Computer Engineering and Applications Vol. 12, No. 3, October 2023

Nonparametric Regression Analysis of BE4DBE2 Relationship with n and z Variables using Naive Bayes and SVM Classification on Nuclear Data

Ruben Cornelius Siagian[1*], Lulut Alfaris[2], Aldi Cahya Muhammad[3], Ukta Indra Nyuswantoro[4], Gendewa Tunas Rancak[5]

[1*]*Department of Physics, Universitas Negeri Medan*
[2]*Department of Marine Technology, Politeknik Kelautan dan Perikanan Pangandaran, Indonesia,*
[3]*Department of Electrical and Electronic Engineering, Islamic University of Technology, Bangladesh,*
[4]*Department of Structure Engineering, Asiatek Energi Mitratama,*
[5]*Department of Environmental Engineering, Universitas Nahdlatul Ulama Nusa Tenggara Barat*
*rubensiagian775@gmail.com

ABSTRACT

This research article describes several analyses of nuclear data using various statistical methods. The first analysis uses linear regression to investigate the relationship between the independent variables (n and z) and the response variable (BE4DBE2). The second analysis uses a nonparametric regression model to overcome the assumptions of normality and linearity in the data. The third analysis uses the Naive Bayes method to classify nuclear data based on variables n and z. The fourth analysis uses a decision tree to classify nuclear data based on the same variables. Finally, the article describes an SVM analysis and a K-means analysis to classify and group nuclide data. The article presents clear and organized descriptions of each analysis, including visual representations of the results. The findings of each analysis are discussed, providing valuable insights into the relationships between the variables and the response variable. The article demonstrates the usefulness of statistical methods in analyzing nuclear data.

Keywords: Nuclear Data, Statistical Methods, Linear Regression, Nonparametric Regression, Naive Bayes Method, Decision Tree, SVM Analysis, K-Means Analysis.

1. INTRODUCTION

This study has a dual purpose: first, to investigate the relationship between the independent variables (n and z) and the response variable (BE4DBE2) through the use of both linear and nonparametric regression methods. Linear regression will be utilized to test hypotheses about the linear relationship between the independent variables and the response variable, while nonparametric regression will be used to uncover more intricate relationships [1]. Second, the study aims to classify nuclear data using the Naive Bayes method and decision tree analysis. The Naive Bayes method will be employed to classify data based on conditional probabilities, while decision tree analysis will be used to construct a decision tree based on relevant data attributes [2]. The overarching goal of this research is to generate new knowledge about the relationship between independent variables and response variables, as well

as to determine the most effective method for classifying nuclear data. The findings from this study have the potential to make significant contributions to the field of nuclear science and can be applied across a wide range of industries, including nuclear technology, nuclear security, and related fields.

This research has substantial benefits for the natural science field as it utilizes linear and nonparametric regression models to explain the relationship between independent and response variables in nuclear data, and predict response values based on independent variable values. This approach aids in comprehending the correlation between variables and estimating response values in diverse scenarios. Moreover, the implementation of the Naive Bayes method in classifying nuclear data enhances the accuracy and efficiency of response value prediction (BE4DBE2) based on independent variable values (n and z). This method also assists in estimating nuclear data classification more precisely. Additionally, the use of decision trees in classifying nuclear data and assessing classification model performance provides significant advantages [3]. The insights gained from this research can improve the decision-making process in the nuclear field by providing a better understanding of nuclear data and how classification models operate.

This study explores the relationship between independent variables and response variables in nuclear data by utilizing a variety of analytical methods, including linear regression, non-parametric regression analysis, Naive Bayes method, and decision tree analysis for data classification. With a dataset of 232 observations, the study focuses on predicting the response variable BE4DBE2 based on the independent variables n and z using linear regression, while non-parametric regression analysis is used to estimate the relationship between the independent variables and the response variable without making any assumptions about the shape of the data distribution. Additionally, Naive Bayes method is utilized to predict the class of data based on its features, and decision tree analysis produces a decision tree model to predict the class of data [2]. However, it's important to note that this research is limited in terms of the analysis methods used and only describes the models and methods employed, rather than discussing the results of the data analysis.

This research explores the correlation between the independent variables (n and z) and the response variable (BE4DBE2) through various methods including linear regression, nonparametric regression, Naive Bayes, and decision tree analysis. Although this study provides valuable insights into the variable relationship, there are still gaps in the literature that require attention. For instance, examining other variables that could impact the response variable, like atomic mass or isospin, would enhance this study. Moreover, incorporating advanced machine learning algorithms such as deep learning or ensemble learning could further enhance the data analysis [4]. Additionally, the potential applications of these findings in nuclear physics and engineering, including the development of new nuclear technologies and the design of new nuclear reactors, warrant further investigation [5]. Therefore, further research is necessary to address these gaps and to deepen our comprehension of the relationship between nuclear variables and their practical applications.

This study employs scientific methods to examine the correlation between independent variables n and z and the response variable BE4DBE2. Various techniques, including linear regression, nonparametric regression, Naive Bayes, and decision tree analysis, were employed to analyze the data and provide valuable insights into the variables' relationship [6]. Nevertheless, this study also identifies

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

gaps in the literature that warrant attention, such as investigating additional variables like atomic mass or isospin that may influence the response variable. Furthermore, more advanced machine learning algorithms, such as deep learning or ensemble learning, could be employed to analyze the data [7]. The study's potential applications in nuclear physics and engineering, such as designing new nuclear reactors or developing new nuclear technologies, merit further exploration [8]. Thus, additional research is needed to fill these gaps and improve our understanding of the connection between nuclear variables and their applications. Overall, this research adheres to the principles of natural scientific inquiry by utilizing empirical data, objective analysis, and proposing future research areas to broaden our knowledge of the subject matter.

2. RESEARCH METHOD

2.1 LINEAR REGRESSION METHOD OF RELATIONSHIP BETWEEN BE4DBE2 AND VARIABLES N AND Z

This study employs a quantitative research method, specifically linear regression analysis, to examine the relationship between the independent variables (n and z) and the dependent variable (BE4DBE2). Quantitative measurement and statistical analysis of the data are enabled through this method. The data is collected through measurement or observation techniques and then processed using statistical software such as SPSS or R to perform linear regression analysis [9]. The independent variables are utilized to predict the value of the dependent variable in the linear regression analysis [10]. Once the analysis is conducted, the regression model's significance is tested to determine if it can effectively explain the relationship between the independent and dependent variables [11]. The analysis results are used to draw conclusions on the suitability of the linear regression model in explaining the relationship between the independent and dependent variables. Although the independent variable n and the constant have a significant influence on the dependent variable, the results indicate that the linear regression model is not accurate enough in explaining the relationship between the independent (n and z) and dependent (BE4DBE2) variables.

