Journal of Electronics, Electromedical Engineering, and Medical Informatics
Multidisciplinary: Rapid Review: Open Access Journal **Vol. 6, No. 2, July 2024, pp: 231-242; eISSN: 2656-8632**

RESEARCH ARTICLE **OPEN ACCESS**

Manuscript received May 2, 2024; revised May 23, 2024; accepted May 27, 2024; date of publication July 8, 2024
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v6i3.446

How to cite: Arya Syifa Hermiati, Rudy Herteno, Fatma Indriani, Triando Hamonangan Saragih, Muliadi, and Triwiyanto, "A Comparative Study: Application of Principal Component Analysis and Recursive Feature Elimination in Machine Learning for Stroke Prediction", Journal of Electronics, Electromedical Engineering, and Medical Informatics, vol. 6, no. 3, pp. 231-242, July 2024.

A Comparative Study: Application of Principal Component Analysis and Recursive Feature Elimination in Machine Learning for Stroke Prediction

Arya Syifa Hermiati[1], Rudy Herteno[1], Fatma Indriani[1], Triando Hamonangan Saragih[1], Muliadi[1], and Triwiyanto[2]

[1] Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia
[2] Department of Medical Electronics Technology, Poltekkes Kemenkes Surabaya, Surabaya, Indonesia

Corresponding author: Rudy Herteno (e-mail: rudy.herteno@ulm.ac.id).

ABSTRACT Stroke is a disease that occurs in the brain and can cause both vocal and global brain dysfunction. Stroke research mainly aims to predict risk and mortality. Machine learning can be used to diagnose and predict diseases in the healthcare field, especially in stroke prediction. However, collecting medical record data to predict a disease usually makes much noise because not all variables are important and relevant to the prediction process. In this case, dimensionality reduction is essential to remove noisy (i.e., irrelevant) and redundant features. This study aims to predict stroke using Recursive Feature Elimination as feature selection, Principal Component Analysis as feature extraction, and a combination of Recursive Feature Elimination and Principal Component Analysis. The dataset used in this research is stroke prediction from Kaggle. The research methodology consists of pre-processing, SMOTE, 10-fold Cross-Validation, feature selection, feature extraction, and machine learning, which includes SVM, Random Forest, Naive Bayes, and Linear Discriminant Analysis. From the results obtained, the SVM and Random Forest get the highest accuracy value of 0.8775 and 0.9511 without using PCA and RFE, Naive Bayes gets the highest value of 0.7685 when going through PCA with selection of 20 features followed by RFE feature selection with selection of 5 features, and LDA gets the highest accuracy with 20 features from feature selection and continued feature extraction with a value of 0. 7963. It can be concluded in this study that SVM and Random Forest get the highest accuracy value without PCA and RFE techniques, while Naive Bayes and LDA show better performance using a combination of PCA and RFE techniques. The implication of this research is to know the effect of RFE and PCA on machine learning to improve stroke prediction.

INDEX TERMS Recursive Feature Elimination, Principal Component Analysis, Support Vector Machine, Random Forest, Naive Bayes, Linear Discriminant Analysis

I. INTRODUCTION

Stroke is one of the leading causes of disability and death in the world [1] has shown that stroke remains the second leading cause of death and the third leading cause of death and disability (in terms of disability-adjusted life years - DALYs) worldwide. Stroke is a clinically defined illness characterized by acute, focal neurological dysfunction generally caused by one of two mechanisms: a blockage of an artery in the brain (ischemic stroke) or a rupture of a blood vessel in the brain (hemorrhagic stroke)[2], [3]. The latter include intracerebral hemorrhage and subarachnoid hemorrhage [4]. The most important modifiable risk factor for stroke is Hypertension. In 2019, in both the 50-74 and 75-plus age groups, ischemic heart disease and stroke were the leading causes of disability-adjusted life years [5]. Stroke research mainly aims to predict risk and mortality. In many health cases, especially stroke, machine learning algorithms can explain human physiology's complex and unpredictable nature. Machine learning (ML) is the academic discipline and set of techniques that enable computers to perform complex tasks. Using ML techniques could improve patient care by personalizing outcome predictions and reducing redundancy in standardized

processes, allowing clinicians to spend more time with patients [6]. Two areas that could benefit from ML techniques in the medical field are diagnosis and outcome prediction [7]. Research by [8] The machine learning algorithms used are decision tree, random forest, and SVM. The results showed that random forest and SVM had an accuracy of 97%. This shows that by using machine learning random forest and SVM, it can very well predict stroke. Another Study [9] performed stroke prediction using five machine learning algorithms: Logistic Regression, Random Forest, Decision Tree, K-nearest neighbors, Support Vector Machine, and Naïve Bayes. The highest accuracy was obtained using naive Bayes, which obtained 82%.

Medical records are often based on the results of various tests and the patient's medical history. However, this type of data collection is usually subject to much noise. Not all attributes in the generated data sets are essential when training machine learning algorithms. Some may be irrelevant, and others may not impact the prediction's outcome. Ignoring or removing these irrelevant or less essential attributes reduces the burden on the machine learning algorithms[10]. Dimensionality reduction is one of the most popular techniques to remove noisy (i.e., irrelevant) and redundant features. Techniques that reduce dimensionality can be subdivided into feature selection and feature extraction [11], [12].

Feature selection involves keeping only pertinent features and discarding redundant and superfluous ones. in this study, Recursive Feature Elimination (RFE) is used for feature selection[13]. In research [14], Combining RFE with seven features using different machine learning techniques gave the best results, namely RFE-XGBoost, which was 97% accurate, followed by RFE-SVM, which gave 95%, and RFE-RF, which gave 93%. Different research [15] using RFE to reduce irrelevant features to diagnose chronic kidney disease results using SVM got an accuracy of 96.67%, KNN 98.33%, Decision Tree 99.17%, and Random Forest got the best accuracy of 100%.

Feature extraction techniques project features into a new, lower-dimensional feature space, and the newly built features are typically a subset of the original features [16]. One of the algorithms for extracting features is Principal Component Analysis (PCA). This study [17] was conducted to determine whether adding PCA feature extraction improves diabetes prediction using logistic regression and *K-Means*. The results showed that adding feature extraction with PCA resulted in a higher accuracy of 89% than that of those who did not use PCA, which was 75%.

In this study, researchers see the novel combination of Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA) as having potential in machine learning for stroke prediction. researchers conducted an experimental study to predict stroke using SMOTE to handle data imbalance, Recursive Feature Elimination (RFE) as feature selection, Principal Component Analysis (PCA) as feature extraction and combining Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA). Four different scenarios were used, and 5, 10, 15, and 20 features were selected using four Machine Learning (ML) algorithms: Support Vector Machine (SVM), Random Forest (RF), naive bayes (NB), and linear discriminant analysis (LDA). This study aims to predict stroke using RFE as feature selection, PCA as feature extraction, and a combination of RFE and PCA. The results of this study are expected to contribute as follows:

a. Novelty by combining Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA) in machine learning to predict stroke.
b. Providing an understanding into the classification performance of machine learning models for stroke prediction through 10-fold Cross-Validation for data splitting, SMOTE for imbalance data handling, utilizing diverse classifiers such as Support Vector Machine, Random Forest, Naive Bayes, and Linear Discriminant Analysis.
c. Providing analysis on noise reduction and prediction accuracy improvement by assessing feature selection and extraction techniques, namely Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA), in stroke prediction.
d. Providing insight into the best classifier for stroke prediction based on the evaluation of classification performance and the effectiveness of feature selection and extraction techniques.
e. Help other researchers use RFE and PCA to reduce noise in machine learning and help clinicians make more accurate and effective predictions when using machine learning to identify patients at high risk of stroke.

