Computer Science and Information Technologies
Vol. 5, No. 2, July 2024, pp. 140~149
ISSN: 2722-3221, DOI: 10.11591/csit.v5i2.p140-149

Improving support vector machine and backpropagation performance for diabetes mellitus classification

Angga Prastyo, Sutikno, Khadijah
Department of Informatics, Faculty of Science and Mathematics, Diponegoro University, Semarang, Indonesia

Article Info

Article history:

Received Dec 16, 2023
Revised Feb 1, 2024
Accepted Feb 15, 2024

Keywords:

Backpropagation
Diabetes mellitus
Imbalanced data
Support vector machine
Synthetic minority oversampling technique

ABSTRACT

Diabetes mellitus is a glucose disorder disease in the human body that contributes significantly to the high mortality rate. Various studies on early detection and classification have been conducted as a diabetes mellitus prevention effort by applying a machine learning model. The problems that may occur are weak model performance and misclassification caused by imbalanced data. The existence of dominating (majority) data causes poor model performance in identifying minority data. This paper proposed handling the problem of imbalanced data by performing the synthetic minority oversampling technique (SMOTE) and observing its effect on the classification performance of the support vector machine (SVM) and Backpropagation artificial neural network (ANN) methods. The experiment showed that the SVM method and imbalanced data achieved 94.31% accuracy, and the Backpropagation ANN achieved 91.56% accuracy. At the same time, the SVM method and balanced data produced an accuracy of 98.85%, while the Backpropagation ANN method and balanced data produced an accuracy of 94.90%. The results show that oversampling techniques can improve the performance of the classification model for each data class.

Corresponding Author:

Sutikno
Department of Informatics, Faculty of Science and Mathematics, Diponegoro University
Prof. Jacob Rais Street, Tembalang, Semarang 50275, Central Java, Indonesia
Email: sutikno@lecturer.undip.ac.id

1. INTRODUCTION

Diabetes is a metabolic disease characterized by increased glucose levels (hyperglycemia) when the pancreas produces insufficient insulin [1]. The International Diabetes Federation (IDF) estimates that there will be at least 537 million adults in the world who have diabetes in 2021, and it is predicted that it will continue to increase to reach 783 million sufferers in 2045. Thus, prevention efforts are needed to reduce the mortality rate caused by diabetes mellitus.

Initial efforts can be made by checking for blood glucose disorders. Prediabetes or borderline diabetes is a condition when glucose and HbA1c levels exceed normal levels but are not high enough to be classified as diabetes [2]. Early detection of this condition will be beneficial for prevention before diabetes mellitus occurs. Machine learning methods can assist in diagnosing and classifying diabetes mellitus [3]-[8]. To increase accuracy, several researchers use other methods, namely support vector machine (SVM) [9], [10], deep neural network [11], [12], nearest neighbor [13], ensemble approach [14], [15], and feature selection [16].

In the case of disease classification, the proportion of data distribution between each class is essential to be noticed [17]. Several researchers have used machine learning methods and imbalanced data to classify diabetes mellitus [2]. One technique for dealing with data imbalance problems is applying minority data oversampling. Research [18] applying the synthetic minority oversampling technique (SMOTE) achieved

better classification performance. Pears *et al*. [19] implemented SMOTE and resulted in a gradual increase in accuracy from 65% to 80%. Research [20] also implemented SMOTE and increased accuracy from 43.27% to 74.38%. Therefore, this paper proposed oversampling techniques using SMOTE to optimize the classification performance of SVM and Backpropagation for diabetes mellitus classification.

2. METHOD

In general, the steps of this research are divided into several processes, as in Figure 1. The first step is data collection. The next step is the main step, which consists of data preprocessing, data splitting, training, and testing. The final step is the evaluation and comparison of the model.

