyperparameters of that classifier [5]. The parameters that are set by the data analysts before the training process is called hyperparameters and it is independent of the training process [6]. Hyperparameter optimization is an optimization that aims to select the best hyperparameters from a particular model that will produce the best performance from the model being built [7]. The hyperparameter optimization algorithm optimizes discrete, ordinal, and continuous variables, but must simultaneously choose which variables to optimize [8]. There are various approaches available for implementing hyperparameters, for example GridSearchCV and Bayesian.

This study was made to compare the accuracy of diabetes prediction using the random forest algorithm with GridSearchCV and Bayesian Hyperparameter optimization. The purpose of this research is to find out the strengths and weaknesses of each optimization when it is applied to predict diabetes using the Random Forest algorithm where a target value of 1 is used to predict a high probability of developing diabetes while a target of 0 is used to predict a low probability of developing diabetes.

2. METHOD

The research was carried out through the stages of study literature, model development using Kaggle Notebook, model testing, and results analysis. The research stage can be seen in Figure 1.

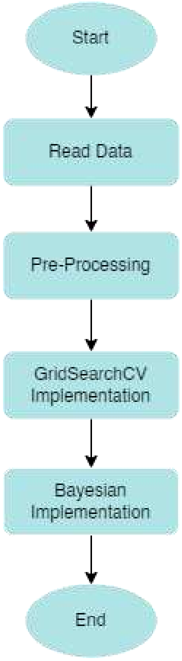

Figure 1. Research stages

Dataset

This study used data from the National Institute of Diabetes and Digestive and Kidney Diseases accessed using the Kaggle database platform [9]. The data is data from a 21-year-old woman who has a total of 8 variables, namely Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, Diabetes Pedigree Function, Age, and Outcome.

.

Pre-processing Stage

Before the data is used, the data needs to go through the pre-processing stage to check the data rows that have a zero value, then the normalization process is carried out. The purpose of the data normalization process is to group data according to different units so that it becomes well structured without data repetition.

GridSeachCV Implementation

This stage is carried out with the ridSearchCV hyperparameter tuning to find parameters that can produce the most optimal performance in the model to be developed. After hyperparameter tuning, the parameters identified as the most optimal parameters are stored for later use in the model development process. The next stage is to build a prediction model with the Random Forest algorithm using the most optimal parameters resulting from the hyperparameter tuning process. Grid search is an approach to parameter tuning that will methodically build and evaluate a model for each combination of algorithm parameters specified in a grid [10].

Bayesian Implementation

This stage is carried out using Bayesian hyperparameter tuning to find parameters that can produce the most optimal performance in the model to be developed. After hyperparameter tuning, the parameters identified as the most optimal parameters are stored for later use in the model development process. The next stage is to build a prediction model with the Random Forest algorithm using the most optimal parameters resulting from the hyperparameter tuning process. Bayesian optimisation can be costly especially when the model is learnt over a large volume of data [11].

Modeling with Random Forest

Random forest is an ensemble learning algorithm in machine learning proposed by Breiman [12] and it is a widely used machine learning method with high prediction accuracy. Random forest is a combination of tree predictors such that each tree depends on random vector values that are sampled independently and with the same distribution for all trees in the forest [13]. The random forest is a classifier consisting of a set of structured tree classifiers where each tree issues a sound unit for the most popular class in the x input [14]. The random forest has been considered one of the most successful ensemble algorithms in machine learning, which builds a large number of random trees one by one and then makes predictions based on an average of the resulting predictions. How the random forest algorithm works can be seen in Figure 2.

