Prevention of cocoa moniliasis using Progressive Web Applications and sensor data in the province of Francisco de Orellana

Darwin Romero [iD] [1]*, *Pilar Oña* [iD] [2]*, *Pedro Aguilar* [iD] [3], *Wilson Chango* [iD] [4]

[1] *Escuela Superior Politécnica de Chimborazo, Sede Orellana, Ecuador.*
[2] *Escuela Superior Politécnica de Chimborazo, Sede Orellana, Ecuador.*
[3] *Escuela Superior Politécnica de Chimborazo, Sede Orellana, Ecuador;*
pedro.aguilar@espoch.edu.e
[4] *Escuela Superior Politécnica de Chimborazo, Sede Orellana, Ecuador;*
wilson.chango@espoch.edu.ec.
* *Correspondence: darwin.romero@espoch.edu.ec , pilar.onia@espoch.edu.ec*
Available from. http://dx.doi.org/10.21931/RB/2024.09.01.15

ABSTRACT

Ecuador is an essential cocoa producer recognized for its quality and aroma. Additionally, it holds a prominent position among the country's traditional export products, making it the third-largest cocoa-producing country in the world. However, the cocoa industry faces challenges due to moniliasis, a fungal disease that affects cocoa trees and causes damage to the fruits, resulting in decreased production. This research aims to prevent cocoa moniliasis by conducting tests with different algorithms to select the best one for predicting moniliasis using sensor data in the progressive web application. Various supervised learning algorithms were applied, including PCA, IPCA, KPCA, Linear Regression, Sci-Kit Learning, and ensemble methods like Bagging and Boosting. Google's Lighthouse is utilized for artifact validation. It is concluded that the Boosting ensemble method with a value of 1.0 and 4 estimators is the algorithm that shows a good fit for prediction. In artifact validation, it yields favorable results with a score of over 90 in various Lighthouse parameters.

Keywords: Moniliasis 1; Progressive Web Application 2; PCA 3; IPCA 4; KPCA 5; Linear Regression 6; Bagging 7; Boosting 8; Lighthouse 9

INTRODUCTION

Thanks to its biodiversity and favorable geographical location, Ecuador is a significant cocoa producer.[1] For the authors[2], cocoa is scientifically known as Theobroma cacao and is prominent among the country's traditional export products. It is the third-largest cocoa producer in the world, earning international recognition and appreciation for its distinctive quality and aroma. As a result, cocoa cultivation extends across various regions of Ecuador, with an annual production of 212,249 tons in 491,221 hectares. Additionally, cocoa production is paramount, sustaining the country's economy and providing employment opportunities.

The province of Francisco de Orellana, located in Ecuador, is renowned for its significant cocoa production. It is cultivated in an environment of exceptional biodiversity and a favorable geographical location, providing ideal conditions for developing high-quality cocoa beans with a distinct flavor and aroma. However, moniliasis, a fungal disease severely affecting cocoa trees caused by the *Moniliophthora perniciosa* and *Moniliophthora roreri* fungi, poses a challenge. Moreover, the authors[3-2] mention that these fungi attack various plant tissues, such as fruits, floral cushions, and buds, leading to witch's broom formation and fruit loss. This disease

is favored by factors such as high temperatures, humidity, and the age of cocoa plants, resulting in economic losses and representing one of the main threats to this industry.

Francisco de Orellana has experienced significant climate changes characterized by highly high heatwaves and variations in humidity. The combination of high moisture and optimal temperatures creates a conducive environment for spore germination and the subsequent development of the disease. Notably, frequent rainfall and insufficient ventilation in cocoa plantations can contribute to maintaining high relative humidity levels, increasing the risk of moniliasis infection.

The most common method to control moniliasis is the use of chemical fungicides. However, prolonged use of these products can lead to resistance to pathogenic fungi, negatively impact the environment, and may not be suitable for organic production.[4] As a result, efforts have been made to implement innovative technological solutions to prevent and control cocoa moniliasis, including testing various supervised algorithms. This allows for identifying the best algorithm for the progressive web application.

The Progressive Web Application (PWA) is a revolutionary approach that transcends boundaries due to its adaptability to different technological devices.[4] It offers a pleasant user experience even with slow or no internet connectivity. As authors 4-5 mentioned, progressive web applications are responsive and independent of connectivity. They can be cached for offline use, providing enhanced security compared to native applications and reducing development costs by adapting to different platforms with a secure connection using the HTTP or HTTPS protocol.