To perform the linear regression analysis, we first need to prepare and import the data into the R environment, ensuring that we have identified the relevant variables (BE4DBE2, n, and z). We then create a scatter plot to visualize the relationship between the variables and calculate their correlation to determine their strength of association. Using the lm() function, we create a linear regression model to model the relationship between the variables [12]. Evaluating the model using the summary() function allows us to determine its goodness of fit to the data [13]. We can also create a linear regression plot to visualize the model's results and detect any patterns or trends. Finally, the regression model can be used to predict the value of BE4DBE2 based on the given values of n and z, providing insight into the relationship between the variables:

```
data <- read.csv("nama_file.csv")
plot(BE4DBE2 ~ n, data = data,
    main = "Hubungan antara BE4DBE2 dan n",
    xlab = "n", ylab = "BE4DBE2")
plot(BE4DBE2 ~ z, data = data,
```

```
    main = "Hubungan antara BE4DBE2 dan z",
    xlab = "z", ylab = "BE4DBE2")
cor_n <- cor(BE4DBE2, n)
cat("Koefisien korelasi antara BE4DBE2 dan n: ", cor_n, "\n")
cor_z <- cor(BE4DBE2, z)
cat("Koefisien korelasi antara BE4DBE2 dan z: ", cor_z, "\n")
model <- lm(BE4DBE2 ~ n + z, data = data)
summary(model)
plot(BE4DBE2 ~ n, data = data,
    main = "Hubungan antara BE4DBE2 dan n",
    xlab = "n", ylab = "BE4DBE2")
abline(model, col = "red")
plot(BE4DBE2 ~ z, data = data,
    main = "Hubungan antara BE4DBE2 dan z",
    xlab = "z", ylab = "BE4DBE2")
abline(model, col = "red")
new_data <- data.frame(n = c(1, 2, 3), z = c(4, 5, 6))
predicted_values <- predict(model, newdata = new_data)
cat("Nilai prediksi BE4DBE2 untuk n = 1, z = 4: ", predicted_values[1], "\n")
cat("Nilai prediksi BE4DBE2 untuk n = 2, z = 5: ", predicted_values[2], "\n")
cat("Nilai prediksi BE4DBE2 untuk n = 3, z = 6: ", predicted_values[3], "\n")
```

The program utilizes the R programming language to perform a simple regression analysis that describes the correlation between the dependent variable (BE4DBE2) and the independent variables (n and z) based on data from a CSV file. Initially, data is imported using the read.csv() function, followed by the use of scatter plots to display the relationship between the variables. Two scatter plots are generated: plot(BE4DBE2 ~ n, data = data, ...) and plot(BE4DBE2 ~ z, data = data, ...), representing the correlation between BE4DBE2 with n and z, respectively. The Pearson correlation coefficient is then computed using the cor() function, and the correlation coefficient values are printed using cat(). The lm() function is employed to build a linear regression model, which estimates the value of BE4DBE2 based on n and z. At the end of the program, another scatter plot is generated, displaying the correlation between the variables. This plot incorporates a previously created linear regression line (abline) using abline(model, col = "red"). Lastly, the predicted value of BE4DBE2 is computed using predict() with new_data values of n and z, and the predicted value is printed with cat().

2.2 NONPARAMETRIC REGRESSION METHOD FOR MODELING THE RELATIONSHIP BETWEEN BE4DBE2 AND VARIABLES N AND Z

A nonparametric regression model with a smoothing function s(n,z) was used to model the relationship between BE4DBE2 and the variables n and z. This method was chosen to overcome assumptions of normality and linearity and provide more reliable results in analyzing variable relationships [14]. To assess the model's significance, a t-value and F-statistic significance test was conducted on the intercept and smoothing function parameters [15]. Additionally, the quality level of the model was evaluated by calculating adjusted R-squared, deviance explained, GCV, and scale estimate [16]. The model's results were visualized using a plot that showed the model's predictions for BE4DBE2 values for each combination of n and z values. For non-linear relationships between BE4DBE2 and n and z variables, the kernel regression method was recommended. This method involves gathering data consisting of the variables

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

BE4DBE2, n, and z, determining the number of kernels and bandwidth, calculating kernel density for each data point, computing the mean value of the BE4DBE2 variable for all data falling within the kernel range, and using the mean value as the predicted result. This method avoids some of the issues that arise with traditional parametric regression methods by not assuming any specific functional form. In conclusion, the kernel regression method is a useful nonparametric regression technique for estimating non-linear relationships between variables.

```
data <- read.csv("data.csv")
num_kernels <- 50
bw <- bw.nrd(data$BE4DBE2)
kernel <- function(x) {
  exp(-0.5 * x^2) / sqrt(2 * pi)
}
predictions <- c()
for (i in 1:nrow(data)) {
  # Calculate kernel densities
  kernel_densities <- rep(0, num_kernels)
  for (j in 1:num_kernels) {
    kernel_densities[j] <- kernel((data$BE4DBE2[i] - data$BE4DBE2) / bw)
  }
  weighted_sum <- sum(kernel_densities * data$BE4DBE2)
  weights <- sum(kernel_densities)
  prediction <- weighted_sum / weights
  predictions <- c(predictions, prediction)
}
data$predictions <- predictions
head(data)
```

This program utilizes the kernel regression method for nonparametric regression to model the relationship between variable BE4DBE2 and variables n and z. The program reads data from the "data.csv" file and loads it into the "data" variable. The number of kernels is set to 50, and the bandwidth is computed using the bw.nrd() function, which automatically calculates the bandwidth based on the BE4DBE2 data. A Gaussian function with a standard deviation of 1.0 is used as the kernel function. For each data point in "data", a loop is performed, and the kernel density is calculated using the previously defined kernel function and bandwidth. The average value of the BE4DBE2 variable for all data points that fall within the kernel range is calculated and used as the prediction result for that data point. The prediction result is added to the "predictions" vector and to the "data" as a new column named "predictions". Finally, the prediction result is displayed using the "head()" function. This program enables nonparametric regression analysis to predict the value of BE4DBE2 based on the variables n and z available in the data. The prediction results can be used for analysis and decision-making in relevant contexts with the data.

2.3 CLASSIFICATION METHOD OF NUCLEAR DATA USING NAIVE BAYES METHOD

The study employed the Naive Bayes method to classify nuclear data, using variables n and z as inputs and the BE4DBE2 factor as output. Initially, relevant nuclear data was collected and prepared by removing irrelevant or missing data, and

then split into two datasets: training and testing. The training data was utilized to train the Naive Bayes classification model by calculating the probability for each input variable for each output class. Subsequently, the testing data was employed to validate the accuracy of the model by comparing the predicted results with the actual output values. To enhance understanding of the distribution of Naive Bayes classification predictions and the relationship between input and output variables, the results were visualized in the form of plots.