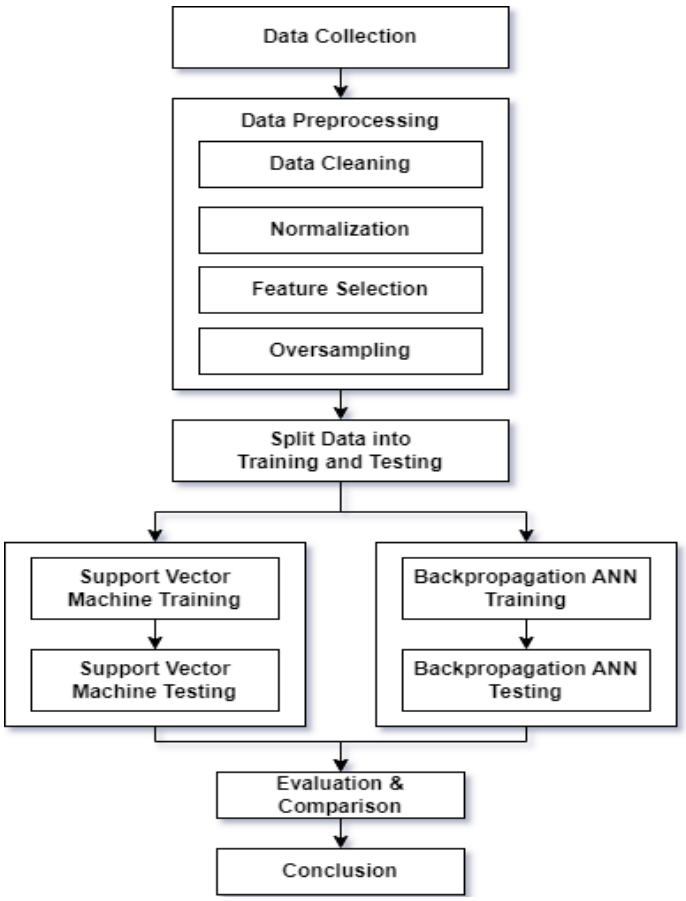

Figure 1. Sequential steps in the proposed methodology

2.1. Data collection

The dataset used in this study was obtained from the Medical City Hospital laboratory and Specialized Center for Endocrinology and Diabetes, Al-Kindy Teaching Hospital, accessed through the Mendeley Data website [21]. This dataset consists of medical information from laboratory analysis. The dataset includes 103 (non-diabetic), 53 (prediabetic), and 844 (diabetic) patients. The attributes are as in Table 1.

2.2. Data cleaning and normalization

The dataset containing 1000 data from laboratory analysis has 12 features, consisting of 11 medical attributes and one feature as the output class target. Meanwhile, two attributes are not needed in the classification process, namely ID and No. Pation. Therefore, both attributes are removed using the drop method. Data cleaning was done: missing values, duplicate data, and encoding labels for the alphabet data type to numeric.

Cleaning is also done by observing anomalies in the data to handle outliers. Outlier handling uses the Interquartile Range (IQR), the difference between the first and third quartiles. IQR is used to obtain each attribute's lower and upper bound values, and then data outside the lower and upper bound will be deleted [22].

In the end, the remaining data became 73 (non-diabetic), 36 (prediabetic), and 523 (diabetic). Next, data normalization changes the data's scale so that each attribute's value has a similar range between 0 and 1.

Table 1. Details of the dataset

Attribute	Description
Gender	Male or Female
Age	In years (min:20, max:79)
Urea	Mg/dl (min:0.5, max:38.9)
Creatinine Ratio	µmol/L (min: 48, max: 80)
HbA1c	mmol/L (min: 0.9, max: 16)
Cholesterol	mmol/L (min: 0.0, max: 10.3)
Triglycerides	mmol/L (min: 0.3, max: 13.8)
HDL	mmol/L (min: 0.2, max: 9.9)
LDL	mmol/L (min: 0.3, max: 9.9)
VLDL	mmol/L (min: 0.1, max: 35)
Body Mass Index	(min: 19, max: 49)
Class	N (non-diabetic), P (prediabetic), Y (diabetic)

2.3. Feature selection

Feature selection is done through a filter method approach using analysis of variance (ANOVA). ANOVA feature selection will compare the variance value between feature groups with the variance within the feature group to obtain the f-ratio value for each feature. Feature selection using ANOVA can also be done based on each attribute's p-value or probability value. A p-value very low or less than α (alpha value) as a significance limit indicates that the feature is more relevant to the class label and should be maintained. The stages of ANOVA feature selection are as follows [23]:

- Create a hypothesis H_1 and H_0
- Count the sum of squares between the groups (SS_B) and the sum of squares within the groups (SS_W).
- Determine the degree of freedom between the groups (df_b) and the degree of freedom within the groups (df_w).
- Count the mean of squares between the groups (MSB) and the mean of squares within groups (MSW).
- Count the f_{ratio}.
- Determine the significance limit α (mostly 0,05).
- Find the probability value or p-value based on the f_{ratio} value in the F distribution table according to the degrees of freedom df_b dan df_w.
- If the result of the p-value is smaller than α, then reject the H_0 so that the feature is suitable for use in the following process.

