doi:10.21608/esju.2024.260404.1025 *Research Article*

The Egyptian Statistical Journal (ESJ), 68 (1): 91-124

Homepage: https://esju.journals.ekb.eg/

The Egyptian Statistical Journal

Print ISSN 0542-1748– Online ISSN 2786-0086

Simulation-Based Assessment of Classification Methods: Statistical Models vs. Machine Learning Algorithms

Reham S. Beram*[1]; Ahmed El-Kotory[1]

Received 03 Jan. 2024; revised 27 May 2024; accepted 31 May 2024

Keywords

Classification;
Logistic regression;
Probit regression;
Discriminant analysis;
Support Vector Machines;
Classification and regression trees;
K-nearest neighbors;
Machine learning.

Abstract

Current studies evaluated the effectiveness of categorization techniques primarily using real datasets with unreported or unknown statistical features. This simulation-based study aims to compare the performance of statistical models (logistic regression, probit regression, and discriminant analysis) with machine learning algorithms (support vector machines, classification and regression trees, and k-nearest neighbors) to comprehensively understand their suitability for classification tasks. Although simulated datasets are used to control their statistical characteristics, the Pima Indian Diabetes real dataset is used to verify the study findings. The outcomes of this study have the potential to guide practitioners and researchers in selecting the most appropriate modeling technique for their specific needs, ultimately enhancing the accuracy and reliability of classification outcomes across various domains. The results revealed that the two statistical models -probit and logit- outperformed in most simulation scenarios. Markedly, the well-grounded, theory-based models of the logit regression and the probit regression models yielded the most accurate predictions in 78.5% and 83.6% of the simulated scenarios, respectively. Interestingly, the performance of the probit model was the best when the binary response variable was balanced (τ=0.50) and when it was too imbalanced (τ=0.90). Notably, the resulting performance metrics of the real dataset refer to the logit, followed by the probit, being the best-predicting models, which resembles the outcome of the simulation study.

1. Introduction

Categorical variables are commonly imperative in many disciplines other than statistics, such as engineering, clinical medicine, genetics, Machine Learning (ML), and social sciences, among others. Notwithstanding their importance in other fields of study, significant progress occurred in statistics more than a century ago regarding the development of regression models where the dependent variable is categorical. These are the qualitative response models (Amemiya, 1981). Such models are also referred to as quantal, discrete, or categorical models. In such models, the researchers' objective may be to estimate the probability of success or failure conditional on a set of regressors (Powers & Xie, 2008). Thus, such techniques are appropriate for the advancement of classification models. Thus, in addition to their resulting probabilities of success and failure, classification models can be employed to classify a new observation into one category or the other

✉ Corresponding author*: reham.beram@gmail.com
[1] Department of Statistics, Mathematics and Insurance, Faculty of Business, Alexandria University, Alexandria, Egypt.

in the case of binary response variables. Although different categorical response models exist, the most commonly applied are the logistic regression, the probit regression, and the Discriminant Analysis (DA).

The term ML was first introduced by Arthur Samuel in 1959 (Arthur, 1959). ML is the field of study that trains computers/systems to operate independently and improve with experience. Accordingly, ML algorithms construct a model based on sample data- training data- to make predictions or decisions. Furthermore, ML utilizes notions from various disciplines: statistics, mathematics, philosophy, computational complexity, and artificial intelligence. Markedly, interest in applying contemporary ML techniques as alternatives to statistical methods is widely increasing (Lynam et al., 2020). For that, colossal improvement has been achieved by ML methods concerning the simple binary discrimination problem that qualitative response models can target.

Further, it was claimed that the successful use of ML in several fields indicates promising applications in other fields. However, the advantages and superiority of ML-based classification methods compared with more traditional statistical ones need to be assessed, validated, and verified in all fields of application (Côté et al., 2022). With that in mind, such alternative ML algorithms include Decision Trees (DTs), Support Vector Machines (SVMs), K-Nearest Neighbors (KNN), Random Forest (RF), Gaussian Process (GP), Naive Bayes (NB) and Artificial Neural Networks (ANN).

Will analyzing the same data set by the aforementioned models reveal the same classification performance? If not, which one will be the best-fit model? Findings related to the superiority of classification accuracy of newer classification approaches compared to traditional, less computer-requiring methods and the stability of the findings are still controversial. Although many previous studies have compared logistic regression to various ML techniques, to the researcher's knowledge, not much research has considered the comparison between ML methods and either probit regression or DA. To conclude, the study will include an empirical analysis represented by a simulation study and the application of a real dataset using the six techniques. The results will be compared to establish the similarities and discrepancies between them. In order to compare the models mentioned above, indicators such as accuracy, sensitivity, specificity, and the Area Under the Receiver Operating Characteristic (ROC) Curve (AUC) will be considered.

This being the objective, the study will be organized such that the relevant literature comparing the six models, whether concerned with the comparison of the statistical models or that related to comparing statistical models with ML classifiers, will be depicted in Section 2. The well-grounded statistical models will be presented in Section 3. Next, Section 4 will be devoted to illustrating the selected ML algorithms. Subsequently, Section 5 will be devoted to the simulation study, including its factors, performance measures, and simulation configuration. Then, the resulting values of the performance measures will be discussed in Section 6. Finally, the study will be concluded with possible future work in Section 7.

2. Literature review

This section presents the relevant literature for this study in chronological order. It will be divided into two parts. The first section, Statistics-based Studies, will introduce some literature that compares the three statistical models, while the second section will depict studies that include logistic regression and various ML algorithms.

2.1 Statistics-based studies

Interest in comparing the different qualitative response models' performance started very early. In 1978, Press and Wilson (1978) examined theoretical arguments for using logistic regression compared to utilizing DA to deal with the classification problem. Two empirical studies of non-normal classification problems were carried out in this research. The results of this comparison revealed that the logistic regression had an average percentage of 67 correct classifications in both the training and test datasets. On the other hand, 63% were correctly classified by DA on average.

Regarding the second study, the outcomes disclosed that the average correctly grouped rate for logistic regression was 80%, while that for DA was 68% for the training sets. The respective rates for the validation sets were 72 and 68 percent. To summarize, the research results revealed that the logistic regression outpaced the DA according to the proportion of correct classification.

