AJAS / SPAS

ISSN 2316-0861

**African Journal of Applied Statistics**
Vol. 10 (1), 2023, pages 1399 - 1418.
DOI: http://dx.doi.org/10.16929/ajas/2023.1399.274

# Empirical Performance of *CART*, *C5.0* and Random Forest Classification Algorithms for Decision Trees.

**Bissilimou Rachidatou Orounla** $^{(1),\star}$ **, Akoeugnigan Idelphonse Sode** $^{(1)}$**, Kolawolé Valère Salako** $^{(1)}$ **and Romain Glèlè KaKaï** $^{(1)}$

$^{(1)}$Laboratoire de Biomathématiques et d'Estimations Forestières, Faculty of Agronomic Sciences, University of Abomey-Calavi, Abomey-Calavi (Benin).

Received on August, 2023; Accepted on September 15; Published on October 19, 2023

**Abstract.** This study compares the performance of *CART*, *C5.0* and Random Forest (*RF*) algorithms. 25 continuous predictors and 25 factors were simulated using a population size of 10,000. Based on this data, sample data were generated by varying the number of predictors, the proportion of categorical versus continuous predictors and the sample size. The performance of the tree algorithms increases with sample size and the number of variables, but for *RF*, it is highly greater than the one of *CART* and *C5.0*. Irrespective of the algorithms, the performance decreases when there are more categorical variables than continuous variables.

**Key words:** accuracy; categorical variables; specificity; non-parametric modeling; simulation.
**AMS 2010 Mathematics Subject Classification Objects :** 97R40; 65Y20; 68Q25.

$^{*}$ Corresponding author: Bissilimou Rachidatou Orounla (rachobis@gmail.com)
Akoeugnigan Idelphonse Sode: sdidelphonse@gmail.com
Kolawolé Valère Salako: salakovalere@gmail.com
Romain Glèlè KaKaï: gleleromain@gmail.com

**Résumé.** (French Abstract) La présente étude compare les performances des algorithmes *CART*, *C5.0* et Random Forest (*RF*). 25 prédicteurs continus et 25 facteurs ont été simulés à partir d'une population de taille 10000. Sur la base de ces données, des échantillons ont été générés en faisant varier le nombre de prédicteurs, la proportion de prédicteurs catégoriels par rapport aux prédicteurs continus et la taille de l'échantillon. La performance des algorithmes augmente avec la taille de l'échantillon et le nombre de variables. Celle de *RF* est nettement supérieure à celle de *CART* et de *C5.0*. Indépendamment des algorithmes, la performance diminue lorsqu'il y a plus de variables catégorielles que de variables continues.

**The authors**.

**Bissilimou Rachidatou Orounla**, M.Sc. in Biostatistics, is preparing a PhD thesis in Biometry at; Laboratoire de Biomathématiques et d'Estimations Forestières, Faculty of Agronomic Sciences, University of Abomey-Calavi (UAC), under the supervision of the fourth author.

**Akoeugnigan Idelphonse Sode**, Msc in Biostatistics, is a Ph.D. Student in Biometry at the: Laboratoire de Biomathématiques et d'Estimations Forestières, Faculty of Agronomic Sciences, University of Abomey-Calavi (UAC), under the supervision of the fourth author.

**Kolawolé Valère Salako**, Ph.D., in Plant Ecology and Conservation, is an agronomist, forester and biostatistician at the: aboratoire de Biomathématiques et d'Estimations Forestières, Faculty of Agronomic Sciences, University of Abomey-Calavi (UAC).

**Romain Glèlè Kakaï**, Ph.D. in Biometry, is a full professor in Biometry and Forest estimations at: Faculty of Agronomic Sciences, University of Abomey-Calavi (UAC). He is the coordinator of Master's Program in Biostatistics and doctoral studies in Biometry. He is also the director of the Laboratoire de Biomathématiques et d'Estimations Forestières.

## 1. Introduction

There is an increasing trend in the use of supervised machine learning (*SML*) to build a concise model of the distribution of class labels in terms of predictor features Kotsiantis, 2007. Many supervised machine learning algorithms have been developed including Decision Trees (*DT*), Neural Networks, Naive Bayes, k-Nearest Neighbors (*KNN*), Support Vector Machine (*SVM*). During the past decades, several algorithms such as statistical classifier, neural network classifier, syntactic classifier and tree-based classifier Zhang et al., 2014 have been proposed for solving real world classification and clustering problems Farid et al., 2014, Liao et al., 2012, Ngai et al., 2009.

Decision trees (*DT*) constitute ones of the most popular methods for classification in various data mining applications and assist the process of decision-making Han et al., 2006. Decision tree algorithms are used to establish non-linear relationship between predictive factors and outcomes as well as for mixed data types (i.e. numerical and categorical). In small and large datasets, classification and regression trees are becoming increasingly popular for partitioning data and identifying local structure Kotsiantis, 2007. Classification trees include those models in which the dependent variable (the predicted variable) is categorical while Regression trees include models in which it is continuous. It provides a modeling technique that is easy for human to understand and simplifies the classification process. The most common classification tree algorithms are Classification And Regression Tree (*CART*) Breiman, 2017, Random Forest (*RF*) Breiman, 2001 and *C5.0*/*C4.5* Bujlow et al., 2012, Salzberg, 1994 since they perform well in terms of execution time, classification accuracy and frequency of use Anyanwu and Shiva, 2009, Sharma and Srivastava, 2016.

These techniques have received a great attention under various aspects during the last decades. Indeed, Ali et al., 2012 compared Random Forest (*RF*) and *C4.5* using the following parameters: correctly classified instances, incorrectly classified instances, F-Measure, Precision, Accuracy and sensitivity. Zhang, 2016 used the air quality data (continuous response) to compare *RF* and *CART* and introduced R functions to perform model based on recursive partition. Moreover, Miller et al., 2016 introduced a multivariate extension to a decision tree ensemble method called Gradient Boosted Regression Trees for finding and interpreting structure in data sets with multiple outcomes and many predictors. Recently, some researchers compared the performance of Classification trees algorithms with generalized linear models (*GLM*s) and some of their extensions. For instance, Jeune et al., 2018 compared the Multinomial Logistic Regression (*MLR*) with Random Forest (*RF*) in the classification of the soil types and found that the classification performance was moderate for both algorithms Based on the Kappa values and *RF* classifier outperformed *MLR* in the validation process.