The statistical results of feature selection with ANOVA for each feature are shown in Table 2. Table 2 shows that the significance level for each feature is BMI, HbA1c, Age, TG, VLDL, Chol, Gender, Urea, LDL, Cr, and HDL. Feature selection is based on a p-value smaller than the alpha value (α = 0.05). For this reason, seven features were selected for the following process: BMI, HbA1c, Age, TG, VLDL, Chol, and Gender.

Table 2. F-statistic result of ANOVA

Attribute	f-ratio	p-value
BMI	173,97	0.00000
HbA1c	151,22	0.00000
Age	109,46	0.00000
TG	22,42	0.00000
VLDL	16,86	0.00000
Chol	11,03	0.00002
Gender	4,73	0.00900
Urea	2,51	0.07800
LDL	0,72	0.38300
Cr	0,65	0.40600
HDL	0,15	0.57500

2.4. Data oversampling

The data used in this study has different numbers; namely, the comparison of non-diabetic: prediabetic: diabetic classes is 73: 36: 523 data. Therefore, the oversampling technique with SMOTE is used

to create synthetic data in the minority class (non-diabetic and prediabetic) to become the same amount of the data in the majority class (diabetic). The oversampling method used in this study is the synthetic minority oversampling technique (SMOTE). This method produces synthetic data between the pairs of nearest neighbors in each minority data as much as the percentage of duplicate minority data [18]. Figure 2 is illustration of SMOTE.

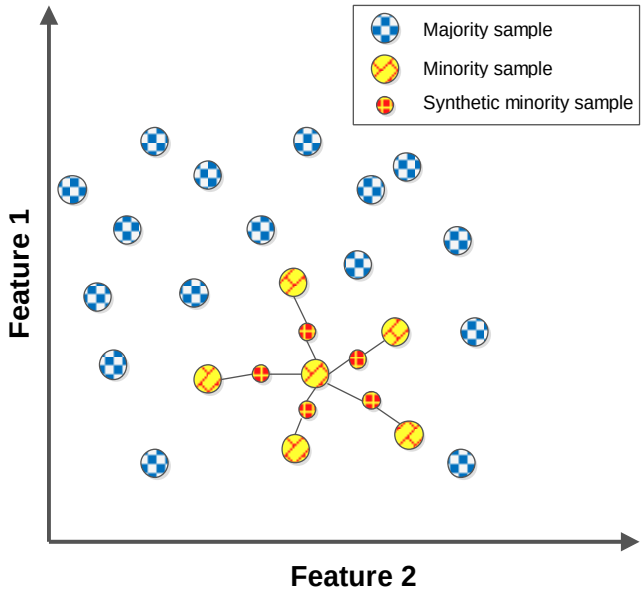

Figure 2. SMOTE illustration

The steps for performing SMOTE for oversampling are as follows [19]:

– Select the k-nearest neighbor value that will be used to generate synthetic data around the reference point.
– Find the nearest neighbors of k reference points used (1).

$$x_{knn} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2} \quad (1)$$

– Calculate each distance difference between the reference point and each of its nearest neighbors, as in (2). Then, multiply the distance by a random number between 0 and 1, and then add the result with the reference point value to produce synthetic data.

$$x_{syn} = x_i + (x_{knn} - x_i) \times \delta \quad (2)$$

Through SMOTE, the number of non-diabetic and prediabetic classes will be as many as the number of majority classes, namely 523 data for each class. Therefore, the amount of data used in the next stage is a total of 1569 data. Table 3 shows the amount of the data after SMOTE.

Table 3. Amount of the data after SMOTE

Step	Amount each class		
	0	1	2
Data Cleaning	73	36	523
SMOTE	523	523	523

2.5. Split data into training and testing

After the data has been balanced, it is divided into training and testing using the K-Fold Cross Validation method with k=10, as in Figure 3. The data division is done with the cross-validation method to assess the ability of the classification model on new data to prevent overfitting. Data divided by ten will be

subsets with the same proportion and class ratio in each subset. In every subset iteration, one subset acts as testing data while the other is training data [24].