Amemiya (1981) introduced some facts about the linear probability model, probit, and logistic regression. It was noted that the probit and logit models often arrive at similar results. For a dataset Theil utilized, Amemiya computed four estimators, and it was concluded that the estimators of LP-least squares and LP-weighted least squares were better than those of the logit and probit. However, the probit estimators outpaced the logit. Regarding the comparison between the logistic regression and DA, the study found that the DA estimator is the genuine MLE; consequently, it should be asymptotically more efficient than its logit counterpart. However, if the normality assumption is not satisfied, the DA estimator misses its consistency, while the logit MLE holds it. Thus, this study concludes that no estimator is the best in all cases. That is, each model has situations where it excels and others where it performs poorly. Similarly, Pohar et al. (2004) inspected the dilemma of selecting between the LDA and the logit. The authors conducted various simulations to scrutinize the performance of the methods. The reference simulation was the one in which all of the requirements for the LDA were met. Then, the effects of varying the sample size, covariance matrix, and the Mahalanobis distance, among others, were examined. The findings showed that the sample size had the most glaring influence on the distinction between approaches. Moreover, the LDA is the best method when the regressors are normally distributed. However, in general, the outcomes of the logit are consistently near to and slightly inferior to those of the LDA in most scenarios. However, when LDA's presumptions are not met, using it is not warranted; in contrast, the logit produces good results regardless of the distribution.

Likewise, Prempeh (2009) addressed whether applying the DA and the logistic regression approaches to the same data set would produce the same conclusion. In terms of the proportion of accurate classifications, the logistic regression resulted in 91.1%, while the DA yielded 88.9%. Consequently, regarding the issue of classification competencies, the first model performed better

than the latter. Further, upon using the hold-out sample to test the efficiency of the estimated models, both the DA and the logit could correctly classify the bankrupt and non-bankrupt firms. Finally, the study concluded that logistic regression yielded superior classification results than DA in the presence of multicollinearity problems in the covariates.

In research conducted by Cakmakyapan and Goktas (2013), a Monte Carlo simulation was undertaken to compare the probit and logit regressions across varying conditions. For instance, different sample sizes, varying correlations between the predictand and the predictors, and different thresholds for converting the latent variable to be binary. For the simulation procedure, the studied latent variable was treated as continuous and impacted by three explanatory variables generated from the multivariate standard normal distribution. Moreover, three distinct variance-covariance matrices - "high," "low," and "no"- were employed to generate data from the multivariate standard normal distribution. To examine the effect of sample size in picking a model, 5 different sample sizes were considered: 40, 100, 200, 500, and 1000. For each of the matrices and sample sizes, data generation was repeated 1000 times. The paper disclosed that the logistic regression was better than the probit regression in "low" and "high" cases for 500 and 1000 sample sizes. In addition, the probit model was better in small sizes (40, 100, and 200). Further, in the "no" case, the two models fitted the data similarly regardless of any conditions.

In a similar but more recent study by Cakmakyapan and Goktas (2013) is that executed by Alsoruji et al. (2018), the researchers conducted a simulation to compare the probit and logit models under various sample sizes, dependent-independent variables' correlation coefficients, and latent response In variable cut points. In the simulation, the regressand is influenced by three covariates from the standard multivariate normal distribution. Three different matrices of the variance-covariance are also considered. Further, five sample sizes were considered: 70, 100, 200, 500, and 1000. The results revealed that the logit was better for large samples and high correlation. Besides, for small sample sizes, the probit was better.

As shown, there was no common consensus in the literature regarding the superiority of one classification method over the other. Instead, each one performed well under different conditions.

2.2 Machine learning-based studies

There is a growing interest in using modern ML techniques as alternatives to statistical methodologies (Lynam et al., 2020). Bichler and Kiss (2004) conducted research that evaluated the effectiveness of logistic regression, KNN, and decision tree algorithms in campaign management. A data collection of 10,054 cases was randomly chosen for this investigation. The data contained a dichotomous dependent variable and a total of 165 regressors. Based on the area under the ROC curve, the logistic regression performance was the best, followed by the DT and then the KNN classifier. According to the predictive accuracy measure, the three approaches were almost equally accurate, with a 97.5% accuracy rate.

Kurt et al. (2008) contrasted the effectiveness of multi-layer perceptrons, logistic regression, and CART in forecasting coronary artery disease occurrence. The study was applied to a total of 1,245

observations. Between the CART and the logistic regression, the CART outperformed according to the various evaluation metrics. In a similar attempt to compare the performance of the logistic regression in the face of the CART and neural networks, Liu et al. (2011) applied the three techniques to a dataset consisting of 1,225 UK males. The three models' overall accuracy ranged from 0.59 to 0.67, with an overall AUC of 0.65 to 0.72. The performance of the neural networks was marginally superior to that of the logistic regression and CART models.

Musa (2013) implemented a comparative study between the logistic regression and the SVM. The data sets used in this study were composed of 13 different datasets with binary class attributes, where the data sizes of the various datasets ranged from 270 to 5,473 observations. The findings demonstrated that, on average, the SVM and the logistic regression performed equally well in balanced and unbalanced data cases across all performance gauges. SVMs, however, might perform better on severely unbalanced data sets. In their study, Settouti et al. (2016) relied on the work presented in 2006 at the International Conference of the IEEE on Data Mining, which identified the 10 topmost ML algorithms appropriate for classification tasks. For instance, Apriori, AdaBoost, Bagging, C4.5, CART, Expectation-Maximization, K-means, KNN, NB, and SVM. After applying them to twelve medical and biological data samples, they used a set of nonparametric statistical tests to rank the selected methods better. The results disclosed that C4.5 is the best algorithm, followed by Bagging. In contrast, CART and Adaboost had the third rank, preceding the SVM, KNN, Apriori, EM, NB, and the K-means, respectively.

In a study to classify patients with type 1 and type 2 diabetes, Lynam et al. (2020) picked the logistic regression in conjunction with ML algorithms: Gradient Boosting Machine (GBM), ANN, KNN, RF, and SVM. The models were applied to 1,378 individuals as the training group, while the validation sample consisted of 566 participants. According to the findings, a slight difference in the performance of the models was revealed. Besides, the calibration slopes applied on the validation sample showed outstanding GBM and ANN performance. Moreover, the logit and SVM have a satisfactory calibration outcome. The study concluded that the logistic regression performed similarly to the selected ML techniques. In 2021, Itoo and Singh (2021) applied logistic regression, NB, and KNN to classify credit card transactions as either fraud or non-fraud. A dataset of the MasterCard transactions was obtained, containing 284,807 observations. According to the outcomes, it was evident that the logistic regression outperformed the other two models. Furthermore, the NB method follows the logistic model. Conversely, the KNN had the lowest accuracy compared to the logistic regression and the NB models.

