United International Journal for Research & Technology
Volume 06, Issue 09, 2025 | Open Access | ISSN: 2582-6832

Comparative Analysis of Random Forest and Support Vector Machine for Classifying Pima Indians Diabetes Dataset

Johanes Eka Priyatma[1] and Mikael Raditya Agung Sasmita[2]
[1]Informatic Department of Sanata Dharma University
[2]Informatic Graduate of Sanata Dharma University

***Abstract*—** This study explored how well two machine learning algorithms—Random Forest (RF) and Support Vector Machine (SVM)—performed in classifying the Pima Indians Diabetes Dataset, which is used to predict the likelihood of individuals developing diabetes. To ensure a fair and reliable comparison, both models were evaluated using 10-fold cross-validation. Their effectiveness was measured through key classification metrics: accuracy, precision, recall, and F1-score. The results highlighted Random Forest as the more stable and reliable model, achieving an average accuracy of 76.3% and consistently strong results across all folds. In contrast, while the SVM with a polynomial kernel delivered slightly better precision (74.57%), it fell short in terms of overall accuracy, recall, and F1-score when compared to Random Forest. Ultimately, Random Forest proved to be better at identifying true positive cases and handling variations in the data, making it a stronger candidate for classifying health-related datasets like this one. That said, with further tuning of its parameters, SVM still holds promise as a competitive alternative.

***Keywords*—** Random Forest, Support Vector Machine, Diabetes Classification, Pima Dataset, Machine Learning.

I. INTRODUCTION

Classification techniques play a pivotal role in modern data science, enabling the categorization of data into predefined labels, particularly when dealing with complex or unlabeled datasets. In healthcare, classification models are especially valuable for diagnostic predictions and disease risk assessments. Among the most prominent machine learning algorithms used for classification tasks are Random Forest (RF) and Support Vector Machine (SVM), both known for their robustness and predictive power.

RF is an ensemble learning method that builds multiple decision trees and merges their outcomes to improve classification accuracy and control overfitting. SVM, on the other hand, is a powerful supervised learning algorithm that works well in high-dimensional spaces by identifying optimal hyperplanes for data separation. These algorithms have been widely applied across domains, including finance, image recognition, and biomedical data analysis (Mishra et al., 2023).

Despite the success of both algorithms, previous research reveals inconsistent findings regarding their comparative performance. For instance, Osisanwo et al. (2017) found that SVM outperformed other classifiers in terms of accuracy and precision. Conversely, studies by Tigga and Garg (2020) and Nahzat and Yağanoğlu (2021) concluded that RF produced better results, especially when dealing with imbalanced or noisy data. Meanwhile, Lyngdoh et al. (2022) applied both methods and found that neither consistently achieved top performance, indicating that algorithm efficacy may depend on dataset-specific characteristics.

Such inconsistencies raise critical questions about the contextual factors influencing classification accuracy. Differences in feature selection, preprocessing techniques, parameter tuning, and dataset characteristics often lead to divergent outcomes (Arshad et al., 2023). Therefore, a head-to-head comparison of RF and SVM on a consistent dataset with standardized evaluation metrics is warranted.

The Pima Indians Diabetes Database is a widely recognized benchmark dataset in the field of medical informatics. It comprises several relevant clinical features such as glucose level, insulin concentration, body mass index, and blood pressure, all of which are critical indicators of diabetes risk. Its broad usage in machine learning research makes it an ideal candidate for testing and comparing classification algorithms under uniform conditions (Alam et al., 2022).

Given the contradictions in prior findings and the medical relevance of the Pima dataset, this study aims to rigorously evaluate and compare the performance of RF and SVM using standardized metrics such as accuracy, precision, recall, and F1-score. The findings will contribute to a clearer understanding of each algorithm's strengths and limitations in the context of health data classification, potentially guiding future applications in clinical decision support systems

II. BRIEF LITERATURE REVIEW

The classification of medical datasets, particularly those related to diabetes, has gained considerable attention due to the growing prevalence of the disease. Among widely used datasets, the Pima Indians Diabetes Dataset (PID) has become a standard benchmark for evaluating machine learning (ML) models in medical diagnosis tasks (Uddin et al., 2019). Two commonly applied classification algorithms in this context are Random Forest (RF) and Support Vector Machine (SVM).

Random Forest, an ensemble-based method, is known for its robustness against overfitting, its capacity to handle high-dimensional data, and its interpretability via feature importance (Zhou et al., 2020). SVM, by contrast, is appreciated for its solid theoretical foundation and effectiveness in handling both linear and nonlinear classification problems using kernel functions (Sharma & Khanna, 2021). Comparative studies have demonstrated that both methods perform competitively on PID, but performance often depends on preprocessing steps and hyperparameter tuning.

Several recent studies have examined the application of RF and SVM to PID. For instance, El-Jerjawi and Abu-Naser (2018) reported higher accuracy using RF over SVM in their classification experiments, attributing the results to RF's ensemble strength. Conversely, Patel et al. (2021) demonstrated that SVM outperformed RF when the dataset was normalized and optimized using grid search. Feature selection also plays a crucial role in improving model accuracy. Techniques like Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA) have shown to enhance both RF and SVM performance (Jain & Choudhary, 2022).