Though there have been many works which focused on classification trees based on either ensembles trees methods or single tree methods, few studies have simultaneously compared the performance of *CART*, *C5.0* and *RF* algorithms under some conditions related to the sample size, the number of variables and the proportion of variables type. Indeed, the effect of sample size on model accuracy is an aspect that is often overlooked Wisz et al., 2008. The classification performance is known to rapidly decrease for sample sizes smaller than 15 records Papeş and Gaubert, 2007, and become dramatically poor for samples sizes smaller than 5 records Pearson et al., 2007. Schratz et al., 2018 illustrated how the machine learning modeling methods including *RF* can be affected by an uneven distribution of the binary response variable, sample size and the number and types of predictors (numeric as well as nominal), the influence of spatial autocorrelation and predictors derived from various sources. Indeed, the proportion of continuous versus categorical variables and the number of predictors constitute some aspects which may impact the quality of classification results. Taking those aspects into account during the classification process may help in the choice of the algorithm to be used considering the configurations and properties of the available data set.

What is the performance of *CART*, *C5.0* and *RF* with multiple outcomes when the number and the proportion of type of predictors change? What is the behavior of these three classification algorithms when the sample size increases? There are some research questions we attempted to answer in this papers.

The paper is organized as follow: the Section 2 recalls the methodology of the study and in Section 3 the main results. We provide our formal analysis (discussion) in Section 4 and concluded the paper in section 5.

## 2. Materials and Methods

### *2.1. Description of selected Algorithms*

### 2.1.1. *CART* algorithm

The *CART* (Classification and regression trees) was introduced by Breiman, 2017. It builds both classification and regression trees. The classification tree construction by *CART* is based on the binary splitting of the data attributes. This algorithm uses the Gini index splitting measure to select the splitting attribute, and the pruning is done using the cross-validation technique Khoshgoftaar and Seliya, 2004. *CART* uses both numeric and categorical attributes to build the decision tree Lewis, 2000. The classification tree subdivides the training data set space into multiple classes (leaves). Each class consists of a set of rules that splits the decision variable spaces Yang et al., 2016. The *CART* Algorithm for DT can be described as follows Lavanya and Rani, 2012:

- Tree building using recursive splitting of nodes:
  1. Selection of splitting attribute: For S attributes, there will be a total of S splits to consider. Find each attribute that takes the best split using a goodness of split

  $$\Delta I(S,T) = I(t) - P_1 I(t_1) - P_2 I(t_2) \tag{1}$$

  $P_1$ and $P_2$ are the probability of the instances of $t$ that go into $t_1$ and $t_2$ respectively. $I(t)$ is the impurity defined as:

  $$\begin{aligned} I(t) &= -\sum_{i\neq j} P(w_i)P(w_j) \\ &= 1 - \sum_{j} P^2(w_j) \end{aligned} \tag{2}$$

  With $P(w_j)$ the conditional probability of class $j$ in $S$
  2. Decide the node to represent a terminal node or to continue splitting the node.
- Stopping the tree-building process when the maximal tree has been produced.;
- Tree pruning: this algorithm used a cross-validation method.
  1. Divide all training data into N disjoint subsets, $R = R_1, R_2, ...., R_N$
  2. For each $j = 1, ....., N$ do
  Test $set = R_j$
  Training $set = R - R_j$
  Using the Training set, Compute the decision tree.
  Decide the performance accuracy $X_j$ with the use of the test set.
- Optimal tree selection: Choose the tree which does not overfit the information but fits this information well in the learning dataset.

### 2.1.2. *C5.0* algorithm

The *C5.0* algorithm is a new generation of Machine Learning (ML) algorithms based on decision trees. It follows the rules of the C4.5 algorithm, which follows the rules of the ID3 algorithm. The *C5.0* classifier was developed as an improved version of a well-known and widely used C4.5 classifier. It has several important advantages, including acknowledging noise and missing data and the error pruning is solved by the *C5.0* algorithm Pandya and Pandya, 2015. This algorithm can offer a powerful boosting method to increase the accuracy of the classification process. The building of a classification tree using the *C5.0* algorithm can be described as follows Pandya and Pandya, 2015:

- Create a root node,
- Check the base case,
- Construct a decision tree using training data,
- Apply cross-validation technique:
  1. Divide all training data into N disjoint subsets, $R = R_1, R_2, ...., R_N$
  2. For each $j = 1, ....., N$ do
  Test $set = R_j$
  Training $set = R - R_j$
  Using the Training set, Compute the decision tree.
  Decide the performance accuracy $X_j$ with the use of the test set.
  3. Reckon the N-fold cross-validation technique to estimate the performance
- Apply Reduced Error Pruning technique: 1. Find the attribute with the highest info gain 2. Classification: for each $t_j \in \mathbb{D}$, apply the $DT$ to determine its class.

### 2.1.3. Random Forest algorithm

Random Forest (*RF*) was developed by Breiman, 2001. It is a group of un-pruned classification or regression trees made from the random selection of training data samples. In *RF*, every decision tree is made by randomly selecting data from available data. For example, a Random Forest for each decision tree can be built by randomly sampling a feature subset and/or by randomly sampling a training data subset for each decision tree. Each tree is grown as described by Ali et al., 2012:

- Sampling N randomly if the number of cases in the training set is N with replacement from the original data.
- For $M$ number of input variables, the number of variables m is selected ($m << M$ is specified at each node), $m$ variables are chosen randomly from the $M$, and the best split on these $m$ variables is used for splitting the node. During the forest growing, the value of $m$ is held constant.
- Each tree is grown to the largest possible extent, and no pruning is used.

## *2.2. Simulation design*

Let's consider a categorical response variable $Y$ with classes $m = 1, ..., M$ and $X(n, p)$ of predictors. To simulate the dataset, fifty predictors were generated with twenty-five continuous and twenty-five categorical predictors. This setting was considered due to the levels of the proportion of categorical versus continuous variables in the design. Fifteen continuous predictors were independently generated with multivariate normal distribution with mean vector 0 and identity covariance matrix Miller et al., 2016, while the other ten continuous variables were generated with uniform distribution Ye and Lord, 2014. The population size was assumed to be very large ($N = 10,000$). Twenty-four binary predictors were generated with Bernoulli distribution where the probability of each case equals 0.5 Steingrimsson and Yang, 2018, and one factor with three levels was also considered.

