IOSR Journal of Business and Management (IOSR-JBM)
e-ISSN:2278-487X, p-ISSN: 2319-7668. Volume 27, Issue 5. Ser. 9 (May. 2025), PP 22-29
www.iosrjournals.org

Analysis Of Naive Bayes And Random Forest Classification Models

Juliana Bruno Pereira[1], Yuri Simões Gomes[2], Jean Mark Lobo De Oliveira[3], David Barbosa De Alencar[4]

FAMETRO University Center, Metropolitan Institute Of Education (IME), AV. CONSTANTINO NERY, 3204, CHAPADA, CEP: 69050-001, MANAUS/AM. Brasil.

Professor, Of The Postgraduate Program In Engineering, Process Management, Systems And Environmental (PGP.EPMSE) - Institute Of Technology And Education Galileo Of The Amazon –ITEGAM - Amazonas – Brazil.

Abstract:
Machine learning (ML) techniques allow you to understand various algorithms and their performance in different applications. The case study presents a comparative analysis between Naive Bayes and Random Forest algorithms through training on a customer review dataset. Using Google Colab and leading libraries for machine learning, such as Scikit-learn, the models were trained and validated. Through metrics, both performed by almost 98%. The results obtained showed that Naive Bayes has greater agility in processing, while Random Forest prioritized predictive robustness. In a real context, these differences can define the final interpretation, the comparison ensures that each model is better suited to the data set.

KeyWords: *Machine Learning; Classification; Naive Bayes; Random Forest; Metrics*.

Date of Submission: 14-05-2025 Date of Acceptance: 24-05-2025

I. Introduction

Machine learning (ML) techniques allow you to understand various algorithms and their performance in different applications. This behavior occurs because each data set has specific characteristics, such as unbalanced data, absence of values, and categorical variables. ML algorithms made it easier to solve problems based on classification and that required more accurate human reasoning to be solved (LUDERMIR, 2021). Some algorithms that stand out are Naive Bayes (NB) and Random Forest (RF), which are often used in various areas, such as fraud control, product recommendation, and clinical analysis.

According to Theobald (2021), choosing the right model for each situation is still a challenge, because although algorithms return a similar result, they differ in some points, such as accuracy, effectiveness, and processing. The comparison between them allows us to identify which model generates better results throughout the training and the ability to generalize in real scenarios.

Naive Bayes is a supervised algorithm, effective on large data sets, known as a probabilistic classifier because it is based on Bayes' theorem. According to Katti et al. (2021), this method allows categorizing the data to a new class based on the data already known. On the other hand, we have Random Forest, which is both a classification and regression technique that uses a hierarchy of decisions to achieve a high degree of predictive accuracy. They are trained on a random subset of the data, with substitution, and on a random subset of resources (FRATELLO et al., 2018).

Therefore, this study proposes the analysis of these two traditional classification models, Naive Bayes and Random Forest in order to identify which one is more efficient, considering the context in question in which they will be trained and validated. The analysis generated from this comparison will be done to show in detail, by means of consolidated metrics, such as, for example, accuracy, which algorithm presented the best performance and precision on the characteristics of the analyzed data, and to present why to take into account the comparison between algorithms before defining the best model, in order to obtain generalization, and with that, validate the model for new implementations.

II. Bibliographic Reference

In order to deepen this study, it is essential to understand the main concepts that form the main structure of the research. Topic 2.1 offers an introduction to the concept of ML and the main types of learning. Topics 2.2 and 2.3 describe the main concept of algorithms and exemplify how each one works. Topic 2.4 introduces the main metrics used to obtain the final result.

Fundamentals Of Machine Learning

The studies by Foote (2022) and Huyen (2024) conceptualize ML as computers that can analyze old data and identify patterns or even make predictions about new data. According to Géron (2021), bullying can be classified according to the amount and type of supervision they receive during training.

The main types of learning are supervened, unsupervised, and reinforcement learning. In supervised learning, The algorithms are trained on data that already has an answer variable, called labeled data. Lee and Shin (2020) state that, By using this already labeled data, the algorithm learns to categorize the information, as it already understands the right response to outputs mapped during training. In unsupervised learning, We have the opposite of what was presented above, we don't have the labeled data to identify the patterns and structure. To train the model using this learning, We need to group the data into so-called groups (clusters) without the need for human intervention, and observe what each grouping represents in the analyzed context. On the other hand, in reinforcement learning, the algorithm does not receive the correct answer, but receives a reinforcement, reward, or punishment signal. The algorithm makes a hypothesis based on the examples and determines whether this hypothesis was good or bad (LUDEMIR, 2021).

