

Groundwater Quality Assessment of Raipur City Using Machine Learning Models



Anushree Shrivastava, Mridu Sahu, and D. C. Jhariya

Abstract Groundwater has been a vital source of water consumption across India. Raipur, the capital city of Chhattisgarh, has been utilizing this resource for water consumption and utilization for Irrigation. While water is a necessity for survival, it is also important that water which we uptake is fit for consumption. Groundwater often gets contaminated by the fertilizers and pesticides by affecting the concentration of the major ions and other parameters present in the water. Groundwater Quality Index is a measure to determine the quality of water, which is calculated based on some physicochemical parameters and ions that water contains. The Water Quality Index (WQI) is a really useful measure for assessing the overall water quality. It simplifies the interpretation of information by lowering a huge number of data points to a single value. The WQI is used to assess whether or not groundwater is suitable for drinking. In this paper, the Water Quality Index was calculated based on pH, TA, TH, Chloride, Nitrate, Fluoride, and Calcium. Further, the quality of groundwater was assessed using various Machine Learning Models, namely, Logistic Regression, Decision Tree Classifier, Gaussian NB, Random Forest Classifier, Linear SVC, and XGB Classifier. The best classification was shown by Random Forest Classifier with an outstanding of 100 percent accuracy.

Keywords Logistic regression · Decision tree classifier · Gaussian NB · Random forest classifier · Linear SVC · XGB classifier

A. Shrivastava · M. Sahu (✉) · D. C. Jhariya
Department of Information Technology, National Institute of Technology, Raipur, India
e-mail: mrisahu.it@nitrr.ac.in

A. Shrivastava
e-mail: ashrivastava.mtech2021.it@nitrr.ac.in

D. C. Jhariya
e-mail: dcjhariya.geo@nitrr.ac.in

1 Introduction

Groundwater quality assessment is essential for determining the quality of water in a region as water is not only a basic necessity in every household but also an important resource for survival. Conservation of water has been a need of the hour, as freshwater is getting contaminated due to various factors like the use of fertilizers, pesticides, and industrial wastes. Thus, the study of freshwater resources like groundwater is becoming important.

The study area for this paper is the capital city of Chhattisgarh, i.e., Raipur. In Raipur, the Kharun River is the only supply of raw water currently available. Groundwater is another source of water, having a capacity of 22 million liters per day, in addition to water from the Kharun River.

The quality assessment of water from the groundwater sources is a tedious task as it involves manual work and analysis. The role of Machine Learning can be crucial in automating such tasks, especially in the scenario where lockdown is implemented due to the pandemic. The prediction using machine learning can prevent the ceasing of analysis of such crucial resources.

In this paper, we have used the concept of the Water Quality Index (WQI) for determining the water quality of the 44 different regions of Raipur. The Water Quality Index was calculated based on pH, TA, TH, Chloride, Nitrate, Fluoride, and Calcium. Further, the quality of groundwater was assessed using various Machine Learning Models, namely, Logistic Regression, Decision Tree Classifier, Gaussian NB, Random Forest Classifier, Linear SVC, and XGB Classifier.

2 Methodology

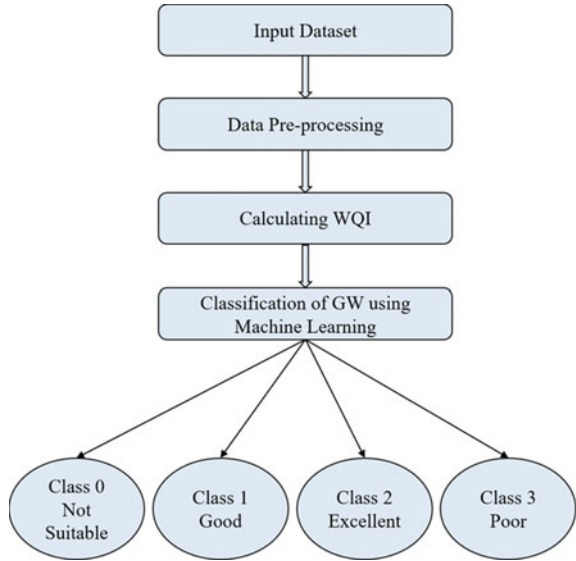
Figure 1 shows the methodology used in this work. The stages include input dataset, data preprocessing, calculating WQI, and classification of groundwater using Machine Learning Models.