Additionally, hybrid models combining RF or SVM with optimization algorithms like Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) have been proposed to improve classification outcomes (Ahmed et al., 2022). These methods report modest increases in performance, suggesting that algorithmic enhancements may benefit both models similarly.

Furthermore, deep learning methods are increasingly used as benchmarks, but RF and SVM remain highly relevant due to their simplicity, lower computational demands, and strong performance on structured tabular datasets like PID (Mohammad et al., 2020). When interpretability is prioritized, RF tends to be favored due to its model transparency (Hosseini et al., 2022).

III. RESEARCH METHOD

A. Data Collection Method

The dataset used in this study was sourced from the Pima Indians Diabetes Database, which was downloaded from Kaggle on November 14, 2024. Kaggle is a well-known platform that offers a wide range of high-quality datasets frequently used in data science and machine learning research. This particular dataset was chosen because it closely aligns with the goals of the study—evaluating the effectiveness of classification algorithms in predicting medical conditions, specifically diabetes.

The data collection process began with identifying a dataset that fit the scope and requirements of the research. After selecting the Pima Indians Diabetes Database, we carried out a thorough review to confirm that the dataset was complete and properly structured. This included checking the number of records, the availability of all required features, and ensuring the data was formatted correctly for machine learning tasks.

The dataset contains 768 records, each representing a medical profile of a Pima Indian woman aged 21 or older. It includes eight input features such as glucose level, blood pressure, insulin level, and body mass index, along with one output label indicating whether the individual was diagnosed with diabetes. A breakdown of these attributes is provided in Table 1.

The Pima Indians Diabetes Dataset includes a range of medical features that are commonly associated with diabetes risk, making it a valuable resource for predictive modelling in healthcare research.

Each attribute in the dataset reflects a specific health indicator that can contribute to identifying the likelihood of diabetes in an individual. Below is a brief overview of these features and their clinical significance.

***Table 1.** Attribute List in Dataset*

No	Attribute	Description
1	Pregnancies	Number of pregnancies
2	Glucose	Plasma glucose concentration 2 hours after an oral glucose tolerance test

3	Blood Pressure	Diastolic blood pressure (mmHg)
4	Skin Thickness	Triceps skinfold thickness (mm)
5	Insulin	Serum insulin concentration 2 hours after glucose ingestion (µU/ml)
6	BMI	Body mass index (kg/m²)
7	Diabetes Pedigree Function	Family history function showing genetic likelihood of diabetes
8	Age	Age of the individual (years)
9	Outcome	Target variable (0 or 1)

a. Pregnancies: This attribute records the number of times a woman has been pregnant. Studies have shown that women with a history of gestational diabetes are at a higher risk of developing type 2 diabetes later in life (Rayanagoudar et al., 2016).

b. Glucose: Refers to the plasma glucose concentration measured two hours after an oral glucose tolerance test. Elevated postprandial glucose levels are a strong indicator of impaired glucose metabolism and an early sign of diabetes (Tabák et al., 2012).

c. Blood Pressure: This is the diastolic blood pressure recorded in millimeters of mercury (mmHg). Hypertension is commonly linked with insulin resistance and an increased risk of developing type 2 diabetes (Cheung & Li, 2012).

d. Skin Thickness: Measures the triceps skinfold thickness in millimeters. Abnormal skin thickness may indicate metabolic irregularities, including higher fat deposits and insulin resistance, particularly in younger patients (Asif, 2021).

e. Insulin: Indicates the serum insulin level two hours after glucose ingestion. Both hypoinsulinemia and hyperinsulinemia are related to diabetes progression, depending on how the body responds to glucose intake (Pankow et al., 2015).

f. Body Mass Index (BMI): A calculation based on weight and height (kg/m²). A BMI over 25 is considered overweight and significantly raises the risk of type 2 diabetes due to increased fat accumulation and insulin resistance (Al-Goblan et al., 2014).

g. Diabetes Pedigree Function: A value indicating the strength of family history of diabetes. A higher pedigree score reflects a stronger genetic predisposition to the disease (Ali, 2013).

h. Age: Records the participant’s age in years. The likelihood of developing type 2 diabetes increases significantly after the age of 45, making age an important non-modifiable risk factor (Zhuo et al., 2014).

The dataset comprises 768 entries, with two outcome classes: 268 individuals diagnosed with diabetes and 500 without the condition. These attributes were specifically selected due to their well-established relevance in diabetes research, offering a comprehensive foundation for training machine learning models to predict disease onset accurately.

B. Random Forest Modeling

Random Forest (RF) is a powerful ensemble learning algorithm widely used for both classification and regression tasks. It operates by constructing a large number of decision trees during training and aggregating their outputs—using majority voting for classification or averaging for regression—to produce the final prediction. As introduced by Breiman (2001), each tree in a Random Forest is trained on a random subset of the dataset, and at each node, a random subset of features is selected to determine the best split. This process introduces randomness that reduces the correlation among trees, improving overall model diversity and minimizing the risk of overfitting.

