Texila International Journal of Academic Research
ISSN: 2520-3088
DOI: 10.21522/TIJAR.2014.12.04.Art005

# Predicting Malaria Incidence in Guinea: A Real Time Machine Learning Tool

Gerard Christian Kuotu[1,2*], Nouman Diakite[1], Dioubate Mouhamed[1], Diallo Abdourahmane[1], Alioune Camara[1,3]

[1]*National Malaria Control Program (NMCP), Conakry, Guinea*
[2]*Aix-Marseille University, Marseille, France*
[3]*Gamal Abdel Nasser University, Conakry, Guinea*

***Abstract***

*Malaria remains a pressing public health issue in Guinea, with approximately 13 million individuals at risk of contracting the disease. Despite efforts to reduce malaria incidence, it remains the leading cause of consultations, hospitalizations, and deaths in the country. To address this challenge, machine learning (ML) techniques have gained traction in epidemiology for predicting disease outbreaks and identifying high-risk areas. During this internship, we aim to use ensemble learning algorithms to develop a predictive model for malaria incidence in Guinea. Our methodology involved data integration, feature engineering, and model training using various ML algorithms, such as logistic regression, random forest, decision tree, support vector machine, gradient boosting machine, artificial neural network and ensemble stacking leveraging diverse datasets, including clinical records, demographic health surveys, and climatic data spanning six years from 2018 to 2023. We evaluated model performance using the F1-score metric. We found that the ensemble stacking method, particularly balanced stacking, demonstrated superior predictive accuracy (F1-score = 0.74). This highlights the importance of interdisciplinary collaboration and data integration in epidemiological research, as well as the potential of ML in informing targeted interventions and resource allocation strategies for malaria control. Challenges such as multicollinearity and imbalanced datasets were addressed through robust statistical techniques and model tuning. This research underscores the significance of translating research findings into actionable insights for malaria control efforts in Guinea. By harnessing the power of ML and deploying user-friendly tools, public health authorities can make informed decisions to mitigate the burden of malaria and improve health outcomes for affected populations.*

***Keywords:*** *Epidemiology, Guinea, Malaria, Machine learning, Predictive Modeling.*

## Introduction

In Guinea, approximately 13 million people are at risk of contracting malaria, with 75% of the population estimated to be at risk according to the Ministry of Health. Malaria is the leading cause of consultations, hospitalizations, and deaths in the general population, with more than 1.5 million cases reported in 2018 [1]. Between 2017 and 2020, the number of malaria cases fell by 9.7%, from 354 to 320 per 1,000 inhabitants at risk nationwide, although the number of deaths rose by 1.6%, from 0.77 to 0.78 per 1,000 inhabitants at risk. The primary vector of transmission is the female Anopheles gambiae mosquito, which has a significant economic burden on the country, leading to decreased productivity and a loss of gross domestic product (GDP). The World Health Organization (WHO) estimates that between 2008 and 2015, malaria control interventions in

***Received: 04.08.2025*** ***Accepted: 06.08.2025*** ***Published on: 31.10.2025***
****Corresponding Author: kgerchrist@gmail.com***

Guinea prevented 1.3 million deaths and 4.6 million clinical cases [2].

Machine learning (ML) is becoming increasingly popular in the field of epidemiology for identifying factors that are associated with the incidence of infectious diseases and predicting their outbreaks. Ensemble learning is an ML algorithm that has been used to pinpoint areas with a high risk of malaria incidence and to determine the contextual factors associated with this risk. However, this research has not fully explored all possible prediction models, making it difficult to establish the most reliable stacking algorithm [3, 4]. Studies have further revealed that these methods can be used to detect spatial clusters of high malaria incidence and to recognize environmental and socioeconomic components that are associated with these clusters. Additionally, they are capable of revealing temporal patterns of malaria incidence and forecasting future incidence trends [4-6]. Although malaria-specific data-driven models are limited due to a lack of structured datasets [7-9], some predictive models for malaria incidence have been created in several West African countries, mostly using climatic features such as relative humidity, rainfall and temperature [10-13]. However, other powerful and versatile ML algorithms, such as the CART, random forest, GBM, SVM and ANN algorithms, could be used to develop a strong and accurate predictive model for the incidence of malaria, taking advantage of each algorithm's unique strengths and ability to capture complex nonlinear patterns in the data.

To reduce the burden of the disease, it is essential to understand the contextual factors that influence its incidence. In addition to climatic factors affecting the intensity, seasonality and geographical distribution of malaria transmission, vector, sociodemographic and economic parameters need to be taken into account to prevent malaria epidemics [14-18]. In addition, the ownership and use of insecticide-treated mosquito nets, combined with access to health care and clean water, are essential for reducing malaria cases [19-22]. In addition, the ownership and use of insecticide-treated mosquito nets, combined with access to health care and clean water, are essential for reducing malaria cases [23-27]. The aim of this work was to predict the incidence of malaria in Guinea at the national level by considering climatic, clinical and demographic factors. More specifically, we aimed to identify factors that are correlated with the incidence of malaria in Guinea, to build a machine learning model that predicts malaria incidence at the national level and to create a web application for malaria incidence prediction.

## Materials and Methods

Our data processing journey commenced with the collection and extraction of six years of diverse data from clinical and demographic health surveys and climatic records, focusing on variables such as malaria cases, mosquito net availability, population demographics, well-being indices, and climatic factors. Following this, the data underwent transformation and integration, ensuring standardized formats, coherence, and consistency across all datasets. Thorough quality assurance checks were conducted to validate the accuracy, completeness, and consistency, and the integrated dataset was explored to discern variable relationships and distribution characteristics. Relevant features were selected for machine learning analysis, including independent variables such as population, temperature, humidity, precipitation and rainfall, with subsequent splitting of the dataset for training and testing, coupled with preprocessing for model readiness. Machine learning models were developed using the prepared dataset to predict malaria incidence, their performance was evaluated, and iterations were made as needed to optimize predictive accuracy. Figures 1 and

2 summarize the data processing and machine learning flow, respectively.

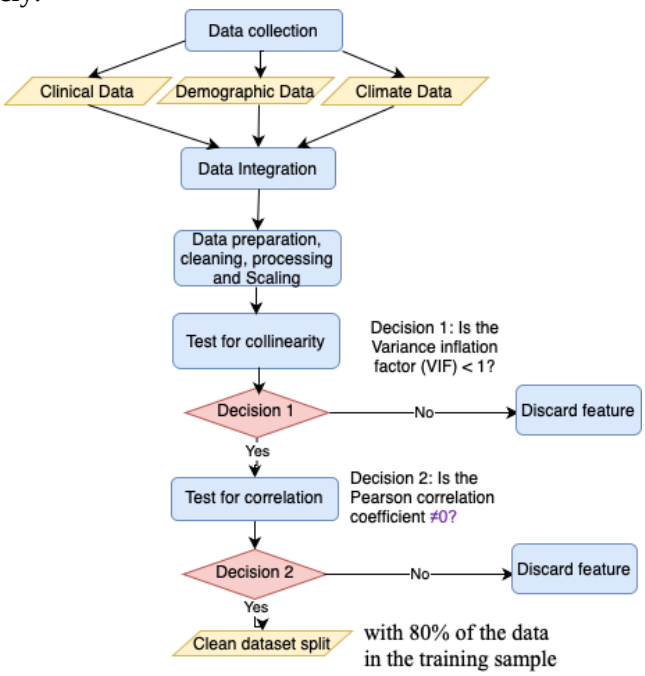


**Figure 1.** Data processing flow

## Data Collection

Our dataset comprises six years of continuous recording and originates from three distinct databases:

1. **Clinical data:** Malaria cases were sourced from the routine DHIS2, the national repository for data on malaria cases and other priority diseases in Guinea. These data were collected at the health facility level and recorded monthly in the DHIS2 across all 38 health districts in the country.
2. **Demographic health survey data:** Information regarding the availability of mosquito bed nets, total population, and socioeconomic well-being index was collected every three years at the national level.
3. **Climatic data:** Monthly average temperature, pressure, precipitation, and humidity values were collected in each of Guinea's eight administrative regions and stored in the national DHIS2 data warehouse.