2.1 Dataset Description

The dataset has been collected from the Geology Department of NIT, Raipur. It consisted of groundwater details of samples collected from 44 locations within Raipur and 24 features, namely, pH, EC, TDS, TH, TA, HCO_3^- , Cl^- , NO_3^- , SO_4^{2-} , F^- , Ca^{+2} , Mg^{+2} , Na^+ , K^+ , Si^{+4} , SSP, SAR, SAR,

KR, RSC, MR, and CR from which seven features have been used, namely, pH, TA, TH, Cl^- , NO_3^- , F^- , and Ca^{+2} for the calculation of WQI, which further form the basis of classification.

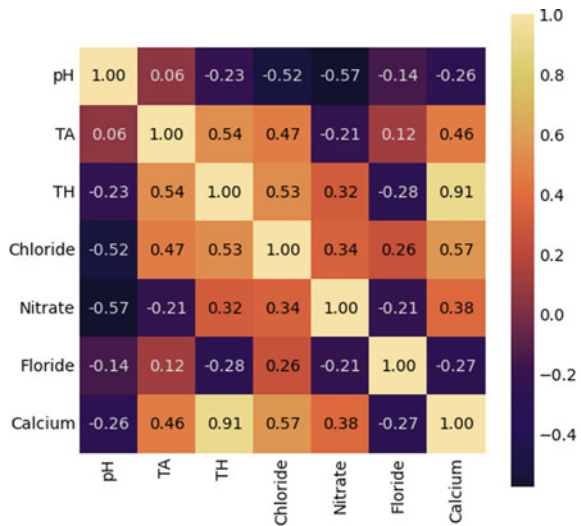
Fig. 1 Methodology used



2.2 Correlation of Features

Figure 2 shows the correlation of the seven parameters used for the calculation of WQI and Water Quality Classification. Correlation shows how the parameters are related to each other and whether the presence or absence of one parameter leads to the presence or absence of another parameter. Correlation is a mechanism used to identify the strength of relationships between features.

Fig. 2 Correlation matrix of the features



Correlation is of three types:

- Positive correlation: If there are two features which are having positive correlation, it means that if the value of one feature is increasing in a scenario, the value of the other feature will also tend to rise in a similar scenario;
- Negative correlation: If there are two features which are having negative correlation, it means that if the value of one feature is increasing in a scenario, the value of the other feature will tend to reduce in a similar scenario;
- No correlation: When two features are said to be in no correlation, it means that the value of the two features is independent, i.e., increasing or decreasing and the value of a particular feature will have no impact on the value of other features.

2.3 Calculation of Water Quality Index

Step 1. Calculation of weightage factor

The weight is assigned for each parameter as per the importance of the parameter in the water consumption. The relative weight is given by:

$$W_i = \Sigma w_i, \tag{1}$$

where W_i denotes relative weight, w_i denotes parameter weight, and n represents the number of parameters. The weight assigned to all the seven parameters and the relative weight is shown in Table 1.

Step 2. Calculating sub-index

For obtaining the Water Quality Index, firstly sub-index is calculated which is given by:

$$SI = W_i * (c/s) * 100, \tag{2}$$

Table 1 Weight, relative, and BIS standard value for each parameter

Parameter	BIS standard	Weightage	Relative weight
pH	6.5–8.5	2	0.133
TA	200–600	1	0.067
TH	200–600	1	0.067
Chloride	250–1000	3	0.133
Nitrate	45	5	0.267
Fluoride	1.0–1.5	4	0.200
Calcium	75–200	1	0.067
Magnesium	30	1	0.067
		Σw_i	ΣW_i

where W_i is the relative weight, c is the value of the parameter in the water sample in mg/l, and s is the standard value of the parameter mentioned in Table 1.