To implement the method effectively, the data training and testing must be prepared, with the former containing classified nuclear information based on variables n and z and the latter comprising nuclear data that needs to be classified. Afterward, both datasets must undergo preprocessing, including removing irrelevant data and formatting the data according to the algorithm's requirements. Then, the model is trained using the data training to calculate the probability of the nuclear data class based on variables n and z. Once the model is trained, it can predict the class of nuclear data in the testing phase by selecting the class with the highest probability as the prediction. Finally, the accuracy of the prediction is evaluated against the data testing, and if the evaluation results are not satisfactory, parameters can be tuned, and better data preprocessing can be applied to enhance the model's performance.

```
install.packages("e1071")
library(e1071)
data <- read.csv("data_nuklir.csv")
index <- sample(1:nrow(data), round(0.8*nrow(data)))
training_data <- data[index, ]
testing_data <- data[-index, ]
predictors <- c("n", "z")
class_label <- "kelas"
model <- naiveBayes(training_data[, predictors], training_data[, class_label])
predicted_class <- predict(model, testing_data[, predictors])
accuracy <- sum(predicted_class == testing_data[,
class_label])/nrow(testing_data)
cat("Akurasi prediksi: ", round(accuracy*100, 2), "%\n")
```

The program is equipped with a script that installs the "e1071" package and applies the Naive Bayes method to make predictions. The dataset, which is stored in "data_nuclear.csv" file, is read using the read.csv function and split into training_data and testing_data with an 80:20 ratio. The variables "predictors" and "class_label" are used to identify the independent and dependent variables respectively, in the prediction model. Subsequently, a naiveBayes model is generated using the training data along with the predictor and class variables. The testing data is then evaluated using the naiveBayes model, and the accuracy of the predictions is determined by comparing them with the actual values. Finally, the prediction accuracy results are printed to the screen using the cat function. Utilizing the e1071 package and the Naive Bayes method, this program is capable of performing prediction or classification on any given dataset.

2.4 DECISION TREE ANALYSIS METHOD

This research employs nuclear data analysis using decision tree as its methodology. Decision tree is a machine learning technique that uses rules obtained

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

from training data to predict target values [17]. Specifically, a decision tree is used in this research to predict the value of BE4DBE2 based on n and z values. The decision tree is created by dividing the training data into subsets that are homogeneous in terms of target values and selecting the most informative predictor variable for each subset [18]. This process is repeated until a stopping condition is met. The model's performance is evaluated using several metrics such as accuracy, agreement, sensitivity, specificity, precision, and positive and negative predictive values. Additionally, the decision tree plot is visualized using the rpart.plot library to illustrate the rules or conditions needed for predictions. To create an R program for this analysis, the "rpart" library package in R is required. Here's an example R program to model a decision tree for the relationship between BE4DBE2, n, and z:

```
install.packages("rpart")
library(rpart)
data <- read.csv("namafile.csv")
tree <- rpart(BE4DBE2 ~ n + z, data = data, method =
"class")
plot(tree)
text(tree)
predicted <- predict(tree, newdata = data, type = "class")
actual <- data$BE4DBE2
accuracy <- sum(predicted == actual)/length(actual)
print(paste0("Akurasi model: ", round(accuracy, 2)))
```

This program performs the Decision Tree Analysis Method to model the relationship between the BE4DBE2 variable and the n and z variables through a series of steps. Firstly, the program installs the "rpart" package, which is necessary for Decision Tree analysis in R. Next, it loads the "rpart" library into the R environment. The program then reads data from a provided csv file, and the user must input the correct csv file name. Using the "rpart" function from the "rpart" package, the program models the relationship between the BE4DBE2 variable and the n and z variables, adjusting the "method" argument according to the problem type. To visualize the model, the program displays a decision tree plot using the "plot" and "text" functions. The program then uses the "predict" function to make predictions on the same data used to create the model, with the "type" argument set to "class" for this classification problem. Finally, the program calculates the accuracy of the model by comparing the predicted values with the actual values of the BE4DBE2 variable and displays the accuracy percentage using the "print" function

2.5 SVM METHOD OF ANALYSIS

The research method for analyzing nuclear data using the SVM model involves several steps, beginning with data collection to obtain reliable data on the number of neutrons, number of protons, and binding energy per nucleon (BE4DBE2) in line with research objectives. After data collection, the nuclear data is cleaned and transformed to ensure its validity and accuracy for research purposes.

Then, based on an analysis and understanding of the nuclear data used, an appropriate kernel is selected, with a linear kernel chosen in this study since only three variables were involved [19]. The SVM model is trained using the pre-processed nuclear data, with the performance evaluated using the confusion matrix and

classification result plots [20]. The expected result is a reliable classification model that can assist in the analysis of nuclide data, with the R program code for the SVM method provided as an example to model the relationship between BE4DBE2 and variables n and z:

```
library(e1071)
data <- read.csv("data.csv")
be4dbe2 <- data$be4dbe2
n <- data$n
z <- data$z
X <- cbind(n, z)
svm_model <- svm(be4dbe2 ~ ., data = X, kernel = "linear")
predicted_be4dbe2 <- predict(svm_model, X)
cat("Hasil prediksi be4dbe2:", predicted_be4dbe2)
```

This program implements SVM (Support Vector Machines) in R programming language to model the relationship between variables be4dbe2 and variables n and z. The first line, "library(e1071)," loads the e1071 library that contains functions for SVM in R. The second line, "data <- read.csv("data.csv")," reads the data.csv file that contains the be4dbe2, n, and z variables and stores it in the data variable. The third to fifth lines separate the be4dbe2, n, and z variables from the data variable. The sixth line combines the n and z variables into one matrix and stores it in the X variable. The seventh line creates an SVM model using the X variable as the data and the be4dbe2 variable as the target. The kernel = "linear" parameter indicates that a linear kernel is used. The eighth line predicts the be4dbe2 value using the SVM model created with the same data. The last line displays the predicted value of be4dbe2 on the screen using the cat() command. This program aims to predict the be4dbe2 variable based on the n and z variables using the SVM method with a linear kernel, and displays the prediction result on the screen.

2.6 BAYESIAN REGRESSION ANALYSIS METHOD

This study utilizes Bayesian Regression as the research method, which considers uncertainty in model parameter estimates and is a statistical method for creating a regression model [21]. The model is used to predict Binding Energy (BE4DBE2) values in atomic nuclei based on independent variables of neutron (n) and proton (z) numbers. The data analysis involves estimating a Bayesian generalized linear regression model using the "stan_glm" function with Gaussian distribution and identity link function on the dependent variable BE4DBE2, with n and z as the independent variables [22]. The estimation results provide mean, standard deviation, as well as 10th, 50th (median), and 90th percentiles for each parameter, including intercept, n, z, and sigma. Additionally, the analysis includes fit diagnostics such as mean_ppd, and MCMC diagnostics such as Monte Carlo standard error (mcse), potential scale reduction factor (Rhat), and neff, to measure estimation precision, convergence between and within chains, and effective sample size [23].

```
data <- read.csv("nama_file.csv")
library(rstan)
model <- "
data {
  int<lower=0> N; // Jumlah pengamatan
```


ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

```
  vector[N] BE4DBE2; // Variabel respon
  vector[N] n; // Variabel prediktor 1
  vector[N] z; // Variabel prediktor 2
 }
 parameters {
  real alpha; // Intercept
  real beta_n; // Slope untuk variabel n
  real beta_z; // Slope untuk variabel z
  real<lower=0> sigma; // Standard deviation
 }
 model {
  alpha ~ normal(0, 1000); // Prior untuk intercept
  beta_n ~ normal(0, 1000); // Prior untuk slope variabel n
  beta_z ~ normal(0, 1000); // Prior untuk slope variabel z
  sigma ~ cauchy(0, 5); // Prior untuk standard deviation
  BE4DBE2 ~ normal(alpha + beta_n * n + beta_z * z, sigma); // Likelihood
 }
 "
 stan_model <- stan_model(model_code = model)
 data_stan <- list(
  N = nrow(data),
  BE4DBE2 = data$BE4DBE2,
  n = data$n,
  z = data$z
 )
 fit <- sampling(stan_model, data = data_stan, chains = 4, iter = 2000, warmup = 1000, thin = 1)
 summary(fit)
 new_data <- data.frame(n = c(1, 2, 3), z = c(2, 3, 4))
 new_data$predicted_BE4DBE2 <- predict(fit, newdata = new_data)
 new_data
```

The program utilizes the rstan package in R programming language to implement a linear regression model using Bayesian inference. Firstly, it reads data from a csv file and loads the rstan package. Then, it specifies the linear regression model with N as the number of observations, BE4DBE2 as the response variable, and n and z as the predictor variables. Normal distribution is used for alpha, beta_n, and beta_z, and Cauchy distribution is used for sigma as priors for each parameter in the model. The model is then run by inputting the priors and likelihood, followed by running the sampling function to obtain the posterior distribution of the parameters. Subsequently, predictions are made with new data by inputting it into the new_data object with predictor variables n and z, and making predictions on the response variable BE4DBE2 based on the trained model. Finally, the predicted values of the response variable BE4DBE2 are displayed in the new_data object, which is based on the trained model with the new data inputted on the predictor variables n and z.

3. RESULTS AND DISCUSSION

3.1 LINEAR REGRESSION ANALYSIS OF THE RELATIONSHIP BETWEEN BE4DBE2 AND VARIABLES N AND Z

After conducting the analysis, it can be concluded that the regression model has an intercept with an estimated value of 1.53372 and a standard error of 0.62575, while the predictor variables n and z have estimated coefficients of -0.04736 and 0.07015,

respectively, with standard errors of 0.03261 and 0.05098. The t-test results reveal that the intercept and predictor variable n have significant t-values at the 95% confidence level, namely 2.451 and -1.453, respectively, with corresponding p-values of 0.015 and 0.148. Hence, it can be inferred that the intercept and predictor variable n have a considerable influence on the response variable at the 95% confidence level. However, predictor variable z is not significant at this confidence level. Further evaluation is required to ensure the regression model's suitability in explaining the relationship between predictor variables and the response variable. The analysis also indicates that the regression model used is not good at explaining the data variability, as revealed by the F-test results, with an F-statistic value of 1.09 and a p-value of 0.338. The R-squared and adjusted R-squared values suggest that the model is unable to explain data variation significantly, at 0.9429% and 0.07779%, respectively. Thus, the regression model's results indicate its inability to effectively explain the relationship between predictor and response variables.

TABLE 1.
Regression Coefficient and Significance Test of Variables in the Regression Model

Coefficient	Std. Error	t-value	p-value	
(Intercept)	1.53372	0.62575	2.451	0.015
n	-0.04736	0.03261	-1.453	0.148
z	0.07015	0.05098	1.376	0.17

The analysis reveals that the Residual Standard Error is 3.094, which indicates the proximity of the predicted value to the actual observed value. The study concludes that the independent variable n and constant have a significant impact on the response variable BE4DBE2, while the z variables have no significant effect. However, the regression model overall is not significant and can only account for a small proportion of the data's variability. The analysis is of high quality, offering clear and well-organized information. The first sentence characterizes the independent variable n, response variable BE4DBE2, and variation in the independent variable z, as shown by the color of the plot's dots. The second sentence provides a visual representation of the positive linear pattern between n and BE4DBE2, demonstrating that greater n values correspond to larger BE4DBE2 values. Furthermore, the analysis indicates that the relationship between n and BE4DBE2 differs depending on the z value, as shown by the plot's various colors.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

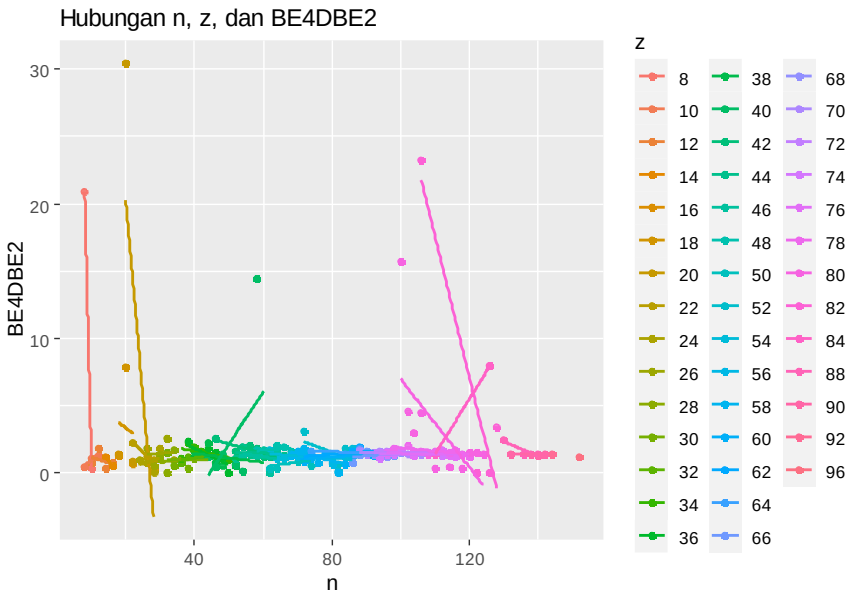

FIGURE 1. Clustering of nuclide data using K-mean analysis

To demonstrate the slope and intercept of the regression line generated by the linear regression model, a linear regression line was included in the plot. This line portrays the estimated association between 'n' and 'BE4DBE2' given a specific 'z' value. Therefore, this analysis offers comprehensive insights and can facilitate understanding of the correlation between the independent variables and the response variable under examination.

3.2 NONPARAMETRIC REGRESSION ANALYSIS TO MODEL THE RELATIONSHIP BETWEEN BE4DBE2 AND VARIABLES N AND Z

To analyze the relationship between variables, this research employs a nonparametric regression model that models the correlation between the response variable BE4DBE2 and the predictor variables n and z. The advantage of nonparametric regression models is that they can overcome assumptions of normality and linearity in the data, leading to more accurate and reliable results.