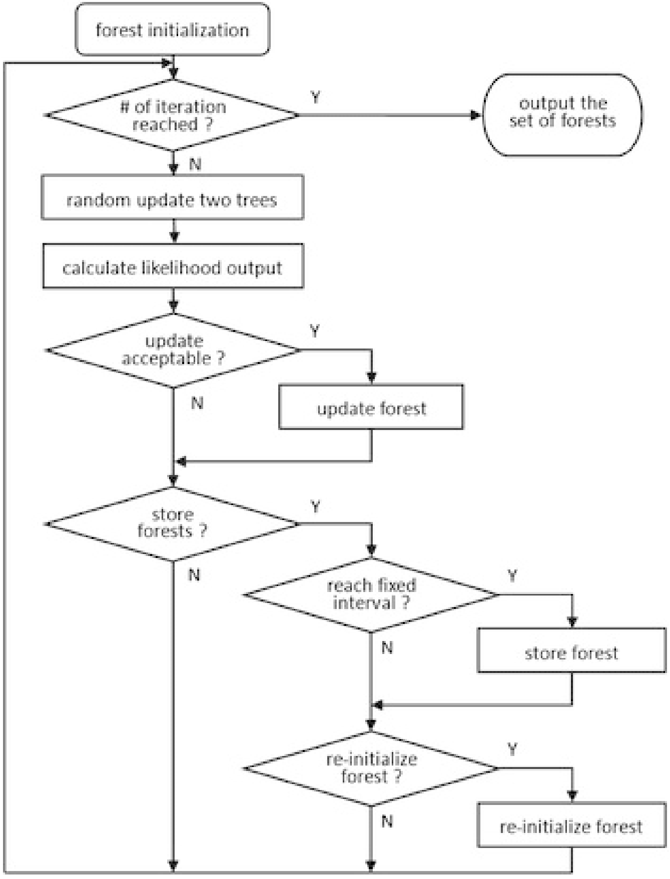

Figure 2. Flowchart random forest algorithm [15]

Hyperparameter Tuning

a. GridSearchCV

GridSearchCV is one part of the scikit-learn module which validates more than one model and provides each hyperparameter automatically and systematically. GridSearch is used to find parameters that can produce the most optimal performance in the model to be developed. s. GS theoretically finds the optimal combination of parameters by the exhaustive method, while the amount of computation required for GS increases exponentially as the parameter dimension increases [16].

b. Bayesian

Bayesian optimization is a very effective algorithm [16]. The Bayesian optimization is an approach to globally optimizing unknown functions [17]. Bayesian optimization creates a probabilistic model of the objective function and uses it to select hyperparameters to estimate the true objective function.

Results Analysis

The results of diabetes prediction using Random Forest with GridSearchCV hyperparameter optimization are compared with Bayesian hyperparameter optimization. In this study, comparisons were made with benchmarks of time and accuracy of the predictions that had been made. Mode performance analysis will be carried out using several evaluation matrices including accuracy, precision, recall and f1-score. Evaluation is carried out to analyze how well the model can perform classification so that later it can be used to help humans predict whether diabetes is detected or not.

3. RESULTS AND DISCUSSIONS

Before the research was carried out, feature analysis was carried out to find out whether there were problems with column collinearity. Analysis was carried out using HeatMap which can be seen in Figure 2.

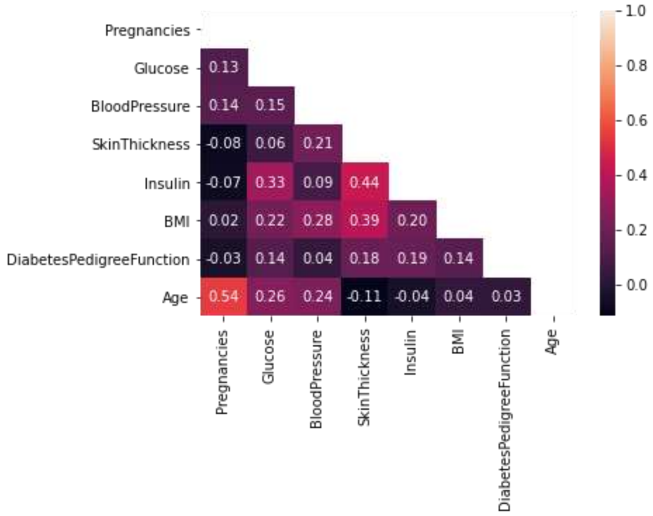

Figure 2. Colleration matrix

From the figure above, it is known that the pregnancies and age, SkinTickness and Insulin, and SkinTickness and BMI columns have a significant level of dependency. The diabetes data above will be divided into two, namely training and test data with a test size of 0.1.

