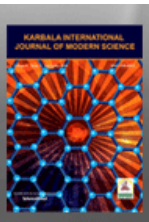

Karbala International Journal of Modern Science

Volume 7 | Issue 4 Article 3

Artificial Intelligence for Para Rubber Identification Combining Five Machine Learning Methods

Chairote Yaiprasert Ph.D.
Walailak University, Thailand, ychairot@mail.wu.ac.th

Artificial Intelligence for Para Rubber Identification Combining Five Machine Learning Methods

Abstract

This study aims to identify Para rubber species using a combination of five machine learning techniques to classify leaf images. The learning process is defined using a dataset for each classification method. Approximately 1,472 leaf images are prepared consisting of various sizes, shapes, quality provided for the model. The classification indicators are defined with the help of an algorithm to identify at least three of the top five potential classification outcomes. The algorithm accurately predicts 100% of the five classification methods. Methods can provide precise and rapid classification of large quantities, without the need for image preprocessing prior to classification.

Keywords

Machine Learning (ML), Para rubber, Classification, Programming

1. Introduction

Para rubber is commonly grown in Southeast Asian countries and part of the genus *Hevea* family *Euphorbiaceae*, of which *Hevea brasiliensis* is the only species grown to produce commercial latex [1]. The yields differ according to the type of rubber species in each cultivated area. Identifying different varieties of Para rubber in cultivation can help to meet productivity objectives. DNA profiling and visual classification is two common methods for classifying Para rubber species. Since the DNA profiling process is time-consuming, the visual classification method for Para rubber is preferred. However, visual classification of rubber seedlings is difficult without specific knowledge of agriculture. In general, trained professionals use the rubber leaf as a visual and morphological characteristic of plant organs for classification purposes. Nevertheless, identifying the organs of this plant is challenging since the leaf of each species is visually similar. Therefore, the lack of specialized taxonomists remains a problem in agricultural farming.

The organs and characteristics of plants can be used to examine their physiological aspects. Some studies have examined the growth factor in plant photosynthesis [2] while the canopy structure has been used in research on carbon-water circulation [3]. Different organ growth states are measured to estimate the various nutrients of plant growth [4]. Traditional study methods often require cutting parts of the tree, and the adoption of technology can reduce the damage caused. Plant classification can be performed using computational models of the leaf recognition system. Most plant species have a unique leaf that differs in shape, color, texture, and margin [5,6]. Various plant leaf identification methods based on shape [7−10] or texture [11,12] have been presented in recent years. These approaches only study a single visual characteristic of a leaf image, resulting in low accuracy. Thus, some leaf identification approaches involve the integration of multi-visual features in leaves for plant species recognition [13−15]. Such classification includes color and shape [16], color and texture analysis [17], surface and contour characteristics [18], fused properties of color, texture, and shape [19], leaf veins [20], color combinations, vein properties, and shape [21]. Each classification method has certain problems which limit its use.

Therefore, artificial intelligence (AI) can be used to automatically learn tasks to answer questions and significantly reduce the time and cost involved in essential practical applications [22]. One of the most cited documents relating to this subject involve predicting the automation capabilities of nearly 1000 different human occupations [23]. Furthermore, the abstract concept of AI in general [24] can bring about more radical change [25]. As an AI method, machine learning (ML) plays an important role in science education for plant recognition and classification [26]. The most common and widely used statistical technique is trend estimation which is very powerful and effective [27]. In ML, a computer algorithm allows computer programs to update automatically through experience [28]. This technology is employed by scientists to analyze large amounts of data, helping them to automate tasks [29]. It can define processes for extracting and interpreting data with a generic automated set of methods.

The ML combination has the same working principles as the taxonomist collaboration. Moreover, competent plant taxonomists specialize in a wide variety of plant species. There is a wide range of ML combinations, including numerical, textual, sounds, images, and hybrids. The comprehensive nature of ML facilitates the work of taxonomists. Various ML techniques have been applied in forecasting. The use of large datasets in forecasting can greatly improve the economic model [30] since accurate prediction reduces operating costs [31]. The ML method exhibits a better performance than conventional forecasting [32] and is also used for classification [33]. The novelty lies in the application of a single ML method. It has also been applied to the classification of a radiology report for medical purposes [34]. Moreover, ML has been used for defect recognition in plant [35], industrial [36,37], and medical applications [38]. The ML method mentioned above works as intended but is still a one-way decision. AI forecasters must ensure that predictions are used positively while minimizing their misuse as much as possible to mitigate any adverse impact [27].

https://doi.org/10.33640/2405-609X.3154

There are reasons to be optimistic that a concerted, holistic research effort could reduce uncertainty about future AI progress. Despite rubber species classification being limited to only two methods, the use of AI is another option. For this reason, the researcher is interested in using the popular supervised ML method and finding ways to increase the accuracy of rubber species classification.

This article proposes a method for classifying the *H. brasiliensis* species to aid taxonomists in defining the unretouched leaf images of Para rubber by combining ML methods. This study uses a combination of multiple judgments to increase accuracy. Leaf images of the Para rubber species are defined according to the association rule in the training and learning model. The Para rubber leaf classification algorithm is then created based on the k-nearest neighbors (k-NN), logistic model, naïve Bayes, neural networks, and random forest of the ML method. All technical issues can be solved by ML methods based on the Wolfram programming language [39]. The results suggest that combined summation yields more accurate classification than the single method.

2. Material and methods

2.1. Experimental data

In the study experiment, the ML algorithm trains and learns the leaf characteristics of Para rubber and memorizes the leaf properties with high precision. The extraction of leaf properties relies on color identification, leaf texture, and shape. The different leaf images are classified into five cultivar classes of Para rubber, namely, PB350, RRIM3001, RRIM600, RRIT251, and RRIT408 as shown in Table 1.