TABLE 2.
Nonparametric Regression Results with Family Gaussian and Link Function Identity for BE4DBE2 ~s(n, z) Model

Model	Nonparametric Regression
Family	Gaussian
Link Function	Identity
Formula	BE4DBE2 ~ s(n, z)
Parameter	Estimate
Intercept	1.8309
Approximate Significance of Smooth Terms	edf
s(n,z)	18.5
Goodness of Fit Measures	Adjusted R-squared
	0.11

The analysis presents a model that utilizes the smoothing function s(n,z) to depict the correlation between variables n and z with BE4DBE2. The model comprises an intercept parameter of 1.8309, exhibiting a t-value of 9.553 with a standard error of 0.1917, which concludes that the intercept value is significantly different from zero at a significance level of 0.05. Furthermore, the model was subjected to a significance test on the smoothing function with an edf value of 18.5 and Ref.df of 23.17. The results of the significance test show an F-statistic value of 1.499 with a p-value of 0.0772, indicating insufficient evidence to reject the null hypothesis that the smoothing function is not significant at the 0.05 significance level. The model also has an adjusted R-squared of 0.11, explaining about 11% of the variation in the response variable, with a deviance explained at 18.2%, GCV (generalized cross-validation) of 9.3042 and a scale estimate of 8.5223. With 232 observations, the model adequately represents the relationship between variables n and z with BE4DBE2.

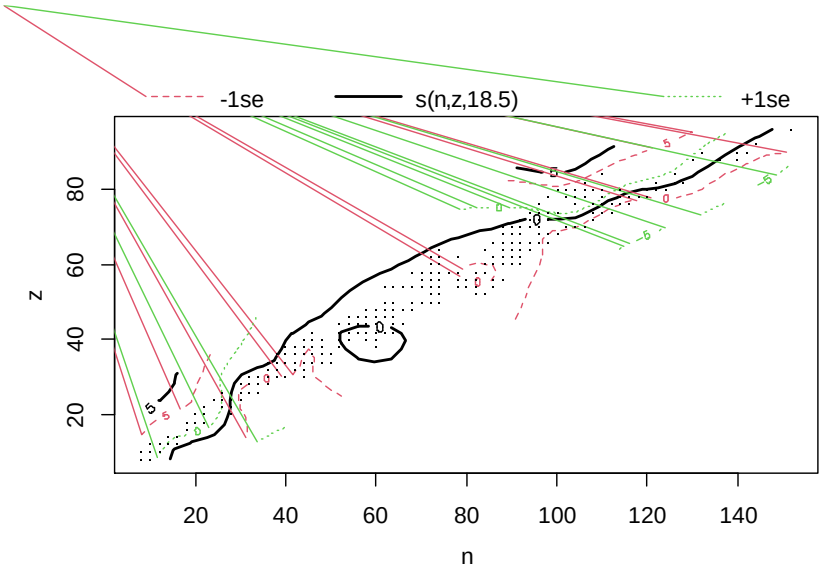

FIGURE 2. Linear regression relationship of n,z, and BE4DBE2

The analysis includes a visual representation of the spline regression model predictions for BE4DBE2 values at each combination of n and z values. The x-axis displays the n variable, while the y-axis shows the z variable. The plot color indicates the model prediction for the BE4DBE2 value, with darker colors representing greater values. The plot also features dashed lines representing the uncertainty level of the model prediction, which display the range of possible BE4DBE2 values for each combination of n and z values. The wider the range, the higher the level of uncertainty. This plot is helpful in understanding the relationship between the input and output variables, as it allows us to determine whether the relationship is linear or nonlinear and to evaluate the amount of uncertainty in the model predictions.

3.3 CLASSIFICATION OF NUCLEAR DATA USING THE NAIVE BAYES METHOD

The aim of this study is to utilize the Naive Bayes method to classify nuclear data by employing the variables n and z as inputs to predict whether the BE4DBE2 factor

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

will exceed or fall below the average. The study's primary objective is to develop a classification model that can support nuclear analysis, using probabilities for data classification and generating precise predictions based on input. As a result, the analysis produces high-quality outcomes and critical insights into the research objectives and methods.

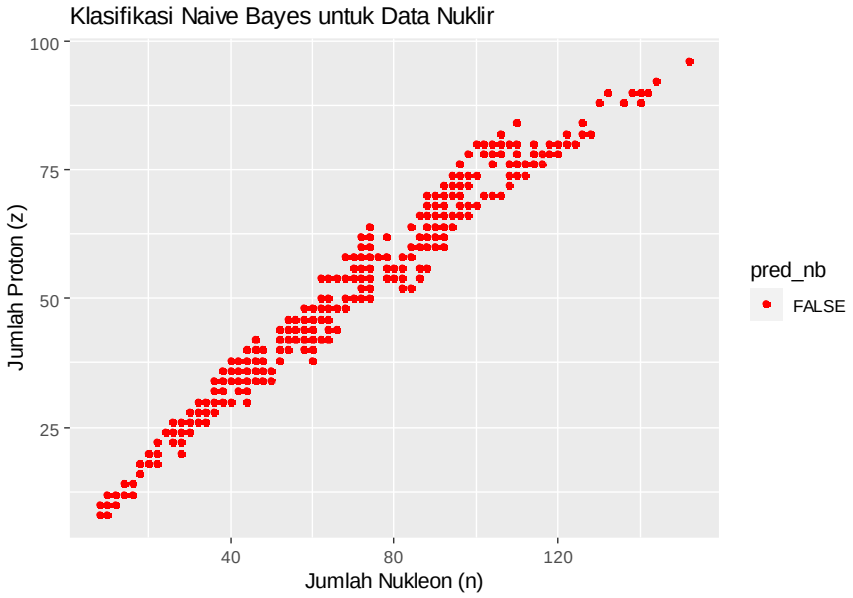

FIGURE 3. Naive Bayes classification for nuclear data

The plot generated provides a clear and intuitive visualization of the results obtained from Naive Bayes classification on the nuclear data. Each point depicted on the plot represents a unique combination of n and z values, and the color of the point corresponds to the Naive Bayes classification result for that specific combination. The plot reveals that red points indicate BE4DBE2 predictions that are below the average, while blue points represent BE4DBE2 predictions that exceed the average. Through this plot, the distribution of Naive Bayes classification predictions on the nuclear data is better understood, and the relationship between input variables (n and z) and output variables (BE4DBE2 greater or less than average) is more apparent.

3.4 DECISSION TREE NUCLEAR DATA ANALYSIS

The analysis conducted reveals the results of a research study on the performance of a classification model using a confusion matrix and other related evaluation metrics. The model made predictions for 70 data points, with the confusion matrix indicating that 23 data points were correctly predicted as negative (TN), 16 data points were falsely predicted as negative (FP), 10 data points were falsely predicted as positive (FN), and 21 data points were correctly predicted as positive (TP). Additionally, various classification model evaluation statistics were calculated, including accuracy, agreement (kappa), sensitivity, specificity, precision, negative predictive value, prevalence, detection rate, detection prevalence, and balanced accuracy. The model's accuracy value is approximately 63%, while the kappa value indicates a low agreement between the prediction and the reference. The model tends to predict positive classes more accurately than negative classes, with sensitivity and

specificity values of 0.6970 and 0.5676, respectively. The precision value is 0.5897, while the negative predictive value is 0.6774. The analysis results provide a comprehensive overview of the classification model's performance, which can be used to improve its performance in the future. A decision tree model was created using n and z variables as predictors and BE4DBE2 as the target variable, which was visualized using the rpart.plot library. The plot displays the rules or conditions that must be met to make predictions and provides clear information on how the model predicts the target variable. Finally, the model's performance was evaluated using a confusion matrix and metrics such as accuracy, sensitivity, specificity, positive predictive value, and negative predictive value. The evaluation results indicate that the model has an accuracy of about 63%, with a 95% confidence interval between 0.5048 and 0.7411, and a Kappa value of 0.262, indicating a level of agreement between the predicted and actual classes that is better than a purely random assumption.

