Support Vector Machine Optimization for Diabetes Prediction Using Grid Search Integrated with SHapley Additive exPlanations

M. Safii[1], **Husain**[2], **Khairan Marzuki**[2]
[1]STIKOM Tunas Bangsa, Pematangsiantar, Indonesia
[2]Universitas Bumigora, Mataram, Indonesia

Article Info

Article history:

Received May 28, 2025
Revised August 01, 2025
Accepted September 24, 2025

Keywords:

Classification;
Diabetes;
Grid Search;
Support Vector Machine;
SHapley Additive exPlanations.

ABSTRACT

The high number of diabetes mellitus sufferers has become a global health issue, and a scientific approach is needed to produce accurate and efficient diagnoses, which can then support decision-making in providing solutions for its management. The goal of this research is to develop a machine learning model that can accurately, efficiently, and transparently diagnose diabetes mellitus for use in clinical practice. This research method involves using the Support Vector Machine (SVM) algorithm, optimized with the Grid Search technique, and evaluated interpretively using the SHapley Additive exPlanations (SHAP) method. This research uses a secondary dataset consisting of the parameters Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, Body Mass Index, DiabetesPedigreeFunction, and Age. Data preprocessing was carried out by performing normalization using a standard scaler and dividing the data into training and testing sets. The results of this study show that the SVM model achieved an accuracy of 0.7532 with the optimal parameters C: 1, gamma: 0.01, and kernel: rbf. Using SHAP, the analysis shows that the parameters Glucose, Body Mass Index, and Age have a significant impact on the results of diabetes classification. The main finding of this study is that Support Vector Machine optimization with SHapley Additive exPlanations can deliver excellent performance in diabetes prediction while also enhancing model transparency. The study's implications suggest that the results can serve as a foundation for developing a medical diagnosis system that is straightforward, accurate, and easy to understand for diabetes mellitus.

Corresponding Author:

M. Safii, 085362234880,
Informatika, STIKOM Tunas Bangsa,
Pematangsiantar, Indonesia,
Email: m.safii@amiktunasbangsa.ac.id

How to Cite:
M. Safii, H. Husain, and K. Marzuki, ”Support Vector Machine Optimization for Diabetes Prediction Using Grid Search Integrated with SHapley Additive exPlanations”, *MATRIK: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, Vol. 25, No. 1, pp. 53-62, November, 2025.

1. INTRODUCTION

Data from the Ministry of Health shows that the number of people with diabetes in Indonesia has now reached 19.5 million. The number is predicted to soar to 28.5 million inhabitants by 2045. The latest data from the International Diabetes Federation (IDF) shows that in 2025, it is estimated that 11.1% of the adult population in Indonesia will have diabetes [1]. A significant increase in IDF projections indicates that by 2050, approximately 853 million people will have diabetes [1]. Projections until 2050 estimate an increase in the number of people with diabetes globally, which could have a major impact on health systems and the economy. Considering that the early symptoms of diabetes are often undetected, an accurate early diagnosis system becomes very urgent. In this context, machine learning algorithms such as Support Vector Machine (SVM) become a promising solution in building a data-driven prediction system based on patients' health history.

Various previous studies have utilized SVM algorithms for medical classification. Research by [2–4] shows that SVM has advantages in handling high-dimensional data and is resistant to overfitting, especially on small-sized datasets. Research conducted by [5] showed an average accuracy of 92.48% in classification using SVM. Parameters such as C (regularization) and gamma (RBF kernel parameter) greatly affect the model's performance [6–8]. For this reason, the Grid Search technique is used as a hyperparameter optimization method. Research by [9, 10] shows that the use of Grid Search can significantly improve the accuracy of diabetes predictions. In a study by [11] the highest accuracy of 98.88% was achieved using optimized SVM, while [12] that the SVM algorithm can achieve the highest accuracy of 98.88%, while gradient enhancement yields an accuracy of 98.08%. This research shows that optimizing the SVM model with GridSearchCV can improve the accuracy in predicting fake job recruitment. Other research by [13] the test results show that the SVM-SMOTE scenario produces the best accuracy. The SVM SMOTE scenario achieves an accuracy of 88% with an error rate of 12% using the RBF kernel, based on a 90:10 split of test and training data. This scenario also results in precision and sensitivity values of 0.880 each. The research results indicate that classifying factors is more accurate when using the support vector machine (SVM) method with inequality data handling (SMOTE). It can be concluded that the distribution of the test and training data affects the testing scenario. Other research by [14] The Grid Search-based Support Vector Machine Classifier is used as a prediction model for predicting diabetic disease. The grid search technique is used here to tune the hyperparameters and improve classification accuracy. The developed framework achieves an accuracy of 98.71% on the Vanderbilt diabetes data.