To carry out predictions, it is necessary to use Supervised Machine Learning (ML), which, according to author[6], is a branch of artificial intelligence focused on learning from experience. It addresses problems such as classification, association, clustering, and feature selection based on large structured data sets. Supervised learning has proven to be helpful in decision-making and predicting potential outcomes. Hence, different supervised algorithms were tested using Python and various libraries. This comparison allowed the selection of the best algorithm for implementation in the progressive web application.

Furthermore, as the data generated by the sensors in cocoa plantations and manually collected data are required, a database is necessary. According to the author[7], NoSQL databases are recommended when handling large amounts of data thanks to their high availability, scalability, and Performance, as they are popular for storing and processing vast data quickly. As a result, one of the most recognized and open-source databases is MongoDB, which allows for dynamic changes in documents and flexible queries, thereby accelerating data retrieval.

This study employed four methods: literature review and documentary research, design science, and Google's Lighthouse.

The questions that have been raised for the development of this research are:

- Question 1: Which supervised machine learning algorithms produce the best predictive results?
- Question 2: What scores or percentages do the Performance, Accessibility, best practices, SEO, and Progressive Web App (PWA) compliance have when validating the artifact?

This study employed four methods: literature review and documentary research, design science, and Google's Lighthouse. The current state of the research field should be carefully reviewed, and key publications cited. Please highlight controversial and diverging hypotheses when necessary. Finally, briefly mention the main aim of the work and highlight the principal conclusions.

MATERIALS AND METHODS

The methodology to be used to develop this research involves a literature review using a documentary research method with a quantitative scope for analyzing supervised learning algorithms related to the issue of cocoa moniliasis. To address this problem, a progressive web application (PWA) will be developed to help farmers prevent this disease using data from sensors and manual inputs. The artifact will be assessed using the Lighthouse tool generated by Google for a more effective evaluation, which provides a scoring mechanism.

Procedure:

Phase I:
Literature Review and Documentary Research

According to the authors, combining both research methods (bibliographic and documentary) allows for a more in-depth investigation of the topic to determine the most suitable tools for developing the progressive web application for predicting cocoa moniliasis. The Design Science Research (DSR) method will also be employed, focusing on creating artifacts as innovative solutions to practical problems in a specific domain. These artifacts are constructed through an iterative design, construction, and evaluation cycle. The goal is to validate the artifact before its implementation, ensuring its utility and effectiveness in addressing the stated problem.[9]

Phase II:
Using the Design Science Research method, the design of the PWA application will be established in detail, including the selection of technologies used for building this application. The intention is to create an application that can be adopted by other applications for predicting cocoa moniliasis, as illustrated in Figure A.
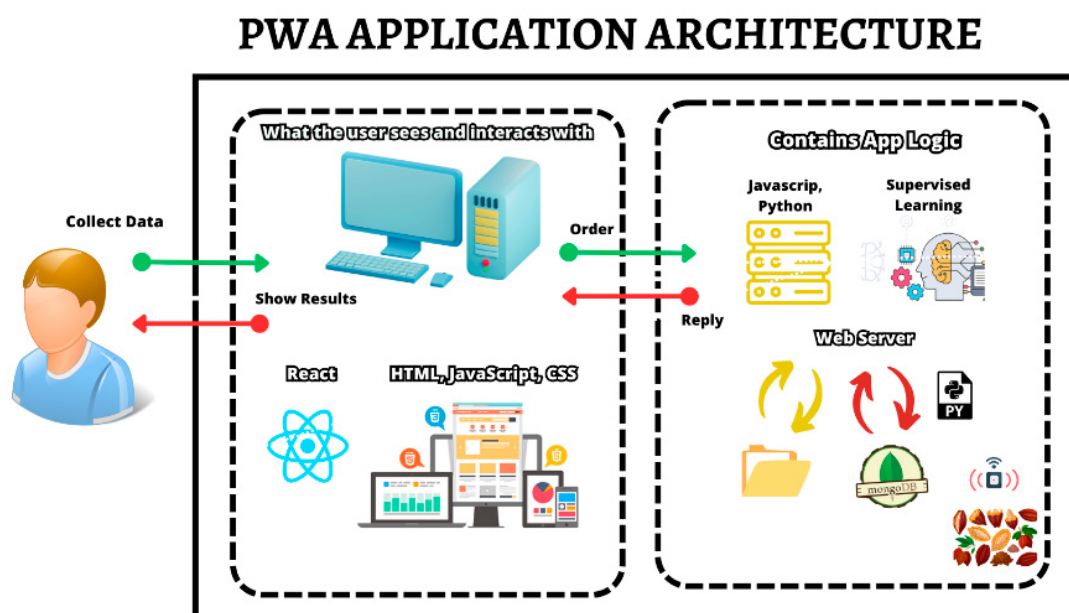

Figure 1. Architecture of the Predictive Progressive Web Application. Source: Conducted by authors, 2023.