Among the few studies that have been carried out on simulated data and real-life datasets is that of Scholz and Wimmer (2021). In this work, 18 classification techniques were compared, including LDA, regularized DA, logistic regression, regularized as well as Bayesian logistic regression, KNN, CART, and SVM with different kernels in addition to a set of ensemble methods (e.g., boosted logistic regression, random forests, bagged CART, Gradient Boosted Trees), among others. A sample size of the training dataset and several features were used to approximate data complexity. Thus, 4 scenarios were generated based on those factors, namely the low-dimension low-sample size, the low-dimension high-sample size, the high-dimension low-sample size (HDLSS), and the high-dimension high-sample size (HDHSS). Under these scenarios, 5 data

characteristics were included as well. Further, 5 performance measures were utilized to compare the selected methods, including the AUC, F1 measure, *H*-measure, and the Brier Score. The results showed that the logistic regression exhibited a moderate performance. Methods such as SVM bagged CART, or C5.0 performed better than the logistic regression in most investigated scenarios. For the HDLSS, a heterogeneous ensemble, kernel-based classifiers with polynomial kernel, or stabilized nearest neighbor classifiers were recommended. Furthermore, comparative analysis using real-world data indicates that the most promising classifiers are ensemble classifiers and SVM in terms of their predictive performance.

Further, Liew et al. (2022) put the Least Absolute Shrinkage and Selection Operator (LASSO), the logistic regression, the gradient boosting (Xgboost), the KNN, and the SVM methods in comparison, among others. The authors applied the selected models to 3,001 individuals. Further, three dichotomous response variables were employed - neck pain, arm pain, and disability. Based on the resultant values of the AUC, the Xgboost attained the highest records in predicting the 3 dependent variables mentioned earlier. Furthermore, the logistic regression had the lowest AUC for predicting neck and arm pain. In comparison, KNN had the lowest AUC for predicting disability. Further, the logit was the most sensitive for predicting arm pain; LASSO and KNN were similarly sensitive for neck pain, while Xgboost and ANN were equally sensitive for disability. Speaking of the SVM, it was the most specific for predicting arm pain. The paper hypothesized that ML will outperform standard logistic regression in predicting recovery status for people suffering from neck pain. Their prediction was partially validated by logistic regression, which was the worst performer for predicting arm and neck discomfort, whereas KNN was the worst performer for disability.

Côté et al. (2022) aimed to discover if ML algorithms outperform statistical models in predicting vegetable and fruit intake using a sample of 1,147. Classification ML techniques such as DT, random forest, and SVM with different kernels -linear, polynomial, radial basis, and sigmoid-, KNN, and Adaboost were tested against the logistic regression and LASSO. The logistic regression and Lasso predicted the dependent variable with an equal accuracy of 64%. Conversely, between the examined ML methods, the SVM with either a radial basis kernel or a sigmoid one predicted acceptable classification with an accuracy of 65%. The SVM with a linear kernel was the least accurate, with an accuracy of 55%.

As has been noted from literature which compared the logistic regression with ML algorithms, some studies concluded either the superiority of the logistic regression performance over the other methods - Itoo and Singh (2021), or at least its equal performance with them - Musa (2013) and Faisal et al. (2020). Other studies have figured out that the employed ML approach outperformed the logit, such as those of Kurt et al. (2008), Liu et al. (2011), and Farhat and Cheok (2021).

3. Statistical-based models

This section focuses on the theoretical foundations of the three widely used statistical models. They are extensively employed in various fields for classification, offering valuable insights into the relationships between predictors and the probability of belonging to a specific class.

3.1 Probit regression

According to Powers and Xie (2008), the probit uses the standard normal distribution transformation to ensure that the resultant probabilities are within the [0,1] range. A binary response model with a dependent variable Y_i which has only two values: one if the event occurs and zero otherwise, and a set of independent variables x_i can be represented such that:

$$\mathcal{P}(Y_i = 1|x_i) = \mathcal{F}(\sum_{k=0}^{K} \beta_k \, x_{ik}) = \phi(\boldsymbol{X\beta}) \tag{1}$$

For a binary regresand Y_i, and a set of predictor variables x_i, the general form of the probit model is expressed by Equation (1), where ϕ is the standard normal distribution CDF, $\phi^{-1}(p_i)$ is its inverse, and p_i is the probability of the i^{th} observation.

$$probit\ (p_i) = \phi^{-1}(p_i) = \sum_{k=0}^{K} \beta_k x_{ik} \tag{2}$$

The Maximum Likelihood Estimation (MLE) technique is employed to estimate the probit regression parameters (Greene, 2012). As regards the classification stage of a new observation, having estimated the β's and calculating the value of $\mathcal{P}_i$ for that new observation, such value should be compared with a prespecified threshold to classify the new observation into one class.

3.2 Logistic regression

As depicted in Powers and Xie (2008), for a binary logistic model, a dependent variable Y_i has only two values, and a set of independent variables x_i, such that:

$$\mathcal{P}(Y_i = 1|x_i) = \mathcal{F}(\sum_{k=0}^{K} \beta_k \, x_{ik}) = \Lambda(\boldsymbol{X\beta}) \tag{3}$$

Where β is a parameter that needs to be estimated, and $\mathcal{F}$ or Λ is the logistic CDF. Further, the general form of the logistic model is:

$$\mathcal{P}_i = \mathcal{F}(\sum_{k=0}^{K} \beta_k \, x_{ik}) = \frac{1}{\left\{1+e^{-\left(\sum_{k=0}^{K} \beta_k x_{ik}\right)}\right\}} = \frac{e^{\left(\sum_{k=0}^{K} \beta_k x_{ik}\right)}}{1+\, e^{\left(\sum_{k=0}^{K} \beta_k x_{ik}\right)}} \tag{4}$$

If $\mathcal{P}_i$ is the success probability, then $(1 - \mathcal{P}_i)$ is the probability of failure. Where $\frac{\mathcal{P}_i}{1-\mathcal{P}_i}$ is the odds ratio in favor of success. The logistic regression's outcome variable is the log of the odds ratio $\mathcal{L}_i$. The MLE technique is applied to estimate the logistic regression parameters. Regarding the classification step of a new observation, upon estimating the β's and calculating the value of $\mathcal{P}_i$ for the new observation, such value should be compared with a prespecified threshold to classify the new observation into one class.