Step 3. Calculating Water Quality Index

$$W_i = \sum SI \tag{3}$$

The Water Quality Index is used to determine the category of water quality, namely, Excellent, Good, Poor, or Not Suitable for consumption. Table 2 shows the different classes of water quality into which the dataset is bifurcated according to the Water Quality Index.

Figure 3 shows the pie chart representation of the occurrence of different categories of water quality. Figure 4 shows the violin plot of the classes of water quality and how they are related to WQI. A violin plot uses density curves to represent numeric data distributions for one or more groups.

Figure 5 shows the distribution and box plot representation of the parameters based on the water quality category. The distribution curves and box plot depict how each feature is contributing to the water quality. In the distribution curve and box plot, green represents excellent water quality, blue represents good water quality, yellow represents poor water quality, and red represents water quality which is not suitable for consumption.

Table 2 Water quality classes

Category	Water quality index	Water quality
0	> 75	Not suitable
1	35–55	Good
2	< 35	Excellent
3	55–75	Poor

Fig. 3 Pie chart representation of the occurrences of different categories of WQ

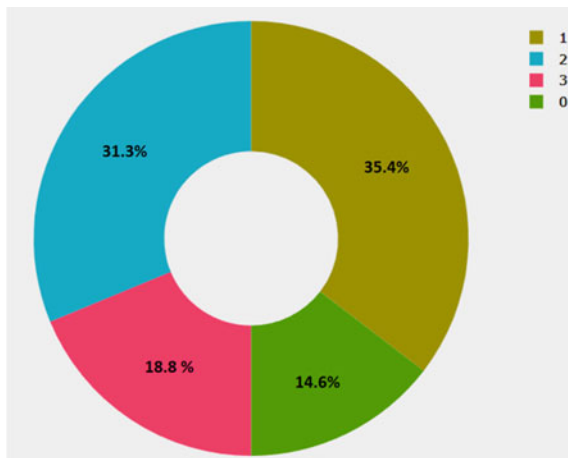
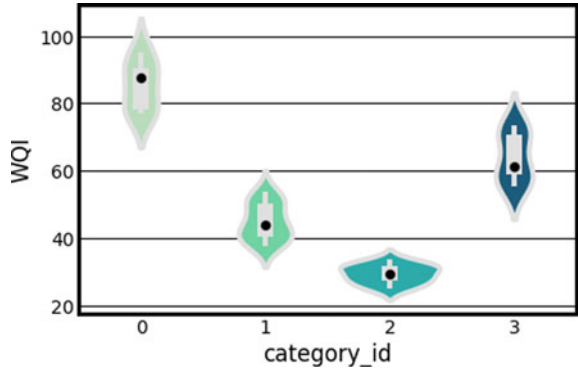


Fig. 4 Violin plot of water quality classes based on WQI



2.4 Machine Learning Models

2.4.1 Logistic Regression

This is a supervised learning-based classification approach. It works with discrete data and produces non-continuous output. However, instead of digital numbers, it outputs probabilistic values between zero and one. The minimum value is the most crucial part of Logistic Regression; it lays the foundation for classification and is used to determine if the outcome is nearer to one or zero. The recall and accuracy settings determine the threshold value. If both recall and accuracy are one, the threshold value is considered to be perfect.

However, this optimum condition does not always occur; thus, there are two possible approaches: one with great precision but low recall and the other with low precision but high recall. The threshold is established [1] based on the system's requirements. The Sigmoid function (see Fig. 6) is an S-shaped curve that is used to calculate the result depending on the threshold value in Logistic Regression. This is accomplished by putting the real values and threshold value onto the curve, then comparing the real values to the threshold value to see if they are closer to 1 or 0. The Logistic Regression model is ideal for binary classes, as can be seen from the preceding description, but it may also be utilized for numerous classes adopting the idea of one versus all [2].