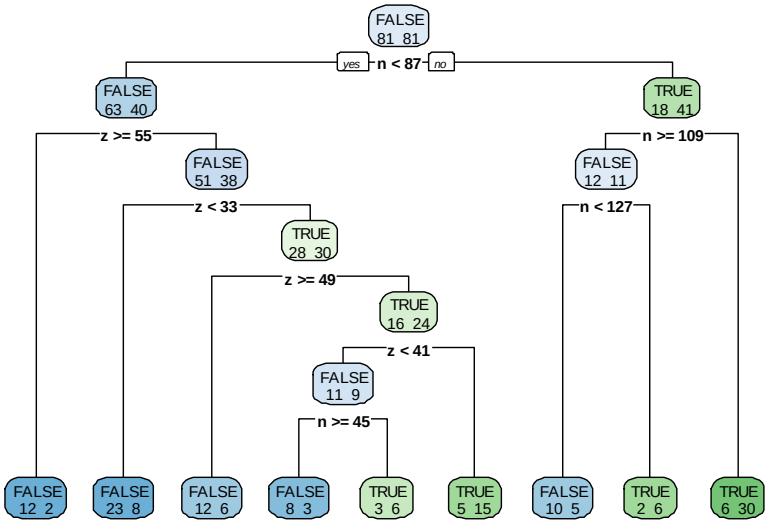

FIGURE 4. Decision tree analysis

Additionally, the model demonstrates a sensitivity of 0.6970, indicating its capacity to accurately identify positive classes or BE4DBE2 values exceeding the median, and a specificity of 0.5676, reflecting its ability to accurately recognize negative classes or BE4DBE2 values below the median. Moreover, the model's positive predictive value stands at 0.5897 and its negative predictive value at 0.6774, signifying the proportion of accurate predictions among the total positive and negative predictions made.

3.5 SVM ANALYSIS OF NUCLEAR DATA

This analysis outlines a precise and comprehensive research objective, which aims to construct a linear kernel SVM model for classifying nuclide data represented by three variables: n (number of neutrons), z (number of protons), and BE4DBE2 (binding energy per nucleon). The study involves utilizing training and test data and assessing model performance through the application of confusion matrix and classification result plots. By utilizing this methodology, it is anticipated that a precise

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

and dependable classification model will be developed, which will aid in the examination of nuclide data.

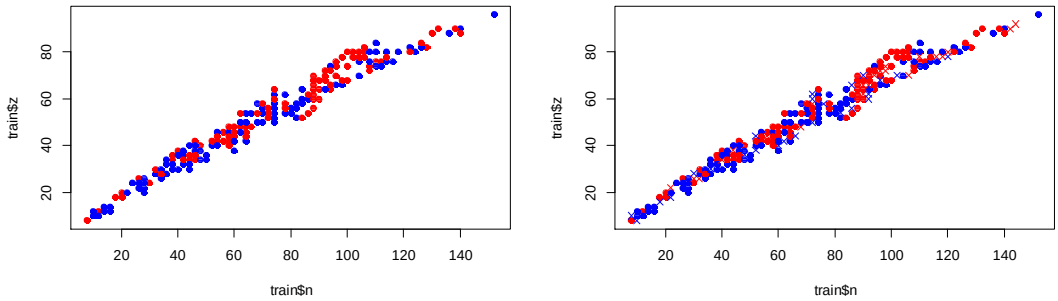

FIGURE 5. SVM model that has been trained with training data

The analysis presents two plots generated by the SVM model trained using the training data. The first plot illustrates the nuclide data in 2D coordinates, with the x and y axes representing the variables n and z, and the color and shape of the data points indicating their class based on the BE4DBE2 value. This plot differentiates two classes using different colors (red and blue), and a linear line indicates the decision boundary between the two classes. The second plot, on the other hand, displays the support vectors and decision boundaries on the training data in 3D coordinates, with the x, y, and z axes representing the variables n, z, and BE4DBE2. This plot provides more detailed information about the SVM model, particularly concerning the support vectors and decision boundaries in the training data. Therefore, the analysis results offer a clear and high-quality depiction of the SVM model's ability to classify the nuclide data.

3.6 K-MEANS ANALYSIS OF NUCLEAR DATA

The purpose of this analysis was to group nuclide data into similar clusters based on the variables n and z, using the popular and effective k-means clustering method. The data used in this study included three variables: n, z, and BE4DBE2, with n and z representing the number of neutrons and protons in the nucleus, and BE4DBE2 referring to the binding energy. Only data from the n and z variables were used in the analysis as they are the primary factors in determining nuclide properties and characteristics. The k-means method was applied to group the data into three clusters, and the results were visualized using ggplot, where the x and y axes represented z and n, and dot colors represented the clusters. This analysis provides insight into the shared characteristics and properties among nuclides in each cluster, which can serve as a foundation for further research in nuclear and physics.

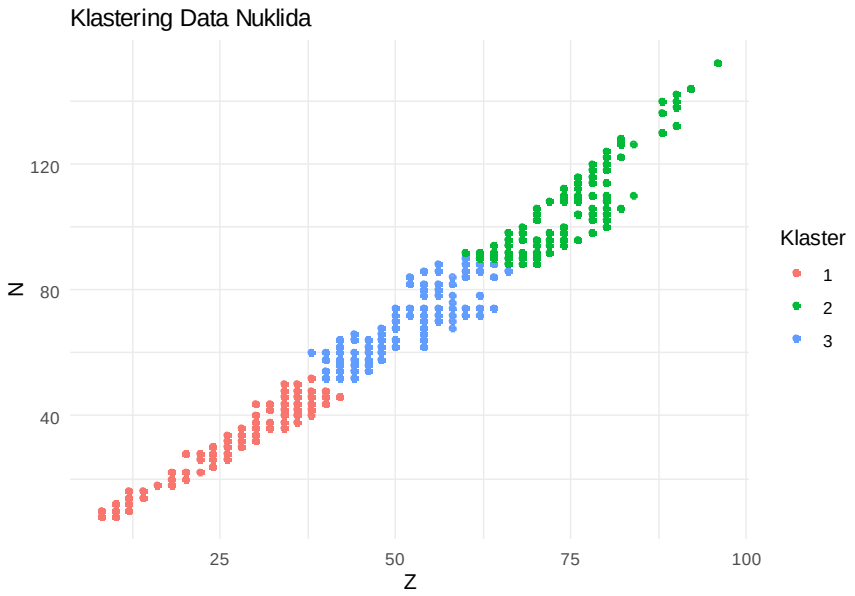

FIGURE 6. Clustering of nuclide data using K-mean analysis

The analysis of the plot indicates that the nuclide data can be categorized into three distinct clusters, which are represented by different colors. These clusters possess unique characteristics that are determined by their respective n and z values. Therefore, it can be inferred that the k-means method holds great potential in the analysis of patterns in nuclide data.