Fig. 1 shows the Para rubber leaf images from a preliminary study to select a suitable ML classification using a background environment. In this study, the accuracy of each method is different. The k-NN, logistic model, naïve Bayes, neural networks, and random forest of the chosen ML method yield a classification accuracy of more than 85%. Whereas four other methods, namely class distribution, decision tree, gradient boosted trees, Markov, and support vector machine yield less than 85% accuracy. A combination of all methods will subsequently be used to increase the accuracy of the results.

Fig. 1. Para rubber leaves with an environmental background.

2.2. Machine learning method

2.2.1. k-nearest neighbors

The k-NN algorithm for local estimation is a non-parameter classification and regression method. The algorithm relies on the classification of distances. The input consists of the k training examples closest to the dataset. The output result depends on whether k-NN is used for the model. All calculations are deferred until

Table 1
Leaf image samples of the Para rubber classes used for training and classification.

Type	PB350	RRIM3001	RRIM600	RRIT251	RRIT408
Training					
Classification					

the function has been evaluated. If the properties represent different physical units or have drastically varying scales, normalizing the training data can significantly improve accuracy [40]. Implementation involves setting a neighbor's weight to the average contribution rather than determining the distance (d). The k-NN classifier sets the k-NN weight to $1/\ k$ and the others to zero.

The order of the nearest neighbors is assigned by weight w_{ni}, with $\sum_{i=1}^{n} w_{ni} = 1$, where $\{w_{ni}\}_{i=1}^{n}$ is determined as the weight of the nearest classifier C_n^{wnn}. The results reveal similarities to the strong consistency of a weighted nearest neighbor classifier. The excess risk under the condition of class distribution consistency shows asymptotic expansion [41] for constants B_1 and B_2 as follows:

$$\mathscr{R}_{\mathscr{R}}\left(C_n^{wnn}\right) - \mathscr{R}_{\mathscr{R}}\left(C^{Bayes}\right) = \left(B_1 s_n^2 + B_2 t_n^2\right)\{1 + o(1)\}, \tag{1}$$

where

$$s_n^2 = \sum\nolimits_{i=1}^{n} w_{ni}^2 \tag{2}$$

and

$$t_n = n^{-2/d} \sum\nolimits_{i=1}^{n} w_{ni} \left\{ i^{1+2/d} - (i-1)^{1+2/d} \right\}. \tag{3}$$

The optimal weighting of $\{w_{ni}^*\}_{i=1}^{n}$ to achieve equilibrium corresponds to the setting:

$$k^* = \left\lfloor B n^{\frac{4}{d+4}} \right\rfloor, \tag{4}$$

$$w_{ni}^* = \frac{1}{k^*}\left[1 + \frac{d}{2} - \frac{d}{2k^{*2/d}}\left\{ i^{1+2/d} - \left(i-1\right)^{1+2/d} \right\}\right], \tag{5}$$

for $i = 1, 2, ..., k^*$ and $w_{ni}^* = 0$ for $i = k^* + 1, ...n$.

The training cases with a class label are vectors in a multidimensional feature space. The algorithm training process involves a collection of vector attributes and an example of a training class label. During classification, k is a user-defined event, with an unlabeled vector assigned a label prior to classification. This is most common among the k training samples closest to the query or test point. The best option for k depends on the data. In general, a large $k-$value reduces the effect of noise in classification [42]. The nearest neighbor algorithm predicts the closest training class to the sample.

2.2.2. Logistic regression

The coefficient is estimated from the data and verified by logistic regression with the given parameters. The logistic regression model consists of two predictors x_1 and x_2, and one binary response variable Y, denoting $p = P(Y = 1)$. A predictor variable and the log-odds (also known as logit) of events at $Y = 1$ indicate a linear relationship for estimation, formulated as follows:

$$\ell = \log_b \frac{p}{1-p} = \beta_0 + \beta_1 x_1 + \beta_2 x_2, \tag{6}$$

where ℓ is the log-odds, b is the base of the logarithm, and β_i are the model parameters.

The purpose of logistic regression is to model whether a random variable Y is 0 or 1 given the experimental data [43]. A generalized linear model function can be parameterized by θ,

$$h_\theta(X) = \frac{1}{1 + e^{-\theta^T X}} = \Pr(Y = 1 | X; \theta). \tag{7}$$

Therefore,

$$\Pr(Y = 0 | X; \theta) = 1 - h_\theta(X), \tag{8}$$

where $Y \in \{0, 1\}$ and $\Pr(y|X;\theta)$ are provided by $\Pr(y|X;\theta) = h_\theta(X)^y (1 - h_\theta(X))^{(1-y)}$.

Assuming all the probability functions observed in the sample are independent, the Bernoulli distribution could be calculated as

$$\begin{aligned} L(\theta|y;x) &= \Pr(Y|X;\theta) \\ &= \prod\nolimits_i \Pr(y_i|x_i;\theta) \quad = \prod\nolimits_i h_\theta(x_i)^{y_i} \left(1 - h_\theta(x_i)\right)^{(1-y_i)}. \end{aligned} \tag{9}$$

Typically, the log-likelihood is maximized,

$$N^{-1} \log L(\theta|y;x) = N^{-1} \sum_{i=1}^{N} \log \Pr(y_i|x_i;\theta), \tag{10}$$

using optimization techniques such as the gradient descent.

2.2.3. Naïve Bayes

Naïve Bayes is a conditional probability model for classifying a given problem, represented by vectors $\mathbf{x} = (x_1, ..., x_n)$ rather than assigning n properties (independent variables) to assess the instance of probability for each K outcome [44] or class C_k following $p(C_k|x_1, ..., x_n)$ [43]. If there are many properties n or values for the attributes, it will be impossible to use the model for probability. The conditional probabilities of the Bayes theorem can be explained in terms of prior

probability and the likelihood that it is proportional to the evidence as follows:

$$p(C_k|\mathbf{x}) = \frac{p(C_k)p(\mathbf{x}|C_k)}{p(\mathbf{x})}. \quad (11)$$

The class probability for the given example is:

$$p(C_k|x_1, \ldots, x_n) \propto p(C_k) \prod_{i=1}^{n} p(x_i|C_k), \quad (12)$$

where $\propto$ denotes proportionality, $p(C_k|x_1, \ldots, x_n)$ is the probability distribution of feature x_i given the class, $p(C_k)$ is the prior probability of the class, $p(\mathbf{x}|C_k)$ is the likelihood, and $p(\mathbf{x})$ is the evidence. The distributions are as expected from the training data. In the current implementation, distributions are modeled using a piecewise-constant function (i.e., a variable-width histogram).