II. MATERIAL AND METHODS

The following is an illustration of the research flow carried out in this study in FIGURE 1. The research process generally involves comparing the results of four classification methods: Support Vector Machine, Random Forest, Naive Bayes, and Linear Discriminant Analysis. All methods used SMOTE, and each method underwent testing under four conditions: Using only Recursive Feature Elimination, using only Principal Component Analysis, Using Recursive Feature Elimination then Principal Component Analysis, and using Principal Component Analysis followed by Recursive Feature Elimination. Seven consecutive stages make up the structure of this research: feature extraction and selection, model training, data preprocessing, SMOTE, data partitioning for training and testing using 10-fold cross-validation, data collection using stroke datasets, and analysis of evaluation outcomes.

A. DATASET

The Kaggle Repository's Stroke Prediction Dataset, accessible at Stroke Prediction Dataset | Kaggle, served as the dataset for this research. The features that are used to categorize patients with stroke disease are listed in this dataset. This dataset has

11 attributes and 5110 occurrences. Ten of the attributes, which reflect the clinical status of the patients, are utilized as predictor variables, while one attribute is used as a target variable. TABLE 1, which offers a thorough summary of the dataset properties, presents the attribute description of the dataset in accordance with earlier research [18], [19].

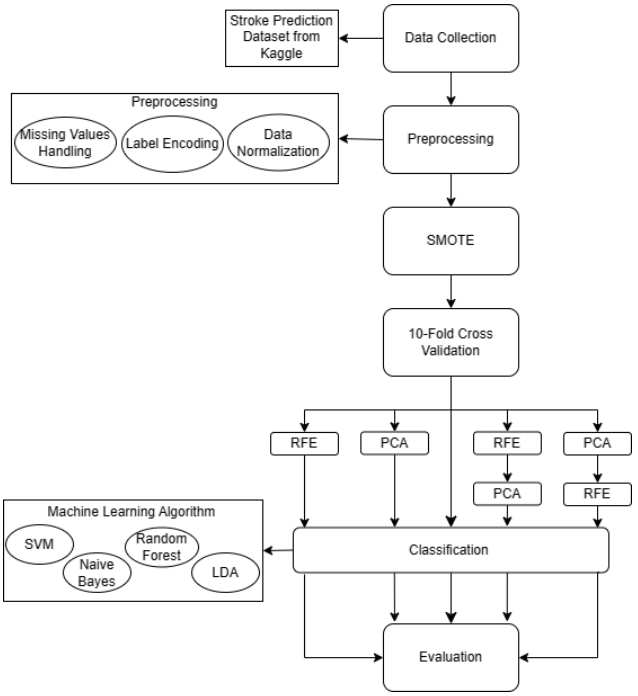

FIGURE 1. Research Flowchart

TABLE 1

Attribute Description

Feature Name	Description	Range
Id	A unique identifier for each patient.	unique identifier
Gender	Gender of the participant	Male, Female
Age	Age of the participant	Float
Hypertension	This feature indicates whether this participant has hypertension. 12.54% of participants have high blood pressure.	0 → No Hypertension 1 → Hypertension
Heart Disease	This feature indicates If the participant has heart disease. The participants' prevalence of heart disease was 6.33%.	0 →→ No Heart Disease 1 → Heart Disease
Ever Married	Marital status of participant, which is 79.84%, is represented by this feature.	Yes–No
Work Type	Job Status of participants	Never_worked, Children, Private, Self-employed, or Govt_job
Residence type	Classification of the patient's place of residence	Urban or Rural
Average glucose level	This feature tracks the participants' average blood glucose level.	Float
BMI	This feature records the participants' BMI.	Float
Smoking Status	Patient's smoking history: never smoked (52.64%), smokes (22.37%), and formerly smoked (24.99%).	Never smoked, smoked, or Formerly Smoked
Stroke	This attribute identifies if the participant has had a stroke in the past. 5.53% of participants have experienced a stroke.	0 → No Stroke 1 → Stroke

B. PRE-PROCESSING

Data pre-processing is essential, as keeping the data in its raw form could adversely affect the quality of the predictions. Tasks such as handling missing values, data transformation, and reducing redundant values are performed because the raw data may contain some missing values, redundancy, and noisy data [20].

1. MISSING VALUES HANDLING.

check the dataset for missing values or null values. Simple imputation was used in this research for missing value handling, which uses the mean (or median) of the available values for the same variable to fill in the gaps left by missing data[21] (Eq. (1)). In the stroke prediction dataset, missing values marked with N/A are missing, so it is necessary to replace them. Missing values frequently result in extreme uncertainty in the classification, which has an impact on prediction, modeling, accuracy, and justice, particularly for subgroups who are protected or sensitive [22].

$$mean = \frac{amount\ data}{lots\ of\ data} \quad (1)$$

TABLE 2

Before and after using missing value *handling*

Before		After	
avg_glucose_level	bmi	avg_glucose_level	bmi
202.21	N/A	202.21	28.89324

2. ONE-HOT ENCODING

one-hot encoding is performed because the dataset contains categorical data. In one-hot encoding, the categorical feature is replaced by k binary features, which can only take the values 0 or 1, with k possible values and $k > 2$. One of these k features, the hot feature, is precisely 1, hence the name one-hot encoding[23]. Examples of features that implement the one-hot encoding algorithm are available in TABLE 3.

TABLE 3

Feature that Implements One-Hot Encoding

bmi	ever_married_No	ever_married_Yes
36.6	0	1
28.89324	0	1
32.5	0	1

3. DATA NORMALIZATION

Data normalization is performed using Standard Scaler to eliminate scale differences between variables in the data, converting it into data with zero mean and unit standard deviation [24] (Eq. (2)).

$$Z = \frac{X-\mu}{\sigma} \quad (2)$$

where Z is the new value of data after scaling, X is the original value of data, μ average of the data, and σ is the standard deviation of the data.

C. SMOTE

Thus, after preprocessing, In order to address the unequal distribution of data between majority and minority groups, the Random Oversampling Technique is utilized. The "Not Stroke" class in the dataset has more samples than the "Stroke" class. SMOTE generates fictitious data based on the similarity of the spatial characteristics of the minority modules[25] (Eq. (3)).

$$X_{new} = x + rand(0,1) \times (y[i] - x) \quad (3)$$

where X_{new} is the new instance, x is the minority class instance, $y[i]$ is the closest neighbor of x. i is 1, 2, ..., N, $rand(0,1)$ are random numbers between 0 and 1 [26]. A comparison of before and after SMOTE implementation is in FIGURE 2.

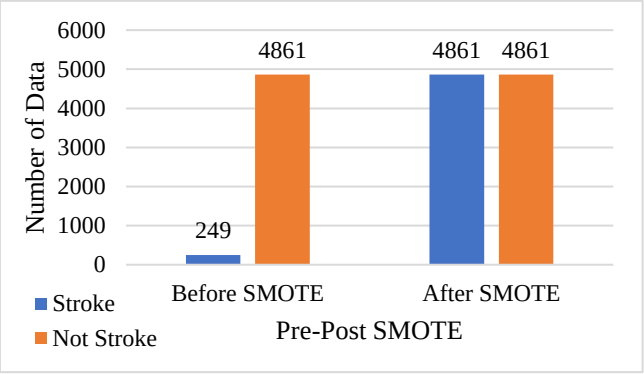

FIGURE 2. **Before and After SMOTE for Stroke Prediction Dataset**

D. DATA SPLITTING

Cross-validation divides the original data into a training set and a testing set. To evaluate the classification performance, the training set is used to train the classification. 10-fold cross-validation is the definition for the K-value, where K=10. The dataset is partitioned into K subsets: K-1 subsets are used as the training set when testing the model, and one subset is used as the validation set [27], [28].