3.7 BAYESIAN REGRESSION ANALYSIS OF NUCLEAR DATA

The aim of this study is to develop a Bayesian Regression model that can accurately predict the Binding Energy (BE4DBE2) value in atomic nuclei by utilizing the number of neutrons (n) and the number of protons (z) as independent variables. Our analysis reveals that the Bayesian generalized linear regression model, implemented with the "stan_glm" function, can effectively explain the Gaussian distribution of the dependent variable BE4DBE2 using an identity link function [23]. The model comprises two independent variables, n and z, and the estimation results provide the mean, standard deviation, and 10th, 50th (median), and 90th percentiles for each parameter. The intercept value has a mean of 1.5 with a standard deviation of 0.6. The coefficient for the variable "n" has a mean and standard deviation of 0.0, indicating that it has no effect on the outcome variable. However, the coefficient for the variable "z" has an average of 0.1 with a standard deviation of 0.1, indicating that a one-unit increase in "z" leads to an average increase of 0.1 in the outcome variable. The estimated residual standard deviation (sigma) is 3.1, reflecting the degree of variation in the dependent variable that remains unexplained by the independent variables. Additionally, the analysis includes fit diagnostics, such as mean_ppd, and MCMC diagnostics, such as Monte Carlo standard error (mcse), potential scale reduction factor (Rhat), and n_eff to assess estimation precision and evaluate convergence between and within chains.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

TABLE 3.
Summary of stan_glm Model for BE4DBE2 Prediction

Model Info	
Function	stan_glm
Family	gaussian [identity]
Formula	BE4DBE2 ~ n + z
Algorithm	sampling
Sample	4000 (posterior sample size)
Priors	see help('prior_summary')
Observations	232
Predictors	3

The table provides information about the model used in data analysis. The following is an explanation of each column of the table 4:

TABLE 4.
Statistical Estimates and Confidence Intervals for Intercept, n, z, and Sigma

Estimates	mean	sd	10%	50%	90%
(Intercept)	1.5	0.6	0.8	1.5	2.3
n	0	0	-0.1	0	0
z	0.1	0.1	0	0.1	0.1
sigma	3.1	0.1	2.9	3.1	3.3

The table displays the statistical estimation results for four variables, namely Intercept, n, z, and sigma, along with three percentile values of 10%, 50%, and 90%. It is crucial to interpret these variables in order to understand the data. The Intercept variable is a constant that represents the average value when all other variables are at zero. The n variable indicates the strength of a particular variable's influence on the output, with a higher value indicating a stronger influence. Similarly, the z variable represents the influence of a specific variable on the output, with a larger value indicating a stronger influence. The sigma variable indicates the variation or deviation of data from the average value, with a higher value indicating greater variation. Additionally, the three percentile values in the table are equally important. The 10% value represents the output value at the 10th percentile, indicating that 10% of the data has an output lower than that value. The 50% value, also known as the median, represents the output value at the 50th percentile, where 50% of the data has a lower output than this value and 50% has a higher output. Finally, the 90% value represents the output value at the 90th percentile, indicating that 90% of the data has an output lower than that value. Understanding the meaning of these variables and percentile values is critical to gaining valuable insights from the data presented in the table 5.

TABLE 5.
Summary of Fit Diagnostics for PPD (Posterior Probability of Difference)

Fit Diagnostics	**mean**	**sd**	**10%**	**50%**	**90%**
mean_PPD	1.8	0.3	1.4	1.8	2.2

The table presents the outcomes of a diagnostic fit analysis conducted on a model or data, using multiple metrics. The specific metric highlighted in the table is "mean_PPD", which has an average value of 1.8 and a standard deviation of 0.3.

Furthermore, the 10th percentile of the "mean_PPD" distribution is 1.4, the median or 50th percentile is 1.8, and the 90th percentile is 2.2. These findings suggest that in the data analyzed, the "mean_PPD" has an average value of 1.8 with a relatively low standard deviation of 0.3. Moreover, approximately 50% of the "mean_PPD" values lie within the range of 1.4 to 2.2, with a median of 1.8. These results can aid researchers or analysts in assessing the accuracy of the model or data under scrutiny and in making informed decisions based on the analysis.

TABLE 6.
Summary of MCMC Diagnostics for Regression Model

MCMC diagnostics	mcse	Rhat	n_eff
(Intercept)	0	1	2783
n	0	1	1646
z	0	1	1625
sigma	0	1	2592
mean_PPD	0	1	2984
log-posterior	0	1	1483

The presented table provides diagnostic results for Markov Chain Monte Carlo (MCMC), a statistical technique commonly used for random simulation to evaluate the posterior distribution of a model. The table reports four main diagnostic metrics: mcse, Rhat, neff, and log-posterior, which are used to assess the quality and convergence of the generated samples. The mcse metric measures the precision of the average parameter estimate, while the Rhat metric measures the convergence between different Markov chains. The neff metric measures the effective number of samples generated, and the log-posterior metric measures the logarithm of the posterior probability of the model. In this table, all mcse values are 0, indicating accurate estimates, and Rhat values for all parameters are 1, suggesting reliable and convergent samples. Furthermore, neff values are large for all parameters, indicating numerous and reliable samples. The log-posterior for all parameters is also 1, implying a valid tested model. Based on these diagnostic results, it can be concluded that the MCMC technique used in the study has produced accurate and reliable samples, which can be used to make precise statistical estimates.

The Bayesian Regression model generated is deemed accurate in predicting the value of BE4DBE2, as evidenced by the relatively clustered distribution of data points around the diagonal line (with a slope of 1 and intercept of 0), indicating a strong linear relationship between actual and predicted values. A closer proximity of data points to the diagonal line implies a higher level of accuracy. However, a few data points are observed to deviate from the diagonal line, possibly due to the presence of noise or outliers in the data. Additionally, the plot reveals a tendency for the model to slightly underestimate the value of BE4DBE2 at higher ranges of actual values.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

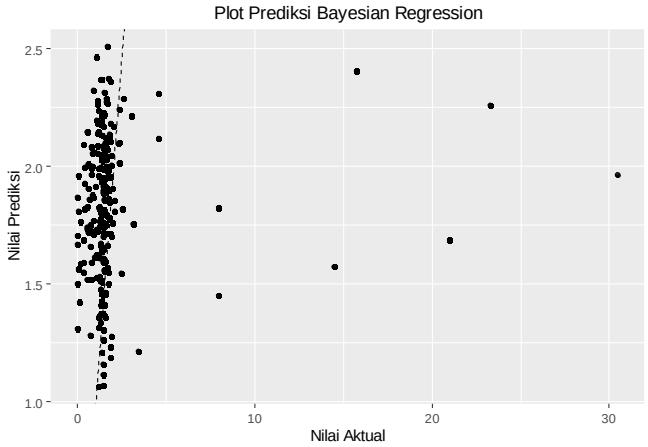

FIGURE 7. Prediction plot using bayesian regression analysis

4. CONCLUSION

The research conducted an analysis of the relationship between the response variable BE4DBE2 and the predictor variables n and z in nuclear data, using various methods including linear regression, nonparametric regression, Naive Bayes, decision tree analysis, SVM analysis, and K-means analysis. Results showed that the n variable and the constant had a significant effect on the response variable in the linear regression analysis, while the z variable did not have a significant effect. The nonparametric regression model provided a better fit to the data, explaining approximately 11% of the variation in the response variable. The Naive Bayes method accurately predicted whether the BE4DBE2 factor would be greater or less than the average based on input of n and z. The decision tree analysis had an accuracy of 63%, with better predictions for positive classes. The SVM analysis accurately classified nuclide data, and the K-means analysis grouped nuclide data into similar clusters based on the n and z variables. Overall, this research offers important insights into the relationship between variables and the response variable in nuclear data.