2.2.4. Neural networks

A neural network consists of interconnected perceptron node layers and is similar to multiple linear regression [45]. Perceptrons feed the signal generated by multiple linear regressions into a potentially nonlinear activation function. The task of identifying something which humans find difficult is easier for neural networks. The neural network procedures help to characterize samples. Given a set of N training examples in the form $(x_1, y_1), \ldots, (x_N, y_N)$, where x_i is a characteristic of the $i-$th sample order and y_i is the class of a Para rubber leaf image. The learning algorithm discovers the $g : X \to Y$ functions, where X is the input data and Y the output class. The possible data attribute Y is a component function G, often called the hypothetical space. It is sometimes convenient to substitute the $g-$value using the grading function $f = X \times Y \to \mathbb{R}$ so g is assigned to return the highest rated $g(x) = \text{argmax } f(x, y)$.

Let F denote the space for the grading function. Although G and F represent the data of any function, the learning algorithm is also a probability model where g is the conditional probability model $g(x) = P(y|x)$, or f is the common probability model $f(x, y) = P(x, y)$. Two basic methods for selecting f or g are used to detect the risk function best suited to the training data or the control of bias exchange. It is assumed that the training set consists of samples with the same independence and evenly distributed pairs (x_i, y_i). $L : Y \times Y \to \mathbb{R}^{\geq 0}$ is defined as a loss function to measure its suitability for the training data. The risk $R(g)$ of the g function represents the expected loss of g, estimated from the following training data $R_{emp}(g) = 1/N \sum_i L(y_i, g(x_i))$, where $g(x_i)$ is the empirical risk.

2.2.5. Random forest

Random forest is a supervised learning algorithm ensemble used in classification. It generally outperforms the decision tree method since it represents the mean of several deep decision trees practiced in different parts of the same training set with the intention of reducing variance. An overfitting training set will correct the random decision forest [46]. The random forest training algorithm uses the common technique of bootstrap aggregation or bagging. The training is set to $X = x_1, \ldots, x_n$ with the solutions $Y = y_1, \ldots, y_n$. A random sample with a change to the training set and tree fitting is repeatedly selected by bagging (B times). Merely training multiple trees in one training set will result in them being significantly related.

The predictions for the invisible sample (x') after training are obtained by taking the forecast averages from each regression tree on $\widehat{f} = 1/B \sum_{b=1}^{B} f_b(x')$. This bootstrapping process leads to better model performance since it reduces variance without increasing bias. Moreover, prediction uncertainty estimation is performed as the standard deviation of the forecast from each regression tree on x' as

$$\sigma = \sqrt{\frac{\sum_{b=1}^{B} (f_b(x') - \widehat{f})^2}{B-1}}. \quad (13)$$

It is common for a few hundred to thousands of trees to be used, depending on the size and nature of the training set.

2.3. Programming procedure

The calculations in this study are performed using a MacBook Pro, 2.3 GHz Intel Core i5 computer processor, with 8 GB 2133 MHz LPDDR3 memory. Mathematica version 12 of the Wolfram Language is used for algorithmic development. Firstly, the machine is assigned to learn image recognition from the association rule, which is a rule-based ML method for finding interesting relationships between variables in large databases. Each type of Para rubber leaf image is imported to designate the primary key of the association rule as shown in **Code Snippet 1.**

```
paraImage = Table[Import[leafImages],
                  {i, 1, Length[leafImages]}]
```

