STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **13**, May 2025, pp 1832–1846.
Published online in International Academic Press (www.IAPress.org)

Acquiring Independent Components through Hybrid PCA and ICA to Enhance the Classification Performance of Decision Tree

Achmad Efendi [1,*], Zuraidah Zuraidah [2], Dewi S. Susanti [3], Naomi N. Debataraja [4], Ratno B.E. Wibowo [5], Samingun Handoyo [1,6]

[1]*Statistics Department, Universitas Brawijaya, Malang, East Java, Indonesia*
[2]*Islamic Banking Study Program, State Islamic Institute of Kediri, Kediri, East Java, Indonesia*
[3]*Statistics Study Program, Lambung Mangkurat University, Banjarbaru, South Kalimantan, Indonesia*
[4]*Statistics Study Program, Tanjungpura University, Pontianak, West Kalimantan, Indonesia*
[5]*Mathematics Department, Universitas Brawijaya, Malang, East Java, Indonesia*
[6]*EECS-IGP Department, National Yang Ming Chiao Tung University, Hsinchu, Taiwan, China*

Abstract The Principal Component Analysis (PCA) is widely used for modeling in both statistical and machine learning domains. However, PCA's orthogonal components may not always be independent. This research aims to compare PCA and Independent Component Analysis (ICA) using simulation and empirical data and to evaluate a Decision Tree (DT) model. Two scenarios of simulation data with linear and nonlinear relationships, along with two empirical datasets were analyzed. PCA was used to project the dataset, while ICA was applied to the 6th to 10th and the 5th to 9th principal components. Both PCA and ICA resulted in projection data with zero correlation values. Scatter plots of PCA projection on nonlinear simulation data indicated consistent underlying patterns, whereas ICA projection revealed sparse patterns on both simulation datasets. The DT model utilizing 7 independent components emerged as the optimal model, displaying superior performance across accuracy, precision, recall, F1 score, Mathew's Correlation Coefficient, and Area Under Curve metrics.

Keywords Confusion matrix, Independent component, Mathew's Correlation, Variable extraction, Empirical data, Simulation

DOI: 10.19139/soic-2310-5070-2175

1. Introduction

Deployment of statistical modeling in real life has wider spread in the few last decades. Statistical modeling must compromise to produce a model that is not only robust but also has a high performance [1]. Statistical models are usually evaluated based on the goodness of fit criteria to find the best one [2]. The modeling tends to focus more on exploration either the mathematical elaboration or the estimation parameters stage [3]. However, the implementation of the acquired best model has not gotten suitable attention, even the model performance in the out-sample data is not under concern. On the other hand, a popular modeling approach known as Machine Learning (ML) employs the model performance on the out-sample data as the criterion to determine the best model [4]. This study will be conducted in a balancing exploration between the modeling process and applying the model by adapting the assessing model in the ML approach.

The ML models are categorized into supervised and unsupervised learning which depend on the availability of the response variable on the building model [5]. All predictive models are supervised type due to they always involve the response variable which can be either a numerical variable leading to the regression model or a categorical variable leading to the classification model [6]. The absence of the response variable in the model input

*Correspondence to: Achmad Efendi (Email: a_efendi@ub.ac.id)

ISSN 2310-5070 (online) ISSN 2311-004X (print)

leads to the unsupervised learning method including the clustering [7], Analytical Hierarchical Process (AHP) model from Bayesian network for data of event logs failure of industrial machine components [8], the variable extraction for dimensionality reduction [9], and so forth. To acquire a model with the best performance, the input variables or predictor variables are expected to have characteristics that are not only independent of each other but also relevance to the response variable [10]. The relevant variables mean that the response variable depends on them [11]. In fitting the multiple regression model, the forward selection [12] or backward elimination [13] is a very popular approach. Unfortunately, When the model candidate has either complicated structure or involved a very high dimension, it was impossible to employ filter variable selection due to the very high computation cost [14]. The variable extraction method such as PCA has an important role in producing the orthogonal components to provide the model input expected for the independent predictor variables [15]. The PCA orthogonal components are independent of each other when the PCA input variables have the joint Gaussian density function [16]. Nevertheless, the real-world data is almost impossible to have the joint Gaussian distribution as the PCA input that will lead the PCA transformation produces orthogonal components that are not always independent of each other. The variable extraction method that can ensure production of the independent components for anything the data input underlying distribution is needed to acquire a better model input (predictor variables) that can lead to a higher model performance. ICA is one of the variable extraction methods that can produce independent components but it cannot serve for dimension reduction purposes [17].

The simple and interpretable model is preferred not only by the model developer but also by the model user. Logistic regression (LR) is one of the classification models that is very popular due to its simple development besides satisfactory performance. Many works implemented successfully the LR model including Handoyo, et al [18] implemented LR to classify fraudulent firms, Nugroho, et al [19] compared the performance between the LR and Learning Vector Quantization on various datasets, Bittencourt et. al. [20] applied both Gaussian Maximum Likelihood and LR to classify hyper-spectral image data, and so forth. The LR performance was satisfactory in some certain datasets employed in [18-20]. Unfortunately, the good performance of the LR model is limited in the datasets with the instance classes separated by a linear decision boundary. The LR model also sensitively drifts the model output when a small change occurs in the threshold value. Meanwhile, a decision tree (DT) is categorized as a nonlinear classification model that is easily developed and interpreted. DT is developed using the principle of dividing and conquer, and it is interpreted by the tree traversal from the root node to the leaf node [21]. It suffers from the over-fitting issue that can be tackled by tree pre- and post-pruning [22]. Many successful implementations of the DT classifier on medical data have been conducted by some researchers including Hu [23], that improved the ID3 algorithm to yield effective of DT in medical data classification, Al Fryan, et al [24] acquired the effectiveness of DT in processing medical data by using internet of things (IoT) and artificial intelligence technologies, Hasan, et al [25] implemented a single DT on three popular medical datasets, and so on. However, almost all the works did not care about the quality of the DT input such as enforcing the DT input into the relevant and independent one.

The PCA variable extraction method projects the set of predictor variables into the orthogonal components that can be employed as a dimension reduction technique. Although the principal components are orthogonal to each other, they are not ensured independent among them. On the other hand, the ICA variable extraction can produce the independent components but it cannot be employed for dimension reduction purposes. Some researches using PCA and ICA separately, but combined it with decision tree. PCA classifier and DT algorithm were used to classify medical record data [26] while ICA and DT were used to classify the ECG signal de-noising [27]. This research aims at demonstrating the distinction of variable extraction produced by both PCA and ICA through simulation data and at comparing the DT performance on the different input data that are produced by either the PCA or ICA transformation. The generating datasets consist of two scenarios namely the first one with a linear pattern and the second one with a nonlinear pattern. The various number of components either the PCA or ICA are employed as the input data to train and evaluate the DT model, and all of the acquired DT models are evaluated in performance metrics namely accuracy, precision, recall, F1 score, Mathew's Correlation Coefficient (MCC), and Area Under Curve (AUC). Furthermore, the remaining part of the paper is organized: Section 2 presents the theoretical framework of the proposed method, Section 3 describes the generating dataset, the empirical dataset, and the research stages, the results and discussion are given in Section 4, and the last section presents the conclusion and recommendation.