Naive Bayes

NB is a supervised algorithm, effective for Great datasets, known as probabilistic classifier because it is based on Bayes' theorem. By applying this algorithm to a dataset, it is possible to observe that certain data were documented and others were not, which may raise doubts about the final analysis (Oliveira and Alencar 2024). The theorem on which NB is based defines probability-based decision-making, as mentioned by the two authors, when there are missing values, There may be misinterpreted information. NB uses the independent variants, which are the values that we know and apply the probability calculation, showing if the data belongs to a certain class. AndThis class (category) is what we are trying to predict. A very common example to understand this algorithm is the spam classifier example. Let's imagine the following example:

A tool company wants to analyze if the number of emails received requesting quotes are, in fact, true or if they are spam. The NB classification algorithm will check, for example, how many times the word "budget" appears in each message, based on the probability calculation, then compare this data, and thus classify the appropriate emails into their respective spam and non-spam classes.

Random Forest

The RF technique is a classification and regression tree technique that uses multiple decision trees, combining techniques such as bagging and feature to achieve a high degree of predictive accuracy. According to Zhigang et at. (2024), RF is one of the widely used ML algorithms. It has low or high classification or correlation accuracy, depending on the amount of nodes created. They are trained on a random subset of the data with substitution and a random subset of features. A very common example to understand this algorithm is the example of the Titanic Crew members. Let's imagine the following example:

The ship was carrying 2244 passengers and crew, and 1500 were lost. The data analysis of this prediction algorithm is to correctly define the number of survivors with the number of people lost on the high seas. The random forest is classified with 500 trees, and 3 variables in each node and 5 boundaries in each variable. Which results in a prediction accuracy of 89%, which is defined according to the group (age, class, gender, etc.) of variables. A group can be a single better tree.

RF analysis can be used in various survival data evaluation decisions, its implementation can be difficult and depends on data input parameters, which ensures the understanding of predictors that individually or collectively influence the prediction (RIGATTI, 2017).

Metrics

Evaluating ML models is essential to validate performance, it is possible to find out the percentage of errors of each model, the training time and the complexity of the model, comparing the results to identify if the model will meet the requirements of the project (GUIMARÃES; MEIRELES; ALMEIDA, 2019).

There are several metrics that make it possible to evaluate the performance of the models, such as Accuracy (A), which according to Shalev et al. (2014), defines as the percentage of correct answers in the examples of classified tests. Its formula can be described below:

$$A = \frac{VP + VN}{VP + VN + FP + FN}$$

Being PV (True Positive), when the positive instances are classified as such, FP (False Positive), when the instances are negative, but are classified positive by the model and FN (False Negative) when the instances were positive and were classified negative.

Agarwall (2020) also argues that Accuracy (P) increases as false positives decrease. Formula below:

$$P = \frac{VP}{VP + FP}$$

Recall or Sensitivity, according to Agarwall (2020) are the instances classified as positive among all other positive ones.

$$S = \frac{VP}{VP + FP}$$

The F1 metric (F1-Score) represents the harmonic mean between accuracy and recall located between 0 and 1. (AGARWALL, 2020).

$$F1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

This approach ensures a balance between the two metrics, penalizing cases where one is significantly lower than the other. For instance, a model with high precision but low recall (or vice versa) will have a lower F1-Score, reflecting its imbalanced performance. The score ranges from 0 (worst performance) to 1 (best performance) and is particularly useful when there is no clear priority between precision and recall or when class distribution is uneven. Due to this characteristic, the F1-Score is commonly used in tasks such as fraud detection, medical diagnosis, and natural language processing, where both false positives and false negatives have significant consequences.

III. Methodology

In this section we will descriptively bring the methodology used for the analysis, including the origin of the datasets used, exploratory analysis, the main libraries used for ML algorithms, model training and validation. The implementation of the analysis was carried out using the Python 3.11.2 programming language, due to its ease of use and the range of libraries aimed at data analysis and machine learning. This Analysis was performed on the free cloud service Google Colab.

```
# Análise Exploratoria
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
import time

# Aprendizado de Máquina
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix, ConfusionMatrixDisplay,
recall_score, accuracy_score, precision_score, f1_score
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_selection import SelectKBest, chi2
import scipy.sparse as sp
```

Figure 1. Main Libraries used
Source: Authors, 2025

The code in this image shows the imports of the libraries required for the machine learning project, highlighting the main libraries used for data manipulation, model training, and performance validation.