Phase III
Data Collection
Data collection will be carried out in a cocoa plantation in the Francisco de Orellana canton for testing and training the cocoa moniliasis prediction algorithms. The data will include readings from sensors such as Rain, Relative Humidity (RH), Dew Point, Wind Speed, Gust Speed, and Wind Direction. Additionally, manual data will be collected, including essential characteristics related to cocoa moniliasis, such as plant name, plant age, fruiting months, humidity, disease severity, and incidence. These collected data will be used for training and validating the supervised learning algorithms. For implementing the algorithms, 10 data points will be selected for training and 1 for testing. In cases where data points are missing, they will be completed with a value of 0 to avoid errors during testing. Regularization and robustness algorithms will utilize the training data and one test data point within the coding.

Phase IV
Testing with Supervised Learning Algorithms
Seven supervised learning algorithms have been selected for testing, including dimensionality reduction techniques PCA, IPCA, and KPCA with standard kernels and ensemble Methods (Bagging and Boosting). These algorithms were used to determine the highest accuracy in predicting cocoa moniliasis. Additionally, Linear

Regression was implemented, focusing on data training regularization and Sci-Kit Learning, focusing on outliers with robust regressions. These tests are crucial for selecting the most effective algorithm before the final implementation of the prediction system.
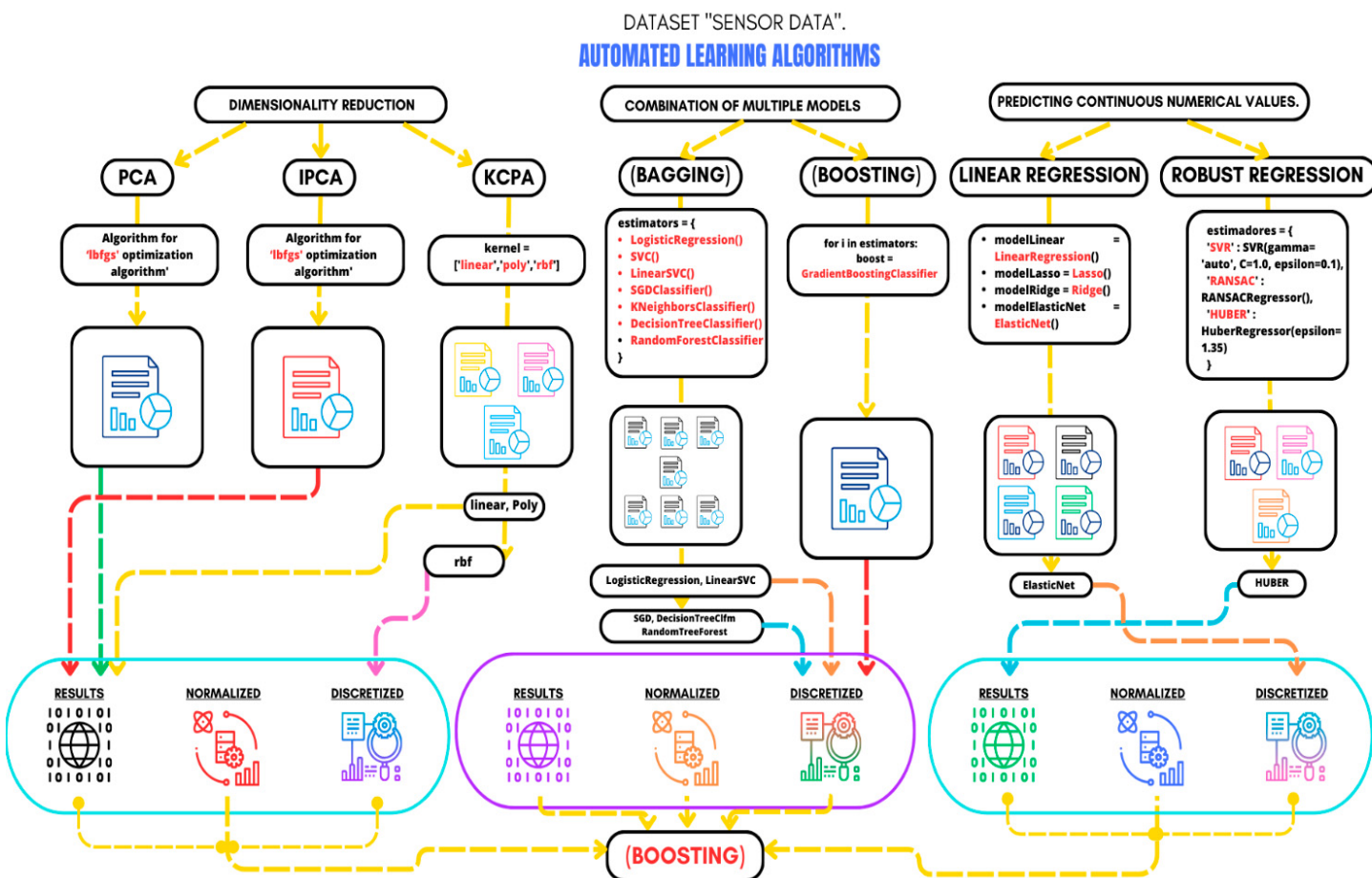