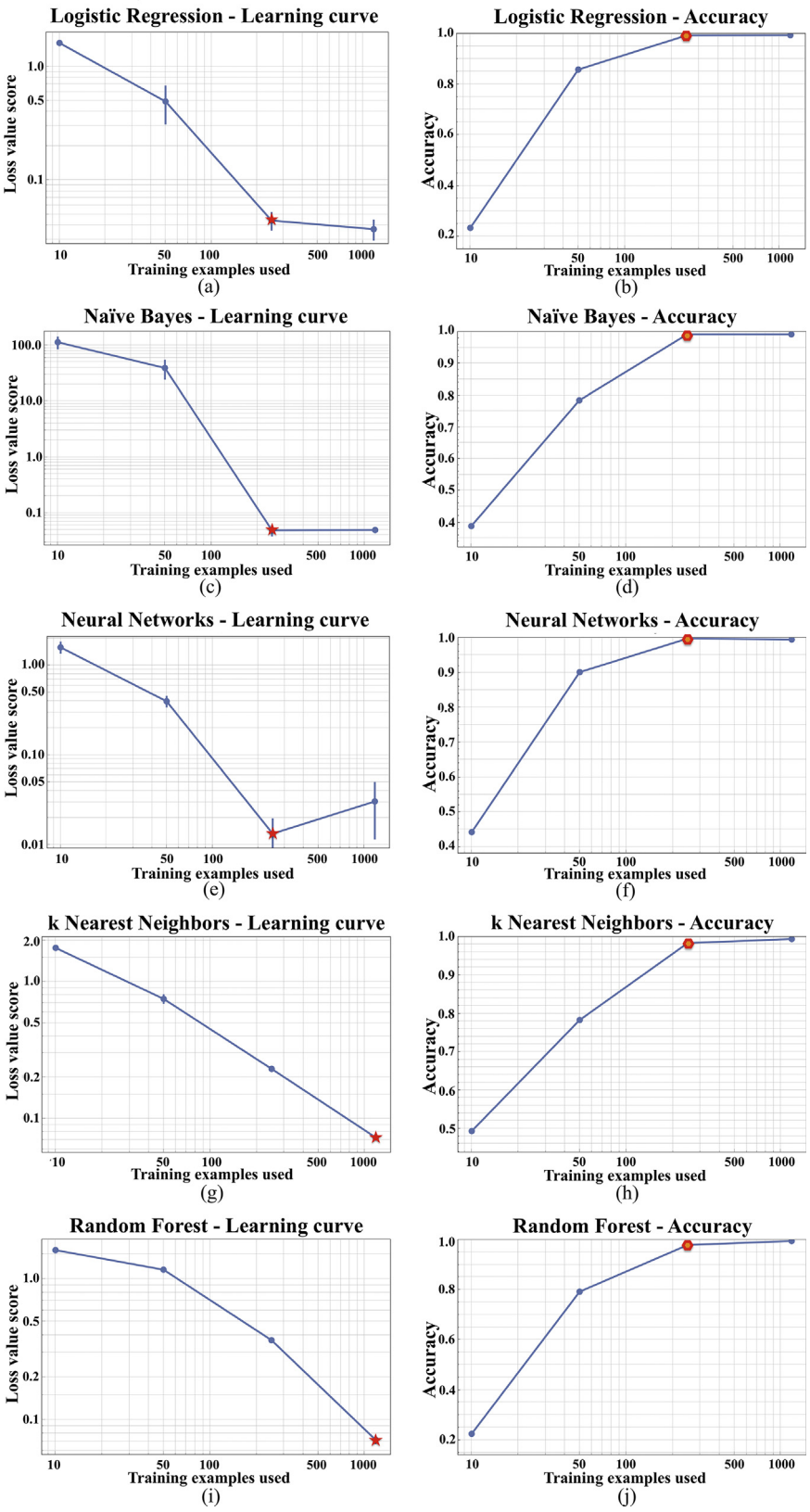

Fig. 2. Learning and accuracy curves for each ML method. (a), (c), and (e) represent learning curves with a loss value score as low as near-zero when learning data is estimated at 250 images. (g) and (i) show that a total of 1472 images of learning data are required to achieve a near-zero loss value score. (b), (d), (h), (f), and (j) represent high accuracy curves approaching 1 for all models using approximately 250 learning images.

Code Snippet 1. Key aspects of the ML method are determined based on each Para rubber leaf image.

The Para rubber leaf images are trained to become the association rule, acting as a symbolically indexed list. The association rule is applied as a training set in the ML method as displayed in **Code Snippet 2.** The classification value result is associated with a given critical component of the specification key.

```
associationRule = Table[{paraImage[[i]] → keyIndex[[i]]},
                        {i, 1, Length[paraImage]}]
trainingSet = Flatten[associationRule]
```

Code Snippet 2. The training dataset is determined from the key ML aspects using a specific key.

The ML programming classification methods k-NN, logistic regression, naïve Bayes, neural networks, and random forest are illustrated in **Code Snippet 3.** The classification function is used to create dataset training rather than explicit programming.

```
knn = Classify [trainingSet, Method →"NearestNeighbors"]
lr = Classify [trainingSet, Method →"LogisticRegression"]
nb = Classify [trainingSet, Method →"NaiveBayes"]
nnw = Classify [trainingSet, Method →"NeuralNetwork"]
rf = Classify [trainingSet, Method →"RandomForest"]
```

Code Snippet 3. The training datasets assigned to learn each ML classification method.

The classification method distinguishes potential leaf features, such as shape, dimension, color, pattern, and background. Each classification method is unique, but all forms aim to find the sample's maximum probability based on the training dataset. This study brings together the best classification possibility for each method. The programming defines the likelihood that at least three out of the five data classification indicators can be included in the results.

3. Results

3.1. Learning model

The learning results for each method are illustrated in Table 2. The classification model is trained from a dataset after modifying the output layer to satisfy the classification requirements. The probability prediction for each sample class follows the interest classification problem of modeling mapping input variables to class labels. Predicting the probability of binary and multiple-class classification problems might indicate the likelihood of an example belonging to the first class and each of the other classes, respectively. A set of model weights that mitigate the difference between the model's predicted probability distribution from the dataset and the probability distribution in the training dataset is known as cross-entropy in maximum likelihood estimation. Cross-entropy is used in this study to estimate the difference between predicted and estimated probability distributions. The model shows perfect training results of almost zero, demonstrating promising improvements in learning for each model classification, yielding an accuracy of over 99% for the different approaches.

Moreover, ML is required to minimize errors. The objective function determines the error measurement and is often called the cost or loss function. The loss function provides the best performance in terms of mean classification accuracy. A near-perfect loss range is approaching zero in this study, confirming the excellent training of the model. Furthermore, each classification method allows speedy computation times while consuming few machine resources, as shown in Table 2 and Fig. 2.

The learning curves indicate the level of development. In the early stages, the model experiences underfitting, meaning that it could not learn the training dataset. Optimal results for the learning algorithm goal exist between the overfitting and

Table 2
Learning results of each method, demonstrating low loss and high accuracy with minimal machine resources.

Method	Loss	Accuracy %	Speed (examples/s)	Training Time
k-NN	0.0703 ± 0.0029	99.48 ± 0.12	12.1	2 min 19 s
Logistic Regression	0.0243 ± 0.0040	99.61 ± 0.13	12.3	3 min 1 s
Naïve Bayes	0.0497 ± 0.0078	99.20 ± 0.34	9.61	2 min 41 s
Neural Networks	0.0131 ± 0.0060	99.80 ± 0.26	11.7	3 min 27 s
Random Forest	0.0865 ± 0.0021	99.20 ± 0.06	12.3	2 min 16 s