2. Literature Reviews

The section discusses about the theoretical framework and associated formula of feature extraction of both PCA and ICA, building the decision tree model; and evaluating the model's performance.

2.1. The Orthogonal Components Produced by the PCA projection

Principal component analysis (PCA) is a feature extraction method that is widely employed as a dimensionality reduction technique. PCA linearly projects the input data from a high-dimensional space to a low-dimensional space [28]. The lower dimensional space acquired by PCA is known as the principal subspace, and a projection direction/vector found by PCA is called the principal component where the projection is composed of one or a few principal components [29]. There are formally two types of the PCA definition. The first one is the orthogonal projection of the data onto a lower-dimensional linear space such that the variance of the projected data maximized, and the other one is the linear projection that minimizes the average projection cost, defined as the mean squared distance between the data points and their projections, However, both definitions lead to the same PCA algorithm [30].

The following elaboration on how PCA works is based on the first PCA definition. Consider a data set of observations $\{xn\}$ where $n = 1, 2, ..., N$ and xn in the Euclidean space with d-dimension. The PCA goal is to project the data onto a space having dimensionality $m \leq d$ while maximizing the variance of the projected data [31]. Firstly, consider the projection onto a one-dimensional space $(m = 1)$, and define the direction of this space using a d-dimensional unit vector u_1 and $u_1^T u_1 = 1$. The data mean and projected data mean are given in the equation (1) while the equation (2) presents its variance.

$$\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n \text{ and } \hat{x} = u_1^T \bar{x} \tag{1}$$

$$VarX = \frac{1}{N}\sum_{n=1}^{N} \langle u_1^T x_n - u_1^T \bar{x} \rangle^2 = u_1^T S u_1 \tag{2}$$

The maximum variance can be acquired through a constrained optimization problem formulated as the following: Maximize $u_1^T S u_1$; Subject to $u_1^T u_1 = 1$, by introducing a Lagrange multiplier to convert the constrained optimization problem to an unconstrained one, where the produced Lagrange function is

$$L(u_1) = u_1^T S u_1 + \lambda_1 (1 - u_1^T u_1) \tag{3}$$

By setting the derivative to u_1 equal to zero, subsequently, it is obtained $Su_1 = \lambda_1 u_1$, where u_1 and λ_1 are respectively the eigen vector and eigen value of S. Furthermore, the objective in the constrained optimization problem of PCA is $u_1^T S u_1 = \lambda_1$. The maximum variance of the projected data is equal to the largest eigenvalue λ_1 of the covariance matrix S. The projection vector u_1 is the eigenvector of S corresponding to the largest eigen value λ_1. The eigenvector u_1 is known as the first principal component, which is used to project a data point x_n via $u_1^T x_n$.

Consider the general case of an m-dimensional projection space. The m eigen vectors u_1,u_2, and u_m are the optimal linear projection of the data covariance matrix S corresponding to the m largest eigenvalues λ_1, λ_2 and λ_m. S is the covariance matrix which is a symmetric matrix that always (orthogonally) diagonalizable. That is, for any symmetric matrix $A \in R^{mxm}$, there exists an orthogonal matrix $Q = [q_1, q_2, ..., q_m]$ and a diagonal matrix $\Lambda = diag(\lambda_1, \lambda_2, ..., \lambda_m)$, both real and square, such that $A = Q\Lambda Q^T$. It has pointed out that λ_i's are the eigen values of A and $q_i's$ the corresponding eigen vectors (which are orthogonal to each other and have unit norm). Thus, such a factorization is called the eigen decomposition of A, and is also called the spectral decomposition of A [31].

2.2. The Independent Components Produced by the ICA Projection

The input variable should be the variables characterized to have a statistically independent manner so that they capture more structure or pattern of data. The statistical independent means $P(x,y) = P(x)P(y)$ that implies $E[f(x)g(y)] = E[f(x)]E[g(y)]$ for all measurable function f and g. Essentially, x and y are independent of each other means that one can't tell anything about x if one observes y and vice versa. Both X and Y are said uncorrelated if $E[XY] = E[X]E[Y]$ that does not imply X and Y independent, but if X and Y are independent that imply both variables are uncorrelated [32].

Consider two signals presented in the simultaneous linear equation system at the equation (4) and the observed on the equation (5) as follows

$$\begin{aligned} x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) \\ x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t) \end{aligned} \tag{4}$$

$$\begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} = \begin{pmatrix} x_1(1) \\ x_2(1) \end{pmatrix}, \begin{pmatrix} x_1(2) \\ x_2(2) \end{pmatrix}, ..., \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} \tag{5}$$

By employing both equation (4) and (5), it desired to acquire the source signals of $s_1(t)$ and $s_2(t)$ given on the equation (6) that are independent of each other, but the a_ij coefficients are also unknown.

$$\begin{pmatrix} s_1(t) \\ s_2(t) \end{pmatrix} = \begin{pmatrix} s_1(1) \\ s_2(1) \end{pmatrix}, \begin{pmatrix} s_1(2) \\ s_2(2) \end{pmatrix}, ..., \begin{pmatrix} s_1(t) \\ s_2(t) \end{pmatrix} \tag{6}$$

The observed signals are used for estimating both of $s_i(t)$ and also a_ij. PCA performs the low-rank matrix factorization for compression while ICA performs the full-rank matrix factorization. Both PCA and ICA perform linear transformation and matrix factorization. Maximum Likelihood ICA algorithm requires knowing densities of hidden sources $f_i(.)$ where the equation involving a matrix form is as follows [33]: $x(t) = As(t)$; $s(t) = Wx(t)$ where $A^{-1} = W = [(w_1 ... w_M)] \in R^{MxM}$.

$$L = \sum_{t=1}^{T} P(x(t)) = \sum_{t=1}^{T} P(As(t)) = \sum_{t=1}^{T} (log|W| + log(Ps(t))) \text{ where } s_i(t) = w_i x(t)$$

$$L = Tlog|W| + \sum_{t=1}^{T}\sum_{i=1}^{M} log f_i(w_i x(t))$$

The maximum likelihood estimator is acquired by maximizing the likelihood function at the equation (6) through its partial derivative to W and setting it up to zero for solving the problem analytically. Unfortunately, it is an impossible task to obtain the analytical solution so the partial derivative is considered as the W change

$$\frac{\partial L}{\partial w_{ij}} = \frac{\partial Tlog|W| + \sum_{t=1}^{T}\sum_{i=1}^{M} log f_i(w_i x(t))}{\partial w_{ij}}$$

$$\frac{\partial L}{\partial w_{ij}} = T(W^T)^{-1}_{ij} + \sum_{t=1}^{T} \frac{f_i^{`}(s_i(t))}{f_i(s_i(t))}(x_j(t))$$

$$\Delta W \propto [W^T]^{-1} + \frac{1}{T}\sum_{t=1}^{T} g(Wx(x)x^T(x)), \text{where } g_i = f_i^{`}/f_i.$$

Subsequently, the W matrix (the ICA projection weights) is calculated using iterative method where the current W can be updated iteratively using ΔW with the μ learning rate until the criteria or determined threshold value is fulfilled [34]. The formula is given at the equation (7) as follows

$$W = W + \mu \Delta W. \tag{7}$$

The numerical solution is acquired when the W value has converged or the stopping criterion has been fulfilled.