We have integrated these three data sources into a unified dataset comprising seven variables and 2,736 observations spanning from 2018 to 2023:

1. Features:
   i. Total population
   ii. Temperature
   iii. Rainfall
   iv. Humidity
   v. Number of mosquito nets
   vi. Well-being index
   vii. Number of malaria cases
2. Target variable: Malaria incidence (quantitative variable).

## Data Integration and Processing

In the data processing pipeline, the initial phase involves data discovery, where the structure, format, and variables of each dataset are identified and understood. Following this, data extraction was carried out using appropriate methods to ensure representativeness and relevance. Subsequently, the data underwent transformation, including standardization, merging, cleaning, and enrichment, to prepare

them for analysis. Once transformed, the dataset was loaded into a CSV format, which was ready for exploratory data analysis and our machine learning application steps. Quality assurance measures were then implemented to validate accuracy, completeness, and consistency, addressing any identified issues. Data governance policies were applied to maintain security, privacy, and compliance, with access controls defined to safeguard sensitive information. Finally, procedures for ongoing maintenance and monitoring were established to ensure that the dataset remains accurate, up-to-date, and free from errors or anomalies, with regular updates from source systems and proactive monitoring of data pipelines.

## Exploratory Data Analysis (EDA)

We conducted a thorough exploratory data analysis (EDA) to understand the relationships between our features and the target variable, starting with scatter plots and line plots to provide insights into which features might be most relevant. We used a test for collinearity (variance inflation factor (VIF)) to identify predictors that had high collinearity. VIF values close to 1 indicate low multicollinearity, suggesting that the variance of the regression coefficient for that feature is not significantly inflated due to correlations with other features. We removed all variables with a VIF above 5 or 10 since they were considered high and indicated that multicollinearity may be a problem. We used Pearson's correlation test to assess the relationships between our feature variables and target variables. Pearson's correlation coefficient lies between -1 and +1, where -1 indicates a negative correlation, 0 indicates no correlation, and 1 signifies a strong positive correlation.

The monthly malaria incidence rate, defined as the number of confirmed malaria cases per 1000 inhabitants in the general population reported in a month, was determined and adjusted by considering the rates of confirmation of biological test diagnosis and attendance at health facilities for each health district. We used the crude incidence that was determined by reporting the cases of malaria confirmed by biological tests per 1000 inhabitants in the general population following this calculation algorithm:

The malaria incidence rate was calculated as the number of new cases/total population × 1000 person-months.

where:

Number of New Cases = Number of individuals who developed malaria in a given month registered at the health center.

Total Population = Total number of individuals living in the given area in the same month.

We will classify malaria incidence using the four WHO standard malaria incidence classes [14] as follows:

1. Very low malaria transmission zone: incidence less than 100 cases per 1000 people;
2. Low malaria transmission zone: incidence between 100 and 250 cases per 1000 people;
3. Moderate malaria transmission zone: incidence between 250 and 450 cases per 1000 people;
4. High malaria transmission zone: incidence greater than 450 cases per 1000 people.

## Feature Engineering

This step is critical for enhancing model performance and involves creating new features from existing features to better capture the underlying patterns in our data. The following steps were performed:

1. **Encode the Target Variable:** Since our newly created variable 'Class' was categorical, to ensure that it was in a format suitable for machine learning algorithms, we converted these categories into numerical codes: Very low = 0, Medium = 1, Medium = 2, and High = 3.

2. **Standardization/Normalization of Numerical Features:** Since our features were likely on different scales (temperature vs. pressure), we found it beneficial to normalize (scale them to a range between 0 and 1). This is particularly important for models such as support vector machines (SVMs) and can also help with gradient descent convergence in neural networks.
3. **Creating interaction terms:** Sometimes, the interaction between two or more features can have a significant impact on the target variable. For example, high temperature combined with high humidity might have a different effect on the 'Class' than each feature individually.
4. **Polynomial Features:** Generating polynomial and interaction features can uncover relationships between features that can help improve model performance.
5. **Missing Values:** If any of the features have missing values, we will need to decide whether to fill them (imputation), remove the rows with missing values, or even use the presence of missing values as a feature itself. Our pipeline performs simple imputation by imputing any missing values.
6. **Feature Selection:** After adding polynomial features, the dimensionality of our data increases, and not all features might be useful for predicting our target variable. We used recursive feature elimination (RFE) to select the most important features.

## Model Training and Tuning

After preprocessing, we split our data into training and testing sets. The training set of 80% of our process dataset and testing set (20%) will contain the scaled original features, their polynomial transformations, and the interaction terms, which are ready to be used for training machine learning models.

To find the optimal hyperparameters for a given model to maximize its performance, we used “grid search and cross validation”, a popular method for hyperparameter tuning in Python's scikit-learn library, which performs an exhaustive search over a specified parameter grid and returns the best parameters.

Since our “Class” variable was imbalanced, we trained our models with both imbalanced and balanced approaches. Then, we compared all two approaches together to select the best model to be used in our application as a predictive model. The modeling, evaluation and selection stages are described in detail in the following figure below (Figure 2).

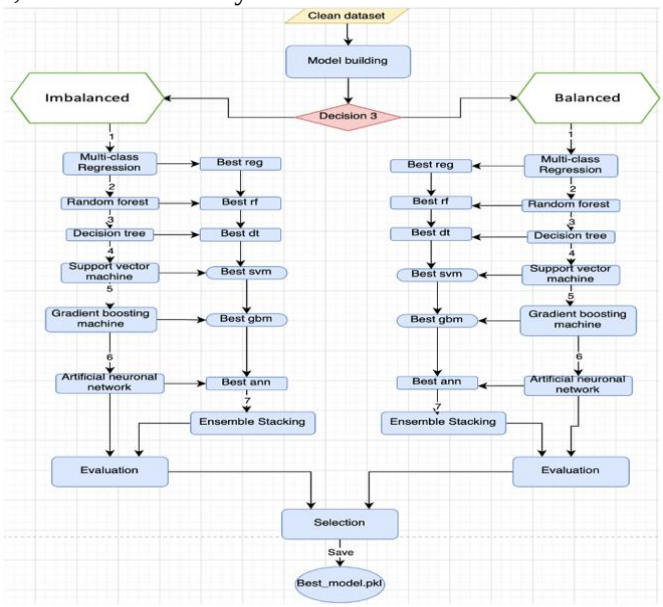


**Figure 2.** Machine Learning Building, Evaluation and Selection Process Flow

Multiclass logistic regression: Through our regression pipeline, we perform hyperparameter tuning for a logistic regression classifier using grid search cross-validation, where parameter "C" is equal to a list of options for the regularization strength, with the values (0.1, 1, 10, 100) representing how strongly our model tries to avoid fitting to noise by penalizing large coefficients. Smaller values specify stronger regularization and parameter "solver", a list of algorithms that the logistic regression model uses for optimization with a list of values equal to newton-cg, lbfgs, liblinear, sag, and saga. We first define a parameter grid containing values for regularization strength and solver algorithms. Then, we initialize a logistic regression classifier with specified parameters. Next, grid search cross validation was initialized with the classifier, parameter grid, and cross-validation settings. It uses cross-validation (cv=5) to assess the performance of each parameter combination, ensuring that the chosen parameters generalize well to unseen data, Verbose = 1 means that the search process will print out progress messages, and n_jobs = -1 allows the process to use all available CPU cores for faster completion. A grid search was then conducted on the training data to find the best combination of hyperparameters (Figure 3).