Figure 3. K-Fold cross validation with k=10

2.6. Training and testing

We use support vector machine (SVM) and backpropagation for classification. Both are machine learning methods that require a learning process using data. These two methods have been widely applied to solve various problems.

2.6.1. Support vector machine

SVM is a technique to obtain an optimal separating function (hyperplane) in the input space to separate two data classes with different target variable values [25]. Nonlinear data problems in SVM can be solved by using kernel functions. Kernel is a parameter that implements a model in a higher feature space [24]. Table 4 shows some kernel functions used in SVM.

Table 4. Kernel function

Kernel	Function Definition
Linear	$K(x_i, x_j) = x_i^T \cdot x_j$
Polynomial	$K(x_i, x_j) = (x_i^T \cdot x_j + 1)^p$
Radial basis function (RBF)	$K(x_i, x_j) = exp(-\gamma \left\| x_i - x_j \right\|^2)$
Sigmoid	$K(x_i, x_j) = tanh(\beta_0 x_i^T \cdot x_j + \beta_1)$

2.6.2. Artificial neural network

ANN is a computational system inspired by neural networks in the human brain. Artificial neural network (ANN) can solve pattern or classification problems by storing knowledge gained from training on past data. ANN has an often-used architecture, namely single-layer perceptron and multilayer perceptron. Backpropagation ANN is one form of multilayer perceptron. This network model connects each unit in the input, hidden, and output layers.

Figure 4 is an example of a Backpropagation network architecture with *n* input (x_1, ..., x_i, ..., x_n) plus one bias unit, a hidden layer consisting of p units (z_1, …, z_j, …, z_p) plus with one bias unit, and there are m outputs (y_1, …, y_k, …, y_m). v_{ij} is the line weight from the input x_i to the hidden layer unit z_j, while v_{0j} is the line weight connecting the bias in the input layer to the hidden layer unit z_j. w_{jk} is the weight of the line from the hidden layer unit z_j to the output unit y_k, while w_{0k} is the weight of the line connecting the bias in the hidden layer to the output unit y_k.

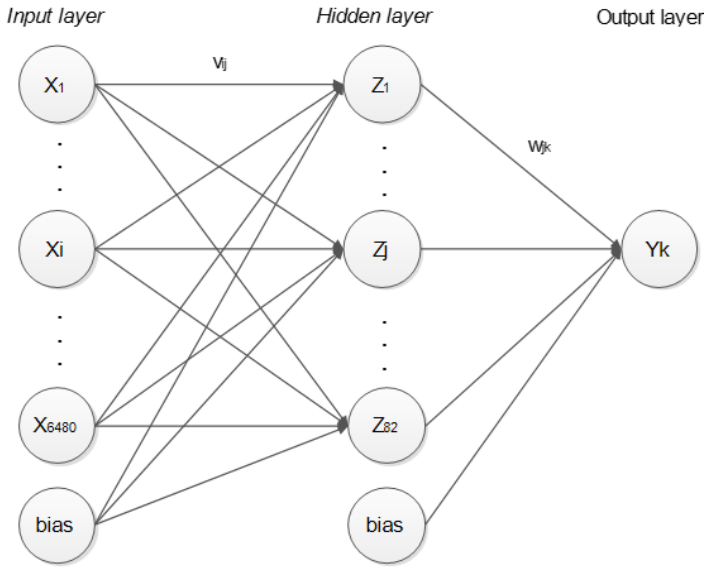

Figure 4. Backpropagation ANN architecture

2.7. Evaluation

The performance of a classification model can be measured using a confusion matrix. Confusion matrix is a table consisting of the number of test data predicted correctly and incorrectly by the classification model [24]. The confusion matrix for multiclass data is shown in Table 5.

Table 5. Confusion matrix for multiclass data

Confusion Matrix		Predicted		
		Class 1	Class 2	Class 3
Actual	Class 1	A	B	C
	Class 2	D	E	F
	Class 3	G	H	I

A: True Positive of Class 1,
D, G: False Positives of Class 1.
B, C: False Negatives of Class 1.
E, F, H, I : True Negatives of Class 1.