2.3. Building Decision Tree Classification Model

The decision tree model implements the basic principle of dividing and conquer to a collection of instances repeatedly to produce some subset that maximizes the overall class purity score [35]. Decision tree construction is carried out by recursively partitioning the set of instances, namely selecting a variable that has the highest ability to separate the set of instances into different classes. The decision tree components include the root node, tree branches, internal node, and leaf nodes. All leaf nodes contain the distribution of class labels and tree branches represent the results of testing instances on a splitting variable in the internal node. At each internal node, a splitting variable is selected to divide the training samples into different subsets. Furthermore, an instance with an unknown class label is classified by tracing the appropriate path from the root node to the leaf node [36]. A splitting variable is determined by using an objective function that measures a degree of purity on each variable and chooses one producing the "purest" nodes. The objective function to measure the nodes purity degree is defined at the equation (8) as follows:

$$S(y) = \sum_{S_i}^{|D|} \sum_{x_j}^{p} 1(y_j \neq \hat{y}_j) \tag{8}$$

for $S = (S_1, S_2, ..., S_k)$, i.e. k subsets; where $|D|$ is the instances number, and each instance has X_i, for $i = 1, 2, 3, \ldots, p$ variables. The constructing of a decision tree is started by picking up a variable and the associated score value that optimizes a criterion such as information gain (IG). The IG is calculated by using the entropy concept which is a value known as the smallest possible number of bits required to transmit a stream of symbols drawn from $X's$ distribution. The dataset D containing C classes has the entropy value denoted by $H(D)$ with the formula at the equation (9) as follows:

$$H(D) = -\sum_{i=1}^{C} p_i log_2(p_i) \tag{9}$$

Where p_i is the i class probability, and the partition process is based on the variable F with the highest purity degree. Subsequently, the partition of the training data produces the $D_1 \sim D_k$ subsets, and the entropy after splitting on the variable F called $H(D, F)$ is defined in the equation (10)

$$H(D, F) = \sum_{i=1}^{k} \frac{|D_i|}{|D|} H(D_i) \tag{10}$$

The IG on all available variables is calculated using the equation (11),

$$IG(D, F) = H(D) - H(D, F). \tag{11}$$

On the proceed node called the child node the above step is repeated until the stopping criteria are fulfilled such as the tree depth has been reached or the minimum instance number on the leaf nodes has been fulfilled [37] .

2.4. The Metrics Model's Performance

After the classifier model candidate was trained using the training data with various inputs, the model performance in the corresponding testing data is a criterion to decide the best model. The performance metrics are calculated based on the confusion matrix produced by the trained model in the testing data. The confusion matrix describes the model capability in the prediction of instances with unknown label classes which are instances that come from the out-sample data. The employing of the model's performance in the testing data as a model goodness of fit is a popular practice in the machine learning approach. Some performance metrics are usually employed to evaluate the classifier model fairly leading to the best one. The simple metric and very popular in assessing the classifier model performance is the accuracy metric. It is calculated as a ratio between the total of instances correctly classified and the total of instances in the testing data [38]. The formula of accuracy metric is given in the equation (12) as

follows:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{12}$$

The MCC (Matthews Correlation Coefficient) metric is widely used in the evaluation of a classifier model performance in biomedical research and it is calculated based on the confusion matrix elements [39]. Another performance metric known as AUC (Area Under ROC Curve) is obtained using a numerical integration approach [40]. Both MCC and AUC metrics have a range value between 0 and 1 describing a binary classifier ability in classifying instances of the positive class (label 1) from instances of the negative class (label 0). Both metrics are calculated by using the equation (13) and (14) as the following:

$$MCC = \frac{(TPTN - FPFN)}{\sqrt{((TP + FP)(TP + FN)(TN + FP)(TN + FN))}} \tag{13}$$

$$AUC = \int_0^1 ROC(x)dx \tag{14}$$

TN, FN, FP, and TP respectively stand for True Negative, False Negative, False Positive, and True Positive. Both MMC and AUC metrics can measure the classifier model sensitivity well.

3. Materials and Methods

The research employs two types of datasets namely the simulated and empirical datasets. We have one simulated dataset and two empirical datasets. The first empirical dateset is about lung cancer while the second are liver cirrhosis data. Indeed, these two datasets are health related data, but the first dataset contain majority of discrete variables while the second contain more about continuous variables. This is aimed at covering all types of data such that this methodology could also be applied to several types of datasets, including areas of banking, industries, socio economies, and so forth.

There are two scenarios for the simulation dataset. The first scenario randomly generates two variables that satisfy a joint Gaussian density function with a high linear relationship. The second scenario is to square both variables in the first dataset to acquire two variables with a high nonlinear relationship. This simulated dataset was prepared to show the distinctions of projection components produced by the PCA and ICA variable extraction in a data source with a high linear relationship while the second dataset was employed to show the effect of both PCA and ICA extracted variables in a data source with a high nonlinear relationship. Both of the generated simulation data can be obtained in the link: https://github.com/saminghan/simulate/blob/main/scenario_1.csv,and https://github.com/saminghan/simulate/blob/main/scenario_2.csv.

The first empirical dataset in this study about lung cancer comes from the Kaggle, the world's largest data science community, which can be downloaded at the link: https://www.kaggle.com/datasets/fdcellat/cancer-prediction-dataset. The dataset represents a collection of responses from a university-conducted survey for studying the potential risk factors for lung cancer. The survey covers a variety of demographic, lifestyle, and health-related questions. The dataset was presented in 12 columns consisting of the ID column as the key index, the Cancer column as the response variable, and 10 columns of predictor variables which are Gender, Age, Marital Status, Children, Smoker, Employed, Years Worked, Income Level, social media, Online Gaming.

While the second empirical dataset are about liver cirrhosis and can be downloaded via this link: https://archive.ics.uci.edu/dataset/225/ilpd+indian+liver+patient+dataset. These are health and medicine, multivariate, containing classification data, with integer and real type data, and containing information of 583 patients and 10 features. The dataset contains records of 416 patients diagnosed with liver disease and 167 patients without liver disease. This information is contained in the class label named 'Selector'.

There are 10 variables per patient: age, gender, total Bilirubin, direct Bilirubin, total proteins, albumin, A/G ratio, SGPT, SGOT and Alkphos. Of the 583 patient records, 441 are male, and 142 are female. The current dataset has been used to study the differences in patients across US and Indian patients that suffer from liver diseases, gender-based disparities in predicting liver disease, as previous studies have found that biochemical markers do not have the same effectiveness for male and female patients. The data contains no missing values.

The research stages, for each type of data, are summarized as the following.