Figure 2 provides a detailed overview of the types of algorithms used and which one had the best fit during the data set testing. Source: Conducted by authors, 2023.

Phase VI
Artifact Validation
Lighthouse is an open-source tool that automates improving web applications' Performance, quality, and correctness. Analyzing a page, it performs a series of tests and generates a detailed report on its Performance. Identifying failed tests provides critical indicators to help developers improve their web applications.[4]

RESULTS

The extraction of sensor data and the collection of manual data took place at the La Belleza Experimental Station, which belonged to the Francisco de Orellana canton and was donated to the Escuela Superior Politécnica de Chimborazo Sede Orellana, covering an area of 37 hectares dedicated to Higher Education. Additionally, the artifact validation will be conducted using Google's tool.

Implementation of the Dimensionality Reduction Algorithm
PCA
According to authors[10], PCA is a method that uses principal components derived from the correlation matrix to simplify relationships between variables. Furthermore, the dataset contains many features, and not all of them are significant. The PCA algorithm was performed from three perspectives: average results, normalized

results, and discretized results, all of which showed a favorable fit with a score of 1 when using the dataset without normalization or discretization. This section may be divided into subheadings. It should provide a concise and precise description of the experimental results, their interpretation, and the experimental conclusions that can be drawn.

Algorithm Type	Types of results	Results
PCA	Normal result	1.0
	Resultado Normalizado	0.88
	Resultado Discretizado	0.9783333333333334

Table 1. Results with PCA Algorithms.Source: Conducted by authors, 2023.

The average results obtained a value of 1.0, indicating a good algorithm fit. The normalized results obtained a value of 0.88, suggesting the algorithm could not fit well. However, the discretized results showed an increased fit with a value of 0.9783333333333334.

IPCA
IPCA (Incremental PCA) is designed to use a memory-independent amount of memory when dealing with datasets that are too demanding[11]. A favorable outcome was achieved when using normal results, indicating a good algorithm fit. However, when implementing normalization and discretization, the algorithm's fit decreased, as shown below:

Algorithm Type	Types of results	Results
IPCA	Normal result	1.0
	Normalized Result	0.8733333333333333
	Discretized result	0.985

Table 2. Results with the IPCA Algorithms.Source: Conducted by authors, 2023.

Upon analyzing the results, it was found that the IPCA algorithm had an excellent fit to the data without any additional transformation, obtaining a value of 1.0. However, when normalization was applied, the algorithm's Performance decreased to 0.8733333333333333, indicating a negative impact. On the other hand, implementing discretization slightly improved the result to 0.985 compared to normalization, suggesting that discretization had a less negative impact on the algorithm's fit.