Based on the confusion matrix, accuracy, precision, recall (sensitivity), specificity, and f1-score are calculated using (3) to (6), respectively.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (3)$$

$$Precision = \frac{\text{TP}}{\text{TP+FP}} \quad (4)$$

$$Recall\ (sensitivity) = \frac{\text{TP}}{\text{TP+FN}} \quad (5)$$

$$F1 - score = \frac{2 \times recall \times precision}{recall + precision} \quad (6)$$

3. RESULTS AND DISCUSSION

The experiment used SVM and backpropagation methods with several scenarios. The first scenario was testing on imbalanced data. The second scenario was testing the data balance using SMOTE. Next, we compared the two scenarios. Apart from that, we also compared the proposed method and previous research.

3.1. Support vector machine

We tested with kernel and regularization parameter (C) variation in SVM. Table 6 shows the result of the experiment in imbalanced data. We can see that the best accuracy and f1-score reached 95.31% and 87.63%, respectively, using the RBF kernel. Table 7 shows the experiment results in balanced data, resulting in the best accuracy and F-1 score reaching 98.85% and 98.85%, respectively. So, we can conclude that adding the SMOTE method for data oversampling can increase the accuracy.

Figure 5 shows that using imbalanced data produces a relatively low fi-score value compared to the f1-score value produced when using balanced data. A low C value will make the margin wider and the hyperplane formed simpler, while a considerable C value will make the margin narrower and the hyperplane more sensitive. Thus, with a more considerable C value, the model will try to classify the data correctly and produce the minimum error value possible.

Table 6. SVM performance on imbalanced data

C	Linear		Polynomial		RBF		Sigmoid	
	Accuracy (%)	f1-score (%)	Accuracy (%)	f1-score (%)	Accuracy (%)	f1-score (%)	Accuracy (%)	f1-score (%)
0.01	82.75	30.19	90.03	57.22	82.75	30.19	82.75	30.19
0.1	82.75	30.19	90.67	58.23	88.29	52.07	82.75	30.19
1	90.82	58.57	90.51	64.63	91.14	58.88	79.27	29.47
10	90.67	58.08	92.41	78.79	91.94	74.49	72.46	27.99
100	90.83	59.13	93.20	80.34	94.31	87.63	71.35	27.74

Table 7. SVM performance on balanced data (SMOTE)

C	Linear		Polynomial		RBF		Sigmoid	
	Accuracy (%)	f1-score (%)	Accuracy (%)	f1-score (%)	Accuracy (%)	f1-score (%)	Accuracy (%)	f1-score (%)
0.01	67.81	68.72	84.96	84.95	72.34	72.70	23.39	16.92
0.1	74.38	74.86	95.35	95.34	85.66	85.82	10.07	8.09
1	94.58	94.58	97.64	97.62	97.00	97.00	7.20	5.71
10	96.05	96.03	98.34	98.33	98.47	98.46	14.47	14.52
100	97.00	96.98	98.66	98.65	98.85	98.85	14.40	14.45

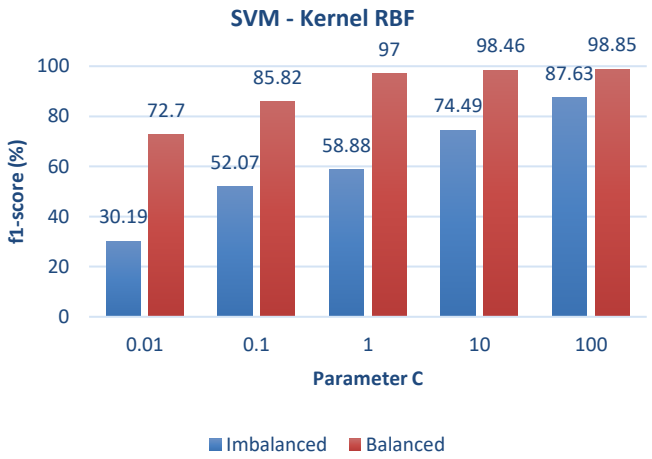

Figure 5. SVM-RBF kernel performance on the data

3.2. Backpropagation

Experiments were conducted to determine the classification performance of the Backpropagation ANN method based on the combination of hyperparameters, including the number of hidden layer neurons and learning rate. The results of Backpropagation ANN experiments on imbalanced and balanced data are shown in Tables 8 and 9, respectively. The best accuracy and F1-score on imbalanced data reached 91.55% and 59.40%, respectively. Table 10 shows that the best accuracy and f1-score of the Backpropagation classification with a balanced data model reached 94.90% and 94.89%, respectively.