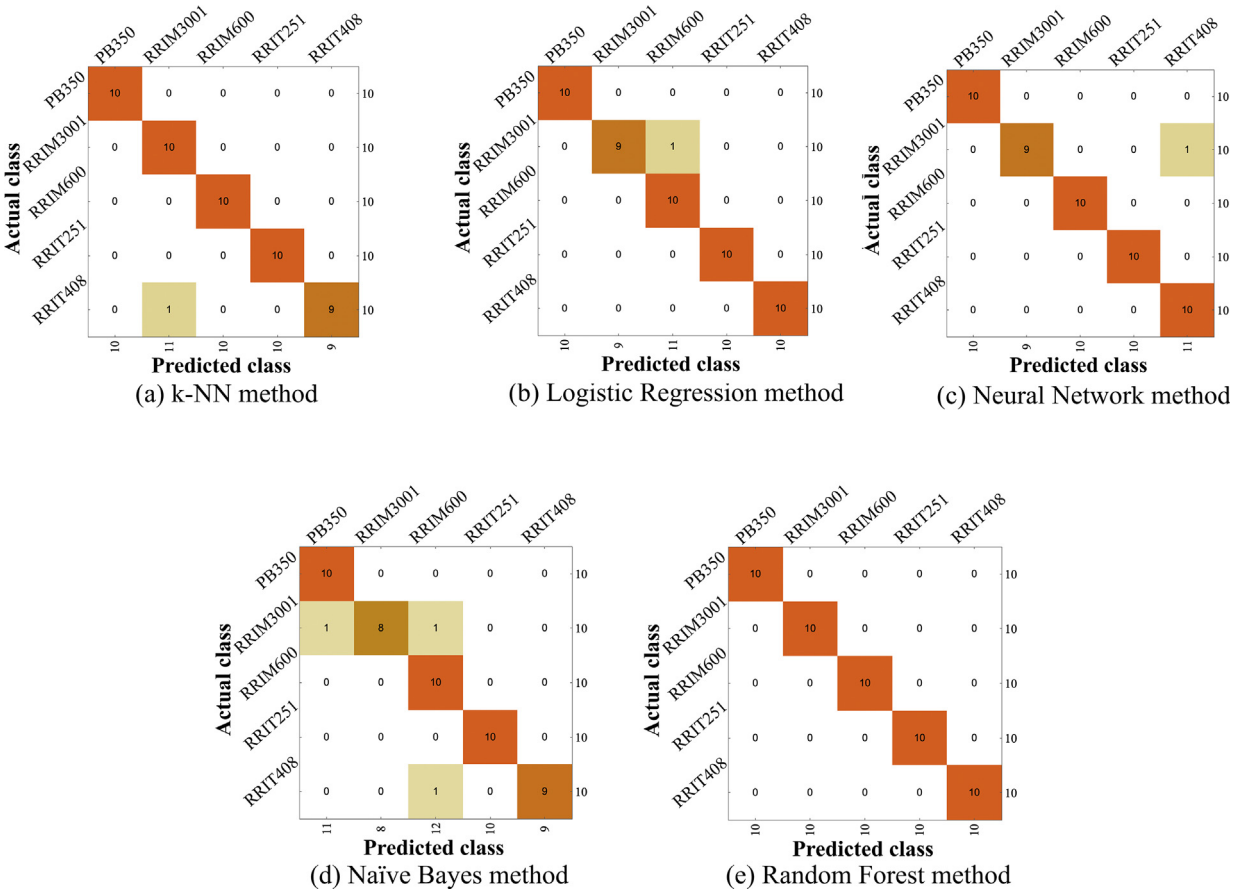

Fig. 3. The confusion matrix plot showing a comparison between the actual and predicted classes of the Para rubber leaves and confirmation of the training data fit for each ML method. (a), (b), and (c) show misclassification for 1 out of 50 images. (d) presents a maximum of 3 out of 50 misclassifications. (e) reveals the accuracy of all classifications.

underfitting models when the loss value of the learning curve starts to approach zero. The results of this study range from 0.0131 ± 0.0060 to 0.0865 ± 0.0021. A good fit is identified when the training and validation losses are reduced to a stable point with minimal mean gaps between the two final loss values. Continuous fit training for logistic regression, naïve Bayes, and neural network methods lead to an overfit when entering 250 training samples. This is an undesirable situation because the resulting overfit does not accurately estimate the response to new observations that do not form part of the original training dataset [47]. Fig. 2(e) indicates overfitting in the neural network method, where learning curves increase the loss value score after learning with 250 images. However, it does not reduce the accuracy, as shown in Fig. 2(f).

3.2. *Classification*

The Para rubber classes were selected based on the most common cultivation in Thailand. The Para rubber leaf images for this study were obtained from the

Table 3
Average probability of each ML method for Para rubber leaf classification.

Actual	k-NN	Logistic Regression	Naïve Bayes	Neural Networks	Random Forest	Summary	Time (s)
PB350	0.98	1.00	1.00	1.00	0.65	PB350	0.000058
RRIM3001	0.98	0.89	0.90	0.86	0.65	RRIM3001	0.000049
RRIM600	0.98	0.99	1.00	0.99	0.88	RRIM600	0.000048
PRIT251	0.97	1.00	1.00	1.00	0.97	RRIT251	0.000047
RRIT408	0.87	1.00	0.90	1.00	0.80	RRIT408	0.000053

Rubber Research Center, Thailand, with a total of 1472 rubber leaf samples used for training. Leaf images with approximately 50 unadorned background problems were used for the test classification. Each model method could recognize characteristics. A total of 250 rubber leaf samples were used in the classification according to the confusion matrix plot shown in Fig. 3. All methods appear to be slightly less effective when examining RRIM3001 and RRIT408. However, the use of a combined analysis technique improved more than three out of the five classification conclusions. The analysis results achieved an accuracy of 100%. Moreover, each class was analyzed rapidly, ranging from 0.000047 to 0.000058 s as shown in Table 3. An example of the classification combining five ML programming methods is demonstrated in **Code Snippet 4**.

```
rrim600 = 
sumIdentify = Tally[ {knn[rrim600, "TopProbabilities"],
      lr[rrim600, "TopProbabilities"],
      nb[rrim600, "TopProbabilities"],
      nnw[rrim600, "TopProbabilities"],
      rf[rrim600, "TopProbabilities"]} ]
{{RRIM600, 5}}
```

Code Snippet 4. An RRIM600 unretouched image sample of the combined five ML programs. Each ML analysis gave the highest probability—finally, all evaluated methods are summarized with the matching results as RRIM600.