FIGURE 3. **10-Fold Cross-Validation**

The measurement of the prediction model's unbiased estimation is done using the 10-fold cross-validation approach. It is employed to prevent overfitting and to compare performance. With the use of 10-fold cross-validation, FIGURE 3 [29] offers a visual depiction of this data partitioning iteration.

E. RECURSIVE FEATURE ELIMINATION (RFE)

Recursive Feature Elimination, a wrapper-type feature selection method, can use various machine learning techniques to choose the best features. Using a backward selection procedure, this approach removes redundant or non-predictive features to discover the ideal feature combination. It determines the significance of each characteristic after first creating a prediction model using all of the features. Secondly, it finds aspects that need to be more relevant by ranking the features. Lastly, it iteratively eliminates the least significant features from the model using measures for model evaluation (such as accuracy, Kappa, and root mean squared error) until the target number of features is retained [15], [30] in this study, using the 'estimator'=LinearRegression() and and 'n_features_to_select' using 5,10,15, and 20.

F. PRINCIPAL COMPONENT ANALYSIS (PCA)

Principal Component Analysis is a linear data processing technique that optimizes information measured by variance and minimizes redundancy measured by covariance. Using the dependencies between the variables, PCA breaks down multi-dimensional data into more manageable, lower-dimensional parts while preserving a significant amount of information. The primary goal of PCA is to reduce the dimensionality of a data set made up of numerous interconnected variables while maintaining as much of the data's variance as is practical [12], [16], [31]. The following are the steps in PCA [32]:

1. Determine the data's mean vector.

$$\bar{x} = \frac{1}{N}\sum_{n=1}^{N} X_n \quad (4)$$

2. From each data point across all data, subtract the means vector:

$$\hat{x}_n = x_n - \bar{x} \quad (5)$$

3. Let $\hat{X} = [\widehat{x_1}, \widehat{x_2}, \ldots, \widehat{x_D}]$ is a data matrix that is orthonormal. The covariance matrix is available.

$$S = \frac{1}{N}\hat{X}\hat{X}^T \quad (6)$$

4. Determine the covariance (or correlation) matrix's eigenvalues and eigenvectors, then arrange them in descending order of eigenvalues.
5. To create the U_k matrix, choose K eigenvectors that match the K greatest eigenvalues. The columns of the U_k matrix will form an orthogonal system. Often referred to as the principal components, these K vectors provide a subspace near to the orthonormal data matrix.
6. Project the orthonormal data matrix onto the identified subspace.

7. The coordinates of the data points on the new space make up the new data.

$$Z = U_K^T \hat{X} \quad (7)$$

The new data can be used to approximate the original data in the following way:

$$x \approx U_k Z + \hat{x} \quad (8)$$

G. SUPPORT VECTOR MACHINE (SVM)

The concept of SVM stems from a two-class classification problem that requires positive and negative training sets; SVM attempts to find the best hyperplane to separate the two classes and maximize the distance between them[33], [34]. The superior performance of the SVM model can be attributed to its ability to find the optimal hyperplane that maximizes the margin between stroke and non-stroke cases [35]. In this study, using "kernel" = linear. The following are the steps in Support Vector Machine [36], [37]:

1. Determine the kernel value on the training data
2. Determine the alpha value

$$\alpha = \frac{sum\ \alpha}{\sum K(x.x_i)} \quad (9)$$

3. Determining the weight

$$w = \alpha_i y_i K(x.x_i) \quad (10)$$

4. Determines the bias value

$$b = \frac{1}{2} w.x^+ + w.x^- \quad (11)$$

5. Determines the test kernel value
6. Determines the prediction result value $f(\emptyset(x))$

$$f(\emptyset(x)) = sign(w.(Xtest.Xi) + b) \quad (12)$$

Dot product or linear kernel parameters are used in this research, and the formula is as follows:

$$K(xi, xj) = xi^T xj \quad (13)$$

H. RANDOM FOREST

Random forest is a machine learning algorithm belonging to the ensembles category, consisting of many trained decision trees, each carrying bootstrapped samples, better known as out-of-bag observations, for each observation. The goal of developing random forest was to enhance decision tree techniques, which frequently have overfitting issues. A majority vote of each individual prediction result determines the ultimate prediction result in the Random Forest process, which focuses on building numerous decision trees. This method successfully addresses the issues that can occur when classification is carried out using a single, frequently suboptimal decision tree [38], [39]. In this study, using the default number of trees in a random forest model in scikit-learn's, which is 100. The following are the steps in Random Forest[40].

1. Forming trees, where each decision tree is formed by applying the Gini index

$$Gini\ Index\ (D) = 1 - \sum_{i=1}^{m} P_i^2 \quad (14)$$

where m is the total number of attributes and P_i is the percentage of attributes in each class. The tree's root node will have the feature with the lowest overall Gini Index value.

2. The total Gini Index at an internal node (e.g., K) is calculated in the following equation (15).

$$Gini\ Index\ sum\ (K) = \frac{T_1}{T} Gini\ Index\ (D_1) + \frac{T_2}{T} Gini\ Index\ (D_2) \quad (15)$$

In this case, T denotes the total records for all classes, T_1 represents the total records for the first class, and T_2 represents the total records for the second class. Until all of the nodes in the tree cannot be divided, this process of child node development continues. The voting method is used to continue the categorization stage after the full tree has been generated.

The Random Forest algorithm goes through the following stages of completion [38].

1. Ascertain how many trees (k) were chosen (k < m) out of all the features (m).
2. Next, for every tree in the dataset, N random samples are extracted.
3. In each tree, a random subset of predictors is chosen, with m < p denoting the number of predictor variables.
4. Next, for k trees, the second and third step procedures are repeated.
5. The most votes from the categorization outcomes of the same number of trees are used to determine the prediction results.

I. NAIVE BAYES

The Naive Bayes algorithm is a probabilistic classification technique based on Bayes' theorem. It assumes that the features used to classify are independent, which may oversimplify the real world[41], [42]. The algorithm calculates the probability that a data point is a member of a particular class based on its feature values. During the training phase, the algorithm learns the probabilities from the data. In the prediction phase, the probabilities of the individual features for each class are multiplied, and the class with the highest probability is selected as the final prediction. Naive Bayes often performs well in text classification and spam filtering despite its simplicity and "naive" independence assumption[27]. In this study, we using the default naive bayes GaussianNB().The formula of naive bayes is in equation (16).

$$(C|F_1, \dots, F_n) = \frac{p\ (C) p(F_1,\dots,F_n|C)}{p(F_1,\dots,F_n)} \quad (16)$$

Where $\mathrm{p}(F_1, \dots, F_n|\mathrm{C})$ is the probability likelihood, $\mathrm{p}(F_1, \dots, F_n)$ is the prior probability of the instance $(F_1, \dots, F_n)$, $\mathrm{p}\,(\mathrm{C})$ is the probability of class C, and $p(C|F_1, \dots, F_n)$ is the posterios probability.