Figure 6 shows the differences in f1-score values produced by the Backpropagation ANN method based on data usage and the number of hidden layer neurons. When using data balanced with SMOTE, the resulting f1 score is relatively equivalent to the accuracy value. These results show that the model works better when using balanced data.

Table 8. Backpropagation ANN result on imbalanced data

Hidden Neuron	Learning Rate	Accuracy (%)	f1-score (%)
3	0.1	91.42	59.18
	0.01	91.55	59.40
	0.001	91.42	59.18
4	0.1	91.41	58.94
	0.01	91.41	58.94
	0.001	91.41	58.94
5	0.1	91.55	59.14
	0.01	91.55	59.14
	0.001	91.55	59.14
6	0.1	91.55	59.34
	0.01	91.41	59.05
	0.001	91.41	59.05
7	0.1	91.55	59.13
	0.01	91.41	59.04
	0.001	91.41	59.04

Table 9. Backpropagation ANN result on balanced data

Hidden Neuron	Learning Rate	Accuracy (%)	f1-score (%)
3	0.1	93.69	93.68
	0.01	93.69	93.70
	0.001	93.63	93.63
4	0.1	94.14	94.14
	0.01	94.52	94.51
	0.001	94.90	94.89
5	0.1	93.88	93.88
	0.01	93.88	93.88
	0.001	94.01	94.01
6	0.1	93.37	93.38
	0.01	93.18	93.19
	0.001	93.56	94.57
7	0.1	93.37	93.37
	0.01	93.63	93.63
	0.001	93.24	93.25

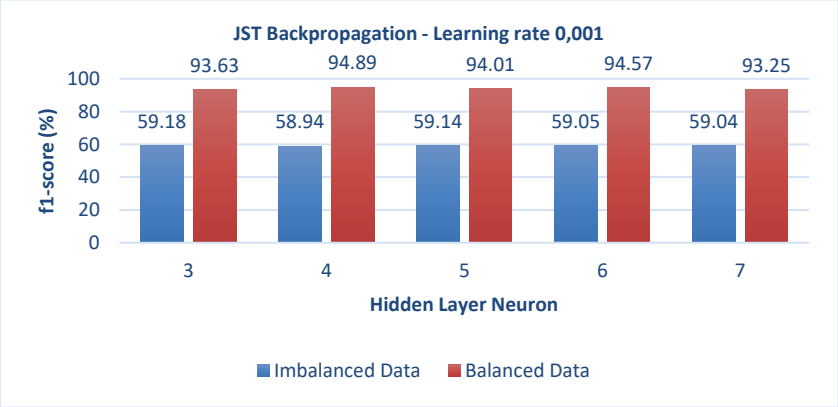

Figure 6. Backpropagation ANN performance on the data

3.3. Comparison of both method's best result

At this stage, the performance of the two methods is compared using data balanced with SMOTE by observing the values of accuracy, precision, recall, f1-score, specificity, and sensitivity. The evaluation value can be calculated through the confusion matrix. The evaluation matrix values generated from the best experiment of the two methods are shown in Table 10.

Table 10. Best result of both methods

Method	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-score (%)
SVM	98.85	98.87	98.85	99.42	98.85
Backpropagation	94.90	95.00	93.08	97.50	94.03

The higher precision value in the SVM method indicates that the model has a good level of accuracy in classifying non-diabetic, prediabetic, and diabetic class data into their respective class categories. In other words, the SVM method has fewer data prediction errors than the ANN Backpropagation method. The higher sensitivity in the SVM method indicates that the model has a good level of sensitivity in detecting each actual class category. The specificity of the SVM method is also higher than the Backpropagation ANN method, which indicates that the SVM method can avoid prediction errors better.

Finally, the proposed method was compared with the previous study, as shown in Table 11. [2] used many methods: Multinomial Logistic Regression, decision tree (DT), random forest (RF), Stochastic gradient Boosting, and naïve Bayes. Compare the result of the proposed method with the others shown in Table 5. We can see that the proposed method outperforms other methods on all performance measures.