Implementation of Common Kernels Algorithm
KPCA
KPCA is a method that uses principal component analysis in a feature space to reduce the data's dimensionality when the data does not have a linearly separable structure, and a KERNEL is found[12]. The standard kernels

analyzed were linear, poly, and rbf, and when conducting training tests, the following results were obtained for average, normalized, and discretized data, as shown below:

Algorithm Type	Types of results	Results
KCPA	Normal result	Score KPCA linear: 0.9916666666666667 Score KPCA poly: 0.9916666666666667 Score KPCA rbf: 0.66
	Normalized Result	Score KPCA linear: 0.8883333333333333 Score KPCA poly: 0.85 Score KPCA rbf: 0.8433333333333334
	Discretized result	Score KPCA linear: 0.8733333333333333 Score KPCA poly: 0.8183333333333334 Score KPCA rbf: 0.8533333333333334

Table 3. Results with KPCA Algorithms.Source: Conducted by authors, 2023.

Upon analyzing the normal results, better fits were obtained in KPCA linear and poly with a value of 0.9916666666666667 compared to the normalized and discretized results, which improved the fit with KPCA rbf with a value of 0.8533333333333334. However, the normal KPCA linear result achieved the best fit.

Implementation of Linear Regression Algorithm
Linear Regression
The authors mention that this algorithm seeks to find the best straight line that fits the data by minimizing a loss function[13]. Regularization techniques include Linear, Lasso, Ridge, and ElasticNet, aiming to reduce model complexity and prevent overfitting by penalizing model coefficients. The following results were obtained for normal, normalized, and discretized data:

Algorithm Type	Types of results	Results
Linear Regression	Normal result	Score Lineal; 0.9337542850425272 Score Lasso; 0.9321976562616098 Score Ridge; 0.9337532615244014 Score ElasticNet; 0.9315081721345164
	Normalized Result	Score Lineal; 0.9144346008057538 Score Lasso; 0.9138052629638189 Score Ridge; 0.9144930223805792 Score ElasticNet; 0.9129496840895489
	Discretized result	Score Lineal; 0.901894410892162 Score Lasso; 0.9008801774658688 Score Ridge; 0.9018951542734359 Score ElasticNet; 0.9005362968682721

Table 4. Results with Linear Regression Algorithms.Source: Conducted by authors, 2023.

The Linear Regression algorithm did not yield favorable results, indicating that it did not fit the dataset well. However, a more favorable fit was achieved when discretizing the data, with ElasticNet obtaining a value of 0.9005362968682721.

Implementation of the Sci-Kit Learning Algorithm
Sci-Kit Learning
In Scikit-learn, a wide variety of robust regression estimators are provided to address problems related to outliers or noise in the data. SVR MSE, Ransac MSE, and Huber MSE are used with loss functions or fitting criteria that are less sensitive to outliers or noise in the data. These robust regression techniques have produced better results than linear regression[13], as shown below:

Algorithm Type	Types of results	Results
Sci-Kit Learning	Normal result	SVR MSE: 0.0212404219 RANSAC MSE: 0.0251234906 HUBER MSE: 0.0291127339
	Normalized Result	SVR MSE: 0.0212404219 RANSAC MSE: 0.3400000000 HUBER MSE: 0.0191127339
	Discretized result	SVR MSE: 0.0212404219 RANSAC MSE: 0.3400000000 HUBER MSE: 0.0291127339

Table 5. Results with Sci-Kit Learning Algorithms Source: Conducted by authors, 2023.

The best robust regression algorithm is SVR MSE, with a value of 0.0212404219. This value indicates a lower mean squared error than the other analyzed algorithms (Ransac MSE and Huber MSE). Therefore, SVR MSE demonstrated better Performance in model fitting to the data than the other mentioned robust regression algorithms.

Implementation of Ensemble Algorithm (Bagging and Boosting).
Bagging
Bagging combines predictions from multiple models trained on bootstrap samples, reducing variance and overfitting. It improves the accuracy and stability of the final model by averaging or voting the projections of the base models[14]. It also provides an estimation of uncertainty in forecasts. The predictive models used are KNN, LogisticRegression, SVC, LinearSVC, SGD, KNN, DecisionTreeClassifier, and RandomForestClassifier. The following results were obtained for average, normalized, and discretized data:

Algorithm Type	Types of results	Results
Ensemble Method (Bagging)	Normal result	Score con KNN: 0.99 Score Bagging with LogisticRegression: 1.0 Score Bagging with SVC: 0.99 Score Bagging with LinearSVC: 1.0 Score Bagging with SGD: 1.0 Score Bagging with KNN: 0.99 Score Bagging with DecisionTreeClf: 1.0 Score Bagging with RandomTreeForest: 1.0
	Normalized Result	Score con KNN: 0.99 Score Bagging with LogisticRegression: 1.0 Score Bagging with SVC: 0.99 Score Bagging with LinearSVC: 1.0 Score Bagging with SGD: 1.0 Score Bagging with KNN: 0.99 Score Bagging with DecisionTreeClf: 1.0 Score Bagging with RandomTreeForest: 1.0
	Discretized result	Score con KNN: 0.99 Score Bagging with LogisticRegression: 1.0 Score Bagging with SVC: 0.99 Score Bagging with LinearSVC: 1.0 Score Bagging with SGD: 1.0 Score Bagging with KNN: 0.99 Score Bagging with DecisionTreeClf: 1.0 Score Bagging with RandomTreeForest: 1.0