4. Discussion

Various researchers use traditional leaf image processing to identify and classify plants [48]. Nevertheless, some have experienced problems with formulating methods and results in general, such as a lack of automated processes regarding the selection of suitable qualifications for classification [49]. In other words, traditional image processing methods are based on hand-designed peculiarities such as feature selection, image property extraction, and image preprocessing. The preprocessing step consists of image reorientation, binary thresholding, contrast stretching, edge recognition, gray scaling, noise removal, and threshold inversion. Rather than using traditional methods, whereby manual feature extraction must be flawless to achieve successful results, there is a need for a model that does not require preprocessing to perform successful classification. However, in this study, the combined five ML methods using automation can be classified precisely without modifying images beforehand, as evidenced by the results.

The study model can be used in agriculture, forestry, rural medicine, and other commercial applications. Identifying Para rubber species is vital for correct crop selection and effective agricultural production. It is also essential to the study of biodiversity and automatic identification in high demand by the agricultural industry [50]. AI computing methods are becoming more automated, enabling machines to perform similar tasks to those requiring human intelligence. AI involves giving computers access to a wealth of information and finding the most suitable solution without applying human-like feelings.

Plant leaf image classification at the species level is often referred to as a granular problem. High homogeneity at the species level and background problems make species identification difficult [51]. The model presented in this study can classify the leaf image of the Para rubber species. The key to obtaining high classification efficiency is to use a combination of ML methods with an advanced function program, leading to a larger variety of properties and higher accuracy than individual depletion.

In the context of model optimization, algorithms are used in this study to evaluate a set of weights, known as an objective function or criterion. The operation can help to maximize or minimize the objective function and identify candidates with the highest or lowest scores, respectively. The optimization process involves the selection of the loss function during the calculation of model errors. This is a challenging issue because the operation must capture the problem properties and be motivated by critical concerns in the study data. These ML programs can help to solve the problem. An automated program can find patterns in data and apply them to new interests. Generally, ML algorithm learning demands large amounts of data and is highly resource intensive. Pre-trained network data helps to reduce data shortages consistent with a specific task (e.g., VGG [52]). This is known as transfer learning [53]. However, this study provides a solution to the problem by considering the loss value.

In addition, many ML methods can be implemented, such as monarch butterfly optimization (MBO), the earthworm optimization algorithm (EWA), elephant herding optimization (EHO), the moth search (MS) algorithm, slime mold algorithm (SMA), Harris Hawks optimization (HHO), and so on. The ML selection method for the task may use complex or

straightforward sample testing and yield satisfactory results with a similar level of accuracy as those obtained from this study.

If this study only considered one method for identifying Para rubber leaves there could be a chance of misclassification. The likelihood of this error could coincide with the use of two methods and would still be classified correctly for three methods. However, if three or four summation methods are applied, an invalid final summation could result. This research suggests that at least five ML methods should be used together to reach an appropriate conclusion. Plant classification using a combination of ML methods can be performed quickly on large quantities. The results confirm that ML is a highly accurate operation and could resolve the problem of a taxonomist shortage. A high impact could be achieved through continuous development, such as implementing a mobile application to determine the image obtained from the planted plots and sending the processed photos for classification via the Internet.

In future works, this study can be developed into a mobile application. Para rubber classification by ML on a mobile device would provide an easily accessible tool to assist farmers in making informed decisions. However, hardware and software architecture should be considered when implementing classification systems on mobile applications to ensure the interoperability of ML methods. Finally, it is unlikely that all ML methods would be used in any study. Therefore, defining the number of ML methods for collective decision-making may require optimization to be performed by AI since it is the most powerful all-purpose technology in this era. The impact of such technology will ultimately benefit agriculture, biodiversity, business, and the economy, while also inspiring further innovation.

5. Conclusion

Combining ML techniques will facilitate the fast and accurate classification of large volumes of information. A program that enables a learning accuracy of more than 99.20% and attains a slight learning loss value approaching zero is the key to successful research. The algorithm was 100% accurate in predicting the five ML methods. The success of the algorithm speed factor in both model training and data classification is due to the use of a computer language with the capability to process and manipulate data. The ML algorithm in the programming language provides advanced ML capabilities, as demonstrated by the fast performance of five analytical computations with five information classes. Each Para rubber class takes less than 0.000054 s to process. This ML program is easy to implement and supports a wide variety of research areas. Specifically, the imaging analysis could abandon the need for the retouching process commonly performed in conventional programming. Therefore, this research provides a guide for data scientists and taxonomists using ML on specific tasks. This technique can also be used in various applications in the future.

Conflicts of interest

The study had no competing financial interests or personal relationships that may influence the work reported in this article.

References

[1] K. Asawatreratanakul, Y.-W. Zhang, D. Wititsuwannakul, R. Wititsuwannakul, S. Takahashi, A. Rattanapittayaporn, T. Koyama, Molecular cloning, expression and characterization of cDNA encoding cis-prenyltransferases from *Hevea brasiliensis*, Eur. J. Biochem. 270 (2003) 4671–4680, https://doi.org/10.1046/j.1432-1033.2003.03863.x.

[2] L.E. Williams, J.E. Ayars, Grapevine water use and the crop coefficient are linear functions of the shaded area measured beneath the canopy, Agric. For. Meteorol. 132 (2005) 201–211, https://doi.org/10.1016/j.agrformet.2005.07.010.

[3] H. Davi, E. Dufrêne, A. Granier, V. Le Dantec, C. Barbaroux, C. François, N. Bréda, Modelling carbon and water cycles in a beech forest, Ecol. Model. 185 (2005) 387–405, https://doi.org/10.1016/j.ecolmodel.2005.01.003.

[4] L. Djouba Conde, Z. Chen, H. Chen, H. Liao, Effects of phosphorus availability on plant growth and soil nutrient status in the rice/soybean rotation system on newly cultivated acidic soils, Am. J. Agric. For. 2 (2014) 309–316, https://doi.org/10.11648/j.ajaf.20140206.23.