The difference between this research and previous studies is that most previous studies focused solely on classification accuracy, without considering model interpretability. This approach did not address the medical world's need for transparency in decision-making based on machine learning. In addition, although there are studies that combine SVM and SHAP, their application in the context of diabetes prediction on secondary data with systematic hyperparameter optimization and class balancing has not been widely encountered. The novelty of this research lies in the integration of the optimized Support Vector Machine method with Grid Search and data balancing using SMOTE. These techniques are then transparently explained using SHAP as an interpretation method based on Shapley's value theory in cooperative games. This approach allows for the development of a diabetes prediction system that is not only accurate but also reliable and understandable for medical personnel. This research aims to develop an SVM-based diabetes prediction model that has been optimized using Grid Search, with data balancing using SMOTE, and interpretation of prediction results through SHAP. The contribution of this research is to produce an accurate and transparent prediction system to support better medical decision-making.

2. RESEARCH METHOD

Based on the background above, this study will develop a classification model for diabetes using the Support Vector Machine algorithm, optimized with Grid Search and integrated with the SHAP (SHapley Additive exPlanations) method to enhance the model's prediction results. This research uses secondary data with eight input parameters and one label parameter consisting of: Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, Age, and Outcome. The methodology in this research is described starting from the data collection stage, data pre-processing, which divides the data into training and testing data, SVM optimization with Grid Search, and the integration process with the SHAP method. This stage of the methodology can develop an optimal predictive model that is visualized interpretively, as seen in the following Figure 1.

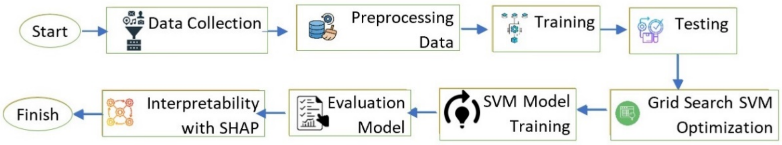
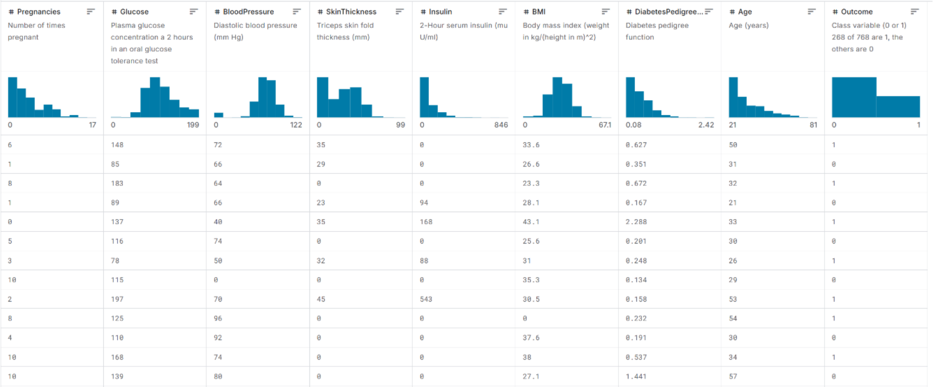


Figure 1. Research flow diagram

The image above is the flow of the research conducted in the development of the machine learning method, with the following stages:

2.1. Data Collection

The data source is secondary data consisting of eight parameters: Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, and Age. It also includes one label, which is the outcome, totaling 768 records. The dataset can be seen in the following Figure 2.