One of RF’s core strengths lies in its ability to combine the predictions of multiple relatively weak models (individual trees) into a single strong model. Its performance is largely influenced by two key factors: the strength of the individual trees and the degree of correlation between them. Less correlation and stronger individual learners result in better overall performance (Breiman, 2001). In addition, Random Forest is highly flexible and can manage high-dimensional data with many features. It also offers built-in tools to assess feature importance, which helps in understanding which input variables have the greatest influence on the predictions—a useful feature for many applied domains including healthcare and finance.

In this study, RF modelling was conducted using the Random Forest Classifier module from the sklearn.ensemble package in Python's scikit-learn library. This implementation is well-suited for efficient and customizable training of Random Forest models. Key hyperparameters used during model development included: n_estimators (number of trees in the forest), max_features (number of features to consider when looking for the best split), max_depth (maximum depth of each tree), min_samples_split (minimum number of samples required to split an internal node), and min_samples_leaf (minimum number of samples required to be at a leaf node). Tuning these parameters helped optimize model accuracy and generalizability for the dataset at hand.

The number of trees was set to 100, as recommended by Breiman (2001), to produce a stable error estimate. For the number of random features selected at each node, this study used two configurations:

$F = 1$ and $F = \mathrm{int}(\log 2(M) + 1)$,

where M is the total number of features in the dataset. With $M = 8$, the value of F was calculated as follows:

$F = \mathrm{int}(\log 2(8) + 1) = 4$.

Other parameters, such as max_depth, were left at their default value (none), meaning there was no limit on the depth of the trees. The min_samples_split was set to 2, so a node would split if it contained at least two samples, and min_samples_leaf was set to 1, meaning each leaf must contain at least one sample.

The dataset was divided into 10 folds using 10-fold cross-validation. In each iteration, 9 folds were used for training, and the remaining fold was used for evaluation. This process was repeated until each fold was used as a test set once. The evaluation results from the 10 folds were averaged for each configuration of F. The best average evaluation result was used as the basis for further analysis, ensuring the selected configuration of F was most suitable for the dataset's characteristics.

C. Support Vector Machine Modeling

Support Vector Machine (SVM) is a machine learning algorithm designed to separate two classes using an optimal hyperplane that maximizes the margin between the classes. The optimal hyperplane is defined as:

$$w \cdot x + b = 0$$

where w is the weight vector, x is the feature vector, and b is the bias. The optimal margin, which indicates the model's generalization to new data, is formulated as (Cortes & Vapnik, 1995) :

$$M = \frac{2}{\|w\|}$$

For data that cannot be linearly separated, SVM uses a kernel function to map the data into a higher-dimensional space, enabling linear separation in that space (Schölkopf & Smola, 2002). Some common kernel functions include:

a. Linear Kernel:

$$k(x, x') = \langle x, x' \rangle$$

b. Polynomial Kernel:

$$k(x, x') = (\langle x, x' \rangle + c)^d$$

c. Gaussian RBF Kernel:

$$k(x, x') = exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right)$$

SVM model training in this study was conducted using the scikit-learn library through the SVC (Support Vector Classifier) class. The model was developed to evaluate three types of kernels: linear, polynomial, and Gaussian (RBF).

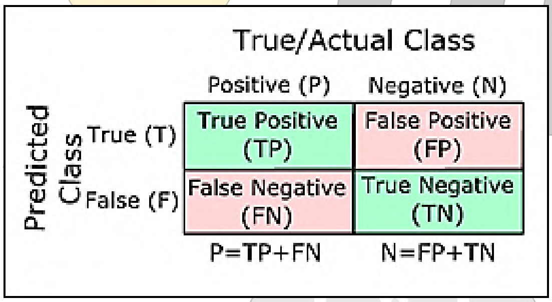
Key parameters used in SVM model training included the C and Gamma parameters. The C parameter controlled the balance between the margin's width and classification errors on the training data. The default value for C was 1.0, which provided a good balance between model complexity and low classification error. Meanwhile, Gamma controlled the contribution of each data point in determining the decision boundary. The default value for Gamma was 'scale', which was computed as $\frac{1}{n\_features}$, where n_features was the number of features in the dataset. A higher Gamma value resulted in a more complex decision boundary, while a lower value simplified the model.

The dataset was divided into 10 folds using 10-fold cross-validation. In each iteration, 9 folds were used for training, and one fold was used for testing. The training process involved selecting a kernel, and evaluation was conducted to measure the model's performance on the test fold. This process was repeated until each fold was used as a test set once. The performance of the three kernels was evaluated, and the kernel with the best average accuracy was selected for further analysis. This ensured that the chosen kernel aligned with the dataset's characteristics, resulting in an optimal SVM model.

D. Evaluation Metrics and Performance Measurement

The performance evaluation of the classification models in this study, specifically Random Forest (RF) and Support Vector Machine (SVM), was conducted using four widely recognized metrics in machine learning: accuracy, precision, recall, and F1-score. These metrics were applied to assess the models' ability to classify unseen data effectively during the testing phase. The evaluation process was designed to provide a comprehensive analysis of the model's performance.

According to Tharwat (2021), the purpose of a classification algorithm is to learn from training data and predict class labels for unseen data during testing. However, testing errors cannot be directly estimated because the actual class labels for the testing samples are not known. To address this, a validation phase was used to evaluate the performance of the trained models. Validation methods play a crucial role in determining the classification accuracy and reliability of the constructed models.