A multinomial logistic model was used to generate the categorical dependent variable $Y$ (with three classes) based on the matrix of independent variables ($X$) generated in the previous step El-Habil, 2012. This model is one of the most commonly used models for analyzing categorical data with more than two levels El-Habil, 2012. For a multinomial model, any class of the response variable can be taken as the reference category El-Habil, 2012. As such, we considered category one as the reference level. To find the relationship between the probability of each class and explanatory variables, we defined the multinomial logistic model in which the log-odds of each individual have a linear link with the predictors given by Biau et al., 2008, El-Habil, 2012:

$$\begin{aligned}\log(\frac{\pi_m(X)}{\pi_1(X)}) &= \alpha_m + X\beta_m \\ &= Z_m\end{aligned} \tag{3}$$

where $X$ is the matrix of predictors, $\pi_m$ is the probability of occurrence of class $m$ and $\beta_m$ the predictors' weights for individuals in class $m$. The probabilities of each class are calculated as follows: for $m = 2, ..., M$ and $i = 1, ..., N$,

$$\pi_m(X) = p(y_i = m|X) = \frac{\exp(Z_{mi})}{1 + \sum_{h=2}^{M} \exp(Z_{hi})} \tag{4}$$

and for the reference category $(m = 1)$,

$$\pi_1(X) = p(y_i = 1|X) = \frac{1}{1 + \sum_{h=2}^{M} \exp(Z_{hi})} \tag{5}$$

with $\sum_{m=1}^{M} \pi_m(X) = 1$, since the denominator acts as a normalizing constant.

The three-level categorical predictor implied two dummy variables in the design matrix El-Habil, 2012. Since all predictors $X$ cannot contribute to generating the true response variable $Y$, the design matrix was constructed with the two dummy variables from the 3-levels factor and forty-six predictors; three variables (one binary and two numeric variables) were excluded. The vectors $\beta_2$ and $\beta_3$ representing the predictors' weights for classes 2 and 3, respectively, are generated from a standard normal distribution Benoit, 2012. The intercepts $\alpha_2$ and $\alpha_3$ were set to 2.5 and 0.2 for classes 2 and 3, respectively, while $\alpha_1 = 0$ (for the reference category). The individual outcomes $Y_i$ were randomly and independently sampled from the population where $Y_i$ has a multinomial distribution with probability defined for each category Williams, 2016. This resulted in the following proportions for the three classes: 21.43 %, 41.12 % and 37.45 % for classes 1, 2 and 3, respectively.

### *2.3. Sampling schemes*

Factors considered were the number of predictors, the proportion of categorical versus continuous variables within the predictors and the sample size. Three ratios (0.25, 0.5, 0.75) of categorical versus continuous variables with a varying number of predictors (4, 8, 12, 16, 20, 24, 28 and 32) were considered. For example, when the proportion of predictors' type is 0.25, it means that 25 % of the variables are categorical versus 75 % which are continuous in the resulting subset. In contrast, when the ratio is 0.75, it means that 75 % of variables are categorical versus 25% which are continuous variables. To obtain the observed data, a random sample of varying sizes (50, 100, 200, 500, 800 and 1000) taken from each subset was considered. The combination of the levels of the different factors resulted in 144 settings of datasets.

Each setting of the dataset was repeated 200 times Kuhn, 2008, and each dataset was split into training (80 % of data) and test (20 %) sets. The three algorithms (*CART*, *C5.0* and *RF*) were used for building different models of classification using respectively rpart Therneau and Atkinson, 2018, *C5.0* Kuhn and Quinlan, 2015 and Random Forest Liaw and Wiener, 2002 R packages. For the *RF* models, the total number of trees to grow was kept to the default value (i.e. ntree = 500). The combination of different settings and replicates resulted in 86,400 models. Fig. 1 summarizes the general workflow of data simulation and processing.

To evaluate the performance of the algorithms, metrics such as accuracy, sensitivity, specificity, and kappa statistics were computed during the classification using the R package caret Kuhn and Quinlan, 2015. The datasets of evaluation metrics were analyzed in the statistical software R version 3.5.2 R Core Team, 2018. Means and standard errors of the performance metrics were calculated and used to evaluate the performance of the three algorithms by observing their variations with respect to the different factor levels. Pairwise multiple comparisons and the Bonferroni method for adjusting p-values from the Agricolae R package de Mendiburu, 2019 were used to classify the mean values of the performance metrics among the algorithms. The formulas to compute different evaluation metrics are defined as follows:

- The Accuracy is the percentage of predictions that are correct and given by:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

- The Sensitivity is the percentage of positive instances that are predicted as positive and given by:

$$Sensitivity = \frac{TP}{TP + FN} \tag{7}$$

- The Specificity is the percentage of negative instances that are predicted as negative and expressed by:

$$Specificity = \frac{TN}{TN + FP} \tag{8}$$

- The Kappa statistic is the measure of agreement relative to what would be expected by chance and expressed by:

$$Kappa = \frac{P_o - P_e}{1 - P_e} \tag{9}$$

where

$$P_o = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

and

$$P_e = P_{yes} + P_{no} \tag{11}$$

with

$$P_{yes} = \frac{(TP + FP)(TP + FN)}{(TP + TN + FP + FN)^2} \tag{12}$$

and

$$P_{no} = \frac{(TN + FP)(TN + FN)}{(TP + TN + FP + FN)^2} \tag{13}$$

Where $P_o$ and $P_e$ are the relative observed agreement (or the accuracy) and the hypothetical probability of chance agreement, respectively. In all these formulas, TP, TN, FP and FN denote the true positives, true negatives, false positives and false negatives, respectively.

### *2.4. Application with cattle breeding dataset*

In this case study, the dataset was collected on feeding strategy and food resources management in Cattle breeding of Nikki, Kalalé and N'Dali districts in northern Benin, with 329 subjects and 48 variables. Three algorithms (*RF*, *C5.0* and *CART*) were applied to this data. Random Forest was used to predict the subjects' classes. However, to obtain the response variable (i.e. the vector of subjects classes), Factorial Analysis on Mixed Data (FAMD) was used, and 25 principal components (PCs) explaining 75 % of the total information in the initial matrix were chosen. Hierarchical Clustering on Principal Components (HCPC) was applied to FAMD results (i.e. the 25 PCs). Then, three classes were obtained, and 34 predictors (23 categorical and 11 continuous variables, i.e. 67,6 % of categorical vs continuous) contributed to building the subjects' classes (i.e. the different clusters).