# Pregnancies	# Glucose	# BloodPressure	# SkinThickness	# Insulin	# BMI	# DiabetesPedigree...	# Age	# Outcome
Number of times pregnant	Plasma glucose concentration a 2 hours in an oral glucose tolerance test	Diastolic blood pressure (mm Hg)	Triceps skin fold thickness (mm)	2-Hour serum insulin (mu U/ml)	Body mass index (weight in kg/(height in m)^2)	Diabetes pedigree function	Age (years)	Class variable (0 or 1) 268 of 768 are 1, the others are 0
0 17	0 199	0 122	0 99	0 846	0 67.1	0.08 2.42	21 81	0 1
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1
1	89	66	23	94	28.1	0.167	21	0
0	137	40	35	168	43.1	2.288	33	1
5	116	74	0	0	25.6	0.201	30	0
3	78	50	32	88	31	0.248	26	1
10	115	0	0	0	35.3	0.134	29	0
2	197	70	45	543	30.5	0.158	53	1
8	125	96	0	0	0	0.232	54	1
4	110	92	0	0	37.6	0.191	30	0
10	168	74	0	0	38	0.537	34	1
10	139	80	0	0	27.1	1.441	57	0

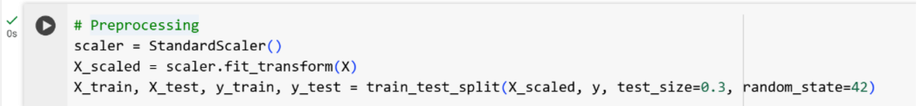
Figure 2. Dataset

2.2. Pre-Processing Data

After collecting secondary data, the next step is data preprocessing, which involves dividing the data into training and testing sets. Before splitting the data, feature scaling is performed using the script scaler = StandardScaler() so that each parameter has a mean of 0 and a standard deviation of 1. The population standard deviation serves to assess how the data is spread from its average value. To calculate it, Equation 2 provided below is used [15].

$$\sigma = \sqrt{\sigma(xi - \mu)^2}/N \tag{1}$$

Where α = Population standard deviation, N = Population Number, xi = each value from the population and μ = average population. The dataset used in this study is secondary data sourced from public repositories and consists of several clinical features relevant to diabetes prediction, with a total of 768 samples. The next step is to divide the data into training and testing sets with a common ratio of 70:30. This process includes feature scaling and standardization. Various steps are aimed at ensuring that the model is trained on representative data and tested on unseen data to measure its generalization capability. Proper preprocessing will help reduce bias, improve model stability, and ensure that prediction outcomes are more reliable. This process can be seen in the following Figure 3.



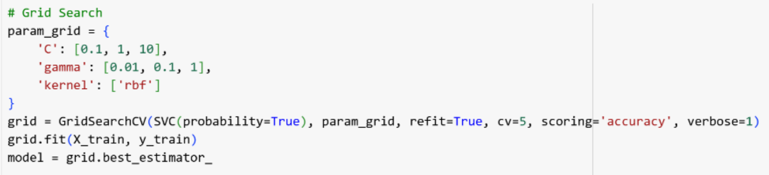
```
# Preprocessing
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3, random_state=42)
```

Figure 3. Data pre-processing

In the script, feature standardization is performed to ensure they have the same scale, which is mean = 0 and standard deviation = 1. Next, the data is divided into training and testing sets, with X_scaled being split into X_train and X_test. Test_size=0.3 means that 30% of the data is used for testing and 70% for training.

2.3. Grid Search Optimization SVM

The process of selecting parameter combinations affects the optimum model results so that the outcomes of the modeling can be trusted. In this study, the optimization process of the Support Vector Machine (SVM) algorithm uses grid search. The grid search method evaluates by combining parameters with cross-validation techniques to find the configurations of C, kernel, and gamma that provide optimum prediction results. Optimization with Grid Search can enhance the model's accuracy using the test data, as illustrated in Figure 4.



```
# Grid Search
param_grid = {
    'C': [0.1, 1, 10],
    'gamma': [0.01, 0.1, 1],
    'kernel': ['rbf']
}
grid = GridSearchCV(SVC(probability=True), param_grid, refit=True, cv=5, scoring='accuracy', verbose=1)
grid.fit(X_train, y_train)
model = grid.best_estimator_
```