Predicted Class		True/Actual Class	
		Positive (P)	Negative (N)
	True (T)	True Positive (TP)	False Positive (FP)
	False (F)	False Negative (FN)	True Negative (TN)
		P=TP+FN	N=FP+TN

Figure 1. *Confusion Matrix 2x2*

The evaluation was based on the confusion matrix (Figure 1), which is used to calculate key classification metrics. In binary classification, the positive class was represented as P, and the negative class as N. Predictions were categorized into four outcomes: True Positive (TP), False Negative (FN), True Negative (TN), and False Positive (FP). These outcomes provided the foundation for calculating the following metrics (Tharwat, 2021):

a. Accuracy: Accuracy measured the proportion of correctly classified samples to the total number of predictions. It was calculated as follow:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

b. Precision: Precision quantified the proportion of correctly predicted positive samples out of all predicted positive samples. It was calculated as:

$$Precision = \frac{TP}{TP + FP}$$

c. Recall, also referred to as sensitivity, measured the proportion of actual positive samples that the model correctly identified. It was calculated as:

$$Recall = \frac{TP}{TP + FN}$$

d. The F1-score provided a balance between precision and recall by computing their harmonic mean, making it especially useful for imbalanced datasets. It was calculated as:

$$F1 - score = \frac{2TP}{2TP + FP + FN}$$

These metrics were critical in evaluating the performance of the models, especially in scenarios where class imbalance might skew the results if accuracy alone was considered.

E. Experimental Setup

The dataset was divided into ten folds for cross-validation. Each fold served as the testing data once, while the remaining nine folds were used for training. For the RF model, features were randomly selected based on predefined values (F = 1 and F = 4), and for the SVM model, three kernel types (linear, polynomial, and radial basis function) were tested. After training, the models were evaluated on the testing fold, and the metrics were calculated. This process was repeated for each fold, and the average metric values across the ten folds were computed to estimate overall model performance.

The metrics were calculated for both algorithms under various parameter settings, allowing for a detailed comparison of their classification performance. This experimental design ensured that the evaluation process was robust and reliable.

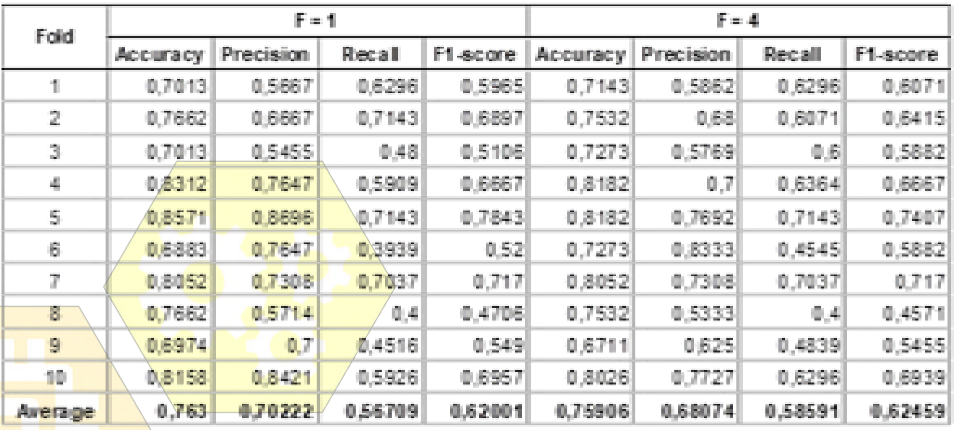
IV. ANALYSIS AND RESULTS

A. Results of the Random Forest Algorithm Implementation

The Random Forest model was tested using two configurations of the max_features parameter (F=1 and F=4) on the diabetes dataset, aiming to evaluate the model's performance across different feature configurations. Each configuration was tested using 10-fold cross-validation to ensure the stability of the results. The evaluation results showed that RF achieved a good accuracy, with an average accuracy of 76.3% for F=1 and a slightly lower accuracy of 75.9% for F=4.

Moreover, RF also showed precision values of 70.22% for F=1, while for F=4, it slightly decreased to 68.1%. The recall for F=1 was recorded at 56.71%, indicating that a significant number of positive samples were not detected. However, using F=4, the recall slightly increased to 58.6%, although the precision slightly decreased. A detailed evaluation of the results is shown in the following Table 2.

Table 2. *Random Forest Model Evaluation Result*



Fold	F = 1				F = 4			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
1	0,7013	0,5667	0,6296	0,5965	0,7143	0,5862	0,6296	0,6071
2	0,7662	0,6667	0,7143	0,6897	0,7532	0,68	0,6071	0,6415
3	0,7013	0,5455	0,48	0,5106	0,7273	0,5769	0,6	0,5882
4	0,8312	0,7647	0,5909	0,6667	0,8182	0,7	0,6364	0,6667
5	0,8571	0,8696	0,7143	0,7843	0,8182	0,7692	0,7143	0,7407
6	0,6883	0,7647	0,3939	0,52	0,7273	0,8333	0,4545	0,5882
7	0,8052	0,7308	0,7037	0,717	0,8052	0,7308	0,7037	0,717
8	0,7662	0,5714	0,4	0,4706	0,7532	0,5333	0,4	0,4571
9	0,6974	0,7	0,4516	0,549	0,6711	0,625	0,4839	0,5455
10	0,8158	0,8421	0,5926	0,6957	0,8026	0,7727	0,6296	0,6939
Average	0,763	0,70222	0,56709	0,62001	0,75906	0,68074	0,58591	0,62459

From the table above, we can see that Random Forest with the F=1 configuration provides more stable results, with higher accuracy compared to F=4. However, despite F=4 slightly decreasing the accuracy, there are small differences in recall and precision that should be considered.