The distribution of the three class subjects was taken as the response variable. According to Breiman, 2017, a single train and test partitions are not reliable estimators of the true error rate of a classification scheme on a limited dataset. So, five-fold cross-validation was applied to the dataset to obtain its partition into five training/test sets. The cross-validation sampling technique splits the dataset into five folds: one fold serves as a test set and the other parts as training sets. The

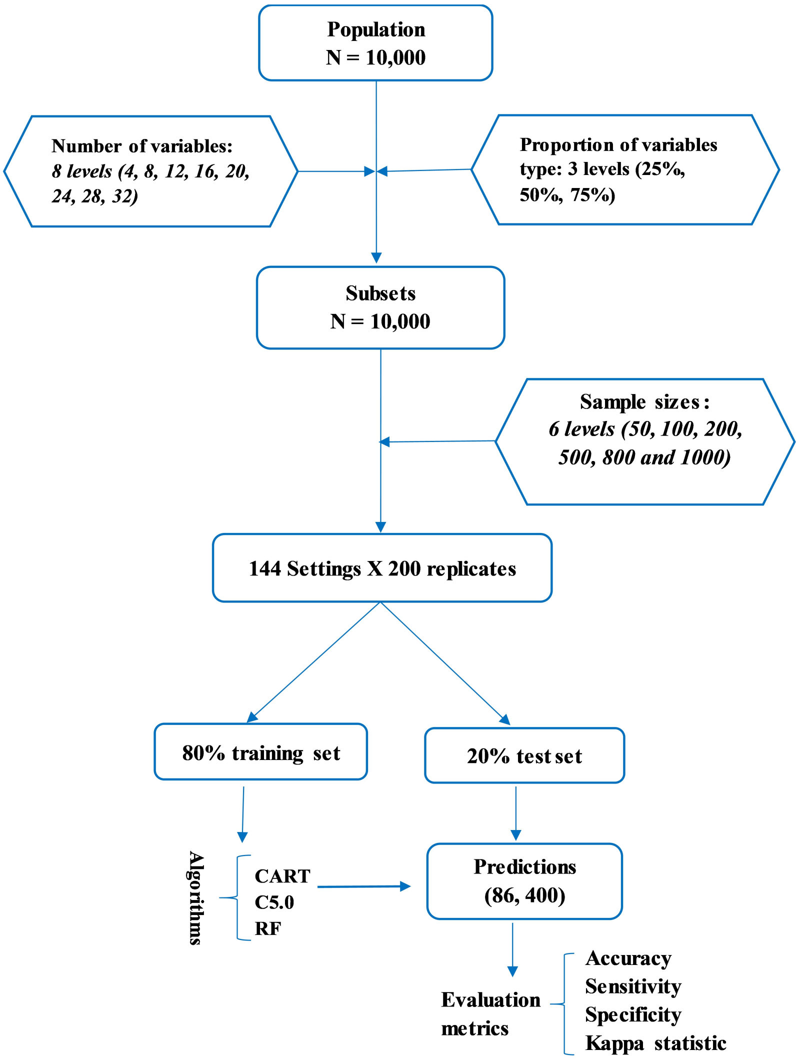

**Fig. 1.** General workflow of the simulation design.

process was repeated five times such that each fold was used as a test set. This ensures that the approximate proportion of each class remains 80 % in the training set and 20 % in the test set Jeune et al., 2018. The metrics such as accuracy, sensitivity, specificity, and kappa statistics were computed and averaged over the 5-fold estimates.

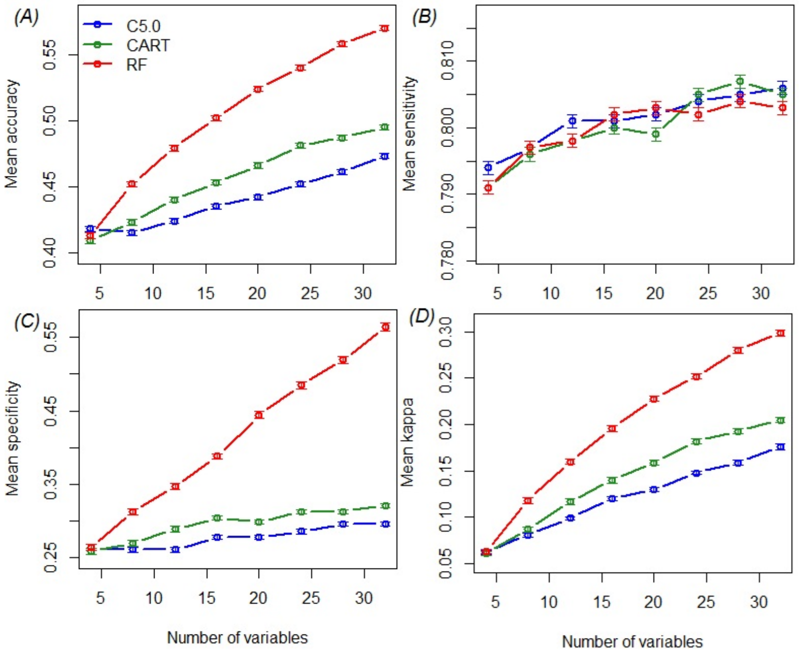

**Fig. 2.** : Performance of three algorithms related to the number of variables. (**A**) accuracy values averaged across the sample size, the proportion of variables type and replicates, (**B**) sensitivity values averaged across the sample sizes, the proportion of variables type and replicates, (**C**) specificity values averaged across the sample sizes, the proportion of variables type and replicates and (**D**) kappa values averaged across the sample sizes, the proportion of variables type and replicates. Error bars represent the standard error of each mean. Orange, green, and blue lines represent *RF*, *CART* and *C5.0* algorithms, respectively.