A parameter search algorithm from a random forest classifier was run to find the optimal set of hyperparameters in terms of classification accuracy. The param_grid dictionary defines the range of hyperparameter values for the Random Forest classifier being searched for.

Grid_SearchCV is a traditional brute-force method that searches the hyperparameter space for the best-tuned model. A GridSearchCV object is sent when a Random Forest Classifier object is sent. The next stage is a pipeline that uses the StandardScaler transformer to standardize the training data before training a model from the training data and standardize the test data before making predictions. Finally, a fitting function is sent out of the pipeline to train each model in the hyperparameter grid and find the best fit. Random Forest classification report with GridSearchCV optimization can be seen in Table 1.

Table 1. Random forest classification report with gridsearchcv optimization

	Precision	Recall	F1-Score	Support
0	0.75	0.92	0.82	51
1	0.71	0.38	0.50	26
Accuracy			0.74	77
Macro Average	0.73	0.65	0.66	77
Weighted Average	0.74	0.74	0.71	77

From Table 1, the highest F1 score, precision, recall was 0.82, 0.75, 0.92 respectively. From the classification report, the resulting accuracy level is 0.74. The time needed to detect diabetes using GridSearchCV optimization is 338,416 seconds.

Bayesian optimization creates a probabilistic model of the objective function and uses it to select hyperparameters to estimate the true objective function. Random Forest classification report with GridSearchCV optimization can be seen in Table 2.

Table 2. Random forest classification report with bayesian optimization

	Precision	Recall	F1-Score	Support
0	0.74	0.90	0.81	51
1	0.67	0.38	0.49	26
Accuracy			0.73	77
Macro Average	0.70	0.64	0.65	77
Weighted Average	0.72	0.73	0.70	77

From Table 2, the highest F1 score, precision, recall was 0.82, 0.74, 0.90 respectively. From the classification report, the resulting accuracy level is 0.74. The time needed to detect diabetes using Bayesian optimization is 338,416 seconds.

The results comparison of diabetes detection using the Random Forest algorithm with GridSearchCV and Bayesian hyperparameter optimization can be seen in Table 3.

Table 3. Comparison results of random forest diabetes detection with gridsearchcv and bayesian hyperparameter optimization

	Fit Time	Accuracy
GridSearchCV	338.416	0.74
Bayesian	177.085	0.73

From research that has been done to detect diabetes, GridSearchCV optimization requires a longer time than Bayesian optimization, namely GridSearchCV optimization for 338,416 seconds and Bayesian optimization for 177,085 seconds. However, with this longer time, GridSearchCV optimization produces a higher level of accuracy than Bayesian optimization, namely GridSearchCV optimization of 0.74 and Bayesian optimization of 0.73.

4. CONCLUSION

From the research conducted, it was found that GridSearchCV and Bayesian hyperparameter optimization has its own advantages and disadvantages. The GridSearchCV hyperparameter excels in terms of accuracy, although it takes longer. On the other hand, Bayesian hyperparameter optimization has a lower accuracy rate than GridSearchCV optimization with a difference of 0.01 and takes less time than GridSearchCV. The GridSearchCV hyperparameter excels in terms of accuracy of 0.74, although it takes longer for 338,416 seconds. On the other hand, Bayesian hyperparameter optimization has a lower accuracy rate than GridSearchCV optimization with a difference of 0.01, which is 0.73 and takes less time than GridSearchCV for 177,085 seconds.

REFERENCES

[1] Z. Punthakee, R. Goldenberg, and P. Katz, "Definition, Classification and Diagnosis of Diabetes, Prediabetes and Metabolic Syndrome," *Can. J. Diabetes*, vol. 42, pp. S10–S15, Apr. 2018, doi: 10.1016/J.JCJD.2017.10.003.

[2] R. A. Pamungkas, A. M. Usman, K. Chamroonsawasdi, and Abdurrasyid, "A smartphone application of diabetes coaching intervention to prevent the onset of complications and to improve diabetes self-management: A randomized control trial," *Diabetes Metab. Syndr. Clin. Res. Rev.*, vol. 16, no. 7, p. 102537, Jul. 2022, doi: 10.1016/J.DSX.2022.102537.