J. LINEAR DISCRIMINANT ANALYSIS (LDA)

Linear Discriminant Analysis was first created by Fisher (1936) as a technique for determining which linear combinations of variables best-divided observations into groups or classes. LDA might be better than multinomial logistic regression for group classification. More precisely, unlike multinomial logistic regression, LDA does not require the definition of a reference group and can be used to classify three or more groups. However, LDA might need to work better. For instance, LDA is not the best option for classification if the data are not multivariate normal and the variance-covariance matrices are not roughly equal [43]. In this study, we using the default LinearDiscriminantAnalysis(). The aim here is to find a linear function [44]

$$y = a_1 x_{i_1} + a_2 x_{i_2} + \cdots + a_q x_{i_q} \quad (17)$$

where:

$$a^T = [\{a_1, a_2, \ldots, a_q\}] \quad (18)$$

is a vector of coefficients that requires calculation, whereas

$$x_i = [x_{i_1}, x_{i_2}, \ldots, x_{i_q}] \quad (19)$$

the patients, as well as

$$x_j = [x_{j_1}, x_{j_2}, \ldots, x_{j_q}] \quad (20)$$

Are the features.

K. EVALUATION

This research uses accuracy in the confusion matrix to evaluate each machine learning. Whether a classification algorithm is used to predict or classify attributes, the Confusion Matrix assesses the performance and accuracy of the method. Its purpose is to assess machine learning methods that address classification issues. Data comparing the system's produced classification results with the anticipated classification results make up the confusion matrix [45]. False Negative (FN), False Positive (FP), True Negative (TN), and True Positive (TP) are the terms used in the Confusion Matrix and are defined in TABLE 4. Equation (21) is the calculation formula of accuracy.

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \quad (21)$$

TABLE 4
Confusion matrix

Actual Class	Predicted Class	
	True	False
True	True Positive (TP)	False Negative (FN)
False	False Positive (FP)	True Negative (TN)

To study the significance of the results of the RFE and PCA algorithm, the framework of permutation-based p-values, which are explored in [46] is followed. In addition, the permutation test (repeated 1000 times) was performed to assess the classification ability of each model. A permutation p-value < 0.05 was considered significant[47].

III. RESULT

This section shows the performance of each machine learning method using Recursive Feature Elimination as the feature selection method and Principal Component Analysis as the feature extraction method. The performance of each machine learning is evaluated based on the accuracy value obtained.

A. SUPPORT VECTOR MACHINE(SVM) PERFORMANCE

This section reveals the experimental findings obtained from the Support Vector Machine (SVM) classification model.

TABLE 5
Accuracy of SVM with RFE and SVM with PCA

Feature	5	10	15	20
RFE	78.91%	79.44%	79.66%	79.58%
PCA	70.57%	71.15%	79.37%	79.46%

From TABLE 5, using RFE feature selection, the SVM algorithm gets the highest accuracy by selecting the best 15 features, 79.66%. Using PCA feature extraction, the SVM algorithm gets the highest accuracy of 79.46% by selecting 20 features. This result is lower than that of SVM using RFE.

TABLE 6
Accuracy of combination SVM with RFE and PCA

Feature	Accuracy	Feature	Accuracy
5 RFE, 5 PCA	78.89%	5 PCA, 5 RFE	70.59%
10 RFE, 5 PCA	74.61%	10 PCA, 5 RFE	71.71%
10 RFE, 10 PCA	79.44%	10 PCA, 10 RFE	71.16%
15 RFE, 5 PCA	68.47%	15 PCA, 5 RFE	78.25%
15 RFE, 10 PCA	73.21%	15 PCA, 10 RFE	78.29%
15 RFE, 15 PCA	79.67%	15 PCA, 15 RFE	79.38%
20 RFE, 5 PCA	69.63%	20 PCA, 5 RFE	78.76%
20 RFE, 10 PCA	70.67%	20 PCA, 10 RFE	79.18%
20 RFE, 15 PCA	79.59%	20 PCA, 15 RFE	79.45%
20 RFE, 20 PCA	79.59%	20 PCA, 20 RFE	79.47%

In TABLE 6, SVM uses feature selection with RFE first, then after that, feature extraction with PCA. The result of the combination is that SVM gets the highest accuracy of 79.67% by using a combination of 15 selected features from RFE, followed by feature extraction with PCA using those 15 features. Furthermore, SVM is combined by PCA feature extraction first, followed by RFE feature selection. The results of this combination obtained SVM's best accuracy of 79.47% by using 20 features from PCA and continued with 20 features passing RFE feature selection.

B. RANDOM FOREST PERFORMANCE

This section reveals the experimental findings obtained using the random forest classification model

TABLE 7
Accuracy of RF with RFE and RF with PCA

Feature	5	10	15	20
RFE	94.56%	89.80%	94.18%	94.84%
PCA	91.53%	93.08%	94.29%	94.42%

From TABLE 7 using RFE feature selection, the random forest algorithm gets the highest accuracy by selecting the best 20 features for an accuracy of 94.84%. Using PCA feature extraction, the random forest algorithm gets the highest accuracy of 94.42% by selecting 20 features. This result is lower than of the random forest using RFE.

TABLE 8

Accuracy of combination Random Forest with RFE and PCA

Feature	Accuracy	Feature	Accuracy
5 RFE, 5 PCA	94.49%	5 PCA, 5 RFE	91.57%
10 RFE, 5 PCA	85.86%	10 PCA, 5 RFE	90.23%
10 RFE, 10 PCA	87.51%	10 PCA, 10 RFE	92.97%
15 RFE, 5 PCA	89.43%	15 PCA, 5 RFE	89.38%
15 RFE, 10 PCA	92.35%	15 PCA, 10 RFE	93.45%
15 RFE, 15 PCA	92.94%	15 PCA, 15 RFE	94.58%
20 RFE, 5 PCA	90.58%	20 PCA, 5 RFE	88.92%
20 RFE, 10 PCA	92.03%	20 PCA, 10 RFE	93.19%
20 RFE, 15 PCA	93.82%	20 PCA, 15 RFE	94.36%
20 RFE, 20 PCA	93.98%	20 PCA, 20 RFE	94.46%

In TABLE 8, Random Forest uses feature selection with RFE first, then after that, feature extraction with PCA. The result of the combination is that random forest gets the highest accuracy of 94.49% by using a combination of 5 selected features from RFE, followed by feature extraction with PCA using those 5 features. Furthermore, random forest is combined by PCA feature extraction first, followed by RFE feature selection. The results of this combination obtained random forest got the best accuracy of 94.58% by using 15 features from PCA and continued with 15 features passing RFE feature selection.

C. NAIVE BAYES PERFORMANCE

This section reveals the experimental findings obtained from the naive bayes classification model.

TABLE 9

Accuracy of Naive Bayes with RFE and Naive Bayes with PCA

Feature	5	10	15	20
RFE	75.26%	72.71%	65.92%	59.34%
PCA	70.20%	70.54%	62.69%	69.87%

From TABLE 9, using RFE feature selection, the naive bayes algorithm gets the highest accuracy by selecting the 5 best features, so that it gets an accuracy of 75.26%. Using PCA feature extraction, the naive bayes algorithm gets the highest accuracy of 70.54% by selecting 10 features. This result is lower than the naive bayes using RFE.