Table 6. Result of the Ensemble Method (Bagging) Algorithm.Source: Conducted by authors, 2023.

The results show a highly favorable fit, with values of 1.0 in most cases, indicating a perfect model fit to the data. This technique is beneficial in classification problems, reducing variance and overfitting.

Boosting

The Boosting algorithm trains weak models, such as simple decision trees, called "weak learners," in successive iterations, focusing on correcting errors[14]. Multiple GradientBoostingClassifier models were trained with different numbers of estimators, and the accuracy of each model was recorded. Then, the algorithm identified the best accuracy result along with the corresponding number of estimators. As shown below:

Algorithm Type	Types of results	Resultado
Ensemble Method (Boosting)	Normal result	{'result':0.6385714285714286, 'n_estimator': 2} {'result': 1.0, 'n_estimator': 4}
	Normalized Result	{'result':0.6385714285714286, 'n_estimator': 2} {'result': 1.0, 'n_estimator': 4}

	Discretized result	{'result':0.6385714285714286, 'n_estimator': 2} {'result': 1.0, 'n_estimator': 4}

Table 7. Result of the Ensemble Method (Boosting) Algorithm.Source: Conducted by authors, 2023.

Los resultados finales muestran que en los tres casos se dio el mismo resultado indicando un buen ajuste del algoritmo de predicción para cada modelo con un valor de 1.0 y con 4 estimadores representativos.

Para más detalle y verificación de las pruebas realizadas se pueden dirigir al siguiente link https://github.com/DarwinBRG/scikit-learn.git

Validación del artefacto

Metodo Lighthouse by Google

Se realizo la validación del artefacto con la herramienta de puntuación de Lighthouse que proporciona Google de código abierto para probar la progresividad de la aplicación web progresiva. Al realizar la el test, Lighthouse da una puntuación de Performance, Accessibility, Best Practices, SEO y PWA, se obtuvo la siguiente puntuación.[15-5]
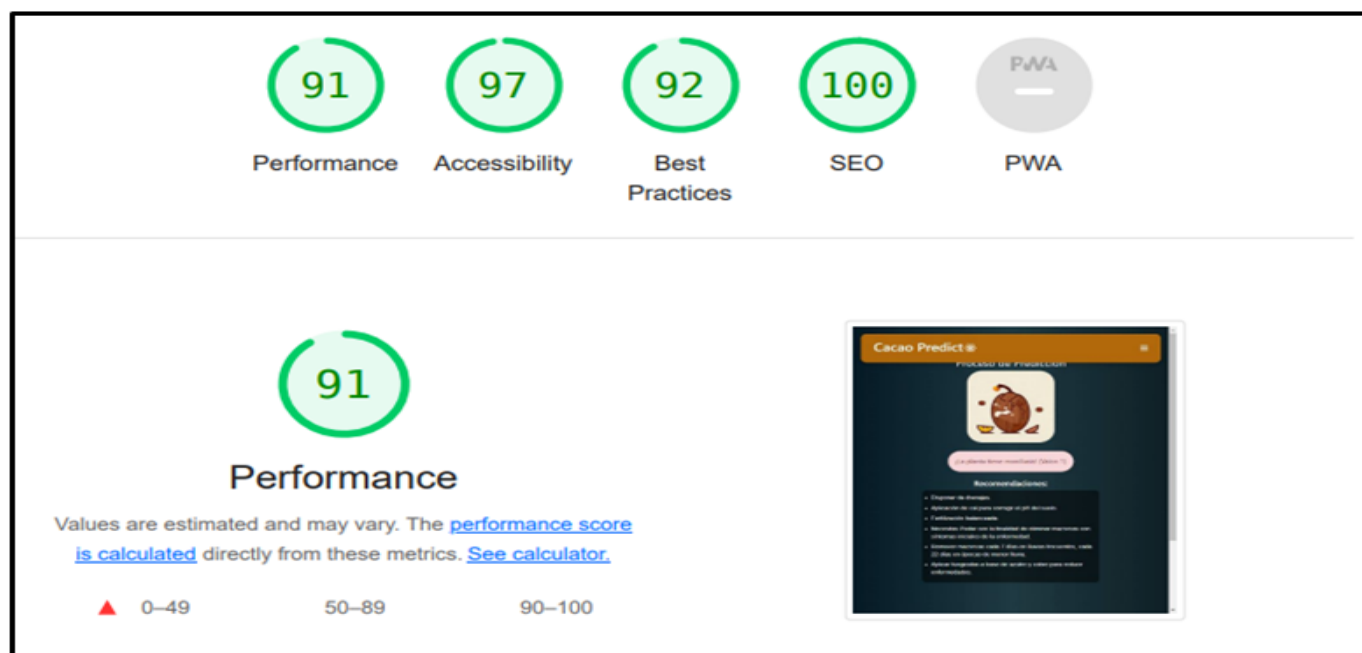