[5] T. Beghin, J.S. Cope, P. Remagnino, S. Barman, Shape and texture based plant leaf classification, Lect. Notes Comput. Sci. 6475 (2010) 345–353, https://doi.org/10.1007/978-3-642-17691-3_32.

[6] D. Casanova, J.J. de Mesquita Sá Junior, O.M. Bruno, Plant leaf identification using Gabor wavelets, Int. J. Imag. Syst. Technol. 19 (2009) 236–243, https://doi.org/10.1002/ima.20201.

[7] C. Zhao, S.S.F. Chan, W.-K. Cham, L.M. Chu, Plant identification using leaf shapes–a pattern counting approach, Pattern Recogn. 48 (2015) 3203–3215, https://doi.org/10.1016/j.patcog.2015.04.004.

[8] C. Yang, H. Wei, Q. Yu, Multiscale triangular centroid distance for shape-based plant leaf recognition, Eur. Conf. Artif. Intell. 285 (2016) 269–276, https://doi.org/10.3233/978-1-61499-672-9-269.

[9] B. Wang, Y. Gao, C. Sun, M. Blumenstein, J. La Salle, Can walking and measuring along chord bunches better describe leaf shapes? Proc. IEEE Comput. Vis. Pattern Recognit. 17355554 (2017) 6119−6128, https://doi.org/10.1109/CVPR.2017.221.

[10] Z. Li, J. Hu, M. Stojmenovic, Z. Liu, W. Liu, Revisiting spectral clustering for near-convex decomposition of 2D shape, Pattern Recogn. 105 (2020) 107371, https://doi.org/10.1016/j.patcog.2020.107371.

[11] M.Z. Rashad, B.S. el-Desouky, M.S. Khawasik, Plants images classification based on textural features using combined classifier, Int. J. Comput. Sci. Inf. Technol. 3 (2011) 93−100, https://doi.org/10.5121/ijcsit.2011.3407.

[12] Z. Tang, Y. Su, M.J. Er, F. Qi, L. Zhang, J. Zhou, A local binary pattern based texture descriptors for classification of tea leaves, Neurocomputing 168 (2015) 1011−1023, https://doi.org/10.1016/j.neucom.2015.05.024.

[13] D.G. Tsolakidis, D.I. Kosmopoulos, G. Papadourakis, Plant leaf recognition using zernike moments and histogram of oriented gradients, Artif. Intell. Method Appl. 8445 (2014) 406−417, https://doi.org/10.1007/978-3-319-07064-3_33.

[14] M. Lukic, E. Tuba, M. Tuba, Leaf recognition algorithm using support vector machine with HU moments and local binary patterns, IEEE 15 Int. Symp. Appl. Mach. Intell. Inform. Proc. 16758125 (2017) 485−490, https://doi.org/10.1109/SAMI.2017.7880358.

[15] X. Zhang, W. Zhao, H. Luo, L. Chen, J. Peng, J. Fan, Plant recognition via leaf shape and margin features, Multimed. Tool Appl. 78 (2019) 27463−27489, https://doi.org/10.1007/s11042-019-07846-0.

[16] A.J. Pérez, F. López, J.V. Benlloch, S. Christensen, Colour and shape analysis techniques for weed detection in cereal fields, Comput. Electron. Agric. 25 (2000) 197−212, https://doi.org/10.1016/s0168-1699(99)00068-x.

[17] R. Pydipati, T.F. Burks, W.S. Lee, Identification of citrus disease using color texture features and discriminant analysis, Comput. Electron. Agric. 52 (2006) 49−59, https://doi.org/10.1016/j.compag.2006.01.004.

[18] J. Chaki, R. Parekh, S. Bhattacharya, Plant leaf recognition using texture and shape features with neural classifiers, Pattern Recogn. Lett. 58 (2015) 61−68, https://doi.org/10.1016/j.patrec.2015.02.010.

[19] J.K. Patil, R. Kumar, Analysis of content based image retrieval for plant leaf diseases using color, shape and texture features, Eng. Agric. Environ. Food 10 (2017) 69−78, https://doi.org/10.1016/j.eaef.2016.11.004.

[20] M.G. Larese, A.E. Bayá, R.M. Craviotto, M.R. Arango, C. Gallo, P.M. Granitto, Multiscale recognition of legume varieties based on leaf venation images, Expert Syst. Appl. 41 (2014) 4638−4647, https://doi.org/10.1016/j.eswa.2014.01.029.

[21] M. Turkoglu, D. Hanbay, Recognition of plant leaves: an approach with hybrid features produced by dividing leaf images into two and four parts, Appl. Math. Comput. 352 (2019) 1−14, https://doi.org/10.1016/j.amc.2019.01.054.

[22] P. Bornet, I. Barkin, J. Wirtz, Intelligent Automation: Welcome to the World of Hyperautomation, World Scientific Publishing Co. Pte. Ltd., Singapore, 2021.

[23] C.B. Frey, M.A. Osborne, The future of employment: how susceptible are jobs to computerisation? Technol. Forecast. Soc. Change 114 (2013) 254−280, https://doi.org/10.1016/j.techfore.2016.08.019.

[24] B. Goertzel, C. Pennachin, Artificial General Intelligence, Springer, Berlin, 2007.

[25] E.A. Gyasi, H. Handroos, P. Kah, Survey on artificial intelligence (AI) applied in welding: a future scenario of the influence of AI on technological, economic, educational and social changes, Proc. Manuf. 38 (2019) 702−714, https://doi.org/10.1016/j.promfg.2020.01.095.

[26] G. Valarmathi, S.U. Suganthi, V. Subashini, R. Janaki, R. Sivasankari, S. Dhanasekar, CNN algorithm for plant classification in deep learning, Mater Today Proc. 46 (2021) 3684−3689, https://doi.org/10.1016/j.matpr.2021.01.847.