Simulation data:

1. Generating both simulation data by setting the mean vector and covariance matrix.
2. Projecting both simulation data with PCA
3. Calculating the mean vectors and covariance matrices of both PCA projection data
4. Drawing the scatter plots of both PCA projection data
5. Projecting both simulation data with ICA
6. Calculating the mean vectors and covariance matrices of both ICA projection data
7. Drawing the scatter plots of both ICA projection data

Empirical data:

1. Encoding the categorical variables into discrete value
2. Commensuration of measures to the numerical variables
3. Normalizing into Z score to predictor variables
4. Projecting PCA to the Z score input data to acquire the orthogonal components
5. Creating the input-output pairs through merging of orthogonal components and the response variable
6. Dividing the input-output pairs into the training and testing data
7. Training the Decision tree model using the training data
8. Computing the confusion matrix on the testing data
9. Calculating the performance metrics on the testing data.
10. Projecting ICA to the PCA transformation data to acquire the independent components
11. Creating the input-output pairs through merging of independent components and the response variable
12. Repeating the stage 6 to 9.

4. Results and Discussion

The simulated datasets are employed to prove the distinction among orthogonality, uncorrelated, and independent concepts between two variables. The empirical datasets were employed as a case study for the variable extraction and implementation of DT classifier model. The various number of components are picked up and merged with the response variable to form the input-output pairs that will be divided into the training and testing data. The DT classifier will be trained and evaluated using the training and testing data respectively.

4.1. The Uncorrelated versus Independent Components

The numerical properties of both datasets produced through simulation are presented in row 1 and row 4 of Table 1 while the properties of both PCA and ICA projection were given in the two succeeding rows of each dataset. The scatter plots between the two variables given in Figure 1 are employed to explore different properties of the projection datasets yielded through the PCA and ICA variable extraction methods.

Consider rows 1 to 3 in Table 1, and the first row of the subplot in Figure 1. Both X and Y variables have a high correlation value of -0.9138. The subplot (1,1) presents the scatter plot of 500 points of $X\&Y$ pairs that form similar to a linear line with a negative gradient. Both PCA and ICA variable extraction yield the X components with variance values that are slightly the same. However, the variances of Y components in both extraction methods are very different. The variance of the Y component extracted by the ICA is greater than two times the variance

Table 1. The statistical properties of simulated datasets and their projected by both PCA and ICA variable extraction

Dataset	Shape	Mean	Covariance	Correlation
$X\&Y$	(500, 2)	[-0.1376, 0.0684]	$\begin{pmatrix} 6.2771 & -4.5442 \\ -4.5442 & 3.9398 \end{pmatrix}$	-0.9138
PCA to $X\&Y$	(500, 2)	[0.0, -0.0]	$\begin{pmatrix} 1.9176 & 0 \\ 0 & 0.0864 \end{pmatrix}$	0
ICA to $X\&Y$	(500, 2)	[[-0.0, 0.0]	$\begin{pmatrix} 1.8266 & 0 \\ 0 & 0.1774 \end{pmatrix}$	0
$X_2\&Y_2$	(500, 2)	[6.2834, 3.9366]	$\begin{pmatrix} 70.6243 & 36.0878 \\ 36.0878 & 27.2255 \end{pmatrix}$	0.823
PCA to $X_2\&Y_2$	(500, 2)	[0.0, 0.0]	$\begin{pmatrix} 0.002 & 0 \\ 0 & 0.002 \end{pmatrix}$	0
ICA to $X_2\&Y_2$	(500, 2)	[0.0, -0.0]	$\begin{pmatrix} 0.002 & 0 \\ 0 & 0.002 \end{pmatrix}$	0

of the Y component extracted by the PCA. Both extraction variable methods yield the same correlation value as 0. The subplot (1,2) shows the scatter plot of the extracted variable by the PCA that displays the first and second components are orthogonal to each other. On the other hand, the subplot (1,3) displays the extracted variables by ICA that are completely sparse which means the ICA method ensures in production of the independent variables.

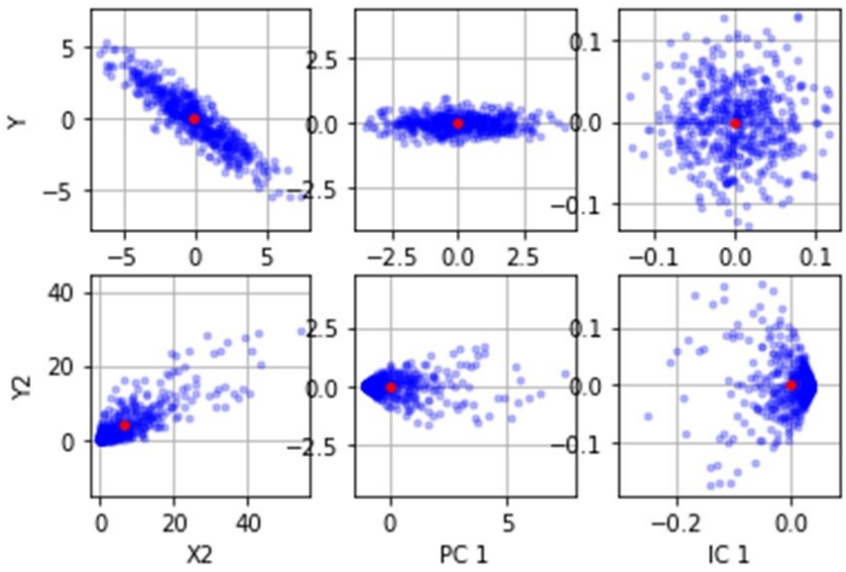

Figure 1. Plot of the actual versus prediction values of the ARDL lag 3 with original data in the testing dataset

Furthermore, consider rows 4 to 6 in Table 1, and the second row of the subplot in Figure 1. Both generated variables of $X_2\&Y_2$ are used for a representation of signals with a nonlinear relationship. They have a correlation value of 0.823. The components produced by both PCA and ICA methods have the same correlation value as 0 where they are on row 5 and row 6 respectively. Nevertheless, the scatter plots on the subplot (2, 2) and subplot (2, 3) describe the different patterns. The components extracted by the PCA have a pattern that is relatively the same

Table 2. Encoding labels of the categorical variable and the class distribution of each variable of lung cancer data

Variable	Original label	New label	Distribution
Gender	['Male', 'Female']	[0, 1]	[515, 485]
Marital Status	['Married', 'Single', 'Widowed', 'Separated']	[1, 0, 2, 3]	[267, 259, 242, 232]
Income Level	['Low', 'High', 'Medium']	[0, 2, 1]	[669, 166, 165]
Smoker	['No', 'Yes']	[0, 1]	[512, 488]
Employed	['No', 'Yes']	[0, 1]	[514, 486]
Social media	['Yes', 'No']	[1, 0]	[524, 476]
Online Gaming	['No', 'Yes']	[0, 1]	[508, 492]
Cancer	['Yes', 'No']	[1, 0]	[776, 224]

Table 3. The Eigen value and total variance explained of lung cancer data

Source	PC1	PC2	PC3	PC4	PC5	PC6	PC7
Eigen	0.2278	0.1113	0.1100	0.1045	0.1025	0.0975	0.0889
Variance	0.2278	0.3391	0.4490	0.5535	0.6560	0.7534	0.8423
Source	PC8	PC9	PC10				
Eigen	0.085	0.0519	0.0207				
Variance	0.9273	0.9793	1				

as the $X_2\&Y_2$ original pattern on the subplot (2,1) projected to 45 degrees. The components produced by the ICA on the subplot (2,3) show the scatter plot with sparse distribution which implies both components independent of each other. The PCA projection cannot produce independent components when the projected data have a nonlinear pattern.