Figure 3. Artifact Validation.<Source: Conducted by authors, 2023.

Algorithm Type	Better result	Results
PCA	Normal result	1.0
IPCA	Normal result	1.0
KCPA	Normal result	Score KPCA linear: 0.9916666666666667
Linear Regression	Discretized result	Score ElasticNet 0.9005362968682721
Sci-Kit Learning	Normal result	SVR MSE: 0.0212404219

| Ensemble Method (Bagging) | Discretized result | Score Bagging with LogisticRegression: 1.0
Score Bagging with LinearSVC: 1.0
Score Bagging with SGD: 1.0
Score Bagging with DecisionTreeClf: 1.0
Score Bagging with RandomTreeForest: 1.0 |
| Ensemble Method
(Boosting) | Discretized result | {'result': 1.0, 'n_estimator': 4} |

Table 8. Analysis of relevant data.Source: Conducted by authors, 2023.

The application scored 91 in Performance, indicating good speed and efficiency. Regarding Accessibility, it scored 97, indicating that the application is accessible and usable for a broad audience, including people with disabilities. Regarding Best Practices, the application scored 92, signifying adherence to industry-recommended best practices for web development, contributing to its quality and security. The application received a perfect score of 100 for SEO, indicating that it is well-optimized for search engines and more likely to appear in relevant search results.

Additionally, an exact score is not provided in the PWA variable, but some noteworthy features are highlighted. The application is installable on any device and incorporates a service worker, allowing it to function offline by caching data. It can automatically update data and has a responsive design, ensuring adaptability to different devices. The application is also configured for a personalized presentation screen, and its content adjusts appropriately to the graphical window size. Furthermore, it establishes a theme color for the address bar and includes a "viewport" tag with the initial scale width. These features enhance the user experience and usability of the progressive web application across various environments and devices.

DISCUSSION

Analyzing the results obtained with the different supervised algorithms and considering the findings according to the authors[13], it can be mentioned that the best algorithm fit is achieved when it has a value greater than 0.80. Therefore, the following results are presented: The findings and their implications should be discussed in the broadest context possible.

It can be stated that favorable results have been obtained independently, indicating a good fit of the algorithms. However, the authors [14] mention that using decision trees helps achieve a better fit when making predictions. Therefore, it has been determined that the best algorithm is the Boosting ensemble method, and implementing this algorithm in the prediction process has yielded favorable outcomes.

CONCLUSIONS

The objective of this research was to prevent cocoa moniliasis by testing different algorithms, such as the dimensionality reduction techniques PCA, IPCA, and KPCA with standard kernels, Ensemble methods (Bagging and Boosting), Linear Regression focusing on data training regularization, and Sci-Kit Learning focusing on outliers with robust regressions. The aim was to compare these algorithms to select the one that fits the prediction well. PCA and IPCA algorithms showed an excellent fit during the tests, obtaining results of 1.0 in their tests with standard data. In contrast, the KPCA algorithm yielded favorable results, especially in its "linear" variant, with an adjustment of 0.9916666666666667. The Linear Regression algorithm demonstrated significant fit by discretizing the data, with the highest result obtained using ElasticNet (adjustment of

0.9005362968682721). The Sci-Kit Learning algorithm (SVR MSE) showed a favorable fit with a value of 0.0212404219. In the ensemble methods (Bagging and Boosting), excellent fit was demonstrated in their discretized tests. Bagging showed results of 1.0 with several estimators while Boosting achieved 1.0 with 4 estimators.