Figure 4. Grid search optimization of SVM

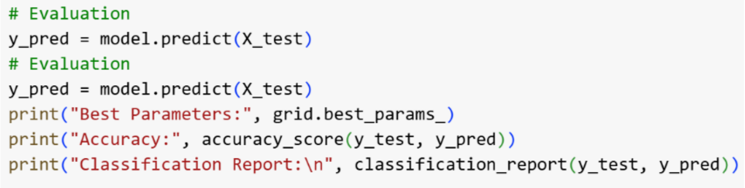
In the above script, parameter optimization on the SVM model is performed using Grid Search with cross-validation to find the best parameter configuration that yields the highest accuracy. The script tests nine parameter combinations, namely 3 C × 3 gamma × 1 kernel, and uses 5-fold cross-validation to evaluate each combination. In the line of code, model = grid. The best_estimator produces the highest accuracy on the training data based on cross-validation and is ready to be used for predictions or further evaluation.

2.4. SVM model training

After the hyperparameter optimization process is completed using Grid Search, the next step is to train the Support Vector Machine (SVM) model using the best parameters obtained. SVM maps input data to a higher-dimensional feature space using kernel functions such as the radial basis function (RBF) to handle data that is not linearly separable. The goal is to minimize classification errors while maximizing the margin, resulting in a model that generalizes well to new data. With this approach, SVM becomes an effective classification algorithm, especially in dealing with high-dimensional and complex datasets. The optimized SVM model is then trained on the training data that has gone through the pre-processing process. This training aims to create a classification model that can optimally separate the target classes based on the available features.

2.5. Model Evaluation

After the model training process is completed, the next step is to evaluate the model's performance in making predictions. The purpose of this model evaluation is to measure the model's ability to generalize to the test data. The metrics for evaluating this model are accuracy, precision, recall, f1-score, and confusion matrix. The values on this metric are considered to objectively assess the model's ability to solve classification problems in this study. This process can be seen in the following Figure 5.



```
# Evaluation
y_pred = model.predict(X_test)
# Evaluation
y_pred = model.predict(X_test)
print("Best Parameters:", grid.best_params_)
print("Accuracy:", accuracy_score(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))
```

Figure 5. Model evaluation script

The model testing step is carried out using the test data (X_test) to generate predictions (y_pred) with the prepared model. After the prediction stage, the best parameters from hyperparameter optimization obtained through the Grid Search process are displayed in the best_params_ attribute. Next, the model's performance is evaluated by calculating the accuracy score using the accuracy_score function, which compares the true labels (y_test) and the prediction results (y_pred). In addition, a more detailed classification report is presented through the classification_report function, which includes evaluation metrics such as precision, recall, f1-score, and support for each class. This evaluation aims to measure how well the model can generalize to data it has never encountered before.

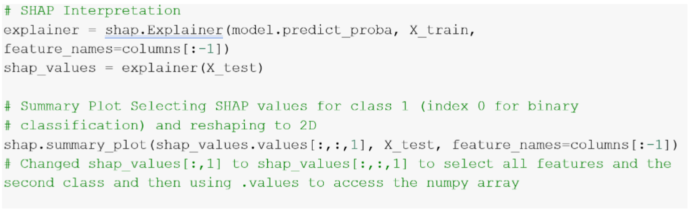
2.6. Interpretability with SHAP

After obtaining the results from the model evaluation, the next process is Interpretability using SHAP (SHapley Additive exPlanations). This method can help provide an understanding of the parameters that influence the prediction results, allowing end users to see their contribution. SHAP is based on the Shapley values from game theory that provide the contribution of each feature

i to the model's predictions. In this approach, the predicted value is generated from the linear sum of the bias and all the utilized features. Each feature is multiplied by the corresponding weight, and then the results are summed with the existing bias [16]. This calculation can be seen in the following Equation 2.