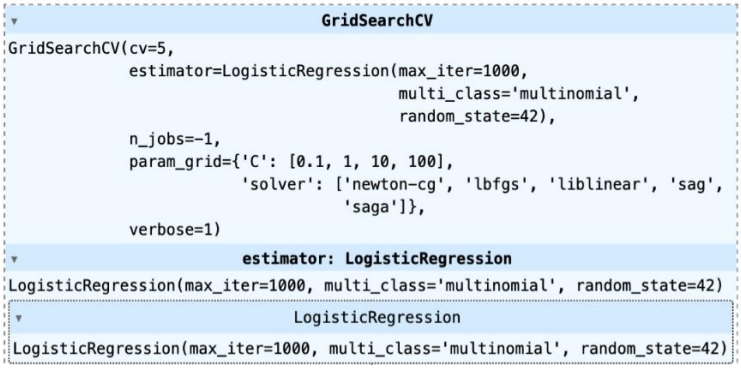


**Figure 3.** Regression Pipeline Flow

**Random Forest:** Through this pipeline, we conducted hyperparameter tuning for a random forest classifier via grid search cross-validation. Initially, a random forest classifier was instantiated with a specified random state. Then, a parameter grid was defined, encompassing values for key hyperparameters such as the number of trees in the forest (n_estimators), maximum depth of trees (max_depth), minimum samples required to split an internal node (min_samples_split), minimum samples required to be at a leaf node (min_samples_leaf), and whether bootstrap samples are used during tree construction (bootstrap). Subsequently, grid search CV was utilized to explore various combinations of these hyperparameters, utilizing a 3-fold cross-validation scheme and assessing performance based on the F1 macro-score (Figure 4).

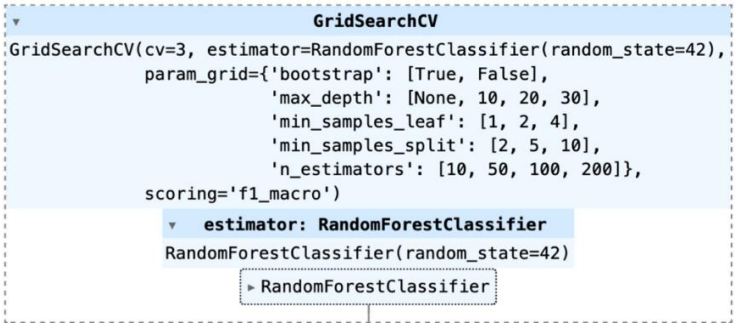


**Figure 4.** RF Pipeline Flow

**Decision Tree:** For this pipeline, we conducted hyperparameter tuning for a decision tree classifier through grid search cross-validation. Initially, a decision tree classifier was instantiated with a specified random state. Then, a parameter grid was defined, including key hyperparameters such as the maximum depth of the tree (max_depth), minimum samples required to split an internal node (min_samples_split), minimum samples required to be at a leaf node (min_samples_leaf), and the criterion for quality measurement of a split (criterion). Subsequently, Grid Search CV was employed to explore various combinations of these hyperparameters, utilizing a 3-fold cross-validation scheme and assessing performance based on the F1 macro score. The grid search is fitted to the prepared training data, and the best-performing decision tree model is extracted from the grid search results and stored in best_dt for subsequent utilization (Figure 5).

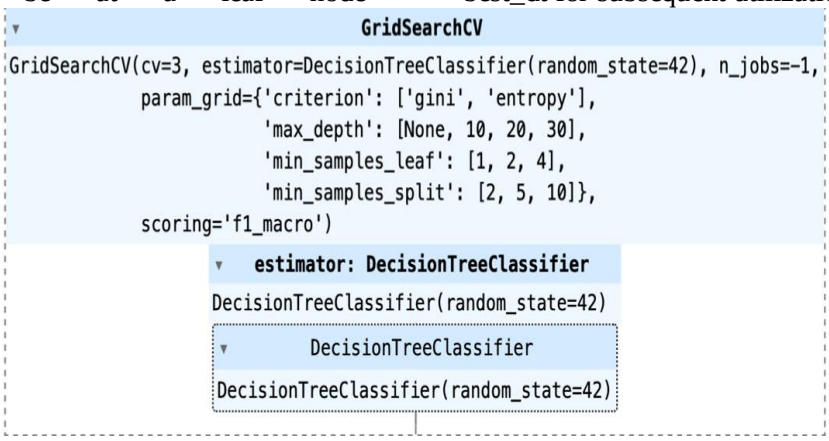


**Figure 5.** Decision Tree Pipeline Flow

**Support Vector Machine:** This pipeline conducts hyperparameter tuning for a support vector machine (SVM) classifier using grid search cross-validation. Initially, a parameter grid was defined, encompassing values for key hyperparameters such as the regularization parameter (C), kernel coefficient for 'rbf', 'poly', and 'sigmoid' kernels (gamma), and the kernel type to be used in the algorithm (kernel). Then, an SVM model is initialized with the specified parameters, including enabling probability estimates and setting a random state. Grid search CV is utilized to explore various combinations of these hyperparameters, employing a 5-fold cross-validation scheme and parallel processing for efficiency. The grid search was fitted to the prepared training data, and the best-performing SVM model was extracted from the grid search results and stored in best_svm for subsequent use (Figure 6).

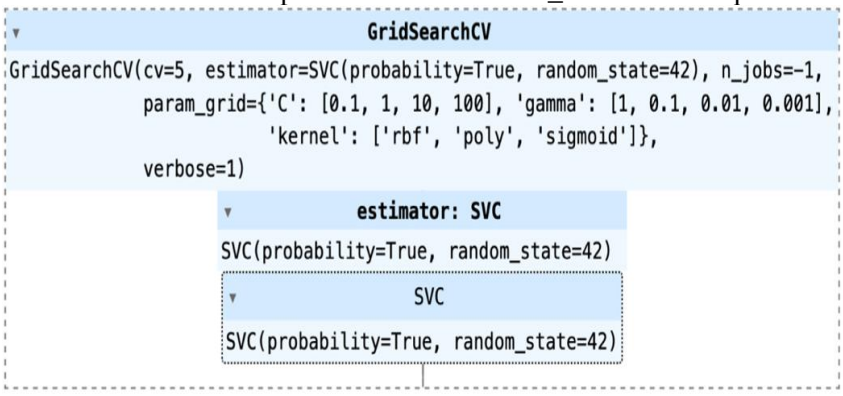


**Figure 6.** Support Vector Pipeline Flow

**Gradient Boosting Machine:** This pipeline performs hyperparameter tuning for a gradient boosting classifier via grid search cross-validation. Initially, a parameter grid was defined, which includes values for key hyperparameters such as the number of boosting stages (n_estimators), the learning rate that controls the contribution of each tree (learning_rate), and the maximum depth of the individual regression estimators (max_depth). Then, a gradient boosting classifier was initialized with the specified parameters, including setting a random state. GridSearchCV is utilized to explore various combinations of these hyperparameters, employing a 5-fold cross-validation scheme and parallel processing for efficiency. The grid search was fitted to the prepared training data, and the best-performing gradient boosting model was extracted from the grid search results and stored in best_gbm for subsequent use (Figure 7).

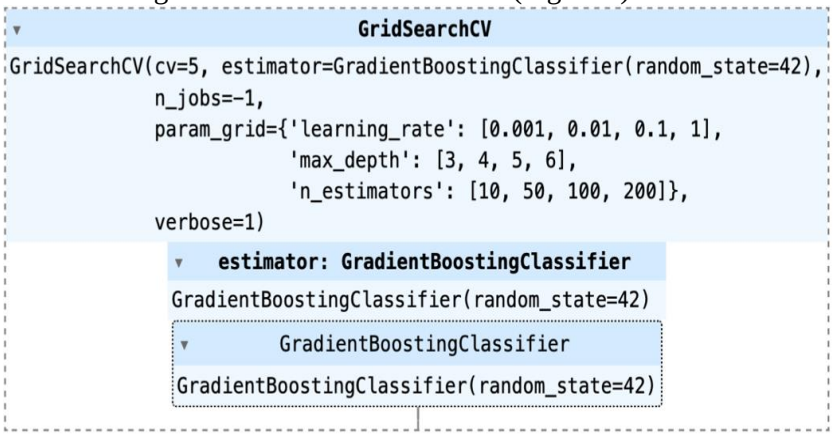


**Figure 7.** Gradient Boosting Pipeline Flow

**Artificial Neuronal Network:** This pipeline conducts hyperparameter tuning for a multilayer perceptron (MLP) classifier using grid search cross-validation. Initially, a parameter grid was defined, encompassing values for key hyperparameters such as the size of the hidden layers (hidden_layer_sizes), activation functions for the hidden layers (activation), and the initial learning rate (learning_rate_init). Then, an MLP classifier was initialized with specified parameters, including setting a random state and increasing the maximum number of iterations for better convergence. Grid search CV was employed to explore various combinations of these hyperparameters, utilizing a 5-fold cross-validation scheme and parallel processing for efficiency. The grid search was fitted to the prepared training data, and the best-performing MLP model was extracted from the grid search results and stored in best_NN for subsequent utilization (Figure 8).