The Database

The database used was extracted from the Kaggle website, specifically olist_customers_dataset.csv. This dataset contains public information regarding a Brazilian e-commerce and has records of 100 thousand orders placed between 2016 and 2018 in various marketplaces in Brazil.

These records were made available by Olist Store, a marketplace company that connects small businesses through one-stop sales contracts. In it, we can view orders from purchase to the customer's final evaluation. Your main tables contain, status price, payment, freight performance, location, product attributes and customer ratings. For this analysis, the focus will be on the customer evaluation table, specifically in column review_score.

Data Processing

Data processing consists of exploratory analysis to identify missing or unbalanced values and treat them correctly before implementing any algorithm. It was observed that the review_comment_title column had null values, because it was a small amount of data, it was decided to remove these values. Column review_score presented an evaluation of 1 to 5 for each purchase made online, this pattern can be complex for the model to understand. Therefore, we transform the evaluations into a binary of 0 and 1 so that the model understands the difference between positive and negative evaluations. That way, we'll have a unique identifier for each category instead of 5 distinct values. After this process we can create the independent and dependent variables.

Conversion Of Texts Into Numerical Representations (TF-IDF)

To work with the textual columns, the Term Frequency and Inverse Document Frequency (TF-IDF) technique was applied, which is used to transform text into numerical vectors and preserve the main semantics for classification. For this application, the sklearn.feature_extraction library is used.text import TfidfVectorizer, with the vectorized data, the data was divided between training and test sets, with 80% of the data destined to training and 20% to testing using the train_test_split function of the Scikit-learn library. This division aims to ensure the impartial evaluation of the models.

```
# Lista de stopwords
portuguese_stopwords = [
    "a", "ao", "aos", "aquela", "aquelas", "aquele", "aqueles", "aquilo", "as",
    "até", "com", "como", "da", "das", "de", "dela", "delas", "dele", "deles",
    "depois", "do", "dos", "e", "ela", "elas", "ele", "eles", "em", "entre",
    "era", "eram", "essa", "essas", "esse", "esses", "esta", "está", "estamos",
    "estão", "estas", "este", "estes", "eu", "foi", "fomos", "for", "foram",
    "havia", "houve", "isso", "isto", "já", "lhe", "lhes", "mas", "me", "mesmo",
    "minha", "minhas", "muito", "muitos", "na", "não", "nas", "nem", "nos",
    "nós", "nossa", "nossas", "nosso", "nossos", "num", "numa", "o", "os",
    "ou", "para", "pela", "pelas", "pelo", "pelos", "por", "porque", "quais",
    "qual", "quando", "que", "quem", "se", "seja", "sejam", "sem", "ser",
    "será", "serão", "seu", "seus", "só", "sua", "suas", "também", "te",
    "tem", "têm", "tendo", "tenha", "tenhamos", "tenhas", "tenho", "terá",
    "terão", "teu", "teus", "tive", "tivemos", "tiver", "tivera", "tiveram",
    "tu", "tua", "tuas", "um", "uma", "você", "vocês", "vos", "à", "às", "é"
]

# TF-IDF
vectorizer = TfidfVectorizer(max_features=5000, stop_words=portuguese_stopwords,)
X_text = vectorizer.fit_transform(clientes_reviews['review_comment_message'])
```

Figure 2. TF-IDF (Term Frequency and Reverse Document Frequency)
Source: Authors, 2025

The code in this image shows the application of TF-IDF to convert text into numerical data, a way for machine learning models to understand the data which are in text format.