4.2. The Empirical Dataset Exploration and Description

The lung cancer dataset employed in the research consists of 10 predictor variables i.e. 7 categorical variables and 3 numerical variables. The encoding of the categorical variables and the commensurate measurement to the numerical variable are conducted to prepare the dataset for the modeling stage. Table 2 presents the categorical classes and their associating labels that are followed by the class distribution. The 'Cancer' variable acts as the response variable with the binary classes i.e. the 'Yes' label means the patient suffered the cancer disease and the 'No' label means otherwise. The class distribution of the response variable is [776, 224] associating to the ['Yes', 'No'], or [1, 0] labels respectively. There are 5 predictors with 2 label categories, 1 predictor variable with 3 categories, and one predictor variable with 4 categories where their label names, encoding, and distribution are given in columns 2 to 4 in Table 2. On the other hand, there are 3 numerical variables namely 'Age', 'Children', and 'Years worked' variables that are conducted by the Min-max transformation to acquire the commensurate measures in the range value of [0,1].

Variable extraction was carried out in the post-processing of predictor variables using PCA and ICA methods. The obtained data projections will act as model input to build and evaluate the decision tree classifier. PCA variable extraction requires standardized input to ensure a zero mean. Table 3 presents the pairs of eigenvalues and the explained variance on the various numbers of PCA components. Because PCA with 6 principal components has explained 75.54% of the variance, this study picks up the number of PCA principal components from 6 to 10. Projection data set PCA has a role not only in building and evaluating decision tree classifiers but also in serving as the input of ICA variable extraction to obtain independent components. Furthermore, there are 10 datasets i.e. 5 datasets from PCA with principal component numbers from 6 to 10, and 5 datasets generated by ICA to the corresponding PCA principal components where they are employed to build and evaluate the decision tree model.

The liver cirrhosis dataset employed in the research consists of 10 predictors i.e. 9 numerical variables and 1 categorical variable. The encoding of the categorical variables and the commensurate measurement to the numerical

Table 4. The Decision tree performance on the components 6 to 10 in PCA for lung cancer data

Comp.	Conf.matrix	Accuracy	Precision	Recall	F1 score	MCC	AUC
PC 6	$\begin{pmatrix} 11 & 10 \\ 28 & 51 \end{pmatrix}$	0.6232	0.6239	0.6230	0.0811	0.1414	0.5847
PC 7	$\begin{pmatrix} 7 & 14 \\ 21 & 58 \end{pmatrix}$	0.6541	0.6523	0.6597	0.2341	0.0612	0.5538
PC 8	$\begin{pmatrix} 7 & 14 \\ 22 & 57 \end{pmatrix}$	0.6450	0.6452	0.6493	0.0682	0.0492	0.5274
PC 9	$\begin{pmatrix} 10 & 11 \\ 23 & 56 \end{pmatrix}$	0.6677	0.6669	0.6633	0.8470	0.1603	0.5925
PC 10	$\begin{pmatrix} 9 & 12 \\ 21 & 58 \end{pmatrix}$	0.6712	0.6720	0.6735	0.0662	0.1447	0.5814

Table 5. The Decision tree performance on the components 6 to 10 in ICA for lung cancer data

Comp.	Conf.matrix	Accuracy	Precision	Recall	F1 score	MCC	AUC
IC 6	$\begin{pmatrix} 9 & 12 \\ 24 & 55 \end{pmatrix}$	0.6454	0.6493	0.6475	0.7641	0.1081	0.5624
IC 7	$\begin{pmatrix} 8 & 13 \\ 24 & 55 \end{pmatrix}$	0.7721	0.7627	0.7607	0.3863	0.2697	0.6272
IC 8	$\begin{pmatrix} 8 & 13 \\ 24 & 55 \end{pmatrix}$	0.6360	0.6396	0.6335	0.1577	0.0674	0.5386
IC 9	$\begin{pmatrix} 10 & 11 \\ 19 & 60 \end{pmatrix}$	0.7008	0.6994	0.6961	0.2870	0.2116	0.6178
IC 10	$\begin{pmatrix} 8 & 13 \\ 17 & 62 \end{pmatrix}$	0.7072	0.7038	0.7103	0.7846	0.1559	0.5829

Table 6. Variables in Liver cirrhosis data

Variable	Minimum	Max.	Mean	Std.Dev
Age	4.00	90.00	44.75	16.19
TB	0.40	75.00	3.30	6.21
DB	0.10	19.70	1.49	2.81
Alkphos	63.00	2110.00	290.58	242.94
Sgpt	10.00	2000.00	80.71	182.62
Sgot	10.00	4929.00	109.91	288.92
Total Proteins	2.70	9.60	6.48	1.09
Albumin	0.90	5.50	3.14	0.80
Albumin and Globulin Ratio	0.00	2.80	0.94	0.33

Table 7. Variables in Liver cirrhosis data [Categorical variables]

Variable	Initial label	Label	Distribution
Gender	['Male', 'Female']	[1,0]	[441, 142]
Outcome/Liver cirrhosis	['Yes', 'No']	[1,2]	[416, 167]

Table 8. The Eigen value and total variance explained of liver cirrhosis data

Source	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
Eigen	0.3130	0.2278	0.1689	0.1168	0.0979	0.0262	0.0199	0.0153	0.0141
Variance	0.3130	0.5409	0.7097	0.8266	0.9244	0.9507	0.9705	0.9859	1.0000

Table 9. The Decision tree performance on the components from 5 to 9 PCA for liver cirrhosis data

Comp.	Conf.matrix	Accuracy	Precision	Recall	F1 score	MCC	AUC
PC 5	$\begin{pmatrix} 34 & 9 \\ 8 & 8 \end{pmatrix}$	0.7119	0.7176	0.7119	0.2854	0.7145	0.3547
PC 6	$\begin{pmatrix} 35 & 8 \\ 8 & 8 \end{pmatrix}$	0.7288	0.7288	0.7288	0.3140	0.7288	0.3430
PC 7	$\begin{pmatrix} 33 & 10 \\ 6 & 10 \end{pmatrix}$	0.7288	0.7523	0.7288	0.3686	0.7373	0.3038
PC 8	$\begin{pmatrix} 31 & 12 \\ 10 & 6 \end{pmatrix}$	0.6271	0.6414	0.6271	0.0926	0.6336	0.4520
PC 9	$\begin{pmatrix} 33 & 10 \\ 12 & 4 \end{pmatrix}$	0.6271	0.6119	0.6271	0.0182	0.6189	0.4913

Table 10. The Decision tree performance on the components from 5 to 9 ICA for liver cirrhosis data