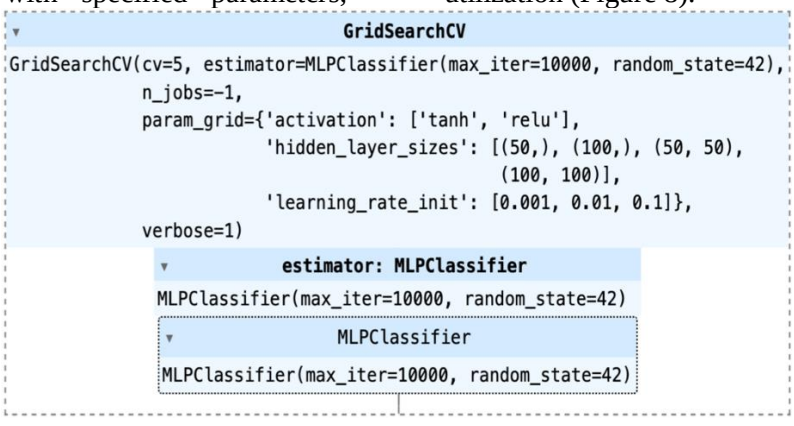


**Figure 8.** Artificial Neural Network pipeline Flow

**Stacking Model:** This pipeline implements a stacking ensemble classifier, which combines predictions from multiple base models using a meta-learner. Initially, a list of base models was retrieved from the previous models, and the corresponding best-performing model was obtained from previous hyperparameter tuning. Then, a meta-learner, in this case, a logistic regression classifier, was defined. Subsequently, a stacking classifier was instantiated with the list of best base models, the meta-learner, and additional parameters such as the number of folds for cross-validation (cv=5) and the method used for stacking (stack_method='auto'). The stacking classifier was fitted to the prepared training data, combining predictions from the best base models and training the meta-learner on these predictions. Finally, the best estimator was extracted from the stacking classifier, representing the entire stacked ensemble model (Figure 9).

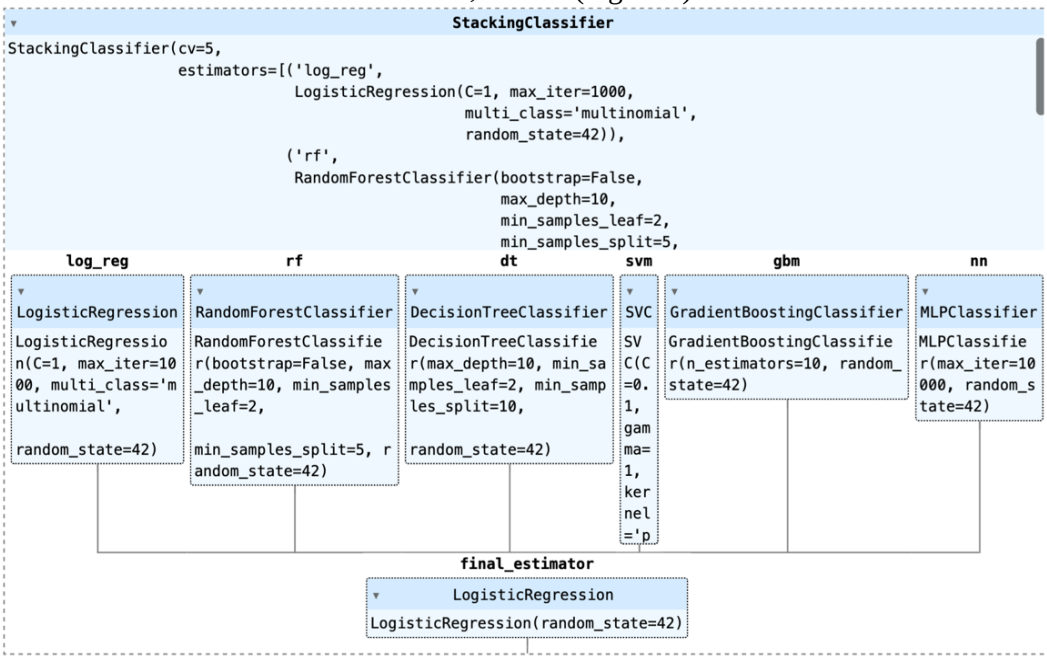


**Figure 9.** Stacking pipeline flow

All these steps were repeated with a balanced approach by setting the "class_weight" parameter to 'balanced' (Figure 10). This automatically adjusts the weights inversely proportional to the class frequencies. The two approaches, balanced and unbalanced, were evaluated and compared with each other to extract the best model.

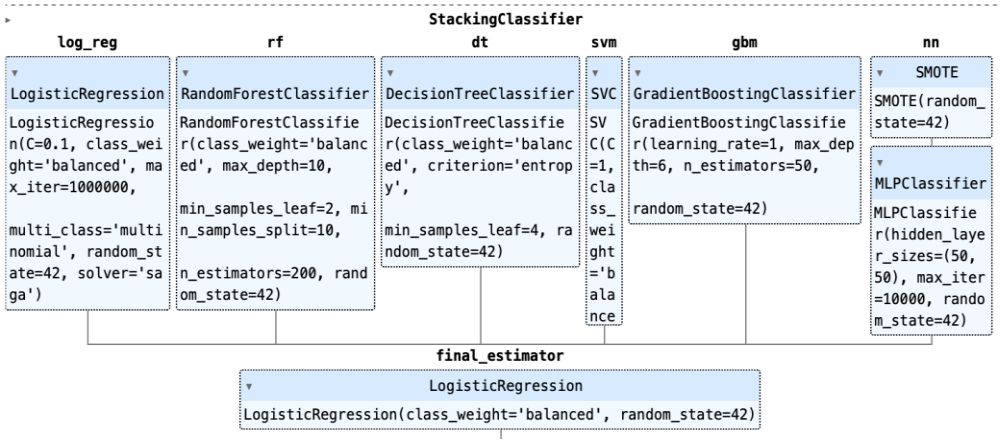


**Figure 10.** Balanced Stacking Pipeline Flow

## Model Evaluation and Selection

Model performance was evaluated using the F1-score, a harmonic mean of precision and recall, providing a balance between the two metrics. This choice is particularly relevant for imbalanced datasets or when the cost of false positives and false negatives is high.

1. **F1-Score Calculation:** The F1-score was calculated for each model to assess its performance following this formula:
    - F1-score = 2 * (precision * recall) / (precision + recall).
2. **Comparison and Selection:** The models' F1-scores were compared, and the model with the highest F1-score was selected as the best performing model.

## Model Deployment

To ensure that we have correctly collected the needs of our future end users, we have adopted the formalism of the Unify Modeling Language (UML)[1] standards to model the static and dynamic views of our future application.

This use case diagram depicts the interactions between users (actors) and a system to predict a given region (Figure 11):

1. The actor here is any authorized user from the NMCP who has access to the system.
2. The use case is “Predict Malaria incidence”.
3. The relationships show how actors interact with these use cases, such as "Users", which are associated with "Predict Malaria incidence" and "Report sharing".
4. The relationships show that "predict" includes the functionality of "manually inputting data".
5. Extended relationships show that "predicts" can be extended to include "shared reports" under certain conditions.