Therefore, when comparing the results with a value of 1.0, indicating a good fit, it was identified that the best algorithm for predicting cocoa moniliasis is Boosting, providing a valuable tool for disease prevention and control in cocoa cultivation. The developed progressive web application demonstrated good Performance in the artifact validation, scoring Performance (91), Accessibility (97), Best Practices (92), and SEO (100). Although the PWA parameter did not receive an exact score, it was highlighted for its ability to be installable on any device, to feature a service worker for offline operation, and to have a responsive design for adaptation to different devices. This showcases good Performance, Accessibility, compliance with best practices, and optimization for search engines, with scores over 90%. This makes it a valuable tool for cocoa producers and the agricultural sector. The relevance of this research lies in its contribution to the field of agriculture and the use of advanced technologies to improve crop production and sustainability.

REFERENCES

1. Venezuela J, Guevara F. Eco-Friendly Biocontrol of Moniliasis in Ecuadorian Cocoa Using Biplot Techniques. Sustainability. 2023;: p. 15. https://doi.org/10.3390/su15054223
2. Ceccarelli V, Lastra S, Loor R, Chacón W, Nolasco M, Conservation and use of genetic resources of cacao (Theobroma cacao L.) by gene banks and nurseries in six Latin American countries. Genet. Resour. Crop Evol. 2022; p. 1283-1302. https://doi.org/10.1007/s10722-021-01304-3
3. Almeida S, Silva S, Lima J, Fim Rosas J, Capelini V. Fuzzy modeling of the risk of cacao moniliasis occurrence in Bahia state, Brazil. SCIELO. 2020. https://doi.org/10.1590/1807-1929/agriambi.v24n4p225-230
4. Tandel S, Jamadar A. Impact of Progressive Web Apps on Web App Development. IJIRSET. 2018. http://dx.doi.org/10.15680/IJIRSET.2018.0709021
5. Adetunji O, Otuneme N, Iyasere E, Opiah O, Enaibe O. Enhanced online book store model: adopting the progressive. IJESRT. 2020. https://doi.org/10.29121/ijesrt.v9.i9.2020.1
6. Jiang T, Gradus J, Rosellini A. Supervised Machine Learning: A Brief Primer. Web of Science. 2020;: p. 675-687. https://doi.org/10.1016/j.beth.2020.05.002
7. Georgi G, Milko M. An approach to storing real-time sensor readings based on NoSQL database systems. 16ª Conferencia sobre Máquinas Eléctricas, Accionamientos y Sistemas de Potencia. 2019: p. https://doi.org/10.1109/ELMA.2019.8771666
8. Josep G, Angela O, Esperanza F. El artículo de revisión. Iberoamericana de Enfermeria. 2020.
9. Brendel A, Lembcke T, Muntermann J, Kolbe L. Toward replication study types for design science research. Journal of Information Technology.36(3). 2021: p. 198–215. https://doi.org/10.1177/02683962211006429
10. Vega J, Guzmán J. Regresión PLS y PCA como solución al problema de multicolinealidad en regresión múltiple. Revista de Matemática: Teoría y Aplicaciones, 18(1). 2021;: p. 9-20.
11. Donoso J, Guillaume G, Thomas A. Clasificación en imagenes hiperespectrales. Uchile.cl. 2017;: p. 1-4.
12. Hernández J. Dimensionality Reduction Methods: Comparative Analysis of methods PCA, PPCA and KPCA. UNICIENCIA Vol. 30, No. 1. 2016;: p. 115-122. https://doi.org/10.15359/ru.30-1.7
13. Sazzadur R, Javed S, Zarrin T, Joy R, Syed H. A Comparative Study On Liver Disease Prediction Using Supervised Machine Learning Algorithms. INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH. 2020;: p. 419-422.
14. Aljamaan H, Alazba A. Software defect prediction using tree-based ensembles. Proceedings of the 16th ACM International Conference on Predictive Models and Data Analytics in Software Engineering. 2020;: p. 1-10. https://doi.org/10.1145/3416508.3417114
15. Cardieri GdA, Zaina LM. Analyzing User Experience in Mobile Web, Native, and Progressive Web Applications: Perspectives from Users and HCI Specialists. Proceedings of the 17th Brazilian Symposium on Human Factors in Computing Systems. 2018;(9): p. https://doi.org/10.1145/3274192.3274201