$$g(\acute{z}) = \phi 0 + \sum_{j=i}^{m} \phi_j \acute{z}_j \tag{2}$$

In that formula, $g(z\acute{z})$ represents the predicted output of the model, while $\varphi 0$ is the initial value (bias) of the prediction when there is no influence from any features. The symbol φj indicates the contribution value from feature j, and $\acute{z}j$ is the value of feature j in the data example byteing analyzed. The sum m represents the total number of features applied in the model. Where g is an explanatory model, $\acute{z} = ((\acute{z}_1 \ldots \acute{z}_M))^T \epsilon \{0,1\}^M$ is a coalition vector, M is the maximum coalition size, and ϕ_j is feature attribution for feature j Shapley value. The interpretation process begins with the initialization of the explainer object using shap.Explainer(), which utilizes the probabilistic prediction method (model.predict_proba) from the XGBoost model and the training data X_train as a reference. Next, SHAP values are calculated by applying the explainer to the test data X_test, resulting in an array of SHAP values (shap_values) for each feature and each observation. To visualize the results, the shap.summary_plot() function is used, which is specifically aimed at displaying the contributions of features to the positive target class (class 1). This process can be seen in the following Figure 6.



```
# SHAP Interpretation
explainer = shap.Explainer(model.predict_proba, X_train,
feature_names=columns[:-1])
shap_values = explainer(X_test)

# Summary Plot Selecting SHAP values for class 1 (index 0 for binary
# classification) and reshaping to 2D
shap.summary_plot(shap_values.values[:,:,1], X_test, feature_names=columns[:-1])
# Changed shap_values[:,1] to shap_values[:,:,1] to select all features and the
second class and then using .values to access the numpy array
```

Figure 6. Script interpretability with SHAP

To obtain an explanation of the understandable model, the SHAP (SHapley Additive exPlanations) method is used. The interpretation process starts by creating an Explainer object from the SHAP library, which is built based on the model's predicted probability function (model.predict_proba) and the training data (X_train). Feature names are added through the feature_names parameter, which in this context does not include the target column. The SHAP values are obtained for the test data (X_test) by calling the explainer object, which produces a shap_values object that indicates the contribution of each feature to the predictions made by the model.

3. RESULT AND ANALYSIS

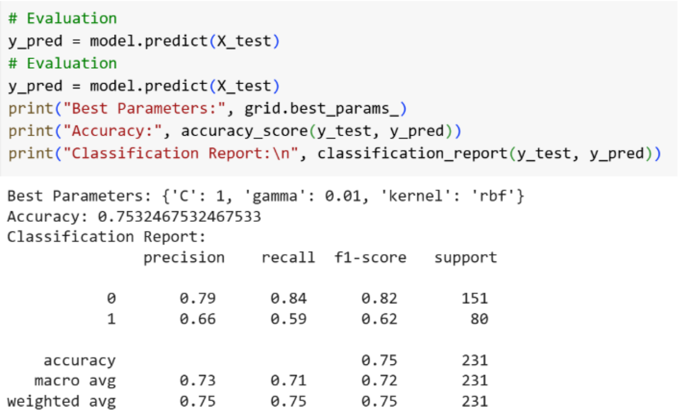
The results of the experiments and analyses conducted using the Support Vector Machine (SVM) method, optimized using the Grid Search technique, for diabetes prediction. The increase in accuracy with the Grid Search method shows that the SVM algorithm used can optimally solve the problems in this research. The optimization results are re-analyzed using SHAP (SHapley Additive exPlanations) techniques to provide interpretability to the model, allowing for the identification of dominant parameters that contribute to the prediction results. The output of this study was obtained from evaluating the SVM model using metrics such as accuracy, precision, recall, and F1-score, which were subsequently interpreted using the SHAP method. This process approach will produce accurate, optimal prediction outputs and ensure transparency in the model's decision results.

3.1. SVM Model Results

The classification process was performed using the Support Vector Machine (SVM) algorithm with a radial basis function (RBF) kernel approach to accommodate data that cannot be linearly separated. The dataset first underwent a preprocessing stage, including normalization using the standard scaler method to ensure all features are on a uniform scale. The SVM model was then optimized using Grid Search techniques to determine the best parameter combination, namely C=1, gamma=0.01, and kernel='rbf'. The evaluation results against the test data are shown in Table 1 below. The performance measurement results can be seen in Figure 7.