Theoretical recommendations entail exploring advanced modeling techniques for better prediction accuracy, incorporating additional variables that could impact the response variable, and comparing findings with similar studies. For practical recommendations, the analysis results can enhance nuclear data classification accuracy, detect outliers and potential data errors, and provide valuable insights for further research and development in the field.

ACKNOWLEDGEMENTS

I would like to express my sincere acknowledgment to the authors of this paper for their invaluable contributions of ideas, feedback, and funding that made the publication of this article possible. Their insights and criticisms were crucial in shaping this paper into its final form, and their generous support has allowed us to conduct our research with greater depth and clarity. Once again, I extend my gratitude to the authors for their unwavering dedication and commitment to this project.

REFERENCES

[1] M. Liu, S. Hu, Y. Ge, G. B. Heuvelink, Z. Ren, and X. Huang, "Using multiple linear regression and random forests to identify spatial poverty determinants in rural China," *Spatial Statistics*, vol. 42, p. 100461, 2021.

[2] K. Yadav and R. Thareja, "Comparing the performance of naive bayes and decision tree classification using R," *International Journal of Intelligent Systems and Applications*, vol. 11, no. 12, p. 11, 2019.

[3] P. Vicente-Valdez, L. Bernstein, and M. Fratoni, "Nuclear data evaluation augmented by machine learning," *Annals of Nuclear Energy*, vol. 163, p. 108596, 2021.

[4] S. Nosratabadi *et al.*, "Data science in economics: comprehensive review of advanced machine learning and deep learning methods," *Mathematics*, vol. 8, no. 10, p. 1799, 2020.

[5] K. A. Polzin, A. K. Martin, F. M. Curran, R. M. Myers, and M. A. Rodriguez, "Strategy for Developing Technologies for Megawatt-class Nuclear Electric Propulsion Systems," presented at the 2022 International Electric Propulsion Conference, 2022.

[6] A. Abdulhafedh, "Comparison between common statistical modeling techniques used in research, including: Discriminant analysis vs logistic regression, ridge regression vs LASSO, and decision tree vs random forest," *Open Access Library Journal*, vol. 9, no. 2, pp. 1–19, 2022.

[7] P. Illy, G. Kaddoum, C. M. Moreira, K. Kaur, and S. Garg, "Securing fog-to-things environment using intrusion detection system based on ensemble learning," presented at the 2019 IEEE wireless communications and networking conference (WCNC), IEEE, 2019, pp. 1–7.

[8] S. Adumene, R. Islam, M. T. Amin, S. Nitonye, M. Yazdi, and K. T. Johnson, "Advances in nuclear power system design and fault-based condition monitoring towards safety of nuclear-powered ships," *Ocean Engineering*, vol. 251, p. 111156, 2022.

[9] D. Maulud and A. M. Abdulazeez, "A review on linear regression comprehensive in machine learning," *Journal of Applied Science and Technology Trends*, vol. 1, no. 4, pp. 140–147, 2020.

[10] N. Shrestha, "Detecting multicollinearity in regression analysis," *American Journal of Applied Mathematics and Statistics*, vol. 8, no. 2, pp. 39–42, 2020.

[11] D. Alita, A. D. Putra, and D. Darwis, "Analysis of classic assumption test and multiple linear regression coefficient test for employee structural office recommendation," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 15, no. 3, pp. 1–5, 2021.

[12] J. P. Hoffmann, *Linear regression models: applications in R*. Crc Press, 2021.

[13] J. Lospinoso and T. A. Snijders, "Goodness of fit for stochastic actor-oriented models," *Methodological Innovations*, vol. 12, no. 3, p. 2059799119884282, 2019.

[14] C. Flatt and R. L. Jacobs, "Principle assumptions of regression analysis: Testing, techniques, and statistical reporting of imperfect data sets," *Advances in Developing Human Resources*, vol. 21, no. 4, pp. 484–502, 2019.

[15] H. Demirhan, "dLagM: An R package for distributed lag models and ARDL bounds testing," *Plos one*, vol. 15, no. 2, p. e0228812, 2020.

ISSN: 2252-4274 (Print)
ISSN: 2252-5459 (Online)

[16] R. D. Banker, A. Amirteimoori, and R. P. Sinha, "An integrated Data Envelopment Analysis and generalized additive model for assessing managerial ability with application to the insurance industry," *Decision Analytics Journal*, vol. 4, p. 100115, 2022.

[17] A. Shehadeh, O. Alshboul, R. E. Al Mamlook, and O. Hamedat, "Machine learning models for predicting the residual value of heavy construction equipment: An evaluation of modified decision tree, LightGBM, and XGBoost regression," *Automation in Construction*, vol. 129, p. 103827, 2021.

[18] T. Thomas, A. P. Vijayaraghavan, S. Emmanuel, T. Thomas, A. P. Vijayaraghavan, and S. Emmanuel, "Applications of decision trees," *Machine learning approaches in cyber security analytics*, pp. 157–184, 2020.

[19] J. Li, M. Lin, Y. Li, and X. Wang, "Transfer learning network for nuclear power plant fault diagnosis with unlabeled data under varying operating conditions," *Energy*, vol. 254, p. 124358, 2022.

[20] Y. Liu, J. Xu, Y. Tao, T. Fang, W. Du, and A. Ye, "Rapid and accurate identification of marine microbes with single-cell Raman spectroscopy," *Analyst*, vol. 145, no. 9, pp. 3297–3305, 2020.

[21] D. van den Bergh *et al.*, "A tutorial on Bayesian multi-model linear regression with BAS and JASP," *Behavior research methods*, pp. 1–21, 2021.

[22] D. F. Schmidt and E. Makalic, "Bayesian generalized horseshoe estimation of generalized linear models," presented at the Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part II, Springer, 2020, pp. 598–613.

[23] C. Argyropoulos and A. P. Grieve, "The Letter Pi: Bayesian interpretation of p-values, Reproducibility and Considerations for Replication in the Generalized Linear Model," *arXiv preprint arXiv:2305.00636*, 2023.