3.3 Discriminant analysis

The main reason for conducting DA is to equip researchers with an approach for classifying an object into one of two or more populations. Hence, in DA, the response variable is a category to which an observation belongs (Park, 2015). Moreover, it is very similar to probit and logistic regressions. In other words, through DA, one can attempt to find a linear function of the set of

independent variables, which comes up with the best discrimination between two categories (Maddala, 1986).

For DA, the form of the equation is as follows:

$$\mathcal{D}_t = \lambda_0 + \lambda_1 x_1 + \lambda_2 x_2 + \ldots\ldots + \lambda_k x_k \quad (5)$$

Where $\mathcal{D}_t$ is the discriminant function, and $\lambda_0, \lambda_1, \ldots\ldots, \lambda_k$ are the discriminant coefficients of the covariates $x_0, x_1, \ldots\ldots, x_k$, respectively. Moreover, given that μ_1 and μ_2 are the means of the X's in the two populations, the corresponding means of $\mathcal{D}_t$ in the two classes, according to Maddala (1986), are $\boldsymbol{\lambda}'\mu_1$ and $\boldsymbol{\lambda}'\mu_2$, respectively. Assuming that the covariance matrices are $\boldsymbol{\Sigma_1}$and $\boldsymbol{\Sigma_2}$ and that they are equal $\boldsymbol{\Sigma_1}$= $\boldsymbol{\Sigma_2}$=$\boldsymbol{\Sigma}$, the variance of $\mathcal{D}_t$ can be expressed by $\boldsymbol{\lambda}'\boldsymbol{\Sigma}\boldsymbol{\lambda}$. The discriminant coefficients are so determined that the ratio (ζ) of the between-group difference -means- relative to the within-group variation -variance- is maximized (Maddala, 1986).

Having estimated $\lambda's$, the value of $\mathcal{D}_t$ should then be calculated upon inserting the values of $\overline{X_1}$ and $\overline{X_2}$ resulting in the means of $\mathcal{D}_t$, namely $\overline{\mathcal{D}}_1$ and $\overline{\mathcal{D}}_2$, respectively. As mentioned by Maddala 0(1986), in order to classify a new observation x_0 into one of two categories, the value of $\mathcal{D}_t$ for that observation should be calculated using the values of its explanatory variable.

At that point assign it to π_1, the first class, if $\mathcal{D}_0$ is nearer to $\overline{\mathcal{D}}_1$ than $\overline{\mathcal{D}}_2$. Further, provided that $\overline{\mathcal{D}}_1$ is larger than $\overline{\mathcal{D}}_2$, $\mathcal{D}_0$ will be closer to $\overline{\mathcal{D}}_1$ than $\overline{\mathcal{D}}_2$ if the following condition holds:

$$|\mathcal{D}_0 - \overline{\mathcal{D}}_1| > |\mathcal{D}_0 - \overline{\mathcal{D}}_2| \quad (6)$$

4. Machine learning-based models

The section is dedicated to the theoretical foundations of three popular machine-learning algorithms: SVM, CART, and KNN. These algorithms are widely used in various domains for classification tasks, offering different approaches to pattern recognition and prediction.

4.1 Support Vector Machines (SVMs)

SVMs are a set of supervised learning methods that can be utilized for data classification, regression analysis, and outliers' detection. According to Awad and Khanna (2015), SVM is a kernel decision machine that averts the calculation of posterior probabilities while building its learning model.

In the SVM context, a hyperplane is chosen to appropriately separate the data points by their respective class, either 0 or 1. While multiple hyperplanes could exist, SVM attempts to find the one that best separates the two categories. In essence, the SVM attempts to maximize the width, the distance, or the margin M. As explained by Hastie et al. (2009), maximizing this width implies to solve the following equations using the Lagrange multipliers' approach

$$\min_{\boldsymbol{\beta},\boldsymbol{\beta_0}} \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C\sum_{i=1}^{N}\xi_i \quad (7)$$

$$Subject\ to\ \boldsymbol{y}(\boldsymbol{x}^T\boldsymbol{\beta}+\boldsymbol{\beta_0}) \geq 1-\xi_i, \xi_i \geq 0\ \forall i \quad (8)$$

Where ξ_i is the distances of the misclassified points, and C is the cost term, which equals ∞ in the case of linearly separable data (Hastie et al., 2009). Further, the *regularization*, C, is a parameter that differs depending on the optimization aim. Provided that the data is not linearly separable, a kernel should be employed to project the data to a higher-dimensional space, kernel space, where it is easier to find a linear boundary between the different classes; hence, the data becomes linearly separable. There are different types of kernels: the linear, the polynomial, the Gaussian radial basic function, the sigmoid function, and the hyperbolic tangent function kernels (Meyer and Wien, 2021).

4.2 Classification and regression trees

The CART classification technique utilizes historical data to build decision trees (Timofeev, 2004). It is a binary recursive separating technique that builds classification trees to predict dummy outcomes in classification tasks. The most critical stages in constructing the model are splitting, stopping, pruning, and optimal tree selection.

In the first place, the CART starts by selecting the first variable to split the data. To achieve this, it calculates the impurity score for each regressor, and the one with the least impurity is picked to divide the observations. To determine the aforementioned best splitting value x_j^R, at each node, the CART solves the hereunder maximization problem of the change in the impurity measure $\Delta i(t)$:

$$\underset{x_j \leq x_j^R, j=1,\ldots,M}{Argmax} \left[i(t_p) - P_l\, i(t_l) - P_r\, i(t_r) \right] \tag{9}$$

Where $i(t_p)$ is the impurity function of the parent node, $i(t_l)$ and (t_r) are the impurity functions of the left and right child nodes, respectively. Further, P_l and P_r are the probabilities of the left and right nodes, respectively (Timofeev, 2004). Equation (9) indicates that the CART explores all the possible values of each independent variable of $\boldsymbol{X}$ until it obtains the best split point "$x_j \leq x_j^R$" which maximizes the change of impurity score $\Delta i(t)$.

A stopping criterion must exist; otherwise, the tree construction will persist until it is impossible to continue. Consequently, the process comes to an end when: (1) there is a sole observation in each of the child nodes; (2) all observations inside each child node possess the same distribution of explanatory variables, making further splitting unattainable; or (3) predetermined limit on the number of levels in the tree has been specified by the user. Consequently, upon stopping tree building, its optimization or pruning should be undertaken to reduce overfitting.