TABLE 10

Accuracy of combination Naive Bayes with RFE and PCA

Feature	Accuracy	Feature	Accuracy
5 RFE, 5 PCA	76.17%	5 PCA, 5 RFE	70.19%
10 RFE, 5 PCA	73.29%	10 PCA, 5 RFE	72.25%
10 RFE, 10 PCA	68.00%	10 PCA, 10 RFE	70.54%
15 RFE, 5 PCA	70.63%	15 PCA, 5 RFE	74.86%
15 RFE, 10 PCA	69.31%	15 PCA, 10 RFE	69.21%
15 RFE, 15 PCA	70.78%	15 PCA, 15 RFE	62.69%
20 RFE, 5 PCA	69.43%	20 PCA, 5 RFE	76.87%
20 RFE, 10 PCA	71.41%	20 PCA, 10 RFE	74.85%
20 RFE, 15 PCA	70.58%	20 PCA, 15 RFE	75.10%
20 RFE, 20 PCA	70.58%	20 PCA, 20 RFE	69.87%

In TABLE 10, naive bayes uses feature selection with RFE first, then, feature extraction with PCA. The result of the combination is that naive Bayes gets the highest accuracy of 76.17% by combining 5 selected features from RFE, followed by feature extraction with PCA using those 5 features. Furthermore, naive bayes is combined by PCA feature extraction first, followed by RFE feature selection. The results of this combination obtained naive bayes with the best accuracy of 76.87% by using 20 features from PCA and continued with 5 features from RFE feature selection.

D. LINEAR DISCRIMINANT ANALYSIS PERFORMANCE

This section reveals the experimental findings obtained from the Linear Discriminant Analysis (LDA) classification model.

TABLE 11

Accuracy of LDA with RFE and LDA with PCA

Feature	5	10	15	20
RFE	78.78%	79.30%	79.41%	79.36%
PCA	71.13%	71.43%	78.33%	79.20%

From TABLE 11, using RFE feature selection, the LDA algorithm gets the highest accuracy by selecting the best 15 features, 79.41%. Using PCA feature extraction, the LDA algorithm gets the highest accuracy, 79.20%, by selecting 20 features. This result is lower than LDA using RFE.

TABLE 12

Accuracy of combination LDA with RFE and PCA

Feature	Accuracy	Feature	Accuracy
5 RFE, 5 PCA	78.78%	5 PCA, 5 RFE	71.13%
10 RFE, 5 PCA	73.70%	10 PCA, 5 RFE	72.47%
10 RFE, 10 PCA	79.29%	10 PCA, 10 RFE	71.43%
15 RFE, 5 PCA	72.10%	15 PCA, 5 RFE	78.01%
15 RFE, 10 PCA	73.94%	15 PCA, 10 RFE	78.19%
15 RFE, 15 PCA	79.44%	15 PCA, 15 RFE	78.33%
20 RFE, 5 PCA	70.15%	20 PCA, 5 RFE	78.68%
20 RFE, 10 PCA	72.45%	20 PCA, 10 RFE	79.18%
20 RFE, 15 PCA	79.35%	20 PCA, 15 RFE	79.39%
20 RFE, 20 PCA	79.63%	20 PCA, 20 RFE	79.20%

In TABLE 12, LDA first uses feature selection with RFE, then feature extraction with PCA. The result of the combination is that LDA gets the highest accuracy of 79.63% by using a combination of 20 selected features from RFE, followed by feature extraction with PCA using those 20 features. Furthermore, LDA is combined by PCA feature extraction first, followed by RFE feature selection. The results of this combination obtained LDA's best accuracy of 79.39% by using 20 features from PCA and continued with 15 selected features from RFE feature selection.

E. PERMUTATION TEST

In this section, we will present the results of the permutation test to test the significance of the features from RFE and PCA. table 2 shows the results of the permutation test.

TABLE 13
p-value of permutation test

Dimentionality reduction	Machine Learning	Feature			
		5	10	15	20
RFE	SVM	0.013	0.0001	0.0001	0.033
	RF	0.04	0.0296	0.034	0.0175
	NB	0.025	0.00001	0.012	**0.087**
	LDA	0.024	0.09	0.027	**0.073**
PCA	SVM	**0.07**	0.047	0.02	0.013
	RF	0.03	0.001	**0.06**	**0.074**
	NB	0.045	0.0003	0.01	**0.08**
	LDA	0.04	0.0043	0.03	0.006

from the permutation test results with P-value, the results obtained p-value < 0.05 can be considered getting significant features, while the results that get p-value ≥ 0.05 can be said to be not significant. from the results of TABLE 13, there are some results from RFE and PCA that are not significant.

IV. DISCUSSION

From the research results previously described, there are four experiments for each machine learning algorithm. The experiments include applying feature selection using RFE, feature extraction using PCA, a combination of RFE with PCA, and PCA with RFE. TABLE 14 compares the highest accuracy evaluation results of all algorithms.

TABLE 14
Comparison of each Algorithm

	SVM	RF	NB	LDA
No PCA & RFE	87.76%	95.11%	60.38%	79.38%
RFE	79.66%	94.84%	75.26%	79.41%
PCA	79.47%	94.42%	70.54%	79.20%
RFE + PCA	79.67%	94.49%	76.17%	79.63%
PCA + RFE	79.47%	94.58%	76.87%	79.39%

From the comparison results above, using RFE and PCA does not affect the increase in model accuracy in SVM and random forest algorithms. On the contrary, using RFE and PCA decreases the model's accuracy due to dimensionality reduction. It can be seen that when not using RFE and PCA, the accuracy of SVM and random forest reaches 87.76% and 95.11%, but when using RFE and PCA, the accuracy decreases to 79% and 94%. In the LDA algorithm, the use of RFE and PCA does not affect the increase in accuracy. When using RFE and PCA, as well as without using RFE and PCA, the accuracy obtained remains around 79%, with only an increase of 0.31% using the RFE+PCA combination. The use of RFE and PCA affects the Naive Bayes algorithm. Before using RFE and PCA, the accuracy obtained was 60.38%, and when PCA+RFE was implemented, the accuracy increased by 21.45% to 76.87%.

The use of scenarios 5, 10, 15, and 20 in this study affects the accuracy of the machine learning algorithm due to the large number of features to be selected. TABLE 4, TABLE 6, TABLE 8, and TABLE 10 show that each scenario's accuracy is different depending on the number of features generated. Random forest is the algorithm with the highest accuracy in all the experiments that have been done. SVM gets its highest accuracy without dimensional reduction, 87.76%, and random forest gets the highest accuracy without dimensional reduction, with a value of 95.11%. Naive Bayes gets its highest accuracy value when using PCA feature extraction with 20 features, and it continues with feature selection using 5 features. LDA gets its highest accuracy value of 79.63% from a combination of RFE feature selection of 20 features and continued with PCA feature extraction of 20 features.
To compare algorithms that use RFE as their feature selection method, TABLE 15 presents the comparison results with previous studies . It is found that, in this study, the random forest algorithm that uses RFE gets better scores.

TABLE 15
Comparison of RFE with Previous Research

Research	Other Research	Proposed Work
SVM	[14] 95%	79.46%
Random Forest	[14] 93%	94.83%
Naive Bayes	[14] 87%	75.26%

TABLE 16 compares algorithms that use PCA as their feature extraction method with previous studies. In this study, the SVM, random forest, and naive Bayes algorithms, which use PCA, scored better than another research.

TABLE 16
Comparison of PCA Accuracy with Previous Research

Algorithms	Other Research	Proposed Work
SVM	[48] 65.3%	79.46%
Random Forest	[49] 78.44%	94.42%
Naive Bayes	[48] 63.5%	70.54%
LDA	[44] 86%	79.20%

To compare the SVM algorithms obtained, TABLE 16 presents the comparison results with previous studies. In this study, SVM using only SMOTE is not better than previous studies that have achieved the best result of 95%.