Table 11. Comparison of the proposed method and previous study

Method	Accuracy (%)	Precision (%)	Recall (%)	f1-score (%)
Multinomial Logistic Regression [2]	86.70	70.00	70.00	70.00
DT [2]	95.07	98.12	78.00	89.67
RF [2]	90.64	75.00	78.00	76.40
Stochastic gradient Boosting [2]	97.04	98.85	81.10	89.00
Naïve Bayes [2]	93.10	89.00	71.86	79.50
Proposed	**98.85**	**98.87**	**98.85**	**98.85**

4. CONCLUSION

Based on the experiments that have been carried out, it can be concluded that using SMOTE to balance the data can improve the classification performance of SVM and backpropagation. The SVM method produced the best performance on the RBF kernel, namely getting an accuracy of 98.85%, f1-score of 98.85%, sensitivity of 98.85%, and specificity of 99.42%. The method is better when compared to the ANN Backpropagation method, which achieved an accuracy of 94.90%, f1-score of 94.03%, sensitivity of 93.08%, and specificity of 97.50% at hidden layer neuron = 4 and learning rate = 0.001. Despite the result, the current study was limited to several experiments on the model's hyperparameter. We hope that further experimentations on the hyperparameter variation will confirm our findings.

REFERENCES

[1] M. Roden and and G. I. Shulman. "The integrative biology of type 2 diabetes," *Nature*, vol. 576, pp. 51-60, 2019, doi: https://doi.org/10.1038/s41586-019-1797-8.

[2] M. R. Rajput and S. S. Khedgikar, "Diabetes prediction and analysis using medical attributes: A Machine learning approach," *Journal of Xi'an University of Architecture & Technology*, vol. XIV, no. 1, pp. 98–103, 2022, doi: 10.37896/JXAT14.01/314405.

[3] A. Kumari and R. Chitra, "Classification of diabetes disease using support vector machine," *International Journal of Engineering Research and Applications*, vol. 3, no. 2, pp. 1797–1801, 2013, [Online]. Available: https://www.researchgate.net/publication/320395340.

[4] N. E. Ordoñez-Guillen, J. L. Gonzalez-Compean, I. Lopez-Arevalo, M. Contreras-Murillo, and E. Aldana-Bobadilla, "Machine learning based study for the classification of Type 2 diabetes mellitus subtypes," *BioData Mining*, vol. 16, no. 1, 2023, doi: 10.1186/s13040-023-00340-2.

[5] A. S. Chauhan, M. S. Varre, K. Izuora, M. B. Trabia, and J. S. Dufek, "Prediction of diabetes mellitus progression using supervised machine learning," *Sensors*, vol. 23, no. 10, 2023, doi: 10.3390/s23104658.

[6] S. Mahajan, P. K. Sarangi, A. K. Sahoo, and M. Rohra, "Diabetes mellitus prediction using supervised machine learning techniques," in *2023 International Conference on Advancement in Computation and Computer Technologies, InCACCT 2023*, 2023, pp. 587 – 592, doi: 10.1109/InCACCT57535.2023.10141734.

[7] S. Gowthami, V. S. Reddy, and M. R. Ahmed, "Type 2 diabetes mellitus: early detection using machine learning classification," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 6, pp. 1191 – 1198, 2023, doi: 10.14569/IJACSA.2023.01406127.

[8] R. Panda, S. Dash, S. Padhy, and R. K. Das, "Diabetes mellitus prediction through interactive machine learning approaches," *Lecture Notes in Networks and Systems*, vol. 445, pp. 143 – 152, 2023, doi: 10.1007/978-981-19-1412-6_12.

[9] B. Shrestha *et al.*, "Enhancing the prediction of type 2 diabetes mellitus using sparse balanced SVM," *Multimedia Tools and Applications*, vol. 81, no. 27, pp. 38945 – 38969, 2022, doi: 10.1007/s11042-022-13087-5.

[10] D. A. Anggoro and D. Permatasari, "Performance comparison of the kernels of support vector machine algorithm for diabetes mellitus classification," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 1, pp. 580 – 585, 2023, doi: 10.14569/IJACSA.2023.0140163.

[11] B. Singh, J. Yadav, M. Singh, and A. Rani, "Deep neural network models for classification of significant attributes to predict Prediabetes Mellitus," 2023, doi: 10.1109/ICPCES57104.2023.10076156.

[12] C. Selvarathi and S. Varadhaganapathy, "Deep learning based cardiovascular disease risk factor prediction among type 2 diabetes mellitus patients," *Information Technology and Control*, vol. 52, no. 1, pp. 215 – 227, 2023, doi: 10.5755/j01.itc.52.1.32008.

[13] L. Testa, M. A. Caruana, M. Kontorinaki, and C. Savona-Ventura, *Predicting the risk of gestational diabetes mellitus through nearest neighbor classification*, vol. 9. 2022.