4.3 K-Nearest neighbours

As depicted in Lynam et al. (2020), KNN is an approach of supervised learning that can be employed for classification and regression purposes. Its primary notion is that observations close together in n-dimensional space will have similar outcomes. Thus, the classification involves searching the entire dataset for the k-points closest in distance (k-neighbors). Hence, KNN selects a set of k objects in the training sample nearest to the test observation and bases the category assignment on the predominance of a particular class in this neighborhood (Wu et al., 2008).

To categorize a new observation, the distance of it to the classified observations is computed, its k-nearest neighbors are identified, and the class of these nearest neighbors is then utilized to determine the category of the observation. Primarily, the choice of the k value is essential. Further, it has been claimed that the optimal value of k is usually $\sqrt{N}$, where N is the total sample size before partitioning the data into training and testing. Having determined the best value for k, the

new observation is classified based on the dominating class of its nearest k neighbors -majority vote rule- as follows:

$$\underset{\pi_k}{Argmax} \sum_{(X_i,Y_i)\in D_{x_0}} I(\pi_k = \mathcal{T}_i) \tag{10}$$

where π_k is a category label, x_0 is the new observation to be classified, D_{x_0} is the list of k closest observations from x_0, $\mathcal{T}_k$ is the class of each k^{th} nearest neighbors while $I(\cdot)$ is an indicator function which gives the value 1 if its argument is correct and 0 otherwise (Wu et al., 2008).

5. Simulation study

The section focuses on the simulation study conducted to investigate the performance of different modeling approaches. It details the factors considered in the simulation, the performance measures used to evaluate the models, and the configuration of the simulation. It is worth mentioning that the simulation study will be performed using the language and environment for statistical computing and graphics, R-4.2.3. Besides, all computations will be performed on the BA High Performance Computing (HPC) cluster, "BA-HPC-C2", The Bibliotheca Alexandrina Supercomputing Facility.

5.1 Simulation study factors

The key objective of this research is to determine whether there is a difference between the most famous and regularly used statistical binary qualitative response models and ML classifiers in prediction and classification accuracy under various conditions. The simulation shall be undertaken with varying sample sizes, changing the number of regressors, diverse levels of correlation between the outcome variable and the explanatory ones, and different cut points for the response variable.

The dependent variable will be continuous and affected by a set of explanatory variables. The regressand and the predictors will be generated from the multivariate standard normal distribution. Hence, the variance-covariance matrix is needed to generate the dataset. In this case, the correlation matrix is the same as the covariance one. These matrices denoted by Σ will be generated to be symmetric and positive semi-definitive and presented in Appendix A. Further, correlations among the regressors shall be as small as possible to avoid the multicollinearity problem. Having generated the data, the continuous outcome variable should be transformed into a binary one through a cut-off point (τ). Such a cut point should change the proportion of the two categories as expressed in Equation (11).

$$Y_i = \begin{cases} 1, Y_i^* \geq \tau \\ 0, Y_i^* < \tau \end{cases} \quad (11)$$

The first threshold is the median so that the two groups are equally distributed. The second will be the 90th decile, so class (1) should represent 10% of the sample, and the rest will belong to class (2). This case should reflect the case of outliers or rare phenomena. Moreover, data generation would be repeated 10,000 times. To sum up, the data generation process is summarized in Table 1.

Consequently, in this study, 84 diverse scenarios will be simulated with 840,000 generated datasets. Every generated dataset will be split into a training set comprising 75% of the primary dataset and a testing set of 25%. Following, the six models of interest will be trained using the training set. Thereupon, the fitted and trained models will be used to predict the outcome of the data points employing the testing set. Subsequently, the performance of the models will be evaluated using a variety of metrics, which will be presented in the subsequent section.

Table 1: Simulation Factors

Cut-Point = Median			**Cut-Point = 90th Decile**		
Sample Sizes	***Number of Regressors***	***Variance-Covariance Matrices***	***Sample Sizes***	***Number of Regressors***	***Variance-Covariance Matrices***
50, 100, 200, 500, 1000, 5000 and 10,000	3, 5 and 10	Moderate - High	50, 100, 200, 500, 1000, 5000 and 10,000	3, 5 and 10	Moderate - High
		Low			Low

5.2 Performance Measures

Following the training step, the models' performance should be evaluated. This can be done using the output of the confusion matrix. As demonstrated by 0it is divided into four quadrants where True Positives (TP) is the number of cases that were predicted to be in class (1) and were actually in that class, False Positives (FP) is the number of observations that were predicted to be in class (1) yet, were actually in the other class, False Negatives (FN) is the number of instances that were labeled to belong to class (2) but were actually in class (1), and True Negatives (TN) is the number of data-points that were predicted to be in class (2) and were actually in that class.

Table 2: Confusion Matrix Representation

		Actual	
		Event (Positive)	***Not-Event (Negative)***
Predicted	***Event (Positive)***	TP	FP
	Not-Event (Negative)	FN	TN

The confusion matrix can calculate several metrics, such as accuracy, sensitivity, specificity, precision, and F1-measure. Accuracy (Acc) is defined as the ratio of the total number of predicted observations that are correctly classified, as depicted in Equation (12).

$$\text{Acc} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{12}$$

As for Sensitivity (Sens), it is the proportion of occurrences (ones) that a model predicted correctly as events. It is also called true positive rate or recall and can be calculated as follows:

$$\text{Sens} = \frac{\text{TP}}{\text{TP+FN}} \tag{13}$$

Speaking of Specificity (Spec) as known as true negative rate, it is the proportion of the non-events (zeros) that a model predicted correctly as non-events which can be in Equation (14).

$$\text{Spec} = \frac{\text{TN}}{\text{TN+FP}} \tag{14}$$

In 1950, Youden provided Youden's Index (YI) expressed in Equation (15), which measures the ability of a classifier to avoid failure (Musa, 2013). The index's value runs from 0 to 1, with zero indicating that the model is ineffective, while 1 implies that it is perfect.

$$\text{YI} = \text{Sens} - (1 - \text{Spec}) \tag{15}$$

Precision (Prec) reflects the percentage of positive predictions that are actually correct.

$$\text{Prec} = \frac{\text{TP}}{\text{TP+FP}} \tag{16}$$

In addition to the previously stated measures, there exists one that encompasses both recall and precision as follows:

$$\text{F} - \text{measure} = 2 \times \frac{\text{Prec} \times \text{Sens}}{\text{Prec} + \text{Sens}} \tag{17}$$

A common method to compare classification models is the AUC (Fawcett, 2006). For the sake of models' comparison, it may be more appropriate to summarize the information provided by the ROC through a single value portraying the models' performance. This can be achieved through the AUC. The value of the AUC will always be in the (0 – 1) range. A value of 0.5 indicates no predictive ability, and 1 indicates perfect predictive ability. Hu et al. (2021) mentioned that, typically, a model is regarded as excellent if its AUC falls between 0.9 and 1.0. While it is considered good if the AUC is between 0.8 and 0.9. Further, a model is judged fair if the AUC falls between 0.7 and 0.8. Lastly, a model AUC less than 0.7 is regarded as poor.