Overall, Random Forest demonstrates stable results and performs better in terms of accuracy compared to other models. The F=1 configuration seems to provide more optimal performance, as evidenced by its better precision and F1-score compared to F=4.

B. Results of the Support Vector Machine Algorithm Implementation

The SVM algorithm was tested using three types of kernels: linear, polynomial, and RBF Gaussian. Each kernel was tested with 10-fold cross-validation and evaluated using the same metrics as RF: accuracy, precision, recall, and F1-score.

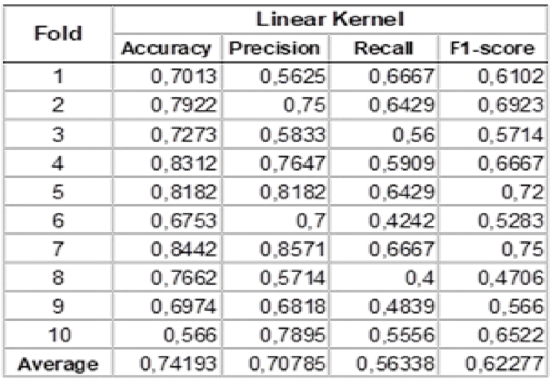
a. Linear Kernel

For the linear kernel, the SVM model achieved an average accuracy of 74.19%, with a precision of 70.78% and a lower recall of 56.34%.

This indicates that while the model performed well in classifying positive samples, a significant number of samples were missed as negatives.

The F1-score for the linear kernel was 62.28%, showing that the model achieved a balanced performance in terms of precision and recall. The results from the linear kernel evaluation are shown in Table 3.

Table 3. *SVM Model Evaluation Results with Linear Kernel*

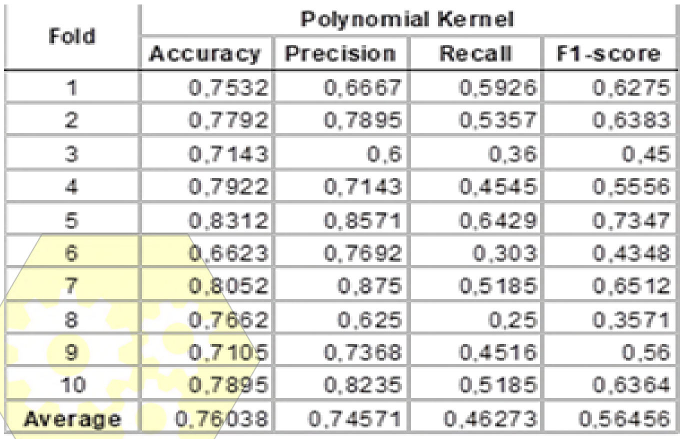


Fold	Linear Kernel			
	Accuracy	Precision	Recall	F1-score
1	0,7013	0,5625	0,6667	0,6102
2	0,7922	0,75	0,6429	0,6923
3	0,7273	0,5833	0,56	0,5714
4	0,8312	0,7647	0,5909	0,6667
5	0,8182	0,8182	0,6429	0,72
6	0,6753	0,7	0,4242	0,5283
7	0,8442	0,8571	0,6667	0,75
8	0,7662	0,5714	0,4	0,4706
9	0,6974	0,6818	0,4839	0,566
10	0,566	0,7895	0,5556	0,6522
Average	0,74193	0,70785	0,56338	0,62277

b. Polynomial Kernel

The polynomial kernel achieved the highest average accuracy among the kernels, with a value of 76.04%. However, despite its higher accuracy, the polynomial kernel showed a lower recall of 46.27%, meaning the model frequently failed to detect patients who truly suffered from diabetes. Precision for the polynomial kernel was recorded at 74.57%, which is relatively high. The average F1-score for the polynomial kernel was 56.46%, reflecting a less optimal balance between precision and recall. The results of the polynomial kernel evaluation can be seen in Table 4.