Comp.	Conf.matrix	Accuracy	Precision	Recall	F1 score	MCC	AUC
IC 5	$\begin{pmatrix} 36 & 7 \\ 12 & 4 \end{pmatrix}$	0.6780	0.6452	0.6780	0.0995	0.6570	0.4564
IC 6	$\begin{pmatrix} 29 & 14 \\ 7 & 9 \end{pmatrix}$	0.6441	0.6932	0.6441	0.2160	0.6602	0.3815
IC 7	$\begin{pmatrix} 32 & 11 \\ 5 & 11 \end{pmatrix}$	0.7288	0.7659	0.7288	0.3969	0.7401	0.2842
IC 8	$\begin{pmatrix} 31 & 12 \\ 10 & 6 \end{pmatrix}$	0.6271	0.6414	0.6271	0.0926	0.6336	0.4520
IC 9	$\begin{pmatrix} 29 & 13 \\ 8 & 8 \end{pmatrix}$	0.6271	0.6698	0.6271	0.1604	0.6426	0.4128

variable are conducted to prepare the dataset for the modeling stage. Table 6 presents the numerical variables in the data, while categorical variable is gender, with 441 men and 142 women. The 'outcome'/liver cirrhosis variable acts as the response variable with the binary classes i.e. the 'Yes' label means the patient suffered liver disease and the 'No' label means otherwise. The class distribution of the response variable is [416, 167] associating to the ['Yes', 'No'], or [1, 0] labels respectively. Categorical variables are summarized in Table 7.

Table 8 presents the pairs of eigenvalues and the explained variance on the various numbers of PCA components for liver cirrhosis data. Since PCA with 5 principal components has explained 91.44% of the variance, this study picks up the number of PCA principal components from 5 to 9. Again, there are 10 datasets i.e. 5 datasets from PCA with principal component numbers from 5 to 9, and 5 datasets generated by ICA to the corresponding PCA principal components where they are employed to build and evaluate the decision tree model.

4.3. The Confusion Matrix and Performance Metrics of the Decision Tree Classifier

The decision tree (DT) classifier models are built using the training data i.e. 90% part, which is selected randomly from each dataset, and the 10% remaining part is used as the testing data for evaluating the DT performance. There are as many as 10 pairs of training and testing data that are obtained from 10 datasets. Each of the training data is

employed to train the DT model and the corresponding testing data is employed to evaluate its performance. The model performance metrics employed including the confusion matrix, the values of Accuracy, MCC, Precision, Recall, F1 Score and AUC are presented in Table 4 and Table 5 as well as Table 9 and Table 10.

For lung cancer data, Table 4 displays the performance metrics of 5 DT classifiers built and evaluated respectively using the training and testing data acquired by PCA projection with 6 to 10 principal components (PC). The DT model with the PCA extraction has an accuracy metric in the range of [62% to 67%] where the lowest and highest accuracy values are obtained respectively by the DT model with 6 and 10 PC inputs. The DT model with 10 PC inputs has an accuracy value of 67% meaning as many as 67 of 100 patients who come from the unknown class label are correctly classified by the model. This is in line with the result of precision and recall, [0.6270 to 0.6669] and [0.6230 to 0.6735] respectively. The DT models with PCA extraction have the MCC and AUC metrics with a range value of [0.0492 to 0.1603] and [0.5274 to 0.5925] respectively. The lowest MCC value is acquired in the DT model with 8 PC inputs, and the highest MCC value is acquired in the DT model with 9 PC inputs. whereas, the lowest AUC value is acquired in the DT model with 8 PC inputs, and the highest AUC value is acquired in the DT model with 9 PC inputs. It is clear, that the DT model with 9 PC inputs has the highest value of both the MCC and AUC metrics where the different result occurs in the accuracy metric that the best DT model has 10 PC inputs.

On the other hand, Table 5 presents the performance metrics of 5 DT classifiers built and evaluated respectively using the training and testing data extracted by the ICA projection with 6 to 10 independent components (IC). The DT models with the ICA extraction acquire the performance metrics in the range [63% to 77%], [0.0674 to 0.2697], and [0.5386 to 0.6272] for the metrics of accuracy, MCC, and AUC respectively. The DT model with 7 IC inputs acquires the highest performance and the DT model with 8 IC inputs acquires the lowest performance in all metrics employed in this study. The best DT model performance with ICA extraction is 77%, 0.269, and 0.6272 for the metrics of accuracy, precision, recall, F1 Score, MCC, and AUC respectively. The best DT model with ICA extraction has around 10% higher in both performance metrics of accuracy and MCC than the best DT model with PCA extraction. While the best DT model with ICA extraction is around 3.5% higher in the AUC metric than the best DT with PCA extraction. The increasing performance of the best DT with the ICA extraction has confirmed significantly the influence of the independence of model input on the model's performance.

While for liver cirrhosis data, Table 9 presents the performance metrics of 5 DT classifiers built and evaluated respectively using the training and testing data acquired by PCA projection with 5 to 9 principal components (PC). The DT model with the PCA extraction has an accuracy metric in the range of [63% to 73%] where the lowest and highest accuracy values are obtained respectively by the DT model with 5 and 9 PC inputs. The DT model with 10 PC inputs has an accuracy value of 73% meaning as many as 73 of 100 patients who come from the unknown class label are correctly classified by the model. This is also in line with the result of precision and recall, [0.6119 to 0.7523] and [0.6271 to 0.7288] respectively. The DT models with PCA extraction have the MCC and AUC metrics with a range value of [0.6189 to 7373] and [0.3038 to 0.4913] respectively. The lowest MCC value is acquired in the DT model with 9 PC inputs, and the highest MCC value is acquired in the DT model with 7 PC inputs. whereas, the lowest AUC value is acquired in the DT model with 7 PC inputs, and the highest AUC value is acquired in the DT model with 9 PC inputs. The different result occurs in the accuracy metric that the best DT model has 7 PC inputs.

On the other hand, Table 10 presents the performance metrics of 5 DT classifiers built and evaluated respectively using the training and testing data extracted by the ICA projection with 5 to 9 independent components (IC). The DT models with the ICA extraction acquire the performance metrics in the range [63% to 73%], [0.6414 to 0.7659], [0.6271 to 0.7288], [0.6336 to 0.7401], and [0.2842 to 0.4564] for the metrics of accuracy, precision, recall, MCC, and AUC respectively. The DT model with 7 IC inputs acquires the highest performance and the DT model with 8 IC inputs acquires the lowest performance in all metrics employed in this study. The best DT model performance with ICA extraction is 73%, 0.7659, 0.7288, 0.7401, and 0.4564 for the metrics of accuracy, precision, recall, F1 Score, MCC, and AUC respectively. The best DT model with ICA extraction has around 1% higher in both performance metrics of accuracy and MCC than the best DT model with PCA extraction. While the best DT model with ICA extraction is around 4% smaller in the AUC metric than the best DT with PCA extraction. The same and a bit decreasing performance of the best DT with the ICA extraction should be subjected in performance discussion where majority of continuous variables included in this second data application (Liver cirrhosis data).