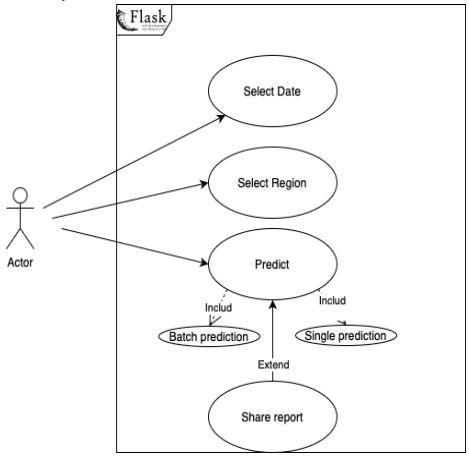


**Figure 11.** Prediction Use Case

Figure 12 shows the interactions between DHIS2 and our Flask application during the prediction process. It shows the sequence of events starting from when the user submits a prediction request through the Flask web interface, triggering a request to the Flask server. The server then interacts with the prediction algorithm to process the input data and generate a prediction. This sequence is represented by messages exchanged between the client (user interface) and the server, as well as between the server and the prediction algorithm. Additional interactions include data validation, error handling during processing, and response delivery back to the user interface. The sequence diagram provides a detailed view of how different parts of the application collaborate to perform the prediction task, aiding in understanding the system's behavior and potential optimizations.

[1] https://en.wikipedia.org/wiki/Unified_Modeling_Language#:~:text=The%20unified%20modeling%20language%20(UML,the%20design%20of%20a%20system.

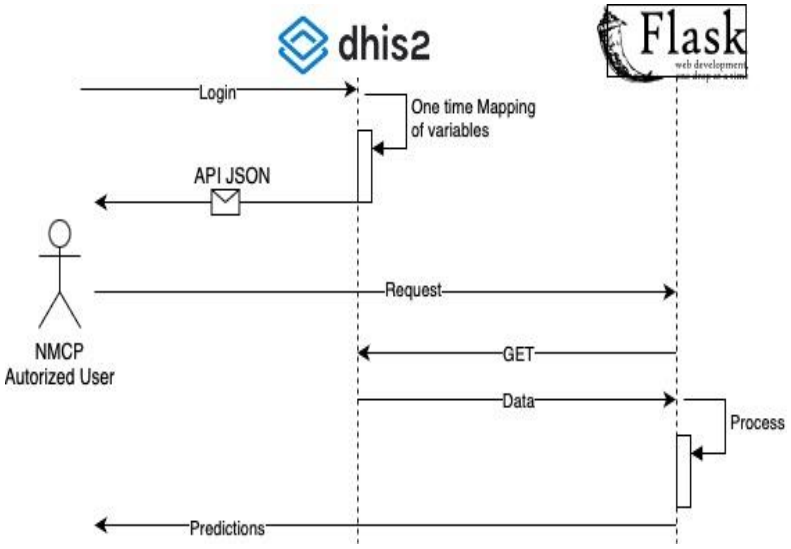


**Figure 12.** Application Sequence Diagram

This state-transition diagram (Figure 13) includes "user logging," "data processing," and "prediction display," representing different stages of our tool's functionality. Transitions depict how the tool moves between these states, triggered by events such as user submission of request of data, processing, and display of the malaria prediction result. The initial state signifies the starting point when the user accesses the tool, while the final state represents the conclusion of the prediction process. Internal transitions occur during data processing, indicating intermediate steps within a single state. Overall, this state diagram offers a clear visualization of the tool's behavior and flow, aiding in the understanding and development of malaria prediction applications.

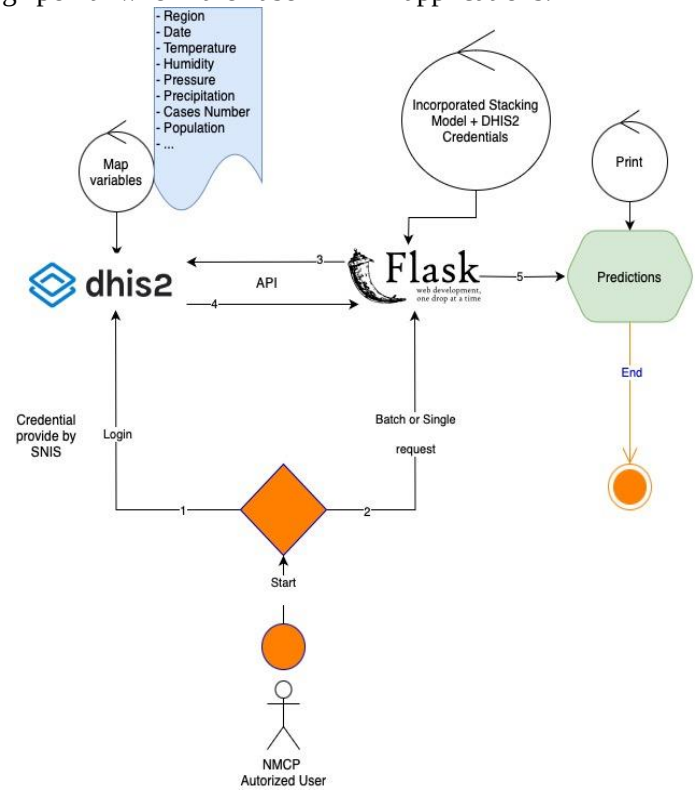


**Figure 13.** Application state-transition diagram

# Results

## Descriptive Analysis

Overall, we find an increasing trend in the mean incidence across most regions from 2018 to 2023. This is evident from the higher mean incidence values in later years compared to earlier years for many regions (Figure 14).

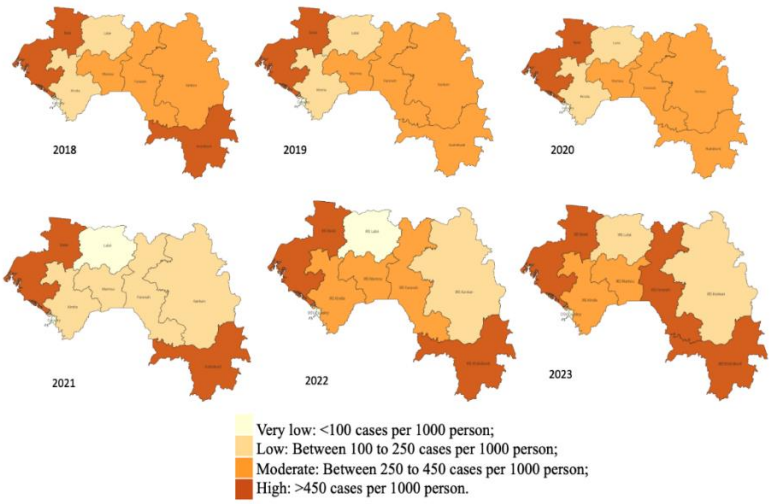


**Figure 14.** Annual Incidence per 1000 Population by Region between 2018 and 2023 in Guinea2

## Variance Inflation Factor

The variance inflation factor (VIF) measures how much the variance of an estimated regression coefficient increases if the predictors are correlated. The table below (Table 1) summarizes our main findings.

**Table 1.** Variance Inflation Factor

| **Features** | **VIF** |
|---|---|
| constant | 14.55 |
| Temperature | 2.67 |
| Humidity | 2.02 |
| Pressure | 2.24 |
| Precipitation | 1.66 |
| Incidence | 1.11 |
| Number of mosquito beds nets | 8.34 |
| Well-being index | 11.22 |

The constant term had a high VIF (14.55), suggesting multicollinearity issues. The temperature, humidity, pressure, precipitation and incidence have relatively low VIF values, indicating low multicollinearity among them. The VIF for "number of mosquito bed nets" was 8.34, which indicates moderate multicollinearity. Although not extremely high, this suggests that the variance of its regression coefficient may be somewhat inflated due to correlations with other predictors.

The VIF for the "well-being index" is 11.22, which is higher and suggests stronger multicollinearity compared to the other features. This indicates that the variance of its regression coefficient is significantly inflated due to correlations with other predictors.

[2] https://portail.sante.gov.gn/base-connaissances/snis-section/

## Correlation Matrix

These results (Table 2) provide insight into the relationships between all our variables. We focus her on the correlation between incidence and all features. The incidence had weak positive correlations with humidity (0.30) (Figure 15B), pressure (0.20) (Figure 15D), and precipitation (0.14) (Figure 15C), indicating a slight tendency for the incidence to increase as humidity, pressure and precipitation increase. There is a very weak negative correlation with temperature (-0.24) (Figure 15A), suggesting a slight tendency for the incidence to decrease as temperature increases.