2.4.2 Decision Tree Classifier

A Supervised Machine Learning Approach called a decision tree produces decisions based on a combination of rules. It constructs decision trees using historical data. Construction of the maximum tree, selection of the appropriate tree size, and classification of fresh data using the established tree are the three elements of this technique.

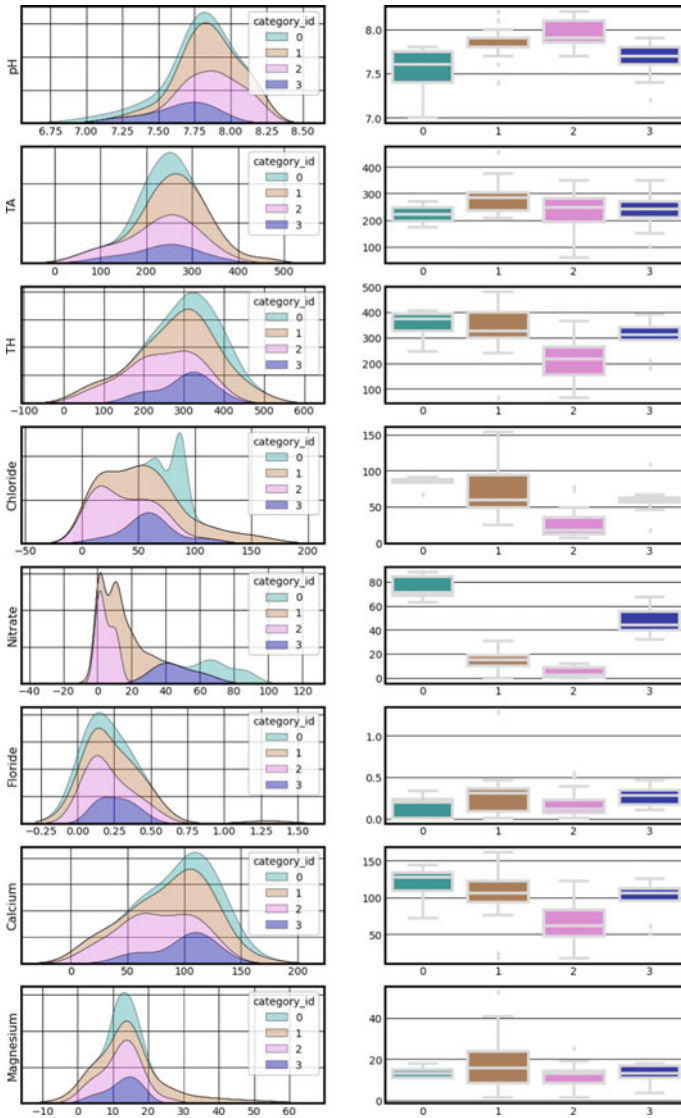


Fig. 5 Density distribution and box plot based on water quality category

Maximum tree creation takes a large amount of time. The observations are separated continually until only one class remains at the lowest possible level. Figure 7 depicts the maximum tree's creation. The figure to the left of the tree's root ought to be smaller than the figure to the right. The values that are referred here are parameters in the learning sample's parameter matrix.