The distinct performance metrics reflect somewhat various tradeoffs in models' predictions, and it is likely for a classification model to perform satisfactorily according to one metric while being suboptimal on others. As a result, it is critical to analyze algorithms over a wide range of performance metrics (Musa, 2013). Consequently, this study will depend on various performance criteria.

6. Simulation Results

This section will present the results and findings of the simulation study.

6.1 Accuracy Measure-based Results

This section will compare the models based on their average accuracy across the 10,000 iterations. Under the median cut-off, it was revealed that no matter the correlation level, the sample size, or the number of predictors, the logit and the probit had the highest accuracy levels in 83% and 92% of the scenarios, respectively. As shown in Table 3, in the high correlation scenario coupled with 3 X's, when the sample size reached 10,000, the six models achieved almost the same accuracy level. Moreover, in all numbers of regressors' scenarios, as the correlation scale drops to low, all models; accuracies fell as well. Yet, the logit and probit are still the best. It was noted that the differences in accuracies of the alternative models diminish as the sample size increases.

Further, it was revealed that the CART accuracy is the worst in 60% of the cases, especially in the high correlation and 3, 5, and 10 regressor scenarios, not only in these scenarios but also in the low correlation case with 10 covariates. Despite this phenomenon, the results demonstrate that *the accuracy of the CART algorithm increases* as the sample size increases in the high correlation case and 3, 5, and 10 regressors scenarios.

Table 3: Accuracy, Median Cut-off, 3 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.920	0.924	0.899	0.902	0.737	0.875
100	0.960	0.960	0.940	0.942	0.799	0.924
200	0.941	0.945	0.900	0.892	0.814	0.886
500	0.950	0.950	0.932	0.932	0.839	0.904
1,000	0.939	0.939	0.924	0.929	0.879	0.927
5,000	0.944	0.944	0.939	0.939	0.910	0.929
10,000	0.936	0.936	0.935	0.934	0.918	0.928
Low						
50	0.728	0.729	0.618	0.614	0.566	0.582
100	0.748	0.748	0.680	0.652	0.607	0.561
200	0.660	0.660	0.580	0.580	0.660	0.640
500	0.734	0.742	0.718	0.694	0.613	0.605
1,000	0.660	0.668	0.624	0.612	0.640	0.572
5,000	0.643	0.644	0.646	0.639	0.617	0.620
10,000	0.665	0.666	0.669	0.662	0.652	0.626

On the other side, unlike the CART, the KNN algorithm attained the highest accuracy among all models in the low correlation scale with 3 and 5 explanatory variables at the 100-sample size. Besides, the most striking observation to emerge from the models' comparison was that at the 200-sample size in the 3 X's and the 5 X's, the accuracy of the logit decreased at all correlation levels.

The same superiority of the logit and the probit models under the 90^{th} decile, with them being the most accurate in 71% and 76% of the scenarios, respectively. Such outcomes are evident from Tables 6-8. As noted in Table 8, the SVM was as accurate as the logit and probit in 10 X's in the high correlation scenario with 100, 1,000 and 10,000 sample sizes, in addition to the low correlation scenario with 50, 500, 5,000 and 10,000 sample sizes.

Table 4: Accuracy, Median Cut-off, 5 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.896	0.910	0.905	0.913	0.695	0.836
100	0.971	0.971	0.935	0.946	0.677	0.836
200	0.810	0.977	0.963	0.975	0.756	0.895
500	0.999	0.999	0.985	0.982	0.792	0.934
1,000	0.995	0.774	0.991	0.990	0.838	0.916
5,000	1.000	1.000	0.986	0.997	0.891	0.951
10,000	0.999	0.999	0.993	0.997	0.907	0.954
Low						
50	0.748	0.748	0.642	0.637	0.558	0.599
100	0.710	0.711	0.701	0.663	0.696	0.554
200	0.625	0.624	0.586	0.604	0.583	0.562
500	0.669	0.662	0.622	0.621	0.653	0.621
1,000	0.720	0.720	0.696	0.696	0.648	0.687
5,000	0.718	0.720	0.715	0.719	0.690	0.676
10,000	0.694	0.694	0.690	0.692	0.662	0.670

Table 5: Accuracy, Median Cut-off, 10 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.849	0.863	0.868	0.886	0.542	0.782
100	0.953	0.979	0.937	0.931	0.708	0.912
200	0.663	0.961	0.925	0.951	0.760	0.846
500	0.853	0.853	0.979	0.978	0.766	0.882
1,000	0.996	0.996	0.970	0.976	0.788	0.890
5,000	0.999	0.999	0.985	0.997	0.829	0.917
10,000	1.000	1.000	0.990	0.996	0.854	0.924
Low						
50	0.776	0.788	0.806	0.797	0.667	0.686
100	0.908	0.903	0.838	0.862	0.542	0.780
200	0.960	0.940	0.920	0.920	0.620	0.720
500	0.895	0.895	0.855	0.855	0.645	0.718
1,000	0.900	0.900	0.888	0.888	0.680	0.796
5,000	0.888	0.887	0.883	0.882	0.712	0.828
10,000	0.901	0.901	0.898	0.901	0.740	0.857

Table 6: Accuracy, Ninetieth Decile Cut-off, 3 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	1.000	1.000	0.917	0.917	0.833	0.833
100	0.960	0.960	0.920	0.920	0.880	0.920
200	0.980	0.980	0.980	0.960	0.860	0.980
500	0.992	0.992	0.960	0.976	0.920	0.952
1,000	0.996	0.996	0.984	0.992	0.948	0.988
5,000	0.973	0.973	0.962	0.969	0.952	0.965
10,000	0.975	0.975	0.956	0.973	0.964	0.970
Low						
50	0.917	0.917	0.917	0.833	0.917	0.833
100	0.920	0.920	0.920	0.920	0.880	0.920
200	0.900	0.900	0.920	0.900	0.940	0.920
500	0.936	0.936	0.936	0.928	0.896	0.944
1,000	0.920	0.916	0.908	0.904	0.888	0.908
5,000	0.901	0.901	0.899	0.900	0.898	0.899
10,000	0.902	0.902	0.899	0.897	0.909	0.891