**Table 2.** Correlation Matrix

| | **Temperature** | **Humidity** | **Pressure** | **Precipitation** | **Incidence** |
|---|---|---|---|---|---|
| Temperature | 1.000000 | -0.610601 | -0.733448 | -0.541672 | -0.238186 |
| Humidity | | 1.000000 | 0.546362 | 0.585471 | 0.304338 |
| Pressure | | | 1.000000 | 0.425031 | 0.204627 |
| Precipitation | | | | 1.000000 | 0.141541 |
| Incidence | | | | | 1.000000 |

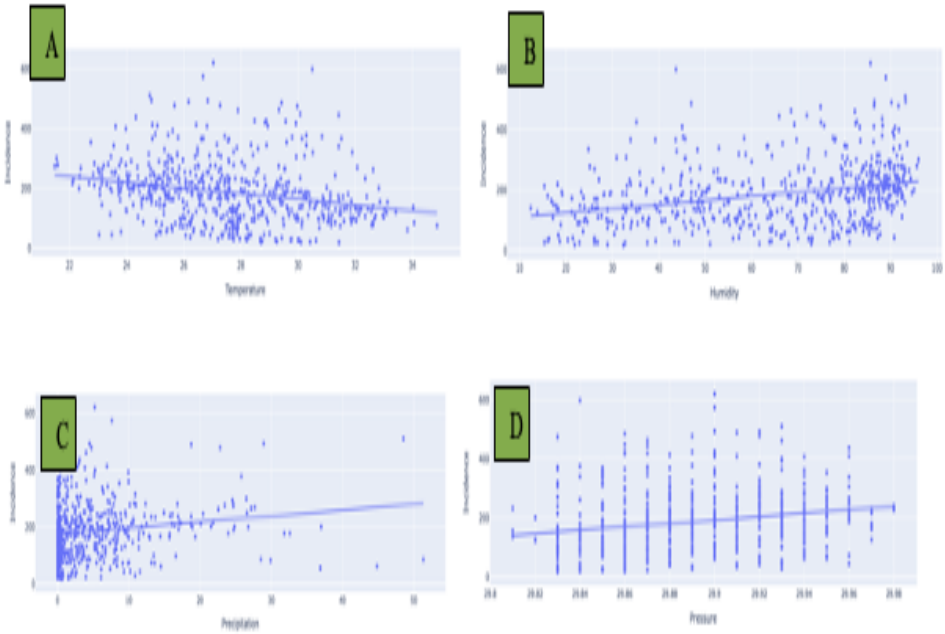


**Figure 15.** Scatterplots of Incidence vs Feature

Our final dataset consists of 576 records for temperature (°C), humidity (%), pressure (kph), precipitation (mm), and incidence (expressed per 1000 people). The mean values indicate an average temperature of 27.52°C, humidity of 64.86%, pressure of 29.89 kph, precipitation of 4.70 mm, and an incidence of 185.43 per 1000 people. The standard deviations reveal the variability within the dataset, with the temperature showing a deviation of 2.63°C, humidity at 22.95%, pressure with a minimal deviation of 0.04 kph, precipitation exhibiting a deviation of 7.17 mm, and incidence displaying a deviation of 104.26 per 1000 people (Figure 16).

| Measure | Numbers of records | Mean | Standard Deviation |
|---|---|---|---|
| Temperature (°C) | 576 | 27.52 | 2.63 |
| Humidity (%) | 576 | 64.86 | 22.95 |
| Pressure (kph) | 576 | 29.89 | 0.04 |
| Precipitation (mm) | 576 | 4.7 | 7.17 |
| Incidence (per 1000 population) | 576 | 185.43 | 104.26 |

**Figure 16.** Descriptive Statistics

Following the stratification of our incidence into four categories (very low, low, medium and high), Figure 17 provides an overview of the frequency of malaria incidence levels within the dataset. This suggests that the majority of instances have either "Low" (323 instances) or "Very Low" (129 instances) malaria incidence, with fewer instances classified as "Moderate" (111 instances) and even fewer instances classified as "High" (13 instances) malaria incidence.

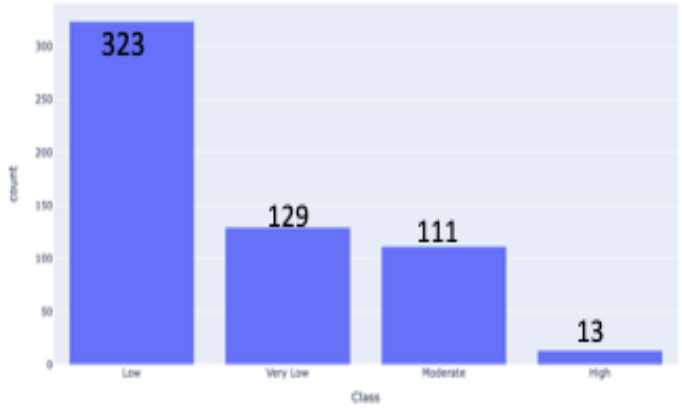


**Figure 17.** Counts of Cases in each Class

## Machine Learning Models

The best estimator, representing the logistic regression model with optimal hyperparameters, was the one with a strength C = 1. The best-performing random forest was the one with bootstrap = False, max_depth = 10, min_samples_leaf = 2, and min_samples_split=5. For the decision tree, our best parameters were max_depth=10, min_samples_leaf=2, and min_samples_split=10. For gradient boosting, we find that n_estimator = 10 is the best parameter. With a support vector machine, we found that the best parameters were C = 0.1, gamma = 1, kernel = 'poly', and probability = True. The model with the default parameter was found to be the best among all the neuronal networks.

Among our models (Table 3 and Figure 18), the balanced stacking model achieved the highest F1-score (0.74), indicating superior overall performance in our classification tasks. The balanced random forest and balanced support vector machine models also performed very well, with F1-scores close to those of the balanced stacking models (0.63 and 0.61, respectively). Although not as high as the balanced stacking model, the unbalanced stacking model still performed well, indicating the effectiveness of the ensemble methods (0.58). The balanced decision tree model performed slightly better than its unbalanced counterpart, indicating that balancing the dataset improved its performance (0.56).

Unbalanced models generally fall below their balanced counterparts. However, some unbalanced models, such as decision tree (dt), random forest (rf), and support vector machine (SVM), still achieved moderate F1-scores. The balanced neural network (0.47), balanced gradient boosting machine (0.44), and balanced logistic regression (0.43) achieved moderate F1-scores, indicating reasonable performance but not as high as that of the top performers. The neural network (0.37), gradient boosting machine (0.34), and logistic regression (0.23) models achieved lower F1-scores, suggesting that they may require further tuning or may not be suitable for this particular dataset.

**Table 3.** Model comparison

| **Models** | **F1-score** |
|---|---|
| balanced_stacking | 0.74 |
| balanced_rf | 0.63 |
| balanced_svm | 0.61 |
| stacking | 0.58 |
| balanced_dt | 0.56 |
| dt | 0.54 |
| balanced_NN | 0.47 |
| balanced_gbm | 0.44 |
| rf | 0.43 |
| balanced_log_reg | 0.43 |
| svm | 0.41 |
| NN | 0.37 |
| gbm | 0.34 |
| log_reg | 0.23 |

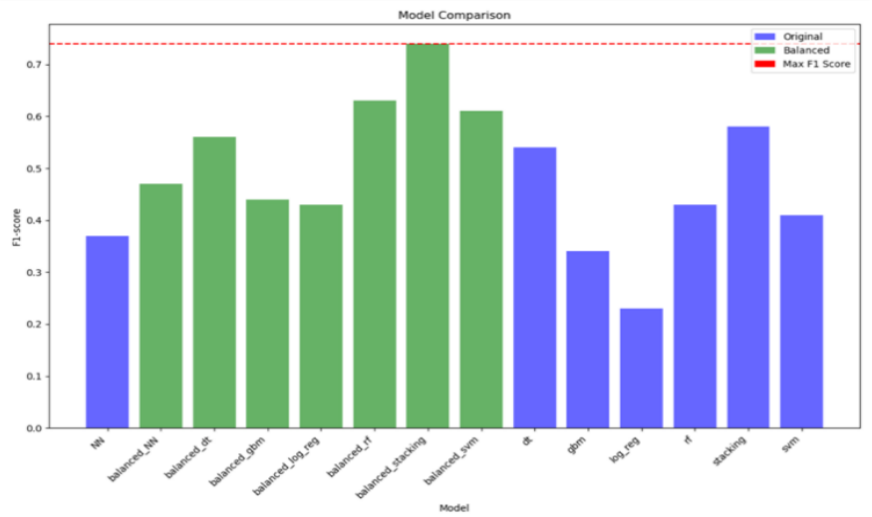


**Figure 18.** Model Comparison

## Web Application

The web application developed (Figure 19, 20 and 21) in this study serves as a user-friendly tool for predicting malaria incidence in Guinea. Building upon the insights gained from machine learning models trained on diverse datasets, including clinical, demographic, and climatic data, the application provides a platform for public health authorities to access predictive insights and make informed decisions.