Fig. 6 Logistic regression

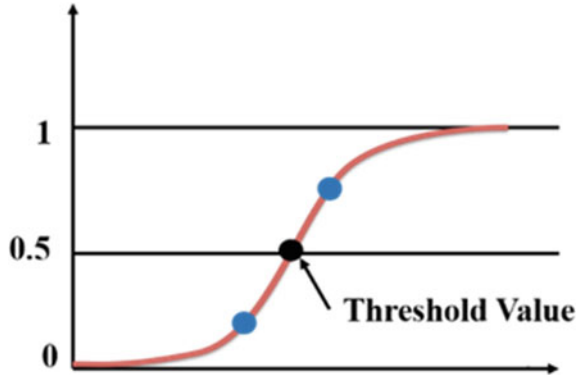
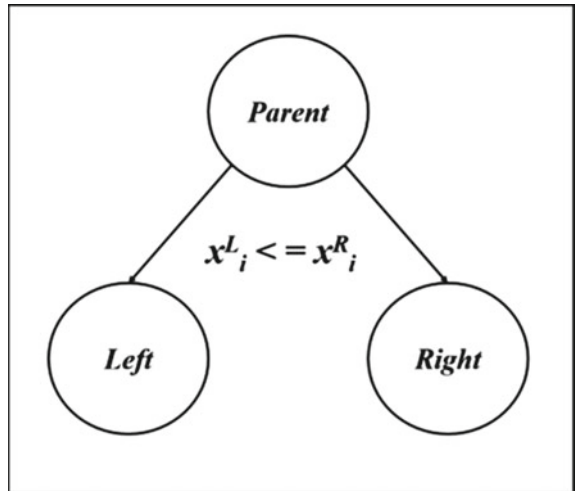


Fig. 7 Construction of maximum tree



The proper size tree must be chosen since the maximum tree might have many layers and be complex; thus, it must be optimized before categorization. This is intended to shrink the tree and remove any extraneous nodes. Finally, classification is completed, with classes allocated to each observation [3].

2.4.3 Gaussian Naïve Bayes

A statistical and supervised learning-based probabilistic classifier is yet another name for GNB. It is based on the probability hypothesis of Bayes. For classification, it employs a probabilistic technique. When used on massive datasets, it produces accurate findings in a shorter amount of time [4].

It has four types of probabilities: likelihood probability, which represents the probability of predictor, such that the class is given; predictor prior probability, which

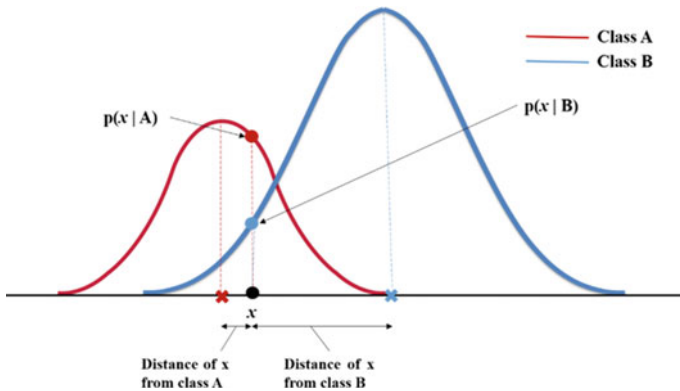


Fig. 8 GNB classifier

represents the probability of the previous predictor; class prior probability, which tells the probability of class; and posterior probability, which represents the probability of class such that the predictor is given. The existence or lack of one characteristic does not impact the existence or lack of another characteristic, according to the Naive Bayes approach. Each characteristic contributes to the categorization process in its own way [5].

We employed GNB, that is a type of NB classifier that is used for continuous data categorization. It is based on probability ideas of Gaussian normal distribution. Classes are believed to be represented by normal distributions with distinct dimensions. It is simple, computationally quick and only takes a modest amount of training data. However, since it presupposes characteristic independence, it may produce inaccurate estimates, thus the label naive. The workings of GNB are shown in Fig. 8. The categorization is based on the shortest distance between each data points (x).

2.4.4 Random Forest Classifier

This approach, as the name implies, uses many trees to arrive at a categorization choice. It is based on the notion of tyranny of the majority. The trees each make their own estimate, and the class with the highest count is chosen. The overall result is successful because the outputs of the multiple decision trees are combined; thus, even if some decision trees produce incorrect predictions, the other trees' outcomes cover up the final estimate [6]. Because it considers different tree options on a majority premise, Random Forest produces superior results. However, when there are a lot of trees, categorization can be stagnant and unproductive. This classifier's training is quick, but predictions are stagnant [7]. Figure 9 depicts the Random Forest Classifier's generic operation.