TABLE 17
Comparison of Highest SVM Model Accuracy with Previous Research

Research	Source Data	Accuracy (%)
[50]	Stroke prediction dataset	95
[9]	Stroke prediction dataset	80
[51]	Baby Cry dataset	83.6
[52]	TikTok Shop closure from Twitter	80.37
Proposed Work (With SMOTE)	Stroke prediction dataset	87.75

TABLE 18 presents the results of a comparison with previous studies to compare the random forest algorithm obtained. In

this study, random forest using only SMOTE gets a better value than previous studies, namely 95.11%.

TABLE 18
Comparison of Highest Random Forest Model Accuracy with Previous Research

Research	Source Data	Accuracy (%)
[50]	Stroke prediction dataset	94.7
[9]	Stroke prediction dataset	73
[51]	Baby Cry dataset	84
[52]	TikTok Shop closure from Twitter	79.14
Proposed Work (With SMOTE)	Stroke prediction dataset	95.11

To compare the naive bayes algorithm obtained, TABLE 19 compares results with previous research. In this study, naive bayes using SMOTE +PCA+RFE are not better than previous studies that have achieved the best result of 87.5%.

TABLE 19
Comparison of Highest Naive Bayes Model Accuracy with Previous Research

Research	Source Data	Accuracy (%)
[50]	Stroke prediction dataset	87.5
[9]	Stroke prediction dataset	82
[51]	Baby Cry dataset	53
Proposed Work (With SMOTE - 20 Feature PCA – 5 Feature RFE)	Stroke prediction dataset	76.86

To compare the results of the obtained LDA algorithm, TABLE 20 compares the results with those of previous studies. In this study, LDA using SMOTE +RFE+PCA is not better than previous studies that have achieved the best result of 86%.

TABLE 20
Comparison of Highest LDA Model Accuracy with Previous Research

Research	Source Data	Accuracy (%)
[44]	coronary artery disease dataset	86
Proposed Work (With SMOTE - 20 Feature RFE – 20 Feature PCA)	Stroke prediction dataset	79.63

This study reveals the improved performance of random forest classification on stroke prediction when machine learning methods are used with SMOTE and without RFE or PCA, allowing the model to classify data more accurately and efficiently. This research also reveals experimental results using RFE feature selection and PCA feature extraction on machine learning algorithms, although some did not experience performance improvements.

In clinical practice, the results of this study can be used to improve the accuracy of stroke prediction. For example, clinicians can use SVM and Random Forest for stroke prediction, especially in situations where limited data is available. In addition, the use of RFE and PCA can help reduce noise and redundant features, thereby improving stroke prediction accuracy. However, the results of this study also show that there are still some areas for improvement, such as the use of larger and more diverse datasets. Therefore, further research that is more specific and broader is needed to improve stroke prediction accuracy.

V. CONCLUSION

This research uses machine learning algorithms to identify and classify stroke diseases. This research is structured using seven sequential stages: data collection using stroke datasets, data preprocessing, SMOTE, data partitioning for training and testing using 10-fold cross-validation, feature selection and feature extraction, model training, and analysis of evaluation results. The results illustrate four experiments for each machine learning algorithm: SVM, random forest, naive bayes, and LDA. The four experiments use RFE feature selection, PCA feature extraction, RFE and PCA combination, and PCA and RFE combination. Data analysis on stroke prediction shows that in the SVM algorithm, the best results are obtained without using dimensional reduction, 87.76%, and random forest, which gets the highest accuracy without dimensional reduction, with a value of 95.11%. Naive bayes gets its highest accuracy value when using PCA feature extraction with 20 features and continued with feature selection using 5 features. LDA gets its highest accuracy value of 79.63% from a combination of RFE feature selection of 20 features and continued with PCA feature extraction of 20 features.

However, to optimize the performance of the method further in predicting stroke, it is crucial to consider the diversity of features in larger datasets in future research. This will enable the model to learn more complex patterns, thereby making more accurate predictions. Moreover, the potential effectiveness of other machine learning algorithms, especially when combined with optimal feature selection and extraction methods, should be emphasized. Lastly, combining feature selection and extraction using other methods should be further explored. By addressing these aspects, future studies hope to achieve more accurate and comprehensive results in stroke prediction.

REFERENCES

[1] V. L. Feigin *et al.*, "Global, regional, and national burden of stroke and its risk factors, 1990–2019: a systematic analysis for the Global Burden of Disease Study 2019," *Lancet Neurol*, vol. 20, no. 10, pp. 795–820, Oct. 2021, doi: 10.1016/S1474-4422(21)00252-0.
[2] S. J. X. Murphy and D. J. Werring, "Stroke: causes and clinical features," *Medicine*, vol. 48, no. 9, pp. 561–566, 2020.
[3] D. Frank, A. Zlotnik, M. Boyko, and B. F. Gruenbaum, "The Development of Novel Drug Treatments for Stroke Patients: A Review," *Int J Mol Sci*, vol. 23, no. 10, p. 5796, May 2022, doi: 10.3390/ijms23105796.
[4] B. C. V. Campbell *et al.*, "Ischaemic stroke," *Nat Rev Dis Primers*, vol. 5, no. 1, p. 70, Oct. 2019, doi: 10.1038/s41572-019-0118-8.
[5] GBD 2019 Diseases and Injuries Collaborators, "Global burden of 369 diseases and injuries in 204 countries and territories, 1990–2019: a systematic analysis for the Global Burden of Disease Study 2019," *The Lancet*, vol. 396, no. 10258, pp. 1204–1222, 2020, doi: 10.1016/S0140-6736(20)30925-9.

[6] A. Pfob, S.-C. Lu, and C. Sidey-Gibbons, "Machine learning in medicine: a practical introduction to techniques for data pre-processing, hyperparameter tuning, and model comparison," *BMC Med Res Methodol*, vol. 22, no. 1, p. 282, Nov. 2022, doi: 10.1186/s12874-022-01758-8.

[7] J. A. M. Sidey-Gibbons and C. J. Sidey-Gibbons, "Machine learning in medicine: a practical introduction," *BMC Med Res Methodol*, vol. 19, no. 1, p. 64, Dec. 2019, doi: 10.1186/s12874-019-0681-4.

[8] I. G. Ivanov, Y. Kumchev, and V. J. Hooper, "An Optimization Precise Model of Stroke Data to Improve Stroke Prediction," *Algorithms*, vol. 16, no. 9, p. 417, Sep. 2023, doi: 10.3390/a16090417.

[9] G. Sailasya and G. L. A. Kumari, "Analyzing the Performance of Stroke Prediction using ML Classification Algorithms," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, 2021, doi: 10.14569/IJACSA.2021.0120662.

[10] G. T. Reddy *et al.*, "Analysis of Dimensionality Reduction Techniques on Big Data," *IEEE Access*, vol. 8, pp. 54776–54788, 2020, doi: 10.1109/ACCESS.2020.2980942.

[11] B. Remeseiro and V. Bolon-Canedo, "A review of feature selection methods in medical applications," *Comput Biol Med*, vol. 112, p. 103375, Sep. 2019, doi: 10.1016/j.compbiomed.2019.103375.

[12] C. Yumeng and F. Yinglan, "Research on PCA Data Dimension Reduction Algorithm Based on Entropy Weight Method," in *2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, IEEE, Oct. 2020, pp. 392–396. doi: 10.1109/MLBDBI51377.2020.00084.