## 3. Results

### *3.1. Performance of algorithms in terms of the number of predictors*

The mean accuracy ranges between 40 % (for four predictors) and 57 % (for 32 predictors) (figure 2A). The mean accuracy increases with the number of predictors for the three algorithms. However, *RF* estimates were higher than those of *CART* and *C5.0* (figure 2A), with *C5.0* having the lowest values. The standard error of the mean was very low (0.005 is the maximum value) for all three models. The sensitivity estimates ranged from 79 % to 81 % (Picture **B** in Fig. 2). The sensitivity values were statistically the same for all three algorithms (Picture **B** in Fig. 2). The specificity (false negative rate) for *RF* increases almost linearly with the number of variables. In contrast, it grew very slowly for *CART* and *C5.0* and stabilized when the number of variables reached 32 (figure 2C). The mean kappa statistic ranged from 6 % to 30 % and increased with the number of predictors for all three algorithms (figure 2D). *RF* estimates grow more rapidly than for *CART* and *C5.0*.

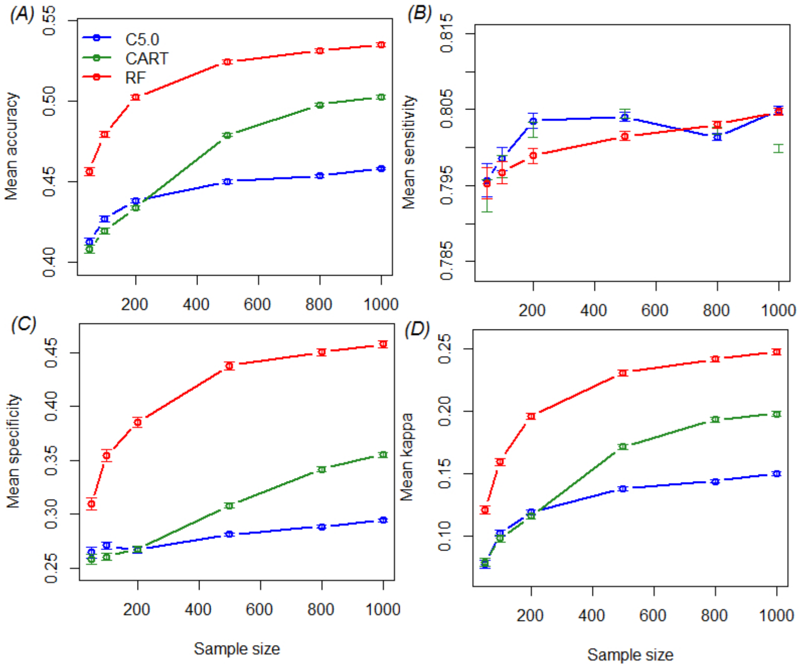

**Fig. 3.** : Performance of three algorithms in relationship with sample size. (**A**) accuracy values averaged across the number of variables, the proportion of variables type and replicates, (**B**) sensitivity values averaged across the number of variables, the proportion of variables type and replicates, (**C**) specificity values averaged across the number of variables, the proportion of variables type and replicates and (**D**) kappa values averaged across the number of variables, the proportion of variables type and replicates. Error bars represent standard errors of means. Orange, green, and blue lines represent *RF*, *CART* and *C5.0* algorithms

### *3.2. Performance of algorithms in terms of sample size*

The mean accuracy ranges from 45 % (for n = 50) to 55 % (for n = 1000) (Picture **A** in Fig. 3), whereas the mean specificity varies between 25 % and 46 % (Picture **C** in Fig. 3), and the mean kappa statistic varies between 6 % and 25 % (Picture **D** in Fig. 3). Moreover, the mean values of accuracy, specificity and kappa statistic increased rapidly, with the sample size for the *RF* and *CART* algorithms. However, *C5.0* estimates for the specificity were almost constant with increasing sample size (Picture **C** in Fig. 3). *RF* estimates were higher than the ones of *CART* and *C5.0* for accuracy, kappa statistic and specificity, while these parameter estimates were low in the *C5.0* model. The SE estimates were very low (0.0005 -0.0055) for the three algorithms. The sensitivity estimates ranged from 79 % to 81 % (Picture **B** in Fig. 3). We observed that, on average, sensitivity values were the same according to the three algorithms. Accordingly, there was no significant difference between the algorithms Bulleted lists look like this:

### *3.3. Performance of algorithms in terms of proportion of categorical versus continuous variables*

The mean accuracy estimates were between 42 % and 55 % (Picture **A** in Fig. 4), whereas the mean specificity ranged from 27 % to 46 % (Picture **C** in Fig. 4). For *kappa* statistics, estimates were between 9 % and 25 % (Picture **D** in Fig. 4). When the number of categorical variables was lower than that of continuous variables (i.e., the proportion of 0.25), the accuracy and kappa statistic values were high for all three algorithms. In contrast, when the number of categorical variables is greater than the number of continuous variables (i.e. the proportion of 0.75), the accuracy and kappa statistic values become lower for the three algorithms. However, for the specificity, *RF* estimates were higher than those of *CART* and *C5.0*, which were relatively invariant across the proportion of the predictor's type. Regarding the sensitivity, all the estimates were the same for the three algorithms regardless of the predictor's type. They ranged between 79 % and 81 %. The values of the performance parameters were the same for all algorithms irrespective of the proportion of the predictor's type specified (Picture **B** in Fig. 4). For all algorithms, the standard errors were also very low (0.0008-0.003) for all parameters.

### *3.4. Application results*

The Hierarchical Clustering on Principal Components (*HCPC*) applied to the *FAMD* results showed that the cattle breeders can be subdivided into three classes. Some variables were not important during the clustering step and have been removed. The breeders of the first class were all from Kalalé (9.12 %) district. They feed their cattle with shrubs chosen according to whether they have good regrowth and can be found everywhere. When the shrubs also have good biomass and good growth, they choose them to feed their cattle. They also practice fallow to graze their animals and food crops like millet, yam, corn, soy and beans. The breeders of the second class were those from Kalalé (23.71%) and N'Dali (32.83 %) districts. These breeders feed their animals with shrubs chosen according to whether they have good growth, resist better to the dry season, and can be found everywhere. They practice food crops such as maize, yams, cassava, beans, and groundnuts. The breeders of the third class were all from Nikki (24.62 %), who feed their cattle with shrubs chosen according to whether they have good regrowth and can be found everywhere. These shrubs must also have good biomass and good growth. The breeders also practice fallow to graze their animals and food-producing crops such as maize, yams, soy, beans, cassava, cotton and groundnuts. The important variables with a high mean accuracy value (15 % and 25 %) in the model fitted to the cattle dataset were district, growth criteria, cultivated areas for cotton and yam, and waste sale. Also, the results showed that the breeders of classes 2 and 3 were predicted with low error rate than those of class 1.