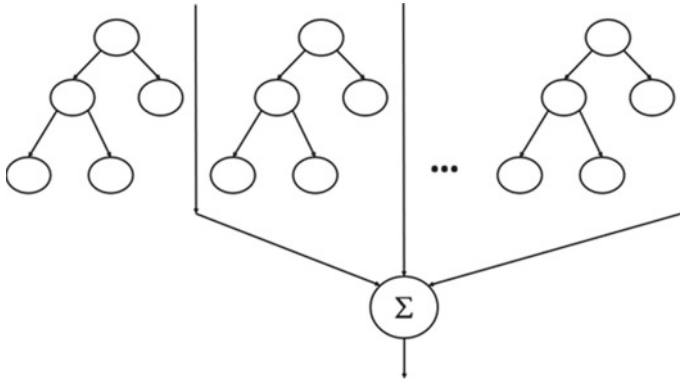
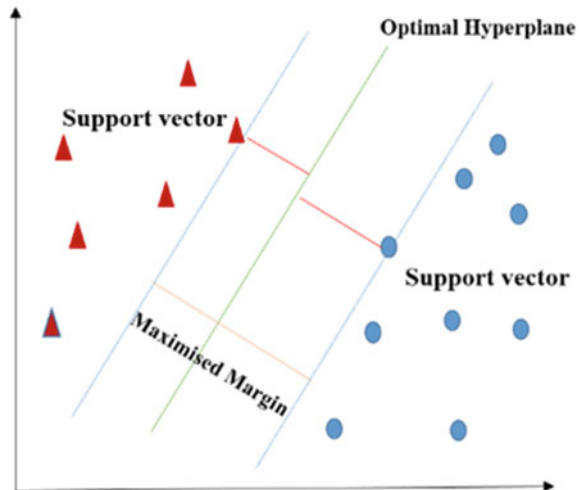


Fig. 9 Random forest classifier

2.4.5 Linear SVC

It is a kernel-based classifier that was designed for linear separation and can divide dataset into two groups. SVM now is utilized to solve a variety of real-world situations [8]. For classification, SVM employs the idea of hyperplane. To partition the data into classes, it creates a line or hyperplane. To achieve maximum gap between the two classes, SVM tries to push the decision boundary much further than possible. It boosts the productivity by using a sub-class of training points called as support vectors. Moreover, because it requires a lot of training time, it performs best with modest datasets. The basic premise of SVM is depicted in Fig. 10, where an optimum hyperplane is used to classify objects into groups based on support vector.

Fig. 10 Linear SVC



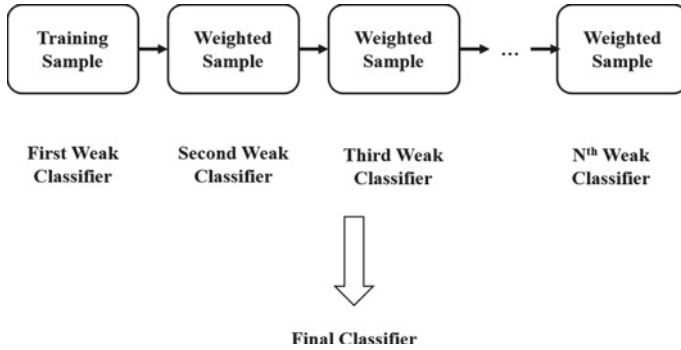


Fig. 11 XGB

2.4.6 XGB

Gradient-boosted decision trees are implemented in XGB. Decision trees are constructed sequentially in this approach. In XGBoost, weights are very significant. All of the independent variables are given weights, which are subsequently put into the decision tree, which predicts outcomes. The weight of variables that the tree predicted incorrectly is raised, and the variables are then put into the second decision tree. Individual classifiers/predictors are then combined to form a more powerful and precise model. Figure 11 shows the working of XGB [9, 10].

2.5 Results

Table 3 shows the comparison of different classifiers used for predicting the groundwater quality using the stated Machine Learning Approach. It can be seen that Random Forest and Decision Tree Classifiers perform the best and give 100 percent accuracy. Also, XGB has performed well in the process of classification, which emphasizes on the fact that when we use labeled data, i.e., supervised learning approach, then classification is better in using XGB, RF, and DT classifiers.