[13] F. E. Bezerra *et al.*, "Impacts of Feature Selection on Predicting Machine Failures by Machine Learning Algorithms," *Applied Sciences*, vol. 14, no. 8, p. 3337, Apr. 2024, doi: 10.3390/app14083337.

[14] B. Zhang, X. Dong, Y. Hu, X. Jiang, and G. Li, "Classification and prediction of spinal disease based on the SMOTE-RFE-XGBoost model," *PeerJ Comput Sci*, vol. 9, p. e1280, Mar. 2023, doi: 10.7717/peerj-cs.1280.

[15] E. M. Senan *et al.*, "Diagnosis of Chronic Kidney Disease Using Effective Classification Algorithms and Recursive Feature Elimination Techniques," *J Healthc Eng*, vol. 2021, pp. 1–10, Jun. 2021, doi: 10.1155/2021/1004767.

[16] J. Ma and Y. Yuan, "Dimension reduction of image deep feature using PCA," *J Vis Commun Image Represent*, vol. 63, p. 102578, Aug. 2019, doi: 10.1016/j.jvcir.2019.102578.

[17] C. Zhu, C. U. Idemudia, and W. Feng, "Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques," *Inform Med Unlocked*, vol. 17, p. 100179, 2019, doi: 10.1016/j.imu.2019.100179.

[18] P. N. Srinivasu, U. Sirisha, K. Sandeep, S. P. Praveen, L. P. Maguluri, and T. Bikku, "An Interpretable Approach with Explainable AI for Heart Stroke Prediction," *Diagnostics*, vol. 14, no. 2, p. 128, Jan. 2024, doi: 10.3390/diagnostics14020128.

[19] M. Alruily, S. A. El-Ghany, A. M. Mostafa, M. Ezz, and A. A. A. El-Aziz, "A-Tuning Ensemble Machine Learning Technique for Cerebral Stroke Prediction," *Applied Sciences*, vol. 13, no. 8, p. 5047, Apr. 2023, doi: 10.3390/app13085047.

[20] N. Alageel, R. Alharbi, R. Alharbi, M. Alsayil, and L. A. Alharbi, "Using Machine Learning Algorithm as a Method for Improving Stroke Prediction," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 4, 2023, doi: 10.14569/IJACSA.2023.0140481.

[21] N. Nezami, P. Haghighat, D. Gándara, and H. Anahideh, "Assessing Disparities in Predictive Modeling Outcomes for College Student Success: The Impact of Imputation Techniques on Model Performance and Fairness," *Educ Sci (Basel)*, vol. 14, no. 2, p. 136, Jan. 2024, doi: 10.3390/educsci14020136.

[22] A. Palanivinayagam and R. Damaševičius, "Effective Handling of Missing Values in Datasets for Classification Using Machine Learning Methods," *Information*, vol. 14, no. 2, p. 92, Feb. 2023, doi: 10.3390/info14020092.

[23] M. Alruily, S. A. El-Ghany, A. M. Mostafa, M. Ezz, and A. A. A. El-Aziz, "A-Tuning Ensemble Machine Learning Technique for Cerebral Stroke Prediction," *Applied Sciences*, vol. 13, no. 8, p. 5047, Apr. 2023, doi: 10.3390/app13085047.

[24] P. Kirichenko, P. Izmailov, and A. G. Wilson, "Last layer re-training is sufficient for robustness to spurious correlations," *arXiv preprint arXiv:2204.02937*, 2022.

[25] C. Tantithamthavorn, A. E. Hassan, and K. Matsumoto, "The Impact of Class Rebalancing Techniques on the Performance and Interpretation of Defect Prediction Models," *IEEE Transactions on Software Engineering*, vol. 46, no. 11, pp. 1200–1219, Nov. 2020, doi: 10.1109/TSE.2018.2876537.

[26] M. K. Suryadi, R. Herteno, S. W. Saputro, M. R. Faisal, and R. A. Nugroho, "Comparative Study of Various Hyperparameter Tuning on Random Forest Classification With SMOTE and Feature Selection Using Genetic Algorithm in Software Defect Prediction," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 2, Mar. 2024, doi: 10.35882/jeeemi.v6i2.375.

[27] R. T. Yunardi, R. Apsari, and M. Yasin, "Comparison of Machine Learning Algorithm For Urine Glucose Level Classification Using Side-Polished Fiber Sensor," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 2, no. 2, pp. 33–39, Jul. 2020, doi: 10.35882/jeeemi.v2i2.1.

[28] H. Wei, C. Hu, S. Chen, Y. Xue, and Q. Zhang, "Establishing a software defect prediction model via effective dimension reduction," *Inf Sci (N Y)*, vol. 477, pp. 399–409, Mar. 2019, doi: 10.1016/j.ins.2018.10.056.

[29] S. A. Sontakke, J. Lohokare, R. Dani, and P. Shivagaje, "Classification of Cardiotocography Signals Using Machine Learning," 2019, pp. 439–450. doi: 10.1007/978-3-030-01057-7_35.

[30] S. Kilmen and O. Bulut, "Scale Abbreviation with Recursive Feature Elimination and Genetic Algorithms: An Illustration with the Test Emotions Questionnaire," *Information*, vol. 14, no. 2, p. 63, Jan. 2023, doi: 10.3390/info14020063.

[31] A. Taner, M. T. Mengstu, K. Ç. Selvi, H. Duran, İ. Gür, and N. Ungureanu, "Apple Varieties Classification Using Deep Features and Machine Learning," *Agriculture*, vol. 14, no. 2, p. 252, Feb. 2024, doi: 10.3390/agriculture14020252.

[32] M. R. Mahmoudi, M. H. Heydari, S. N. Qasem, A. Mosavi, and S. S. Band, "Principal component analysis to study the relations between the spread rates of COVID-19 in high risks countries," *Alexandria Engineering Journal*, vol. 60, no. 1, pp. 457–464, Feb. 2021, doi: 10.1016/j.aej.2020.09.013.

[33] S. Ma, W. Cao, S. Jiang, J. Hu, X. Lei, and X. Xiong, "Design and implementation of SVM OTPC searching based on Shared Dot Product Matrix," *Integration*, vol. 71, pp. 30–37, Mar. 2020, doi: 10.1016/j.vlsi.2019.11.007.

[34] V. Umarani, A. Julian, and J. Deepa, "Sentiment Analysis using various Machine Learning and Deep Learning Techniques," *Journal of the Nigerian Society of Physical Sciences*, pp. 385–394, Nov. 2021, doi: 10.46481/jnsps.2021.308.

[35] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, Sep. 2020, doi: 10.1016/j.neucom.2019.10.118.

[36] B. Richhariya, M. Tanveer, and A. H. Rashid, "Diagnosis of Alzheimer's disease using universum support vector machine based recursive feature elimination (USVM-RFE)," *Biomed Signal Process Control*, vol. 59, p. 101903, May 2020, doi: 10.1016/j.bspc.2020.101903.

[37] B. Gaye, D. Zhang, and A. Wulamu, "Improvement of Support Vector Machine Algorithm in Big Data Background," *Math Probl Eng*, vol. 2021, pp. 1–9, Jun. 2021, doi: 10.1155/2021/5594899.

[38] N. H. Arif, M. R. Faisal, A. Farmadi, D. Nugrahadi, F. Abadi, and U. A. Ahmad, "An Approach to ECG-based Gender Recognition Using Random Forest Algorithm," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 2, Mar. 2024, doi: 10.35882/jeeemi.v6i2.363.