Applying the three algorithms to the data showed that *RF* outperformed *CART* and *C5.0* for all performance criteria except sensitivity. The results showed that the value of mean accuracy is 0.9635 ± 0.0038, 0.9333 ± 0.0071, and 0.9182 ±

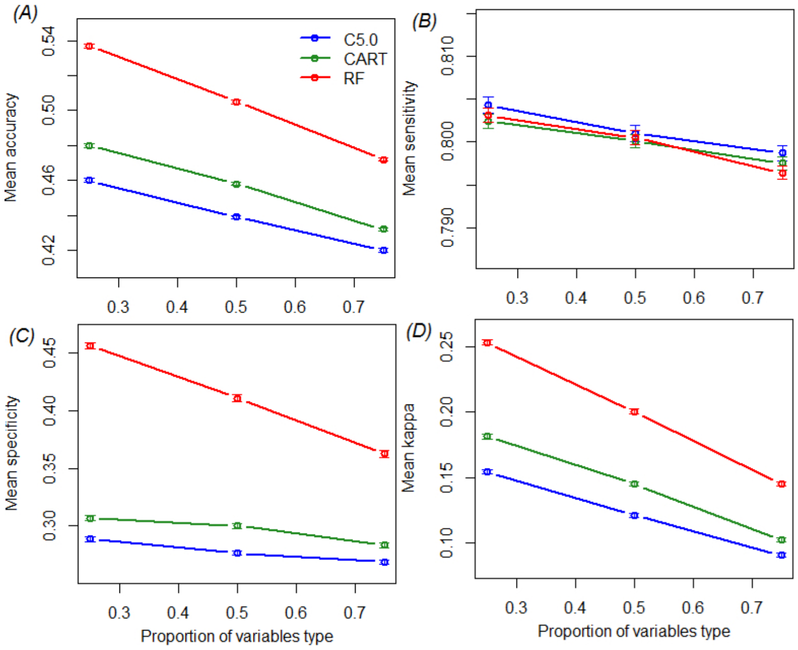

**Fig. 4.** Performance of three algorithms in relationship with the proportion of predictor's types.
(**A**) accuracy values averaged across the number of variables, sample sizes and replicates, (**B**) sensitivity values averaged across the number of variables, sample sizes and replicates, (**C**) specificity values averaged across the number of variables, sample sizes and replicates and (**D**) kappa values averaged across the number of variable sample sizes and replicates. The error bar represents the standard error of each mean. Orange, green, and blue lines represent *RF*, *CART* and *C5.0* algorithms.

0.0161 for Random Forest, *CART* and *C5.0*, respectively. So, by using *RF*, 96.34 % of predictions are correct. However, 99 ± 0.6 % (sensitivity) of cattle breeders are classified correctly within classes 2 and 3, and 100 ± 0 % (specificity) are classified in class 1. The value 0.951 ± 0.02 of the kappa statistic indicates a perfect agreement between the predicted classes and the observations in the test dataset. The standard errors were very low, indicating a good precision of estimates.

## 4. Discussion

### *4.1. Merits and limitations*

We used simulations to evaluate the relative performance of classification algorithms such as *CART*, *C5.0* and Random Forest. The different simulation settings we proposed allowed us to mimic some case studies that can happen in real-world situations. Also, the simulations allowed us to choose the different values of intercepts for the desired proportion of different classes of the response variable.

Standard normal and uniform distributions were assumed to generate continuous predictors. These are some scenarios that we could have in practice. However, in a real-world situation, the predictors can also have other distributions such as chi-square, Poisson, negative binomial, or log-normal distribution, which we did not consider in this study. Furthermore, apart from the linear effect of predictors on the response variable, quadratic or interaction effects on the response are also plausible. Still, these cases were not considered in this study since the number of parameters of the simulated model will become high, and the interpretation could be complicated Miller et al., 2016. Models such as classification or regression trees might be resistant to highly correlated predictors, but multicollinearity may negatively affect the interpretability of the model Kuhn, 2008. The predictors that are more correlated can be identified and removed.

### *4.2. Performance of algorithms*

The classification results from *CART*, *C5.0* and *RF* algorithms revealed how well the algorithms perform according to the sample size, the number of variables and the proportion of the predictor's type. The performance of the algorithm is improved when the number of variables and sample size increase. However, *RF* performs better than the other algorithms in general. This better performance of *RF* may be due to its predictive efficiency Ali et al., 2012, its ability to make feature selections and its non-parametric property for various types of datasets Qi, 2012. Accurate predictions and better generalizations are achieved using ensemble strategies and a random sampling Qi, 2012. This generalization property is due to the bagging scheme, which improves the method by decreasing the variance, while similar techniques like boosting achieve this by reducing the bias Breiman, 1996, Lavanya and Rani, 2012. The feature selection used by *RF* provides accurate predictions on many applications and can measure the importance of each feature with model training. Unlike classical decision trees, there is no need to prune trees in *RF* since the ensemble and bootstrapping schemes allow *RF* to overcome overfitting issues Qi, 2012. *RF* performs feature selection while classification rules are built, increasing its use for variable selection (e.g., selecting a subset of genetic markers relevant for predicting a certain disease) Lavanya and Rani, 2012, Qi, 2012. An empirical comparison conducted by Caruana and Niculescu-Mizil, 2006 also showed that the Random Forest algorithm achieved excellent performances compared to numerous other supervised learning algorithms. Also, *RF* generally exhibits a significant performance improvement compared to single tree classifiers such as C4.5 Ali et al., 2012, Dahinden, 2011. From the simulation results, the slight variation (approximately 1 % - 2 %) of the sensitivity for the three algorithms means that the subjects within classes 2 and 3 were correctly classified by the three models, regardless of the number of predictors and the sample size of the data. However, the best performance of *RF* in terms of specificity means that the first class of subjects was correctly predicted. Regarding the kappa statistic, there is a good association between the observed and forecast data for the three algorithms. Likewise, in terms of sample size, a dataset with 500 subjects allows *RF* to perform more or less optimally. However, *CART*'s performance over *C5.0* may be linked to its computational efficiency Field Lewis, 2000; in other words, it could be due to the Gini index used for selecting the best predictor attribute and the measure of the impurity of a node.