Implementation Of The Models

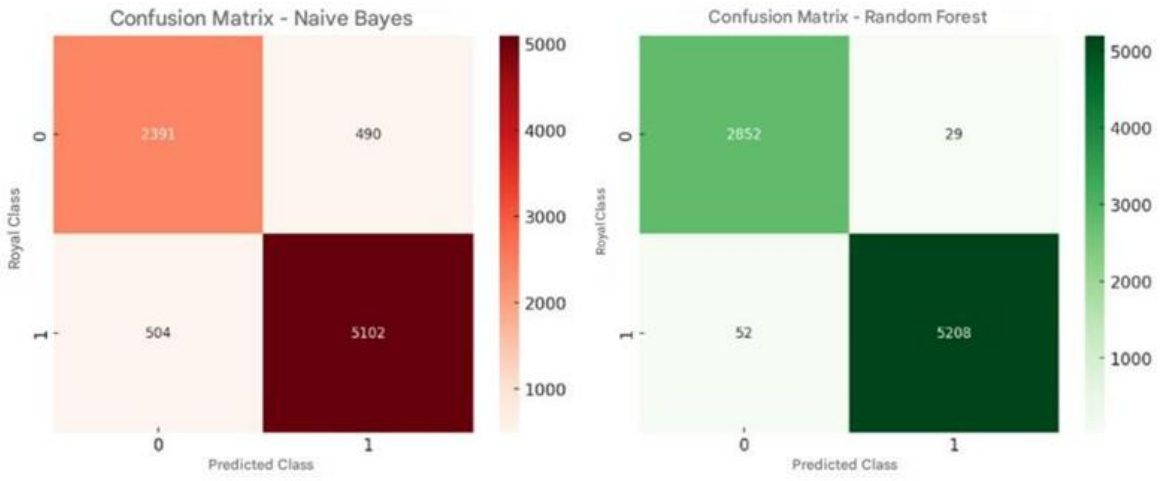
NB was trained using the MultinomialNB classifier, which is best suited for data with textual characteristics and works well with fractional counts such as TF-IDF. The RF was trained using the RandomForestClassifier, which fits a series of decision trees into several subsamples. We set the n_estimators=100, that is, this defines that the RF will create 100 decision trees to compose a tree, while the Random parameter in state=42 ensures that the process of randomness in the division of data for the process is consistent and that it can be reproduced in other executions.

Figure 3. NB and RF Implementation Code
Source: Authors, 2025

This code snippet shows how the models train to classify customer reviews as negative and positive. After this training, the models were able to make new predictions about new data sets.

Performance Evaluation

The sklearn.metrics library is used to validate the models through the metrics. After the model is trained (X_train and y_train) and tested (X_test and y_test) we use the train_and_evalute function that automatically returns the calculation of accuracy, precision, recall and F1-score. Below is the graph of the results.

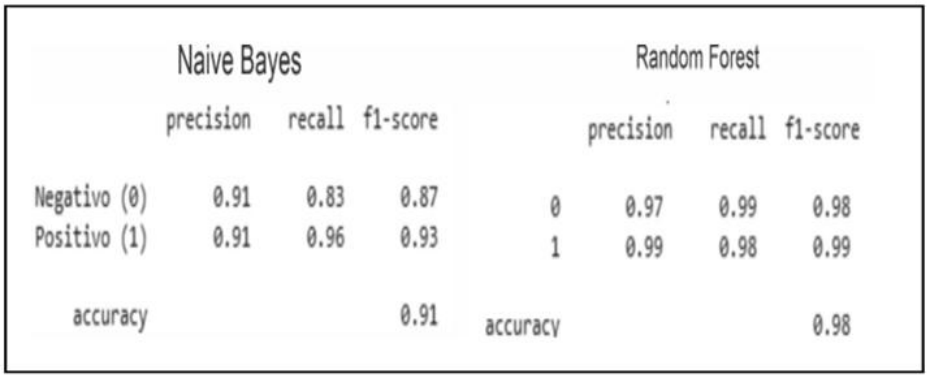


Graph 1. Confusion Matrix
Source: Authors, 2025

The confusion matrix graph was used to graphically represent the successes and errors of classification of the models, this matrix brought a more detailed approach to the distribution of true positives, true negatives, false positives and false negatives, offering a complementary view to the numerical metrics and enabling a better view of the comparison of the performance of the two models.

IV. Results

To evaluate the performance of the Naive Bayes (NB) and Random Forest (RF) models, widely recognized metrics in the area of machine learning, accuracy, precision, recall, F1-score, and execution time were used. Each of these metrics offers a complementary perspective on the performance of the algorithms, allowing for a broader analysis of each model's ability to correctly classify the data. The tests were conducted based on a dataset containing customer reviews, and the results obtained are presented and discussed below.