[14] O. O. S. Abe, O. O. Obe, O. K. Boyinbode, and N. Biodun Olagbuji, "Early gestational diabetes mellitus diagnosis using classification algorithms: an ensemble approach," 2023, doi: 10.1109/AFRICON55910.2023.10293603.

[15] R. Ashtagi, P. Dhumale, D. Mane, H. M. Naveen, R. V. Bidwe, and B. Zope, "IoT-based hybrid ensemble machine learning model for efficient diabetes mellitus prediction," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, no. 10s, pp. 714 – 726, 2023, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85171965186&partnerID=40&md5=45b837ea06854eb0e37f380a29114551.

[16] O. S. Zargar, A. Bhagat, and T. A. Teli, "Feature selection, importance and missing value imputation in diabetes mellitus prediction," in *Proceedings of the 17th INDIACom; 2023 10th International Conference on Computing for Sustainable Global Development, INDIACom 2023*, 2023, pp. 914 – 919, [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-

85159603898&partnerID=40&md5=1fb91b770a0efe4ec3e83690a84db067.
[17] V. S. Spelmen and R. Porkodi, "A Review on Handling Imbalanced Data," *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, Coimbatore, India, 2018, pp. 1-11, doi: 10.1109/ICCTCT.2018.8551020.
[18] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique nitesh," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002, [Online]. Available: https://arxiv.org/pdf/1106.1813.pdf%0Ahttp://www.snopes.com/horrors/insects/telamonia.asp.
[19] R. Pears, J. Finlay, and A. M. Connor, "Synthetic minority over-sampling technique (SMOTE) for predicting software build outcomes," in *Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE*, 2014, pp. 546–551.
[20] A. Fernadez, S. Garcia, F. Herrera, and N.V. Chawla, "SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary," *Journal Of Artificial Intelligence Research*, vol. 61, pp. 863-905, 2018, doi: https://doi.org/10.1613/jair.1.11192
[21] A. Rashid, "Diabetes Dataset," *Mendeley Data*, 2020. https://data.mendeley.com/datasets/wj9rwkp9c2/1.
[22] H. P. Vinutha, B. Poornima, and B. M. Sagar, "Detection of outliers using interquartile range technique from intrusion dataset," in *Information and Decision Sciences*, 2018, pp. 511–518.
[23] K. Johnson and R. Synovec, "Pattern recognition of jet fuels: Comprehensive GC × GC with ANOVA-based feature selection and principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 60, pp. 225–237, 2002, doi: 10.1016/S0169-7439(01)00198-8.
[24] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining, (First Edition)*. USA: Addison-Wesley Longman Publishing Co., Inc., 2005.
[25] Z. L. Wang, Z. G. Zhou, Y. Chen, X. T. Li, and Y. S. Sun, "Support vector machines model of computed tomography for assessing lymph node metastasis in esophageal cancer with neoadjuvant chemotherapy," *Journal of Computer Assisted Tomography*, vol. 41, no. 3, pp. 455–460, 2017, doi: 10.1097/RCT.0000000000000555.

BIOGRAPHIES OF AUTHORS

Angga Prastyo received a Bachelor's degree with honors (cumlaude) in Computer Engineering (S.Kom) from the Faculty of Sciences and Mathematics, Universitas Diponegoro, Indonesia in 2023. His main interests include machine learning, software development, and UI/UX design. He can be contacted at email: anggaprastyo010@gmail.com.

Sutikno received a Doctor of Philosophy (Ph.D) degree in computer science, faculty of mathematics and natural sciences from the University of Gadjah Mada, Indonesia. Now, he is an Assistant Professor at the Department of Informatics, Faculty of Sciences and Mathematics, University of Diponegoro. His research interests include machine learning, computer vision, and artificial intelligence. He can be contacted at email: sutikno@lecturer.undip.ac.id.

Khadijah received a Bachelor of Informatics Engineering (S.Kom) from the Universitas Diponegoro, Indonesia, in 2011 and a Master of Computer Science (MCs) from the Universitas Gadjah Mada, Indonesia, in 2014. She has been a lecturer with the Department of Informatics, Universitas Diponegoro, since 2014. Her main research interests are artificial intelligence and machine learning. She can be contacted at email: khadijah@live.undip.ac.id.