Table 3 Comparative results of different classifiers

Model	Train accuracy	Test accuracy
LR	0.70833	0.666667
DT	0.950000	0.920000
GNB	0.791667	0.833333
RF	0.850000	1.000000
Linear SVC	0.71667	0.583333
XGB	0.941667	0.916667

Fig. 12 Confusion matrix of RF

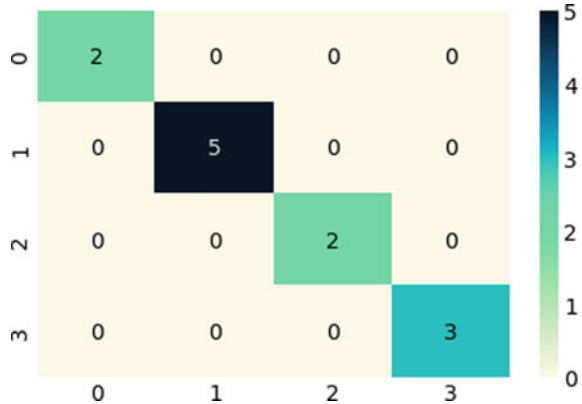


Figure 12 shows the confusion matrix of random forest classification.

2.6 Conclusion and Future Work

The groundwater quality prediction and classification were a multiclass problem. It can be observed that Extreme Gradient Boost, Random Forest, and Decision Tree Classifiers have been proved to perform best in this problem. The dataset for such problems is usually raw data, with some missing values and uneven scales, so data preprocessing needs to be done in order to get the best classification results. Further, Deep Learning techniques can be used in future for this domain, especially when the dataset is large. Time series forecasting.

References

1. "Introduction to Logistic Regression—by Ayush Pant—Towards Data Science." <https://towardsdatascience.com/introduction-to-logisticregression-66248243c148>. Accessed 11 Jan 2021
2. Thomas WE, Manz DO (2017) Chapter 4—exploratory study. In: Thomas WE, Manz DO (eds) Research methods for cyber security, Syngress, 2017. ISBN 9780128053492 pp 95–130. <https://doi.org/10.1016/B978-0-12-805349-2.00004-2>. <https://www.sciencedirect.com/science/article/pii/B9780128053492000042>
3. Timofeev R (2004) Classification and regression trees (CART) theory and applications. Humboldt University, Berlin, p 54
4. Islam MJ, Wu QJ, Ahmadi M, Sid-Ahmed MA (2007) Investigating the performance of naive-bayes classifiers and k-nearest neighbor classifiers. In: 2007 international conference on convergence information technology (ICCIT 2007), IEEE, pp 1541–1546
5. Jiang L, Wang D, Cai Z, Yan X (2007) Survey of improving naive bayes for classification. In: international conference on advanced data mining and applications. Springer, Berlin, Heidelberg, pp 134–145

6. "Understanding Random Forest. How the Algorithm Works and Why it Is... — by Tony Yiu — Towards Data Science." <https://towardsdatascience.com/understanding-random-forest58381e0602d2>. Accessed 11 Jan 2021
7. Chaudhary SK, Kamal R (2016) An improved random forest classifier for multi-class classification. *Inf Process Agric* 3(4):215–222. <https://doi.org/10.1016/j.inpa.2016.08.002>
8. Akhtar AK, Khan SA, Shaikat A (2013) Automated plant disease analysis (APDA): Performance comparison of machine learning techniques. In: *Proceeding—11th international conference frontiers of information technology. FIT 2013*, pp 60–65. <https://doi.org/10.1109/FIT.2013.19>
9. Aggarwal P (2019) ML: XGBoost (extreme gradient boosting). GeeksforGeeks. Retrieved 18 May 2022, from <https://www.geeksforgeeks.org/ml-xgboost-extreme-gradient-boosting/>
10. Friedman JH (2002) Stochastic gradient boosting. *Comput Stat Data Anal* 38(4):367–378