Naive Bayes

	precision	recall	f1-score
Negativo (0)	0.91	0.83	0.87
Positivo (1)	0.91	0.96	0.93
accuracy			0.91

Random Forest

	precision	recall	f1-score
0	0.97	0.99	0.98
1	0.99	0.98	0.99
accuracy			0.98

Figure 4. Ranking Report
Source: Authors, 2025

The classification report is a key tool in evaluating the performance of machine learning models applied to classification tasks. It provides a detailed overview of the model's performance when dealing with different classes, making it possible to analyze metrics such as accuracy, precision, recall, and F1-score. These metrics allow you to understand not only the model's hit rate, but also its ability to avoid false positives and negatives.

Acurace

Accuracy is one of the most widely used metrics in the evaluation of classification models. It represents the proportion of correct answers of the model in relation to the total number of predictions made, and is useful for measuring the overall effectiveness of the approach adopted. However, its interpretation should be done with caution, because, in unbalanced data sets, it can provide a false impression of good performance. In the present study, both models obtained satisfactory results, with emphasis on the Random Forest, which achieved a 98% success rate. Naive Bayes, on the other hand, obtained 91%, which also represents a consistent performance. The difference of 7 percentage points indicates a slight superiority of RF in the ability to generalize from training data.

Table 1. Accuracy of Random Forest and Naive Bayes models

Model	Accuracy (%)
Random Forest	98
Naive Bayes	91

Source: Authors, 2025

These results suggest that Random Forest was more effective at dealing with data complexity, better capturing patterns present in assessments. However, it is worth noting that accuracy, by itself, does not guarantee the overall quality of the model and should be analyzed in conjunction with other more specific metrics.

Recall And Accuracy

Recall and precision are metrics that complement the analysis of accuracy. Recall measures the model's ability to correctly identify positive cases (true positives), and is essential in contexts where losing a positive instance can bring great losses. Accuracy, on the other hand, evaluates how much of the positive classifications made by the model are correct, that is, how many of the positive predictions actually belong to the positive class. When analyzing these metrics, the Random Forest model performed better, with a recall of 98% and accuracy of 97%. Naive Bayes also performed well, achieving 96% recall and 91% accuracy, proving effective in identifying positive reviews, although with a higher propensity for false positives.

Table 2. Comparison between Recall and Model Accuracy

Model	Recall (%)	Accuracy (%)
Random Forest	98	97
Naive Bayes	96	91

Source: Authors, 2025

These data show that, although both models are competent for the classification task, Random Forest stands out for offering greater reliability in its positive predictions. This is particularly relevant in applications that require high security in automated decisions, such as screening critical evaluations or monitoring sensitive feedback.

F1-Score and Runtime

The F1-score is a derived metric that combines precision and recall in a single indicator, being especially useful when there is a need to balance the two dimensions. It is calculated as the harmonic mean between recall and accuracy, favoring models that perform well in both criteria. Execution time, on the other hand, represents the computational efficiency of the model, a fundamental aspect when dealing with large volumes of data or real-time applications.

Table 3. F1-Score and Runtime of the evaluated models

Model	F1-Score (%)	Runtime
Random Forest	99	1.64 seconds
Naive Bayes	93	0.01 seconds

Source: Authors, 2025

The results show that the Random Forest model obtained an F1-score of 99%, while the Naive Bayes achieved 93%. While RF demonstrated a greater ability to capture relevant patterns in assessments, NB excelled in execution time: only 0.01 seconds to train and predict the data, versus 1.64 seconds for Random Forest.

V. Conclusion

When analyzing the comparison between the models, we realized that the results are similar in numerical terms, the RF performed at 98% and the NB achieved 91% in the trained data. Despite this proximity, some factors show a difference between them. While NB took 0.01 seconds to process the data, RF took 1.64 seconds. This time difference says a lot about the functionality of each one on the data. NB, being a probabilistic model that treats variables independently, has a greater efficiency in text classification, having advantages in sets that contain textual data. RF, on the other hand, uses several decision trees, combining the results to generate a new final tree that is more robust than the initial one. The delay in training indicates that the model considers complex word relationships, such as the general structure of the text. This results in the creation of a larger number of nodes in the trees, which generates a more detailed but slower model.

When we look at the other metrics, we observe important differences. In accuracy, Random Forest performed better, demonstrating a lower number of errors made when ranking reviews. In Recall, Naive Bayes stood out by being able to better identify positive ratings with a percentage of 96%. In the real world, it's not about applying the model with the highest metric, but rather understanding the limitations and advantages of each. This comparative analysis reinforces the importance of deeply evaluating the performance of the models considering the context of the data and understanding the results through the metrics. Other techniques can be explored to ensure an in-depth study of the model choice, such as applying HyperParameter tuning using GridSearch CV to test various combinations and optimize the results of the models. Another approach would be the use of Natural Language Models (LLMs) used for sentiment analysis, which can be exploited to detect reactions and mood swings in texts, such as anger, hatred, or dissatisfaction, increasing the complexity of the analysis of evaluations and making the interpretation of data accurate.