Table 4. *SVM Model Evaluation Result with Polynomial Kernel*

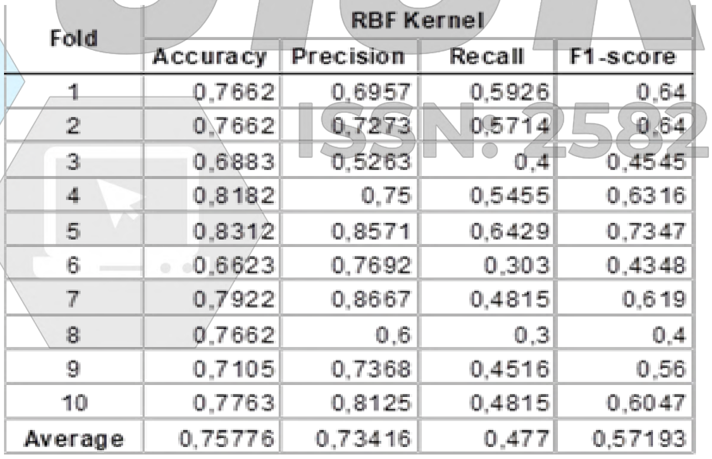


Fold	Polynomial Kernel			
	Accuracy	Precision	Recall	F1-score
1	0,7532	0,6667	0,5926	0,6275
2	0,7792	0,7895	0,5357	0,6383
3	0,7143	0,6	0,36	0,45
4	0,7922	0,7143	0,4545	0,5556
5	0,8312	0,8571	0,6429	0,7347
6	0,6623	0,7692	0,303	0,4348
7	0,8052	0,875	0,5185	0,6512
8	0,7662	0,625	0,25	0,3571
9	0,7105	0,7368	0,4516	0,56
10	0,7895	0,8235	0,5185	0,6364
Average	0,76038	0,74571	0,46273	0,56456

c. RBF Gaussian Kernel

The RBF Gaussian kernel achieved an average accuracy of 75.78%, with good precision at 73.42%. However, the recall for this kernel only reached 47.70%, indicating that the model also struggled to detect all positive cases. The F1-score was recorded at 57.19%, showing a balance between precision and recall, though many positive samples were still missed. The results of the RBF Gaussian kernel evaluation are shown in Table 5.

Table 5. *SVM Model Evaluation Result with RBF Kernel*



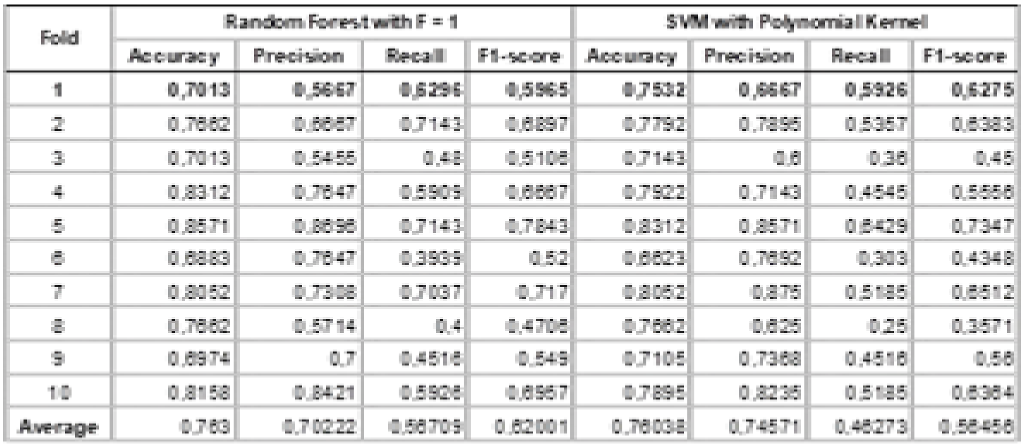
Fold	RBF Kernel			
	Accuracy	Precision	Recall	F1-score
1	0,7662	0,6957	0,5926	0,64
2	0,7662	0,7273	0,5714	0,64
3	0,6883	0,5263	0,4	0,4545
4	0,8182	0,75	0,5455	0,6316
5	0,8312	0,8571	0,6429	0,7347
6	0,6623	0,7692	0,303	0,4348
7	0,7922	0,8667	0,4815	0,619
8	0,7662	0,6	0,3	0,4
9	0,7105	0,7368	0,4516	0,56
10	0,7763	0,8125	0,4815	0,6047
Average	0,75776	0,73416	0,477	0,57193

From the evaluation results, the polynomial kernel showed the highest accuracy among the three kernels, with an average value of 76.04%, slightly better than both the linear and RBF Gaussian kernels. The polynomial kernel proved effective for handling more complex datasets, as it captured non-linear patterns better than the linear kernel. However, the polynomial kernel had a lower recall (46.27%) compared to the others, indicating that the model struggled to detect all positive cases. This suggests that despite the higher accuracy, the polynomial kernel missed several positive cases that should have been detected.

d. Comparison of Best Evaluation Results from Each Algorithm

A comparison between the Random Forest and Support Vector Machine models reveals that both have their own strengths and weaknesses. The comparison results can be seen in Table 6 below.