4.4. Discussion

In order to take part in the resource identification initiative, projection components acquired by both PCA and ICA methods have the same correlation values as either the linear or nonlinear of the simulation data. The correlation value is used as a criterion to infer the independence between two variables, it will lead to mis-inference when both variables have a nonlinear relationship between both PCA and ICA. The PCA produces the orthogonal and uncorrelated components but the ICA produces the independent and uncorrelated ones [32]. Based on the scatter plots in Figure 1 uncorrelated components are not independent components when the underlying data have a nonlinear pattern. The uncorrelated variables are not guaranteed that both variables are independent each of other [34]. Two components are linearly uncorrelated i.e. the orthogonal components acquired by the PCA are not always two independent components, although they have zero correlation value [32]. On the other hand, the ICA extracted components always are not only zero correlation values but also independent components.

The DT models with the more principal components as predictor variables tend to have a higher performance metric than the DT models with the less principal component as predictor variables. The more principal components in PCA projection mean the more explained variance [29]. The DT model with 100% explained variance of principal components as the predictor variables achieved the highest accuracy metric and the DT model with 98% explained variance of principal components as the predictor variables achieved the highest performance on both metrics of MCC and AUC. This result creates a contradiction that PCA projection fails as a tool for dimension reduction [31]. However, the DT model with 7 independent components outperforms in 3 performance metrics compared to the DT models with other numbers of independent components. by the ICA projection to 7 principal components which explain the 85% variance in the dataset. We provide two examples with a set of data with majority of discrete variables as well as a set of data with majority of continuous variables. And this result proves successfully that not only does the ICA projection produce the independent components [33] but also the PCA projection acts as the dimension reduction method [30].

5. Conclusion

Both PCA and ICA variable extraction on the linear simulation data produced a distinction of data pattern i.e. one scatter plot forms an ellipse shape and the other scatter plot forms a sparse pattern. Both projection data have the same correlation values as zero which can be interpreted as the orthogonal components are also the independent components. The second simulation data have a nonlinear underlying pattern. Although both PCA and ICA projection data have the same correlation value as zero, the scatter plot of PCA extraction data remains a nonlinear pattern but the scatter plot of ICA extraction data forms the sparse pattern.

For analysis to our datasets, the performance of DT models on the different input data that are produced by the PCA projection achieves the highest accuracy metric in 10 principal components and the highest in both MMC and AUC metrics in 9 principal components. On the other side, the DT model with 7 independent components acquired by the ICA projection achieves the highest performance in all six-performance metrics. The accuracy performance gap between the best DT model with PCA projection and the best DT model with ICA projection is around 10% for discrete majority data and quite balanced for continuous majority data. An interesting issue for the next research is the hybrid of variable selection and variable extraction on high-dimensional data to obtain a set of independent predictors of the nonlinear models of machine learning.

Conflict of Interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Author Contributions

AE: Conceptualization and formulation of ideas as well as research aims, Methodology, Statistical analysis, Writing – original draft, review & editing, Conclusion; ZZ: Writing – review & editing, Data Curation; DSS: Writing – review & editing, Formal Analysis; NND: Writing – review & editing, Formal Analysis; RBEW: Writing – review & editing; SH: Writing – original draft, Conceptualization, Methodology, Statistical analysis.

Data Availability

Data are available online, with links in Section 3 “Materials and Methods”. While python codes are available as per request.

Funding and Acknowledgement

We are grateful for the funding from Universitas Brawijaya, with the grant of Hibah Doktor Lektor Kepala 2024, contract number: 2174.2/UN10.F0901/B/KS/2024.

REFERENCES

1. N. Bettenburg, M. Nagappan, and A. E. Hassan, Towards improving statistical modeling of software engineering data: think locally, act globally!, *Empirical Software Engineering*, Vol. 20, No. 2, 2015, doi: 10.1007/s10664-013-9292-6.
2. R. Rossi, A. Murari, P. Gaudio, and M. Gelfusa, Upgrading model selection criteria with goodness of fit tests for practical applications, *Entropy*, Vol. 22, No. 4, 2020, doi: 10.3390/e22040447.
3. S. Luo, H. Sun, Q. Ping, R. Jin, and Z. He, A review of modeling bio-electrochemical systems: Engineering and statistical aspects, *Energies*, Vol. 9, No. 2. 2016. doi: doi.org/10.3390/en9020111.
4. M. I. H. Khan, C. P. Batuwatta-Gamage, M. A. Karim, and Y. T. Gu, Fundamental Understanding of Heat and Mass Transfer Processes for Physics-Informed Machine Learning-Based Drying Modelling, *Energies*, Vol. 15, No. 24. 2022. doi: doi.org/10.3390/en15249347
5. S. Karunaratne and H. Gacanin, An overview of machine learning approaches in wireless mesh networks, *IEEE Communications Magazine*, Vol. 57, No. 4. 2019. doi: 10.1109/MCOM.2019.1800434.
6. P. Chatterjee, M. Yazdani, F. Fernández-Navarro, and J. Pérez-Rodríguez, *Machine Learning Algorithms and Applications in Engineering*, CRC Press, 2023. doi: 10.1201/9781003104858.
7. T. M. S. Filho, B. A. Pimentel, R. M. C. R. Souza, and A. L. I. Oliveira, Hybrid methods for fuzzy clustering based on fuzzy c-means and improved particle swarm optimization, *Expert System with Applications*, Vol. 42, No. 17-18, 2015, doi: 10.1016/j.eswa.2015.04.032.
8. E. Lima, E. Gorski, E. F. R. Loures, E. A. P. Santos, and F. Deschamps, Applying machine learning to AHP multicriteria decision making method to assets prioritization in the context of industrial maintenance 4.0, *IFAC Papers Online*, 52-13, 2019, 2152-2157. doi: 10.1016/j.ifacol.2019.11.524
9. M. O. Arowolo, M. O. Adebiyi, A. A. Adebiyi, and O. J. Okesola, A hybrid heuristic dimensionality reduction methods for classifying malaria vector gene expression data, *IEEE Access*, Vol. 8, 2020. doi: 10.1109/ACCESS.2020.3029234.
10. Y. Bouchlaghem, Y. Akhiat, and S. Amjad, *Feature Selection: A Review and Comparative Study*, E3S Web of Conferences, 10th International Conference on Innovation, Modern Applied Science & Environmental Studies (ICIES) 2022. doi: 10.1051/e3sconf/202235101046.
11. S. Handoyo, N. Pradianti, W. H. Nugroho, and Y. J. Akri, A Heuristic Feature Selection in Logistic Regression Modeling with Newton Raphson and Gradient Descent Algorithm, *International Journal of Advanced Computer Science and Applications*, Vol. 13, No. 3, 2022, doi: 10.14569/IJACSA.2022.0130317.
12. Y. Xie, Y. Li, Z. Xia, and R. Yan, An Improved Forward Regression Variable Selection Algorithm for High-Dimensional Linear Regression Models, *IEEE Access*, Vol. 8, 2020, doi: 10.1109/ACCESS.2020.3009377.
13. K. Luijken, R. H. H. Groenwold, M. van Smeden, S. Strohmaier, and G. Heinze, A comparison of full model specification and elimination of potential confounders when estimating marginal and conditional causal effects on binary outcomes from observational data, *Biometrical Journal*, Vol. 66, No. 1, 2024, doi: 10.1002/bimj.202100237.
14. C. Lataniotis, S. Marelli, and B. Sudret, Extending classical surrogate modeling to high dimensions through supervised dimenbackwardsionality reduction: A data-driven approach, *Int Journal Uncertain Quantif*, Vol. 10, No. 1, 2020. doi: 10.1615/Int.J.UncertaintyQuantification.2020031935.
15. S. Surono, K. W. Goh, C. W. Onn, and F. Marestiani, Developing an optimized recurrent neural network model for air quality prediction using K-means clustering and PCA dimension reduction, *International Journal of Innovative Research and Scientific Studies*, Vol. 6, No. 2, 2023, doi: 10.53894/ijirss.v6i2.1427.