[39] I. Yoo, J. Bi, and X. Hu, "2019 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2019, San Diego, CA, USA, November 18-21, 2019," I. Yoo, J. Bi, and X. Hu, Eds., IEEE, 2019. [Online]. Available: https://ieeexplore.ieee.org/xpl/conhome/8965270/proceeding

[40] S. Bhanumathi and S. N. Dr. Chandrashekara, "Impute, Select, Decision Tree and Naïve Bayes (ISE-DNC): An Ensemble Learning Approach to Classify the Lung Cancer," *SSRN Electronic Journal*, 2020, doi: 10.2139/ssrn.3667438.

[41] P. Sandhya, V. Spoorthy, S. G. Koolagudi, and N. V. Sobhana, "Spectral Features for Emotional Speaker Recognition," in *2020 Third International Conference on Advances in Electronics, Computers and*

Communications (ICAECC), IEEE, Dec. 2020, pp. 1–6. doi: 10.1109/ICAECC50550.2020.9339502.
[42] T. T. A. Putri, S. Sriadhi, R. D. Sari, R. Rahmadani, and H. D. Hutahaean, "A comparison of classification algorithms for hate speech detection," *IOP Conf Ser Mater Sci Eng*, vol. 830, no. 3, p. 032006, Apr. 2020, doi: 10.1088/1757-899X/830/3/032006.
[43] P. Boedeker and N. T. Kearns, "Linear Discriminant Analysis for Prediction of Group Membership: A User-Friendly Primer," *Adv Methods Pract Psychol Sci*, vol. 2, no. 3, pp. 250–263, Sep. 2019, doi: 10.1177/2515245919849378.
[44] C. Ricciardi *et al.*, "Linear discriminant analysis and principal component analysis to predict coronary artery disease," *Health Informatics J*, vol. 26, no. 3, pp. 2181–2192, Sep. 2020, doi: 10.1177/1460458219899210.
[45] D. Valero-Carreras, J. Alcaraz, and M. Landete, "Comparing two SVM models through different metrics based on the confusion matrix," *Comput Oper Res*, vol. 152, p. 106131, Apr. 2023, doi: 10.1016/j.cor.2022.106131.
[46] M. Ojala and G. C. Garriga, "Permutation Tests for Studying Classifier Performance," *Journal of Machine Learning Research*, vol. 11, no. 62, pp. 1833–1863, 2010, [Online]. Available: http://jmlr.org/papers/v11/ojala10a.html
[47] N. Thanh Nhu, D. Y.-T. Chen, and J.-H. Kang, "Identification of Resting-State Network Functional Connectivity and Brain Structural Signatures in Fibromyalgia Using a Machine Learning Approach," *Biomedicines*, vol. 10, no. 12, p. 3002, Nov. 2022, doi: 10.3390/biomedicines10123002.
[48] T. R. Gadekallu, N. Khare, S. Bhattacharya, S. Singh, P. K. R. Maddikunta, and G. Srivastava, "Deep neural networks to predict diabetic retinopathy," *J Ambient Intell Humaniz Comput*, vol. 14, no. 5, pp. 5407–5420, May 2023, doi: 10.1007/s12652-020-01963-7.
[49] S. Cheon, J. Kim, and J. Lim, "The Use of Deep Learning to Predict Stroke Patient Mortality," *Int J Environ Res Public Health*, vol. 16, no. 11, p. 1876, May 2019, doi: 10.3390/ijerph16111876.
[50] Md. Ashrafuzzaman, S. Saha, and K. Nur, "Prediction of Stroke Disease Using Deep CNN Based Approach," *Journal of Advances in Information Technology*, vol. 13, no. 6, 2022, doi: 10.12720/jait.13.6.604-613.
[51] P. A. Riadi, M. R. Faisal, D. Kartini, R. A. Nugroho, D. T. Nugrahadi, and D. B. Magfira, "A Comparative Study of Machine Learning Methods for Baby Cry Detection Using MFCC Features," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 1, Jan. 2024, doi: 10.35882/jeeemi.v6i1.350.
[52] N. Z. Al Habesyah, R. Herteno, F. Indriani, I. Budiman, and D. Kartini, "Sentiment Analysis of TikTok Shop Closure in Indonesia on Twitter Using Supervised Machine Learning," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 2, Apr. 2024, doi: 10.35882/jeeemi.v6i2.381.

AUTHOR BIOGRAPHY

Arya Syifa Hermiati, originally from Martapura, South Kalimantan. After graduating high school, she continued her education at the university level. Since 2020, she has been studying in the Computer Science Study Programme at Lambung Mangkurat University. Her current field of research is in data science. In addition, she had an internship as a junior researcher who used her skills to analyze data and conduct research in the field of education. Her final project was to conduct research centered on stroke disease prediction.

Rudy Herteno is currently a lecturer in the Faculty of Mathematics and Natural Science, at Lambung Mangkurat University. He received his bachelor's degree in Computer Science from Lambung Mangkurat University and a master's degree in Informatics from STMIK Amikom University. His research interests include software engineering, software defect prediction, and deep learning.

Fatma Indriani is a lecturer in the Department of Computer Science, Lambung Mangkurat University. She completed her undergraduate program in the Informatics Department at Bandung Institute of Technology. In 2008, she started working as a lecturer in the Computer Science department at Lambung Mangkurat University. She then obtained her master's degree at Monash University, Australia (2012) and a doctorate degree in Bioinformatics at Kanazawa University, Japan (2022). Her research interest is applied Data Science.

Triando Hamonangan Saragih is a lecturer at the Department of Computer Science, Lambung Mangkurat University. His research interests center on Data Science and Computer Networks. He completed his bachelor's degree in Computer Science at Brawijaya University, Malang, in 2016 and a master's degree in Computer Science at Brawijaya University, Malang, in 2018. His research field is Data Science.

Muliadi is a lecturer in the Department of Computer Science at Lambung Mangkurat University, where he specializes in Artificial Intelligence, Decision Support Systems, and Data Science. His academic journey began with a bachelor's degree in Informatics Engineering from STMIK Akakom in 2004 and a master's degree in Computer Science from Gadjah Mada University in 2009. With expertise in Data Science, he also brings valuable skills in Start-up Business Development, Digital Entrepreneurship, and Data Management Staff.

Dr. Triwiyanto received the B.S. degree in Physics from Airlangga University, Indonesia, M.S. degree in Electronic Engineering from the Institut Teknologi Sepuluh Nopember Surabaya, Indonesia, in 2004, and a Ph.D. degree in Electrical Engineering from Gadjah Mada University, Yogyakarta, Indonesia, in 2018. From 1998 to 2004, he was a Senior Lecturer with the Microcontrollers Laboratory. Since 2005, he has been an Associate Professor with the Medical Electronics Technology Department, Health Polytechnic Ministry of Health Surabaya, Indonesia. In 2018, Triwiyanto received the best Doctoral Student award from Gadjah Mada University. Additionally, he is Editor-in-chief in several peer review journals and chairman Technical Programme Committee at several International Conferences. His current research interests include a microcontroller, electronics, biomedical signal processing, machine learning, rehabilitation engineering, and surface electromyography (sEMG)-based physical human-robot interactions. I have published over 80 papers in reputable journals and conferences. My Scopus h-index is 13 (June, 2024), which reflects the impact of my research. I am a proud member of the IEEE (Institute of Electrical and Electronics Engineers). You can find my complete publication record and more details on my Scopus profile here