The application interface is intuitive and easy to navigate and is designed to accommodate users with varying levels of technical expertise. Users can input relevant parameters such as temperature, humidity, pressure and precipitation. The application utilizes advanced algorithms to process input data and generate accurate forecasts, helping stakeholders identify high-risk areas and allocate resources effectively. Additionally, the application offers visualization capabilities, allowing users to explore Guinea maps and interact to select specific regions.

Overall, the web application represents a valuable tool for enhancing malaria surveillance and control efforts in Guinea. By democratizing access to predictive analytics, the application empowers public health authorities to proactively address malaria transmission and improve health outcomes for communities across the country.

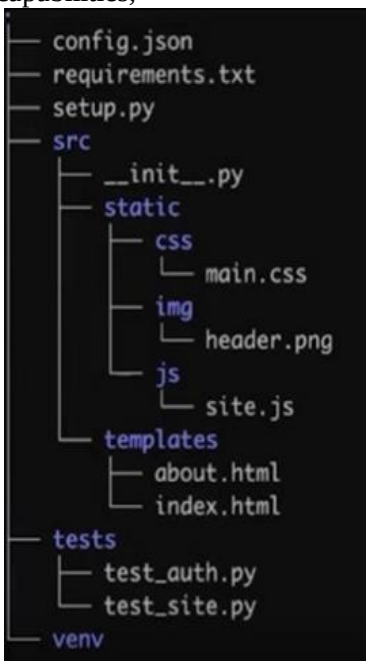


**Figure 19.** Flask Application Structure

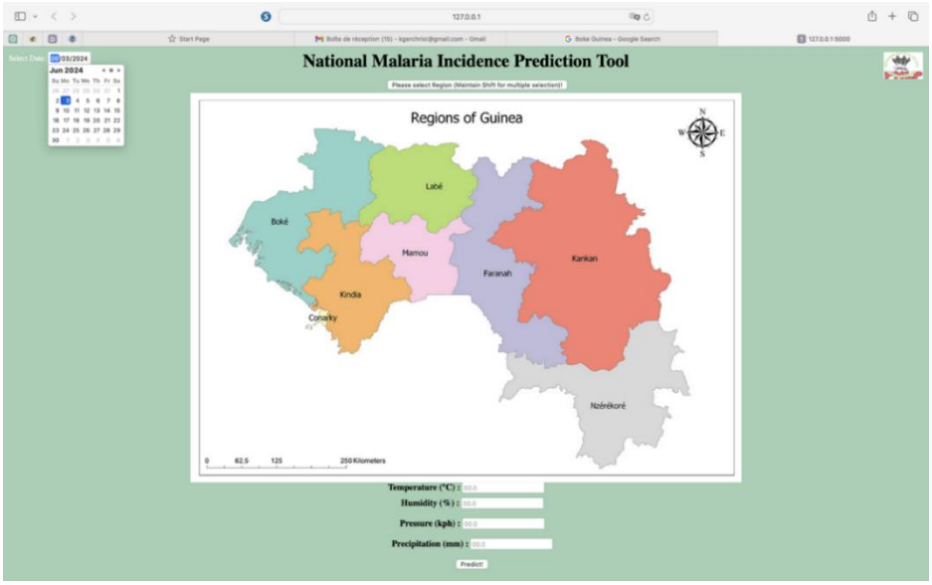


**Figure 20.** Application Interface

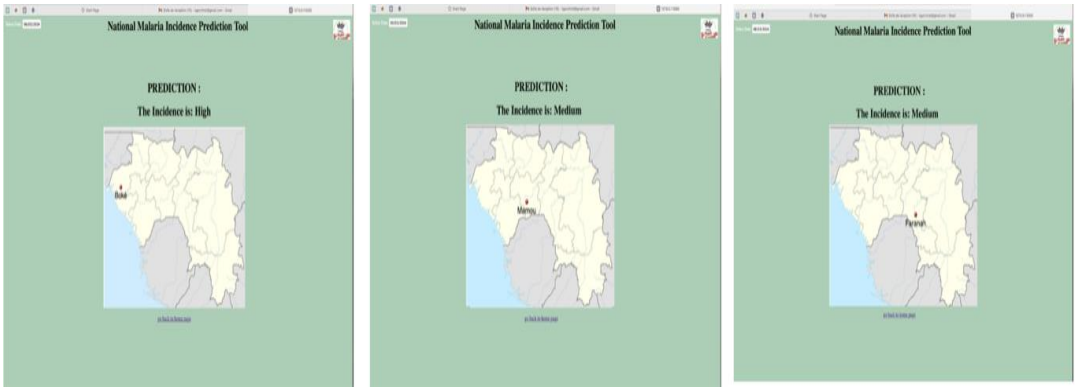


**Figure 21.** Prediction Outputs for the Boke, Mamou and Faranah Regions

## Discussion

The present internship contributes to understanding the dynamics of malaria in Guinea and demonstrates the potential of machine learning in predicting disease incidence. By integrating diverse data sources, including clinical, demographic, and climatic data, this study aimed to predict malaria incidence at the national level. The results indicate promising avenues for leveraging machine learning techniques to improve malaria control strategies.

In machine learning, especially in classification tasks, the distribution of classes in the dataset might be imbalanced, meaning that some classes have significantly more instances than others. This class imbalance can lead to biased models that favor the majority class, often resulting in poor performance for the minority classes. The scikit-learn[3] "balanced" mode automatically adjusts the weights to be inversely proportional to the class frequencies in the input data. It internally calculates the class weights based on the class distribution in the training data and assigns higher weights to minority classes and lower weights to majority classes. During the training of the model, these class weights are incorporated into the algorithm's objective function (such as the loss function). This means that errors on the minority classes are penalized more heavily during training, effectively making the model more sensitive to minority class instances. By adjusting the class weights, the model is encouraged to pay more attention to minority classes, potentially improving its ability to correctly classify these instances. This approach is particularly useful when the dataset is highly imbalanced.

One key finding of the study is the identification of factors correlated with malaria incidence. Through exploratory data analysis, the study revealed relationships between climatic variables such as temperature, humidity, pressure and precipitation. These insights underscore the complex interplay between environmental, demographic, and health-related factors in malaria transmission, highlighting the importance of multifaceted interventions in malaria control efforts.

Our study employed various machine learning algorithms, including logistic regression, random forest, decision trees, support vector machines, gradient boosting machines, neural networks, and stacking models, to predict malaria incidence. Among these, the balanced stacking model emerged as the top performer, achieving the highest F1-score (0.74). This underscores the effectiveness of ensemble methods in capturing the complex patterns inherent in malaria transmission dynamics. However, it is worth noting that model performance varies across different algorithms, indicating the importance of selecting appropriate techniques based on dataset characteristics and the problem domain.

[3] https://scikit-learn.org/stable/

Furthermore, this study addresses practical considerations in deploying predictive models for malaria incidence prediction. By developing a user-friendly web application, our internship aims to facilitate the utilization of predictive insights by public health authorities. This underscores the importance of translating research findings into actionable tools that can inform decision-making and resource allocation in real-world settings.