Table 1. Evaluation Results

Class	Precision	Recall	F1-Score	Support
0	0.79	0.84	0.82	151
1	0.66	0.59	0.62	80



```
# Evaluation
y_pred = model.predict(X_test)
# Evaluation
y_pred = model.predict(X_test)
print("Best Parameters:", grid.best_params_)
print("Accuracy:", accuracy_score(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))
```

```
Best Parameters: {'C': 1, 'gamma': 0.01, 'kernel': 'rbf'}
Accuracy: 0.7532467532467533
Classification Report:
               precision    recall  f1-score   support

           0       0.79      0.84      0.82       151
           1       0.66      0.59      0.62        80

    accuracy                           0.75       231
   macro avg       0.73      0.71      0.72       231
weighted avg       0.75      0.75      0.75       231
```

Figure 7. Classification report SVM

Based on the script in the image above, it can be explained that:

1. Accuracy: 0.7532467532467533. An accuracy of 75.3% indicates that out of all predictions made by the model on the test data, about 75.3% are correct (for both patients without diabetes and with diabetes).
2. The precision for class 0 (patients without diabetes) is 0.79, and the precision for class 1 (patients with diabetes) is 0.66.
3. The recall (sensitivity/TPR) for class 0, representing patients without diabetes, is 0.84, meaning the model successfully identified 84% of patients who truly do not have diabetes. Meanwhile, the recall for class 1, representing patients with diabetes, is 0.59, indicating that the model correctly detected only 59% of diabetic patients.
4. The F1-score for class 0 (patients without diabetes) is 0.82, and the F1-score for class 1 (patients with diabetes) is 0.62.

3.2. Integration with SHAP

To enhance the transparency and interpretability of the optimized Support Vector Machine (SVM) model, the next step is to integrate it with the SHAP (SHapley Additive exPlanations) method. SHAP is a model explanation technique based on Shapley value theory from game theory, which can measure the contribution of each feature individually to the prediction outcome. By integrating SHAP, not only is model performance prioritized, but it also provides a deep understanding of how input parameters influence the model's decision-making process. This makes it easier for practitioners and medical experts to interpret diabetes prediction results more transparently and accurately. The result of the integration can be seen in Figure 8 below.

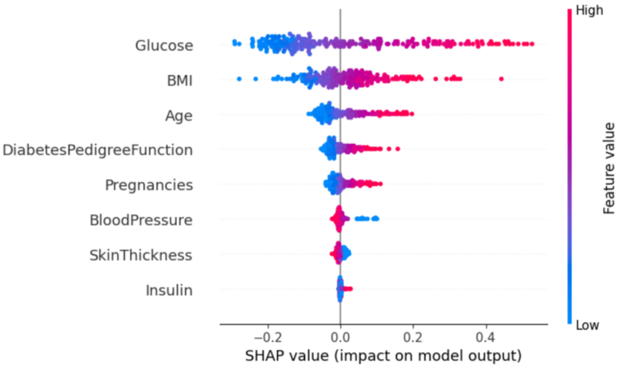


Figure 8. SHAP summary plot

Based on Figure 8, the contribution of each feature to the SVM model's predictions in diabetes classification illustrates how each parameter individually affects the model output. SHAP (SHapley Additive exPlanations) was used to analyze and improve the interpretability of the SVM model. The SHAP summary figure in Figure 8 above illustrates how each factor contributes to the model's predicted output for the classification of diabetes. The model's features are shown on the vertical axis, arranged according to their importance for making predictions. The SHAP values for each parameter are plotted against the model output in the horizontal line. Each parameter's values are shown by the colors blue and red, with blue denoting low numbers and red denoting high numbers. According to the graph, the study concludes that age, body mass index (BMI), and glucose have the greatest effects on the outcomes of diabetes prediction. Other parameters, such as blood pressure, pregnancy, and diabetes pedigree, have a moderate impact, while skin thickness and insulin have very little. This finding emphasizes that blood sugar levels, obesity conditions, and age are the main indicators in the early diagnosis of diabetes. The results of this study are consistent with or supported by previous studies, such as [17–19] which indicate that blood glucose, BMI, and age are dominant predictors in diabetes classification. Blood sugar and obesity in particular dominate the input to categorization judgments, according to research [20] that incorporates SHAP into machine learning algorithms. Therefore, the study's findings support the notion that medical prediction systems that combine optimized SVM with SHAP interpretation offer both accuracy and transparency.