Table 6. *Comparison of Random Forest and Support Vector Machine Evaluation Results*



Fold	Random Forest with F = 1				SVM with Polynomial Kernel			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
1	0,7013	0,5667	0,6296	0,5965	0,7532	0,6667	0,5926	0,6275
2	0,7662	0,6667	0,7143	0,6897	0,7792	0,7895	0,5357	0,6383
3	0,7013	0,5455	0,48	0,5106	0,7143	0,6	0,36	0,45
4	0,8312	0,7647	0,5909	0,6667	0,7922	0,7143	0,4545	0,5556
5	0,8571	0,8696	0,7143	0,7843	0,8312	0,8571	0,6429	0,7347
6	0,6883	0,7647	0,3939	0,52	0,6623	0,7692	0,303	0,4348
7	0,8052	0,7308	0,7037	0,717	0,8052	0,875	0,5185	0,6512
8	0,7662	0,5714	0,4	0,4706	0,7662	0,625	0,25	0,3571
9	0,6974	0,7	0,4516	0,549	0,7105	0,7368	0,4516	0,56
10	0,8158	0,8421	0,5926	0,6957	0,7895	0,8235	0,5185	0,6364
Average	0,763	0,70222	0,58705	0,62001	0,76038	0,74571	0,46273	0,56456

Based on the results, Random Forest (RF) consistently showed more stable and reliable performance, achieving an average accuracy of 76.3%. It balanced precision, recall, and F1-score well—especially in the F=1 configuration—reaching its highest accuracy at 85.71% in the fifth fold. In contrast, Support Vector Machine (SVM) with a polynomial kernel, while comparable in accuracy (76.04%), struggled with recall, indicating it missed several positive cases.

One key reason for SVM's lower performance may be the lack of advanced data preprocessing in this study. A previous study by Nahzat and Yağanoğlu (2021) achieved 87% accuracy using the same dataset, largely due to more thorough data cleaning and transformation. They addressed zero values in key features (e.g., glucose, blood pressure) by replacing them with class-based means or medians, treated them as missing values, and normalized the dataset—an essential step for SVM performance. These preprocessing steps were not included in the current study, potentially limiting the SVM's ability to perform well.

Another notable difference was in the data splitting approach. While Nahzat and Yağanoğlu (2021) used a 70/30 train-test split, this study applied cross-validation, which may affect the comparability of the results, especially given the dataset's relatively small size.

The findings confirm that data preprocessing significantly influences algorithm performance. RF's ensemble approach makes it robust to outliers and inconsistent data, while SVM's effectiveness is more dependent on clean, normalized inputs and appropriate kernel selection. The polynomial kernel used in this study did not perform as well without preprocessing, reinforcing the importance of tuning and preparation for SVM models.

In summary, RF proved to be the more stable and reliable algorithm for classifying the Pima Indians Diabetes dataset under the given conditions. With better preprocessing, however, SVM could potentially close the performance gap or even outperform RF in certain scenarios.

IV. CONCLUSION

This study compared the performance of Random Forest (RF) and Support Vector Machine (SVM) in classifying the Pima Indians Diabetes dataset. Key findings include:

a. RF showed slightly better performance, achieving 76.3% accuracy compared to SVM with a polynomial kernel at 76.04%. RF also demonstrated greater consistency across folds, making it more reliable under data variability.

b. SVM performed well with non-linear patterns but was highly sensitive to kernel parameters, leading to less stable results than RF.

c. Preprocessing significantly impacts model performance. For example, Nahzat and Yağanoğlu's (2021) study achieved 87% accuracy using extensive preprocessing, such as data normalization and handling invalid values, which was not applied here.

d. RF outperformed SVM in recall and F1-score, indicating better detection of positive cases. Although SVM showed slightly higher precision, it came at the expense of missing more positives.

e. RF is recommended for this dataset due to its robustness and effectiveness in handling class imbalance and variation.

f. These results underscore the importance of preprocessing and algorithm selection. Future research should explore optimized preprocessing methods and hybrid models to leverage the strengths of both RF and SVM.

REFERENCES

[1] Ahmed, A., Saeed, F., & Rehman, S. (2022). A hybrid GA–SVM and GA–RF approach for diabetes classification. Healthcare Technology Letters, 9(2), 45–52. https://doi.org/10.1049/htl2.12046

[2] Al-Goblan, A. S., Al-Alfi, M. A., & Khan, M. Z. (2014). Mechanism linking diabetes mellitus and obesity. Diabetes, Metabolic Syndrome and Obesity: Targets and Therapy, 7, 587–591. https://doi.org/10.2147/DMSO.S67400

[3] Alam, M. M., Islam, M. T., Hossain, M. S., & Kabir, M. N. (2022). Performance evaluation of machine learning algorithms for health prediction. International Journal of Advanced Computer Science and Applications, 13(5), 88–95. https://doi.org/10.14569/IJACSA.2022.0130512

[4] Ali, O. (2013). Genetics of type 2 diabetes. World Journal of Diabetes, 4(4), 114–123. https://doi.org/10.4239/wjd.v4.i4.114

[5] Arshad, M., Aslam, M. U., Saleem, K., Raza, B., & Zubair, M. (2023). Performance evaluation of supervised machine learning algorithms for medical data classification. IEEE Access, 11, 29320–29332. https://doi.org/10.1109/ACCESS.2023.3252998

[6] Asif, M. (2021). The role of skinfold thickness in the prediction of diabetes mellitus. Journal of Taibah University Medical Sciences, 16(2), 270–276. https://doi.org/10.1016/j.jtumed.2021.02.005

[7] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

[8] Cheung, B. M. Y., & Li, C. (2012). Diabetes and hypertension: Is there a common metabolic pathway? Current Atherosclerosis Reports, 14(2), 160–166. https://doi.org/10.1007/s11883-012-0227-2

[9] Cortes C, & Vapnik V. (1995). Support-vector Networks. Kluwer Academic Publishers;: 273-297.