Acknowledgements

To FAMETRO University Center, Metropolitan Institute of Education (IME), and to the Postgraduate Program in Engineering, Processes, Systems and Environmental Management of the Galileo Institute of Technology and Education of the Amazon (PPG.EGPSA/ITEGAM)..

References

[1] Agarwal, R. The 5 Classification Evaluation Metrics Every Data Scientist Must Know. Torwards Data Science, 2020. Available At: Https://Www.Kdnuggets.Com/2019/10/5-Classification-Evaluation-Metrics-Every-Data-Scientist-Must-Know.Html. Accessed On: 23 Apr.2025

[2] Faceli, Katti Et Al. Artificial Intelligence: A Machine Learning Approach. Rio De Janeiro: Ltc. Available At: Https://Repositorio.Usp.Br/Item/002208293. Accessed On: 23 Apr. 2025. , 2011

[3] Foote, Kd. A Brief History Of Deep Learning, 2022. Available At: Https://Www.Dataversity.Net/Brief-History-Deep-Learning/. Accessed On: 23 Apr.2025

[4] Frajacomo, H. C. Selection Of Snps Using Random Forests. Federal University Of São Carlos, 2020.

[5] Fratello, M.; Tagliaferri, R. Decision Trees And Random Forests. Encyclopedia Of Bioinformatics And Computational Biology: Abc Of Bioinformatics, V.1, P.374–383, 2018. Available At: Doi:10.1016/B978-0-12-809633-8.20337-3. Accessed On: 23 Apr.2025
[6] Géron, A. Mãos A Obra: Aprender De Máquina Com Scikit-Learn, Keras Tensorflow. Publisher: Alta Books, 2021.
[7] Guimarães, L. M. S.; Meireles, M. R. G.; Almeida, P. E. M. D. Evaluation Of The Pre-Processing And Training Stages In Text Classification Algorithms In The Context Of Information Retrieval. Perspectives In Information Science, Scielo Brazil, V. 24, P. 169–190, 2019.Accessed On: 23 Apr.2025
[8] Hu J, Szymczak S. A Review On Longitudinal Data Analysis With Random Forest. Brief Bioinform. 2023 Mar 19; 24(2):Bbad002. Available At: Doi: 10.1093/Bib/Bbad002. Pmid: 36653905; Pmcid: Pmc10025446. Accessed On: 23 Apr.2025.
[9] Huyen. C. Designing Machine Learning Systems: An Interactive Process For Production-Ready Applications . [S.L.: S.N.]: "Alta Books", 2024.
[10] Lee, I., And Shin, Y. J. Machine Learning For Enterprises: Applications, Algorithm Selection, And Challenges. Business Horizons, Volume 63, 2020. Https://Doi.Org/10.1016/J.Bushor.2019.10.005. Accessed 23 Apr.2025
[11] Ludermir, T. B. Artificial Intelligence And Machine Learning: Current State And Trends. Estudos Avançados, Scielo Brasil, V. 35, P. 85–94, 2021. Available At: Https://Doi.Org/10.1590/S0103-4014.2021.35101.007. Accessed On: 23 Apr.2025
[12] Rigatti, Sj. Random Forest. J Insur Med. 2017; 47(1):31-39. Available At: Doi: 10.17849/Insm-47-01-31-39.1. Pmid: 28836909.Accessed On: 23 Apr.2025
[13] Schade, G. Azure Machine Learning - Part 2. Available At: Https://Gabrielschade.Github.Io/2018/01/17/Azure-Machine-Learning-2.Html. Accessed On: 30 Mar. 2025.
[14] Theobald, O. Machine Learning For Absolute Beginners. [S.L.: S.N.]: "Scatterplot Press; 3rd Edition", 2021.
[15] Zhigang, S.,Wang, G, Li, P.,Wang H, Zhang, M., Liang X., An Improved Random Forest Based On The Classification Accuracy And Correlation Measurement Of Decision Trees, Expert Systems With Applications. Volume 237, Part B, 2024.Available At: 10.1016/J.Eswa.2023.121549. Accessed On: 23 Apr.2025.