From the case study, Random Forest was the best method and classified almost all the breeders in their different classes well. This method has shown to be a powerful statistical classifier in many other fields, such as computational ecology; computational drug screening, where panels of cell lines are used to test drug candidates for their ability to inhibit proliferation Riddick et al., 2011. Cutler et al., 2007 compared the accuracies of *RF* with those of four other commonly-used statistical classifiers on three different ecological datasets. *RF* showed high classification accuracy in all three applications. The comparison of the performance of the Multinomial Logistic Regression (*MLR*) with Random Forest (*RF*) in the classification of the soil types showed that the *RF* classifier outperformed *MLR* in the validation process (Kappa values were 0.33 and 0.55, respectively) Jeune et al., 2018. These values of the Kappa statistic were very low compared to the one obtained from our case study, which was 0.951. The fluctuating trend of the error rate of the out-of-bag (OOB) or feature permutation importance Qi, 2012 given by *RF* revealed that the estimation of accuracy loss varies when the number of trees increases. Despite the high performance of Random Forest, it presents some limitations, such as the large computation time and the interpretability issue, especially in the case of multiple outcome variables Miller et al., 2016. In general, decision tree ensembles exchange interpretability for the prediction performance Miller et al., 2016. The computation time is also important for evaluating different classification models Zhang et al., 2016. A slow statistical procedure can be a great challenge in the case of *big data*.

One of the main aims of modeling is to estimate some metrics without bias or with minimal error. Thus, the low standard error observed for the mean estimates of evaluation metrics means that the predictive performance of the different algorithms is stable, probably because of the properties of the machine learning methods in general. Model estimates with high prediction could lead to good conclusions and effective and reliable decision-making afterwards.

### *4.3. Practical implications and suggestions for further research*

In most cases, the results from the ensemble tree model (*RF*) were better than those from the individual tree models (*CART* and *C5.0*) Dahinden, 2011 in terms of accuracy, specificity and kappa statistic. Further accuracy improvement involves a good model that performs well in predictions. A classification model with 100 % sensitivity means that all the trees with the disease are correctly identified. High sensitivity is crucial when the study identifies a severe but treatable disease (e.g., an attack on plants). However, Kappa is an excellent performance measure when the classes are highly unbalanced Kuhn, 2008. The importance of the cost associated with an incorrect classification could vary according to the field of application of the classification tree method and the evaluation metric applied in the study Hassouna et al., 2016. Extending this study to the comparison of *RF*, Gradient Boosted Regression Tree, *CART* combined with shuffled cross-validation scheme Yang et al., 2016 and possibly the Bayesian classification, i.e. Naïve Bayes algorithm Farid et al., 2014, according to the same metrics and features studied here could be considered for further evaluation. The effect of interactions and collinearities between predictors, as well as the variation of tree complexity and learning rate, on the predictive performance of these algorithms are additional issues not covered in this study, but that should be further investigated.

## 5. Conclusion

We assessed the performance of two single tree and one ensemble tree methods according to the number of predictors, the sample size and the proportion of categorical vs continuous variables. The performance of the three algorithms was improved when the number of variables and sample size increased, but *RF* showed an excellent performance compared to *CART*, which was, in turn, better than *C5.0*. When the number of continuous variables in a dataset is much larger than the categorical variables, we can expect a good performance of algorithms in terms of accuracy and agreement between the observed and predicted classes for all algorithms. For the three algorithms, the estimates' precision was very good, which may be due to the properties of machine learning methods in general. Applying the three algorithms showed that *RF* remains the best classification method for decision trees. The result of *RF* on the real dataset showing three categories of cattle breeders revealed that *RF* classified well the different categories of cattle breeders identified in Northern Benin. We suggest using the *RF* method for classification problems to gain significant performance and accurate predictions.

**Contributions:** OBR, SAI, SKV, and GKR conceived and designed the study. OBR and SAI designed the simulation plan and performed data analysis. OBR wrote the first draft of the manuscript. OBR, SAI, SKV, and GKLR gave input to the data analysis and contributed to the manuscript's writing. All authors provided substantial input to the writing of the manuscript and critically reviewed it for final approval.

**Acknowledgments:** The first author thanks the Regional Universities Forum for Capacity Building in Agriculture (RUFORUM) for funding her tuition fees during the two years of the Master's program.

We would also like to thank the members of the Editorial Board and the anonymous referees for taking the necessary time and effort to review the manuscript. We sincerely appreciate all your valuable comments and suggestions, which helped us in improving the quality of the manuscript.

## References

Ali, J., Khan, R., Ahmad, N., and Maqsood, I. (2012). Random Forests and Decision Trees. *International Journal of Computer Science*, 9(5):7.

Anyanwu, M. and Shiva, S. (2009). Comparative Analysis of Serial Decision Tree Classification Algorithms. *International Journal of Computer Science and Security*, 3(3).

Benoit, K. (2012). Multinomial and Ordinal Logistic Regression. *Lecture notes ME104 Linear Regression Analysis, London School of Economics and Political Science*.

Biau, D. J., Kernéis, S., and Porcher, R. (2008). Statistics in Brief: The Importance of Sample Size in the Planning and Interpretation of Medical Research. *Clinical Orthopaedics and Related Research*, 466(9):2282.

Breiman, L. (1996). Bagging predictors. volume 24, pages 123–140. Machine Learning.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32.

Breiman, L. (2017). *Classification and Regression Trees*. Routledge, New York, 1 edition.

Bujlow, T., Riaz, M., and Pedersen, J. (2012). A method for classification of network traffic based on C5.0 machine learning algorithm. pages pp. 237–241. In Proc. of the International Conference on Computing, Networking and Communications (ICNC).

Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pages 161–168. ACM Press. Place: Pittsburgh, Pennsylvania.

Cutler, D., Edwards, J., Beard, K., and Cutler, A. (2007). Random forests for classification in ecology. *Ecology*, 11(88):2783–2792.

Dahinden, C. (2011). An improved Random Forests approach with application to the performance prediction challenge datasets. *Hands-on Pattern Recognition, Challenges in Machine Learning*, 1(2):223–30.

de Mendiburu, F. (2019). agricolae: Statistical Procedures for Agricultural Research. R package version 1.3-0.