4. CONCLUSION

This study fundamentally aims to combine optimization of predictive models and interpretability in the context of early diabetes diagnosis. The application of hyperparameter tuning techniques on the Support Vector Machine algorithm has been shown to improve classification accuracy while maintaining a balance of key evaluation metrics. Beyond merely enhancing performance, the integration of SHAP methods provides transparency in model decision-making, addressing the need for clarity in predictive logic within healthcare systems. The results of this study emphasize that accurate predictive modeling alone is not sufficient; understanding the factors that influence prediction outcomes is essential for clinical acceptance and user trust. For further development, this model can be expanded into external validity testing using real-world data from health institutions, and it should also be evaluated alongside medical practitioners to assess its clinical utility directly. Additionally, integrating it into visual interface-based decision support systems would be a strategic step to bridge the use of the model by non-technical medical personnel in the field.

5. ACKNOWLEDGEMENTS

Collate acknowledgements in a separate section at the end of the article before the references and do not, therefore, include them on the title page, as a footnote to the title or otherwise. List here those individuals who provided help during the research (e.g., providing language help, writing assistance, or proofreading the article, etc.).

6. DECLARATIONS

AI USAGE STATEMENT

The authors acknowledge that Artificial Intelligence tools, including ChatGPT developed by OpenAI, were utilized to support language refinement, grammar correction, and paraphrasing in the manuscript preparation process. The authors confirm that all ideas, data interpretations, and conclusions are their own and not generated by the AI tool.

AUTHOR CONTIBUTION

M. Safii was responsible for designing the research idea and objectives, as well as writing the initial draft of the article. Husain was responsible for developing and implementing the Grid Search method on the SVM algorithm, and analyzing the results. Khairan Marzuki was responsible for conducting data exploration, integrating SHAP for model interpretation, and revising and editing the article.

FUNDING STATEMENT

This research did not receive special funding from any funding agency, whether private or governmental.

COMPETING INTEREST

The author declares that the entire research, analysis, and manuscript preparation process was carried out without any conflict of interest that could affect the academic and scientific integrity of this article.

REFERENCES

[1] B. Hidayat, R. V. Ramadani, A. Rudijanto, P. Soewondo, K. Suastika, and J. Y. Siu Ng, “Direct Medical Cost of Type 2 Diabetes Mellitus and Its Associated Complications in Indonesia,” *Value in Health Regional Issues*, vol. 28, pp. 82–89, March,2022, https://doi.org/10.1016/j.vhri.2021.04.006.

[2] I. B. Sya’idah, S. Surono, and G. Khang Wen, “Dynamic Weighted Particle Swarm Optimization - Support Vector Machine Optimization in Recursive Feature Elimination Feature Selection,” *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 23, no. 3, pp. 627–640, 2024, https://doi.org/10.30812/matrik.v23i3.3963.

[3] R. Guido, S. Ferrisi, D. Lofaro, and D. Conforti, “An Overview on the Advancements of Support Vector Machine Models in Healthcare Applications: A Review,” vol. 15, no. 4, pp. 1–36, April, 2024, https://doi.org/10.3390/info15040235.

[4] Q. Wang, “Support Vector Machine Algorithm in Machine Learning,” in *2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, August, 2022, pp. 750–756, https://doi.org/10.1109/ICAICA54878.2022.9844516.

[5] N. W. S. Saraswati and I. G. A. A. Diatri Indradewi, “Recognize The Polarity of Hotel Reviews using Support Vector Machine,” *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 22, no. 1, pp. 25–36, 2022, https://doi.org/10.30812/matrik.v22i1.1848.

[6] G. Anyanwu, C. Nwakanma, J. M. Lee, and D.-S. Kim, “Optimization of RBF-SVM Kernel using Grid Search Algorithm for Detecting DDoS Attack in SDN-Based VANET,” *IEEE Internet of Things Journal*, vol. 10, no. 10, pp. 8477–8490, may 2023, https://doi.org/10.1109/JIOT.2022.3199712.

[7] C.-A. Tsai and Y.-J. Chang, “Efficient Selection of Gaussian Kernel SVM Parameters for Imbalanced Data,” pp. 1–13, February,2023, https://doi.org/10.3390/genes14030583.