In conclusion, this study highlights the potential of machine learning in malaria epidemiology and underscores the importance of interdisciplinary collaborations among researchers, healthcare practitioners, and policymakers in addressing public health challenges. By harnessing the power of data-driven approaches, we can continue to advance our understanding of malaria dynamics and improve intervention strategies for malaria elimination.

## Conclusion

In conclusion, our internship provides valuable insights into the epidemiology of malaria in Guinea and demonstrates the potential of machine learning in predicting and understanding disease dynamics. Despite progress in reducing malaria cases, the disease remains a significant public health concern, with substantial morbidity and mortality rates.

The integration of diverse data sources and the application of advanced machine learning algorithms have enabled the development of predictive models for malaria incidence, with ensemble methods showing promising results. However, challenges such as multicollinearity and imbalanced datasets underscore the need for careful data preprocessing and model tuning to ensure robust and accurate predictions.

Moving forward, the findings from this internship can inform targeted interventions and resource allocation strategies to reduce the burden of malaria in Guinea. By leveraging predictive models and deploying user-friendly tools, public health authorities can improve decision-making and allocate resources more efficiently to areas at highest risk of malaria transmission.

Moreover, ongoing collaboration between Guinean researchers, healthcare practitioners, and policymakers is essential for advancing malaria research and implementing evidence-based interventions. By harnessing the power of data and machine learning, we can continue to make progress toward malaria elimination and improve health outcomes for communities in Guinea and beyond.

## Conflict of Interest

The author declares that there is no conflict of interest.

## References


[1]. President's Malaria Initiative (PMI). 2023, *Guinea Malaria Operational Plan FY 2023*. https://d1u4sg1s9ptc4z.cloudfront.net/uploads/2023/01/FY-2023-Guinea-MOP.pdf

[2]. World Health Organization (WHO). 2020, *Malaria in the African Region*. https://www.afro.who.int/health-topics/malaria

[3]. Hamilton, A. J., Strauss, A. T., Martinez, D. A., et al., 2021, Machine learning and artificial intelligence: Applications in healthcare epidemiology. *Antimicrobial Stewardship & Healthcare Epidemiology, 1*(1), e28. https://doi.org/10.1017/ash.2021.191

[4]. Harvey, D., Valkenburg, W., & Amara, A., 2021, Predicting malaria epidemics in Burkina Faso with machine learning. *PLOS ONE, 16*(6), e0253302. https://doi.org/10.1371/journal.pone.0253302

[5]. Ji, C., Zou, X., Hu, Y., et al., 2019, XG-SF: An XGBoost classifier based on shapelet features for time series classification. *Procedia Computer Science, 147*, 24–28. https://doi.org/10.1016/j.procs.2019.01.087

[6]. Huang, J., & Ling, C. X., 2005, Using AUC and accuracy in evaluating learning algorithms. *IEEE*

*Transactions on Knowledge and Data Engineering, 17*(3), 299–310. https://doi.org/10.1109/TKDE.2005.50

[7]. Kalipe, G., Gautham, V., & Behera, R. K., 2018, Predicting malarial outbreak using machine learning and deep learning approach: A review and analysis. In *2018 International Conference on Information Technology (ICIT)* (pp. 33–38). IEEE. https://doi.org/10.1109/ICIT.2018.00017

[8]. Yaa, E. A., Quaye, I. K., Osei, P. P., et al., 2021, Malaria prediction model using machine learning algorithms. *Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12*(11), 7488–7496.

[9]. Nkiruka, O., Prasad, R., & Clement, O., 2021, Prediction of malaria incidence using climate variability and machine learning. *Informatics in Medicine Unlocked, 22*, 100508. https://doi.org/10.1016/j.imu.2020.100508

[10]. Kim, Y., Ratnam, J. V., Doi, T., et al., 2019, Malaria predictions based on seasonal climate forecasts in South Africa: A time series distributed lag nonlinear model. *Scientific Reports, 9*, 17882. https://doi.org/10.1038/s41598-019-54250-3

[11]. Higuchi, D., 2014, Characteristics of coping strategies for dysesthesia in preoperative patients with compressive cervical myelopathy. *Asian Spine Journal, 8*(4), 393–400. https://doi.org/10.4184/asj.2014.8.4.393

[12]. Castro, M. C., 2017, Malaria transmission and prospects for malaria eradication: The role of the environment. *Cold Spring Harbor Perspectives in Medicine, 7*(9), a025601. https://doi.org/10.1101/cshperspect.a025601

[13]. El-Hasnony, I. M., Elzeki, O. M., Alshehri, A., et al., 2022, Multi-label active learning-based machine learning model for heart disease prediction. *Sensors, 22*(4), 1184. https://doi.org/10.3390/s22031184

[14]. World Health Organization (WHO). 2017, *A framework for malaria elimination*. https://iris.who.int/handle/10665/254761

[15]. Weiss, D. J., Lucas, T. C. D., Nguyen, M., et al., 2019, Mapping the global prevalence, incidence, and mortality of Plasmodium falciparum, 2000–17: A spatial and temporal modelling study. *The Lancet, 394*(10195), 322–331. https://doi.org/10.1016/S0140-6736(19)31097-9

[16]. Garske, T., Ferguson, N. M., & Ghani, A. C., 2013, Estimating air temperature and its influence on malaria transmission across Africa. *PLoS ONE, 8*(2), e56487. https://doi.org/10.1371/journal.pone.0056487

[17]. Bhatt, S., Weiss, D. J., Cameron, E., et al., 2015, The effect of malaria control on Plasmodium falciparum in Africa between 2000 and 2015. *Nature, 526*(7572), 207–211. https://doi.org/10.1038/nature15535

[18]. Karuri, M. K., & Snow, R. W., 2016, Forecasting malaria burden in Africa using satellite meteorological data. *Frontiers in Public Health, 4*, 112. https://doi.org/10.3389/fpubh.2016.00112

[19]. Bousema, T., Griffin, J. T., Sauerwein, R. W., et al., 2012, Hitting hotspots: Spatial targeting of malaria for control and elimination. *PLoS Medicine, 9*(1), e1001165. https://doi.org/10.1371/journal.pmed.1001165

[20]. Reiner, R. C., Perkins, T. A., Barker, C. M., et al., 2015, A systematic review of mathematical models of mosquito-borne pathogen transmission: 1970–2010. *Journal of the Royal Society Interface, 12*(106), 20140921. https://doi.org/10.1098/rsif.2014.0921

[21]. Chang, H. H., Davis, G. M., & Waller, L. A., 2014, Mining spatio-temporal data on malaria for exploratory analysis and model building. *International Journal of Health Geographics, 13*(1), 31. https://doi.org/10.1186/1476-072X-13-31

[22]. Sturrock, H. J. W., Hsiang, M. S., Cohen, J. M., et al., 2013, Targeting asymptomatic malaria infections: Active surveillance in control and elimination. *PLoS Medicine, 10*(6), e1001467. https://doi.org/10.1371/journal.pmed.1001467

[23]. Snow, R. W., & Marsh, K., 2002, The consequences of reducing Plasmodium falciparum transmission in Africa. *Advances in Parasitology, 52*, 235–264. https://doi.org/10.1016/S0065-308X(02)52005-X

[24]. Omumbo, J. A., Hay, S. I., Goetz, S. J., et al., 2002, Updating historical maps of malaria transmission intensity in East Africa using remote sensing. *Photogrammetric Engineering & Remote Sensing, 68*(2), 161–166.

[25]. Osei, P., Frempong, G. A., & Nettey, O. E. A., 2020, Spatial analysis of malaria incidence and associated risk factors in Ghana. *Geospatial Health, 15*(1), 13–22. https://doi.org/10.4081/gh.2020.859

[26]. LeCun, Y., Bengio, Y., & Hinton, G., 2015, Deep learning. *Nature, 521*(7553), 436–444. https://doi.org/10.1038/nature14539

[27]. Müller, A. C., & Guido, S., 2016, Introduction to machine learning with Python: A guide for data scientists. *O'Reilly Media.*