Table 7: Accuracy, Ninetieth Decile Cut-off, 5 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.834	0.834	0.917	0.834	0.833	0.833
100	0.960	0.960	0.920	0.920	0.720	0.960
200	0.980	1.000	0.960	0.940	0.920	0.920
500	0.996	0.996	0.956	0.976	0.920	0.940
1,000	0.950	0.950	0.970	0.990	0.912	0.952
5,000	1.000	1.000	0.962	0.998	0.942	0.968
10,000	0.999	0.999	0.968	0.996	0.952	0.976
Low						
50	1.000	1.000	0.917	0.917	0.833	0.917
100	0.840	0.840	0.880	0.880	0.880	0.880
200	0.860	0.880	0.860	0.880	0.920	0.880
500	0.936	0.936	0.928	0.928	0.920	0.936
1,000	0.916	0.916	0.908	0.916	0.892	0.892
5,000	0.911	0.911	0.910	0.904	0.889	0.906
10,000	0.909	0.908	0.906	0.901	0.893	0.899

Table 8: Accuracy, Ninetieth Decile Cut-off, 10 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	1.000	1.000	1.000	0.917	0.917	0.917
100	1.000	1.000	1.000	1.000	0.960	0.960
200	1.000	1.000	0.960	0.960	0.900	0.900
500	0.992	0.992	0.944	0.976	0.880	0.936
1,000	0.992	0.992	0.960	0.992	0.920	0.936
5,000	0.950	0.950	0.968	0.996	0.916	0.947
10,000	1.000	1.000	0.965	0.997	0.925	0.955
Low						
50	0.933	0.934	0.867	0.934	0.917	0.933
100	0.926	0.926	0.889	0.852	0.880	0.852
200	0.940	0.940	0.920	0.900	0.920	0.880
500	0.952	0.952	0.944	0.944	0.896	0.928
1,000	0.964	0.964	0.956	0.948	0.888	0.940
5,000	0.959	0.959	0.948	0.955	0.897	0.919
10,000	0.956	0.956	0.947	0.954	0.898	0.925

6.2 Sensitivity Measure-based Results

The results and comparison based on the average sensitivity will be discussed in this section. As depicted earlier, the sensitivity criterion measures the ability of the classification model to determine the positive cases. A sensitivity value of 1 means the model did not classify any observation as false negative. That is, no actual positive observation ($Y = 1$) was predicted as being negative ($Y = 0$). In the extreme case, its 0 value means that the model failed to detect any true positive cases while simultaneously committing false negative predictions.

Under the $\tau = 0.5$ cases and the 3 predictors, the results revealed that the differences in models' accuracy lessen as the sample size upsurges. Besides, CART has the worst sensitivity, followed by the KNN. However, the sensitivity of the CART rises as the sample size surges at a high correlation level. For the 10 X's scenarios, the performance of the logit and the probit was not the best among other models. Besides, their performance was too volatile. On the other hand, the sensitivity of the CART model continued to increase as the sample size rose in the high correlation case. Another fundamental note is that the KNN was the only model whose sensitivity did not decrease under the 40% sensitivity threshold, unlike the other models.

In the $\tau = 0.9$ case, it is imperative to mention that the six models of the logit, the probit, the DA, the SVM, the CART as well as the KNN failed to detect any true positive instances in 27%, 22%, 43%, 57%, 63% and 52% of the scenarios, respectively, with an average sensitivity of zero. In the group of 3 predictors, the logit and the probit methods were the best in the high correlation scenario, specifically at small sample sizes from 50 to 500. Likewise, the same models performed

best in the low correlation scenario at the low sample sizes. In the low correlation, the KNN became comparable to the logit and the probit at the 5,000 and 10,000 sample sizes.

Regarding the 5 X's group of scenarios, in the high correlation case with a 50-sample size, the DA was the only model that achieved 50% sensitivity. In contrast, all the other models had zero sensitivity. At the same time, the logit model was the only one to have a value of 33% and 50% for the sensitivity in the 50 and 100 sample sizes, respectively. Contrary to the previous results, in the low correlation case, with a 50-sample size, a value of 1 was obtained by the logit and the probit. In the 10 X's case, at the high correlation with a sample size of 50, the probit model performed better than its logit counterparts with a 1 sensitivity value of the first compared to a 0 value for the latter. Moreover, at sample sizes of 5,000 and 10,000, the SVM became comparable with its logit and probit counterparts. It further beat them in the 10,000-sample size. In the low correlation case, the performance of the logit and the probit was the best across all sample sizes.

6.3 AUC Measure-based Results

The AUC represents the overall quality of the model's predictions or its discrimination power. The results of the AUC measure will be discussed in this section and portrayed in Table 9 to Table 14. The AUC outcomes reveal that regardless of the cut point, the correlation level, the number of regressors, or the sample size, the logit, probit, DA, SVM, CART, and KNN *achieved an AUC of 1 or near 1* (0.99) thus excellent discriminatory performance in almost 15%, 14%, 3%, 8%, 0 and 0 of the total number of cases, respectively. Specifically, under the *median* cut-off, they attained AUC values higher than the 0.7 threshold in 54%, 55.5%, 49%, 46%, 27%, and 43% of those cases.

In the 3 regressors sample with a high correlation, the six models had good to excellent discriminatory performance for all sample sizes. Such a finding can be seen in Table 9. In the low correlation scenarios, the results indicated that the only fairly performing models are the logit and probit models in the scenarios of the 50, 100, and 500 sample sizes. In addition, the DA and the SVM performed reasonably well at 500 data sizes.

Comparatively, it is noted from Table 10 that the performance of the six models improved in the high correlation scenario when the number of covariates increased to **5**, except for the CART. That is, the performance of the CART was better in the 3-X's case. Furthermore, it could be seen that under the low correlation scenario and the sample sizes -1,000, 5,000, and 10,000, all the fitted models performed better when the number of regressors changed to 5.