[3] A. Viloria, Y. Herazo-Beltran, D. Cabrera, and O. B. Pineda, "Diabetes Diagnostic Prediction Using Vector Support

Machines," *Procedia Comput. Sci.*, vol. 170, pp. 376–381, 2020, doi: 10.1016/j.procs.2020.03.065.

[4] S. C. Gupta and N. Goel, "Predictive Modeling and Analytics for Diabetes using Hyperparameter tuned Machine Learning Techniques," *Procedia Comput. Sci.*, vol. 218, pp. 1257–1269, 2023, doi: 10.1016/j.procs.2023.01.104.

[5] M. Ramadhan, I. Sitanggang, F. NASUTION, and A. GHIFARI, "Parameter Tuning in Random Forest Based on Grid Search Method for Gender Classification Based on Voice Frequency," *DEStech Trans. Comput. Sci. Eng.*, vol. 11, no. 9, Oct. 2017, doi: 10.12783/dtcse/cece2017/14611.

[6] S. G. C. G and B. Sumathi, "Grid Search Tuning of Hyperparameters in Random Forest Classifier for Customer Feedback Sentiment Prediction," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 9, 2020, doi: 10.14569/IJACSA.2020.0110920.

[7] V. Shalamov, V. Efimova, and A. Filchenkov, "Faster Hyperparameter Optimization via Finding Minimal Regions in Random Forest Regressor," *Procedia Comput. Sci.*, vol. 212, pp. 378–386, 2022, doi: 10.1016/j.procs.2022.11.022.

[8] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for Hyper-Parameter Optimization," in *Advances in Neural Information Processing Systems*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2011. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf

[9] M. AKTURK, "Diabetes Dataset," 2020. https://www.kaggle.com/datasets/mathchi/diabetes-data-set (accessed May 19, 2023).

[10] G. S. K. Ranjan, A. Kumar Verma, and S. Radhika, "K-Nearest Neighbors and Grid Search CV Based Real Time Fault Monitoring System for Industries," in *2019 IEEE 5th International Conference for Convergence in Technology (I2CT)*, IEEE, Mar. 2019, pp. 1–5. doi: 10.1109/I2CT45611.2019.9033691.

[11] T. T. Joy, S. Rana, S. Gupta, and S. Venkatesh, "Hyperparameter tuning for big data using Bayesian optimisation," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, IEEE, Dec. 2016, pp. 2574–2579. doi: 10.1109/ICPR.2016.7900023.

[12] J. You, S. A. S. van der Klein, E. Lou, and M. J. Zuidhof, "Application of random forest classification to predict daily oviposition events in broiler breeders fed by precision feeding system," *Comput. Electron. Agric.*, vol. 175, p. 105526, Aug. 2020, doi: 10.1016/j.compag.2020.105526.

[13] L. Breiman, "Random Forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[14] J. Jumanto, M. A. Muslim, Y. Dasril, and T. Mustaqim, "Accuracy of Malaysia Public Response to Economic Factors During the Covid-19 Pandemic Using Vader and Random Forest," *J. Inf. Syst. Explor. Res.*, vol. 1, no. 1, pp. 49–70, Dec. 2022, doi: 10.52465/joiser.v1i1.104.

[15] Q. Li and G. Clifford, "Signal Processing: False Alarm Reduction," in *Secondary Analysis of Electronic Health Records*, 2016, pp. 391–403. doi: 10.1007/978-3-319-43742-2_27.

[16] J. Wu, "Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization," *Journal of Electronic Science and Technology*, vol. 17, no. 20190104. p. 26, 2019. doi: 10.11989/JEST.1674-862X.80904120.

[17] Y. Chen *et al.*, "Bayesian optimization based random forest and extreme gradient boosting for the pavement density prediction in GPR detection," *Constr. Build. Mater.*, vol. 387, p. 131564, Jul. 2023, doi: 10.1016/j.conbuildmat.2023.131564.

.