16. A. Ifarraguerri and C. I. Chang, Unsupervised hyperspectral image analysis with projection pursuit, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 38, No. 6, 2000, doi: 10.1109/36.885200.
17. M. Midya, P. Ganguly, T. Datta, and S. Chattopadhyay, ICA-Feature-Extraction-Based Fault Identification of Vehicular Starter Motor, *IEEE Sensors Letters*, Vol. 7, No. 2, 2023, doi: 10.1109/LSENS.2023.3242814.
18. S. Handoyo, Y. P. Chen, G. Irianto, and A. Widodo, The varying threshold values of logistic regression and linear discriminant for classifying fraudulent firm, *Mathematics and Statistics*, Vol. 9, No. 2, 2021, doi: 10.13189/ms.2021.090207.
19. W. H. Nugroho, S. Handoyo, and Y. J. Akri, An influence of measurement scale of predictor variable on logistic regression modeling and learning vector quantization modeling for object classification, *International Journal of Electrical and Computer Engineering*, Vol. 8, No. 1, 2018, doi: 10.11591/ijece.v8i1.pp333-343.
20. H. R. Bittencourt, D. A. O. Moraes, and V. Haertel, *A binary decision tree classifier implementing logistic regression as a feature selection and classification method and its comparison with maximum likelihood*, 2007 IEEE International Geoscience and Remote Sensing Symposium, 23-28 July 2007. doi: 10.1109/IGARSS.2007.4423159.
21. J. Zhang, *Rethink Decision Tree Traversal*, 2022, doi: https://arxiv.org/abs/2209.04825.
22. M. A. Ahmed, A. Rizaner, and H. U. Ali, A novel decision tree classification based on post-pruning with Bayes minimum risk, *PLoS One*, Vol. 13, No. 4, 2018, doi: 10.1371/journal.pone.0194168.
23. R. Hu, Medical Data Mining Based on Decision Tree Algorithm, *Computer and Information Science*, Vol. 4, No. 5, 2011, doi: 10.5539/cis.v4n5p14.
24. L. H. Al Fryan, M. I. Shomo, M. B. Alazzam, and M. A. Rahman, Processing Decision Tree Data Using Internet of Things (IoT) and Artificial Intelligence Technologies with Special Reference to Medical Application, *Biomed Research International*, Vol. 2022, 2022, doi:10.1155/2023/9762317
25. M. R. Hasan, N. A. A. Bakar, F. Siraj, M. S. Sainin, and M. S. Hasan, *Single decision tree classifiers' accuracy on medical data*, Proceedings of the 5th International Conference on Computing and Informatics, ICOCI 2015, No. 188, 2015.
26. M. A. Hambali, M. D. Gbolagade, Y. A. Olasupo, Heart Disease Prediction Using Principal Component Analysis and Decision Tree Algorithm, *Journal of Computer Science and Eng.*, Vol. 4, No. 1, 2023, doi: icsejournal.com/index.php/JCSE/article/view/617/16.
27. J. Kuzilek, V. Kremen, F. Soucek, L Lhotska, Independent Component Analysis and Decision Trees for ECG Holter Recording De-Noising, *PLoS ONE*, Vol. 9, No. 6, 2014, doi: 10.1371/journal.pone.0098450.
28. Y. Taguchi, *Unsupervised Feature Extraction Applied to Bioinformatics: A PCA Based and TD Based Approach*, Springer, 2019.
29. J. Jiang, J. Ma, C. Chen, Z. Wang, Z. Cai, and L. Wang, SuperPCA: A Superpixelwise PCA Approach for Unsupervised Feature Extraction of Hyperspectral Imagery, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 56, no. 8, 2018, doi: 10.1109/TGRS.2018.2828029.
30. L. F. Sánchez-Peralta, A. Picón, J. A. Antequera-Barroso, J. F. Ortega-Morán, F. M. Sánchez-Margallo, and J. B. Pagador, Eigenloss: Combined PCA-based loss function for polyp segmentation, *Mathematics*, Vol. 8, No. 8, 2020, doi: 10.3390/MATH8081316.
31. C. Croux, P. Filzmoser, and M. R. Oliveira, Algorithms for Projection-Pursuit robust principal component analysis, *Chemometrics and Intelligent Laboratory Systems*, Vol. 87, No. 2, 2007, doi: 10.1016/j.chemolab.2007.01.004.
32. J. Yang and Q. Cheng, A comparative study of independent component analysis with principal component analysis in geological objects identification, Part I: Simulations, *Journal of Geochemical Exploration*, Vol. 149, 2014, doi: 10.1016/j.gexplo.2014.11.013.
33. C. F. Beckmann and S. M. Smith, Probabilistic Independent Component Analysis for Functional Magnetic Resonance Imaging, *IEEE Trans Med Imaging*, Vol. 23, No. 2, 2004, doi: 10.1109/TMI.2003.822821.
34. A. Hyvärinen and E. Oja, Independent component analysis: Algorithms and applications, *Neural Networks*, Vol. 13, No. 4–5, 2000, doi: 10.1016/S0893-6080(00)00026-5.
35. J. Franklin, J. Rogan, S. R. Phinn, and C. E. Woodcock, *Rationale and Conceptual Framework for Classification Approaches to Assess Forest Resources and Properties*, Remote Sensing of Forest Environments, pp 279–300, 2003. doi: $10.1007/978-1-4615-0306-4_10$.
36. B. Charbuty and A. Abdulazeez, Classification Based on Decision Tree Algorithm for Machine Learning, *Journal of Applied Science and Technology Trends*, Vol. 2, No. 01, 2021, doi: 10.38094/jastt20165.
37. Y. Y. Song and Y. Lu, Decision tree methods: applications for classification and prediction, *Shanghai Arch Psychiatry*, Vol. 27, No. 2, 2015, doi: 10.11919/j.issn.1002-0829.215044.
38. M. Makhtar, D. C. Neagu, and M. J. Ridley, *Binary classification models comparison: On the similarity of datasets and confusion matrix for predictive toxicology applications*, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2011. doi: $10.1007/978-3-642-23208-4_11$.
39. S. Boughorbel, F. Jarray, and M. El-Anbari, Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric, *PLoS One*, Vol. 12, No. 6, 2017, doi: 10.1371/journal.pone.0177678.
40. D. J. Hand, Measuring classifier performance: A coherent alternative to the area under the ROC curve, *Mach Learn*, Vol. 77, No. 1, 2009, doi: 10.1007/s10994-009-5119-5.