[10] El-Jerjawi, A., & Abu-Naser, S. (2018). Diabetes prediction using artificial neural network. International Journal of Advanced Science and Technology, 117, 111–119.

[11] Hosseini, R., Ahmed, M., & Khan, S. (2022). Explainable AI for medical diagnosis: A case study on diabetes prediction using Random Forest. Journal of Healthcare Informatics Research, 6(1), 1–17. https://doi.org/10.1007/s41666-021-00102-4

[12] Jain, S., & Choudhary, A. (2022). Performance evaluation of machine learning algorithms on diabetes dataset. Materials Today: Proceedings, 56, 313–318. https://doi.org/10.1016/j.matpr.2021.07.398

[13] Lyngdoh, H. D., Nongrum, E. L., & Laitflang, A. D. (2022). Comparative analysis of machine learning classifiers for diabetes prediction. Procedia Computer Science, 218, 2712–2719. https://doi.org/10.1016/j.procs.2022.01.298

[14] Mishra, M., Tiwari, R., & Pradhan, M. (2023). A comprehensive comparison of classification algorithms for heart disease prediction. Materials Today: Proceedings, 74, 2674–2680. https://doi.org/10.1016/j.matpr.2023.01.179

[15] Mohammad, R. M., Ibrahim, R. W., & Md Nor, R. B. (2020). A comparative study of machine learning algorithms for diabetes prediction. Indonesian Journal of Electrical Engineering and Computer Science, 20(2), 623–629. https://doi.org/10.11591/ijeecs.v20.i2.pp623-629

[16] Nahzat, A., & Yağanoğlu, A. M. (2021). Diabetes prediction using random forest and logistic regression methods. Journal of Intelligent Systems, 30(1), 471–479. https://doi.org/10.1515/jisys-2021-0057

[17] Osisanwo, F. Y., Akinsola, J. E. T., Awodele, O., Hinmikaiye, J. O., Olakanmi, O., & Akinjobi, J. (2017). Supervised machine learning algorithms: Classification and comparison. International Journal of Computer Trends and Technology, 48(3), 128–138. https://doi.org/10.14445/22312803/IJCTT-V48P126

[18] Pankow, J. S., Duncan, B. B., Schmidt, M. I., Ballantyne, C. M., Couper, D. J., Hoogeveen, R. C., & Schmidt, R. (2015). Fasting glucose and insulin resistance predict cardiovascular disease risk: The ARIC Study. Diabetes Care, 38(3), 439–444. https://doi.org/10.2337/dc14-2036

[19] Patel, H., Patel, R., & Yadav, N. (2021). Performance analysis of SVM and Random Forest algorithms for diabetes prediction. Journal of Physics: Conference Series, 1913(1), 012031. https://doi.org/10.1088/1742-6596/1913/1/012031

[20] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–

2830. https://www.jmlr.org/papers/v12/pedregosa11a.html

[21] Rayanagoudar, G., Hashi, A. A., Zamora, J., Khan, K. S., Hitman, G. A., Thangaratinam, S., & Cooray, S. D. (2016). Quantification of the type 2 diabetes risk in women with gestational diabetes: A systematic review and meta-analysis of 95,750 women. BMJ Open, 6(12), e013450. https://doi.org/10.1136/bmjopen-2016-013450

[22] Schölkopf B, & Smola AJ. (2002). Learning with Kernels—Support Vector Machines, Regularization, Optimization and Beyond. London: The MIT Press.

[23] Sharma, V., & Khanna, P. (2021). SVM-based classification for early diagnosis of diabetes. Procedia Computer Science, 173, 70–78. https://doi.org/10.1016/j.procs.2020.06.009

[24] Tabák, A. G., Herder, C., Rathmann, W., Brunner, E. J., & Kivimäki, M. (2012). Prediabetes: A high-risk state for diabetes development. The Lancet, 379(9833), 2279–2290. https://doi.org/10.1016/S0140-6736(12)60283-9

[25] Tigga, N. P., & Garg, S. (2020). Prediction of type 2 diabetes using machine learning classification methods. Procedia Computer Science, 167, 706–716. https://doi.org/10.1016/j.procs.2020.03.297

[26] Tharwat A. (2021). Classification Assessment Methods. Appl Comput Informatics; 17(1), 168-192.

[27] Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. BMC Medical Informatics and Decision Making, 19(1), 1–16.

[28] Zhuo, X., Zhang, P., & Hoerger, T. J. (2014). Lifetime direct medical costs of treating type 2 diabetes for people diagnosed at age 25, 45, or 65 years. Diabetes Care, 37(9), 2607–2615. https://doi.org/10.2337/dc13-2054

[29] Zhou, Y., Wang, F., & Xiao, H. (2020). Feature selection and classification for high-dimensional data using random forest. Expert Systems with Applications, 140, 112896. https://doi.org/10.1016/j.eswa.2019.112896

[30] Zou, Q., Qu, K., Luo, Y., Yin, D., Ju, Y., & Tang, H. (2018). Predicting diabetes mellitus with machine learning techniques. Frontiers in Genetics, 9, 515. https://doi.org/10.3389/fgene.2018.00515