El-Habil, A. M. (2012). An Application on Multinomial Logistic Regression Model. *Pakistan Journal of Statistics and Operation Research*, 8(2):271–291.

Farid, D. M., Zhang, L., Rahman, C. M., Hossain, M., and Strachan, R. (2014). Hybrid decision tree and naïve Bayes classifiers for multi-class classification tasks. *Expert Systems with Applications*, 41(4):1937–1946.

Han, J., Kamber, M., and Pei, J. (2006). *Data Mining: Concepts and Techniques, Second Edition*. Morgan Kaufmann, Amsterdam ; Boston : San Francisco, CA, 2 edition edition.

Hassouna, M., Tarhini, A., Elyas, T., and AbouTrab, M. S. (2016). Customer churn in Mobile Markets: A Comparison of Techniques. *arXiv preprint arXiv:1607.07792*.

Jeune, W., Francelino, M. R., Souza, E. d., Fernandes Filho, E. I., Rocha, G. C., Jeune, W., Francelino, M. R., Souza, E. d., Fernandes Filho, E. I., and Rocha, G. C. (2018). Multinomial Logistic Regression and Random Forest Classifiers in Digital Mapping of Soil Classes in Western Haiti. *Revista Brasileira de Ciência do Solo*, 42.

Khoshgoftaar, T. M. and Seliya, N. (2004). Comparative Assessment of Software Quality Classification Techniques: An Empirical Case Study. *Empirical Software Engineering*, 9(3):229–257.

Kotsiantis, S. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Informatica (Ljubljana)*, 31.

Kuhn, M. (2008). Building Predictive Models in R Using the caret Package. *Journal of Statistical Software*, 28(1):1–26.

Kuhn, M. and Quinlan, R. (2015). C50: C5.0 Decision Trees and Rule-Based Models. *CRAN, UTC*.

Lavanya, D. and Rani, K. (2012). Ensemble Decision Tree Classifier for Breast Cancer Data. *International Journal of Information Technology Convergence and Service*, 2(1):17–24.

Lewis, B. (2000). Book reviews. *Journal of the American Society for Information Science*, 51(5):490–491.

Liao, S.-H., Chu, P.-H., and Hsiao, P.-Y. (2012). Data mining techniques and applications - A decade review from 2000 to 2011. *Expert Syst. Appl.*, 39:11303–11311.

Liaw, A. and Wiener, M. (2002). Classification and Regression by randomForest. *R news*, 2(3):18–22.

Miller, P. J., Lubke, G. H., McArtor, D. B., and Bergeman, C. S. (2016). Finding structure in data using multivariate tree boosting. *Psychological Methods*, 21(4):583.

Ngai, E. W. T., Xiu, L., and Chau, D. C. K. (2009). Application of data mining techniques in customer relationship management: A literature review and classification. *Expert Syst. Appl.*, 36:2592–2602.

Pandya, R. and Pandya, J. (2015). C5.0 Algorithm to Improved Decision Tree with Feature Selection and Reduced Error Pruning. *International journal of Computer Applications*, 117(16):18–21.

Papeş, M. and Gaubert, P. (2007). Modelling ecological niches from low numbers of occurrences: assessment of the conservation status of poorly known viverrids (Mammalia, Carnivora) across two continents. *Diversity and Distributions*, 13(6):890–902.

Pearson, R. G., Raxworthy, C. J., Nakamura, M., and Peterson, A. T. (2007). Predicting species distributions from small numbers of occurrence records: a test case using cryptic geckos in Madagascar. *Journal of Biogeography*, 34(1):102–117.

Qi, Y. (2012). Random Forest for Bioinformatics.

R Core Team (2018). *R: A language and environment for statistical computing.* R Foundation for Statistical Computing, Vienna, Austria.

Riddick, G., Song, H., Ahn, S., Walling, J., and Borges-Rivera, D. (2011). Predicting in vitro drug sensitivity using random forests. *Bioinformatics*, 27(2):220–224.

Salzberg, S. L. (1994). C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993. *Mach Learn*, 16(3):235–240.

Schratz, P., Muenchow, J., Iturritxa, E., Richter, J., and Brenning, A. (2018). Performance evaluation and hyperparameter tuning of statistical and machine-learning models using spatial data. *arXiv preprint arXiv:1803.11266*.

Sharma, A. and Srivastava, A. (2016). Understanding Decision Tree Algorithm by using R Programming Language. *In ACEIT conference proceeding*, 2016:6.

Steingrimsson, J. A. and Yang, J. (2018). Subgroup Identification using Covariate Adjusted Interaction Trees. *arXiv e-print*, pages arXiv–1806.

Therneau, T. and Atkinson, B. (2018). rpart: Recursive Partitioning and Regression Trees. *R package version*, 4:1–9.

Williams, R. (2016). Understanding and interpreting generalized ordered logit models. *The Journal of Mathematical Sociology*. Publisher: Routledge.

Wisz, M. S., Hijmans, R. J., Li, J., Peterson, A. T., Graham, C. H., and Guisan, A. (2008). Effects of sample size on the performance of species distribution models. *Diversity and Distributions*, 14(5):763–773.

Yang, T., Gao, X., Sorooshian, S., and Li, X. (2016). Simulating California reservoir operation using the classification and regression-tree algorithm combined with a shuffled cross-validation scheme. *Water Resources Research*, 52(3):1626–1651.

Ye, F. and Lord, D. (2014). Comparing Three Commonly Used Crash Severity Models on Sample Size Requirements: Multinomial Logit, Ordered Probit and Mixed Logit Models. *Analytic methods in accident research*, 1:72–85.

Zhang, Y., Lu, S., Zhou, X., Yang, M., Wu, L., Liu, B., Phillips, P., and Wang, S. (2016). Comparison of machine learning methods for stationary wavelet entropy-based multiple sclerosis detection: decision tree, k-nearest neighbors, and support vector machine. *Simulation*, 92(9):861–871.

Zhang, Y., Wang, S., Phillips, P., and Ji, G. (2014). Binary PSO with mutation operator for feature selection using decision tree applied to spam detection. *Knowledge-Based Systems*, 64:22–31.

Zhang, Z. (2016). Decision tree modeling using R. *Annals of Translational Medicine*, 4(15).