[8] I. S. Al-Mejibli, J. K. Alwan, and D. H. Abd, “The effect of gamma value on support vector machine performance with different kernels,” *International Journal of Electrical and Computer Engineering*, vol. 10, no. 5, pp. 5497–5506, 2020, https://doi.org/10.11591/IJECE.V10I5.PP5497-5506.

[9] Y. Zhao, W. Zhang, and X. Liu, “Grid search with a weighted error function: Hyper-parameter optimization for financial time series forecasting,” *Applied Soft Computing*, vol. 154, p. 111362, March, 2024, https://doi.org/10.1016/j.asoc.2024.111362.

[10] D. Belete and M. D H, “Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results,” *International Journal of Computers and Applications*, vol. 44, pp. 1–12, sep 2021, https://doi.org/10.1080/1206212X.2021.1974663.

[11] D. Anggreani, Hamdani, and Nurmisba, “Grid Search Hyperparameter Analysis in Optimizing The Decision Tree Method for Diabetes Prediction,” *Indonesian Journal of Data and Science*, vol. 5, pp. 190–197, dec 2024, https://doi.org/10.56705/ijodas.v5i3.190.

[12] R. Rofik, R. A. Hakim, J. Unjung, B. Prasetiyo, and M. A. Muslim, “Optimization of svm and gradient boosting models using gridsearchcv in detecting fake job postings,” *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 23, no. 2, pp. 419–430, 2024, https://doi.org/10.30812/matrik.v23i2.3566.

[13] F. Yunita Sari, M. S. Kuntari, H. Khaulasari, and W. Ari Yati, “Comparison of Support Vector Machine Performance with Oversampling and Outlier Handling in Diabetic Disease Detection Classification,” *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 22, no. 3, pp. 539–552, 2023, https://doi.org/10.30812/matrik.v22i3.2979.

[14] S. Amutha and J. Raja Sekar, “An Optimized Framework for Diabetes Mellitus Diagnosis Using Grid Search Based Support Vector Machine BT - Computer, Communication, and Signal Processing. AI, Knowledge Engineering and IoT for Smart Systems,” pp. 153–167, 2023.

[15] M. Rakrak, “Exploring Variability in Data: The Role of Range, Variance, and Standard Deviation,” *International Journal of Multidisciplinary Research and Analysis*, vol. 08, no. 03, pp. 1327–1331, mar 2025, https://doi.org/10.47191/ijmra/v8-i03-47.

[16] D. Fryer, I. Strümke, and H. Nguyen, "Shapley Values for Feature Selection: The Good, the Bad, and the Axioms," *IEEE Access*, vol. 9, pp. 144 352–144 360, 2021, https://doi.org/10.1109/ACCESS.2021.3119110.

[17] S. Ahmed, M. S. Kaiser, M. Hossain, and K. Andersson, "A Comparative Analysis of LIME and SHAP Interpreters with Explainable ML-Based Diabetes Predictions," *IEEE Access*, vol. PP, p. 1, jan 2024, https://doi.org/10.1109/ACCESS.2024.3422319.

[18] M. Islam, H. R. Rifat, M. S. Shahid, A. Akhter, M. A. Uddin, and K. M. Mohi Uddin, "Explainable Machine Learning for Efficient Diabetes Prediction Using Hyperparameter Tuning, SHAP Analysis, Partial Dependency, and LIME," *Engineering Reports*, vol. 7, dec 2024, https://doi.org/10.1002/eng2.13080.

[19] S. U. Hassan, S. J. Abdulkadir, M. S. M. Zahid, and S. M. Al-Selwi, "Local interpretable model-agnostic explanation approach for medical imaging analysis: A systematic literature review," *Computers in Biology and Medicine*, vol. 185, p. 109569, 2025, https://doi.org/10.1016/j.compbiomed.2024.109569.

[20] S. I. Oktora, D. Matualage, K. A. Notodiputro, and B. Sartono, "Data-Driven Insights Into Underdeveloped Regencies: SHAP-Based Explainable Artificial Intelligence Approach," *International Journal of Artificial Intelligence Research; Vol 9, No 1 (2025): June*, vol. 9, pp. 1–13, 2025, https://doi.org/10.29099/ijair.v9i1.1399.

[This page intentionally left blank.]