[27] R. Gruetzemacher, F.E. Dorner, N. Bernaola-Alvarez, C. Giattino, D. Manheim, Forecasting AI progress: a research agenda, Technol. Forecast Soc. Change 170 (2021) 120909, https://doi.org/10.1016/j.techfore.2021.120909.

[28] T.M. Mitchell, Machine Learning, McGraw-Hill, Singapore, 1997.

[29] A.D.J. van Dijk, G. Kootstra, W. Kruijer, D. de Ridder, Machine learning in plant science and plant breeding, IScience 24 (2021) 101890, https://doi.org/10.1016/j.isci.2020.101890.

[30] H. Zhang, H. Nguyen, D.-A. Vu, X.-N. Bui, B. Pradhan, Forecasting monthly copper price: a comparative study of various machine learning-based methods, Resour. Pol. 73 (2021) 102189, https://doi.org/10.1016/j.resourpol.2021.102189.

[31] L.-L. Li, Z.-F. Liu, M.-L. Tseng, K. Jantarakolica, M.K. Lim, Using enhanced crow search algorithm optimization-extreme learning machine model to forecast short-term wind power, Expert Syst. Appl. 184 (2021) 115579, https://doi.org/10.1016/j.eswa.2021.115579.

[32] K. Maehashi, M. Shintani, Macroeconomic forecasting using factor models and machine learning: an application to Japan, J. Jpn. Int. Econ. 58 (2020) 101104, https://doi.org/10.1016/j.jjie.2020.101104.

[33] D. Borg, G.S. Sestito, M.M. da Silva, Machine-learning classification of environmental conditions inside a tank by analyzing radar curves in industrial level measurements, Flow Meas. Instrum. 79 (2021) 101940, https://doi.org/10.1016/j.flowmeasinst.2021.101940.

[34] P. López-Úbeda, M.C. Díaz-Galiano, T. Martín-Noguerol, A. Luna, L.A. Ureña-López, M.T. Martín-Valdivia, Automatic medical protocol classification using machine learning approaches, Comput. Method Progr. Biomed. 200 (2021) 105939, https://doi.org/10.1016/j.cmpb.2021.105939.

[35] G. Zhao, L. Quan, H. Li, H. Feng, S. Li, S. Zhang, R. Liu, Real-time recognition system of soybean seed full-surface defects based on deep learning, Comput. Electron. Agric. 187 (2021) 106230, https://doi.org/10.1016/j.compag.2021.106230.

[36] I. Pastor-López, B. Sanz, A. Tellaeche, G. Psaila, J.G. de la Puerta, P.G. Bringas, Quality assessment methodology based on machine learning with small datasets: industrial castings defects, Neurocomputing 456 (2021) 622−628, https://doi.org/10.1016/j.neucom.2020.08.094.

[37] A. Niccolai, D. Caputo, L. Chieco, F. Grimaccia, M. Mussetta, Machine learning-based detection technique for NDT in industrial manufacturing, Mathematics 9 (2021) 1251, https://doi.org/10.3390/math9111251.

[38] B. Wu, J. Zhou, H. Yang, Z. Huang, X. Ji, D. Peng, Y. Yin, X. Shen, An ameliorated deep dense convolutional neural network for accurate recognition of casting defects in X-ray

images, Knowl. Base Syst. 226 (2021) 107096, https://doi.org/10.1016/j.knosys.2021.107096.
[39] S. Wolfram, An Elementary Introduction to the Wolfram Language, Wolfram Media Inc., Champaign, 2017.
[40] S.M. Piryonesi, T.E. El-Diraby, Role of data analytics in infrastructure asset management: overcoming data size and quality problems, J. Transp. Eng. B Pvmt. 146 (2020) 121−129, https://doi.org/10.1061/jpeodx.0000175.
[41] R.J. Samworth, Optimal weighted nearest neighbour classifiers, Ann. Stat. 40 (2012) 2733−2763, https://doi.org/10.1214/12-aos1049.
[42] B. Everitt, E. Al, Cluster Analysis, Wiley, London, 2011.
[43] P. Harrington, Machine Learning in Action, Manning Publications, Shelter Island, 2012.
[44] M.N. Murty, V.S. Devi, Pattern Recognition, Springer, London, 2011.
[45] C.M. Bishop, Neural Networks and Machine Learning, Springer, Berlin, 1998.
[46] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, second ed., Springer, New York, 2009.
[47] G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning: with Applications in R, Springer, New York, 2013.
[48] K.K. Thyagharajan, I. Kiruba Raji, A review of visual descriptors and classification techniques used in leaf species identification, Arch. Comput. Method Eng. 26 (2018) 933−960, https://doi.org/10.1007/s11831-018-9266-3.
[49] B. Pratomo, l. Lisnawita, T.C. Nisa, M. Basyuni, Short communication: digital identification approach to characterize *Hevea brasiliensis* leaves, Biodiversitas 22 (2021) 1006−1013, https://doi.org/10.13057/biodiv/d220257.
[50] J. Ahmad, K. Muhammad, I. Ahmad, W. Ahmad, M.L. Smith, L.N. Smith, D.K. Jain, H. Wang, I. Mehmood, Visual features based boosted classification of weeds for real-time selective herbicide sprayer systems, Comput. Ind. 98 (2018) 23−33, https://doi.org/10.1016/j.compind.2018.02.005.
[51] S. Tekkesinoglu, M. Shafry Mohd Rahim, A. Rehman, I. Mat Amin, T. Saba, *Hevea* leaves boundary identification based on morphological transformation and edge detection features, Res. J. Appl. Sci. Eng. Technol. 7 (2014) 2447−2451, https://doi.org/10.19026/rjaset.7.551.
[52] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, Comput. Vis. Patter. Recognit. 6 (2015) 1−14. https://arxiv.org/pdf/1409.1556.pdf.
[53] A. Kamilaris, F.X. Prenafeta-Boldú, Deep learning in agriculture: a survey, Comput. Electron. Agric. 147 (2018) 70−90, https://doi.org/10.1016/j.compag.2018.02.016.