In the 10 X scenarios – Table 11, under the high correlation, all models' performance was good to excellent under all sample sizes except the CART. It showed poor performance from sample sizes 50 through 200, then its performance has enhanced. Likewise, in the low correlation level, the same trend holds, yet the respective performance of the models decayed from being fair to good. Additionally, the CART did not deliver any performance at all sample sizes except at the 10,000-sample size.

Table 9: AUC, Median Cut-off, 3 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.920	0.924	0.899	0.902	0.739	0.875
100	0.960	0.960	0.940	0.942	0.799	0.924
200	0.941	0.945	0.900	0.892	0.814	0.886
500	0.950	0.950	0.932	0.932	0.839	0.904
1,000	0.939	0.939	0.924	0.929	0.879	0.927
5,000	0.944	0.944	0.938	0.939	0.910	0.929
10,000	0.936	0.936	0.935	0.934	0.918	0.928
Low						
50	0.728	0.729	0.641	0.639	0.606	0.619
100	0.748	0.748	0.656	0.652	0.616	0.561
200	0.660	0.660	0.580	0.580	0.660	0.640
500	0.734	0.742	0.702	0.694	0.613	0.605
1,000	0.660	0.668	0.624	0.612	0.640	0.572
5,000	0.643	0.644	0.640	0.639	0.617	0.620
10,000	0.665	0.666	0.663	0.662	0.652	0.626

Speaking about the 90th decile, at the 3 covariates case -plotted in Table 12- and at the high correlation level and the 50-sample size, the logit, the probit, the DA, and the SVM attained an average AUC above 0.7. Notably, the logit and the probit had perfect discrimination with a 1 AUC value. At size 100, only the logit and the probit attained almost an AUC value of 1, while all other models' AUC was at 0.5 and the CART AUC was less than 0.5.

Table 10: AUC, Median Cut-off, 5 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.896	0.910	0.905	0.913	0.700	0.836
100	0.971	0.971	0.935	0.946	0.677	0.836
200	0.810	0.977	0.956	0.975	0.756	0.895
500	0.999	0.999	0.985	0.982	0.792	0.934
1,000	0.995	0.774	0.991	0.990	0.838	0.916
5,000	1.000	1.000	0.986	0.997	0.891	0.951
10,000	0.999	0.999	0.993	0.997	0.907	0.954
Low						
50	0.748	0.748	0.659	0.654	0.601	0.627
100	0.710	0.711	0.701	0.663	0.696	0.554
200	0.625	0.624	0.586	0.604	0.583	0.562
500	0.669	0.662	0.622	0.621	0.653	0.621
1,000	0.720	0.720	0.696	0.696	0.648	0.687
5,000	0.718	0.720	0.715	0.719	0.690	0.676
10,000	0.694	0.694	0.690	0.692	0.662	0.670

Table 11: AUC, Median Cut-off, 10 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.849	0.863	0.869	0.886	0.622	0.783
100	0.953	0.979	0.937	0.931	0.671	0.912
200	0.663	0.961	0.925	0.951	0.666	0.846
500	0.853	0.853	0.979	0.978	0.725	0.882
1,000	0.996	0.996	0.970	0.976	0.722	0.890
5,000	0.999	0.999	0.985	0.997	0.779	0.917
10,000	1.000	1.000	0.990	0.996	0.788	0.924
Low						
50	0.776	0.789	0.806	0.798	0.667	0.691
100	0.908	0.903	0.838	0.862	0.542	0.780
200	0.960	0.940	0.920	0.920	0.620	0.720
500	0.895	0.895	0.855	0.855	0.645	0.718
1,000	0.900	0.900	0.888	0.888	0.680	0.796
5,000	0.888	0.887	0.883	0.882	0.712	0.828
10,000	0.901	0.901	0.898	0.901	0.740	0.857

Table 12: AUC, Ninetieth Decile Cut-off, 3 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	1.000	1.000	0.750	0.750	0.500	0.500
100	0.978	0.978	0.500	0.500	0.478	0.500
200	0.989	0.989	0.900	0.889	0.561	0.989
500	0.996	0.996	0.792	0.875	0.731	0.750
1,000	0.977	0.977	0.909	0.954	0.779	0.932
5,000	0.917	0.916	0.823	0.912	0.806	0.891
10,000	0.938	0.930	0.796	0.909	0.898	0.895
Low						
50	0.950	0.950	0.750	0.500	0.500	0.500
100	0.500	0.500	0.500	0.500	0.500	0.500
200	0.500	0.589	0.600	0.500	0.500	0.600
500	0.658	0.658	0.556	0.500	0.500	0.611
1,000	0.658	0.656	0.539	0.500	0.496	0.539
5,000	0.529	0.513	0.512	0.500	0.500	0.533
10,000	0.541	0.541	0.517	0.500	0.500	0.529

Evidently, the differences between models' AUC values decrease as the sample size rises. Further, from the 500 to 10,000 sample size all the models' AUCs were above the 0.7 threshold, indicating their discriminatory power increases when the sample size rise. At the low correlation level, sample size 50, the three statistical models yielded an average AUC of more than 0.7, whilst all ML

algorithms AUC stood at 0.5. Concerning the 5 X's set of data, it is revealed in Table 13 that at sample size 50 generated from the high correlation scenario, only the DA performed well with an AUC of 0.75. Furthermore, from a 500 sample size, the AUC value of all the fitted models was higher than 0.7. Hence, their performance is enhanced with a growing sample size. At the 5,000 and 10,000 sizes, the logit, probit, and SVM attained a 1 AUC.

Regarding the *low* correlation state, at the 50 size, only the logit and the probit models' AUC were 1. Otherwise, all the models performed poorly. According to Table 14, depicting 10 predictors, at the *high* correlation level and all sample sizes, the logit, probit, DA, and SVM performed fairly with AUC higher than 0.7. An exception at the 50-sample size, the SVM AUC dropped. Moreover, the logit and probit AUC values stood at almost 1 across all sample sizes except 5,000.

In conclusion, when the binary response variable was balanced -under the median cut-point- the discrimination power of the six models, especially for the low correlation scenarios, was higher than their counterparts when the data became more imbalanced (90^{th} decile cut-offs). Furthermore, the discrimination performance was the highest at the median cut-off, followed by the 90^{th} decile.

Table 13: AUC, Ninetieth Decile Cut-off, 5 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.501	0.501	0.751	0.501	0.500	0.500
100	0.978	0.978	0.957	0.957	0.530	0.978
200	0.989	1.000	0.800	0.878	0.489	0.600
500	0.983	0.984	0.812	0.896	0.752	0.777
1,000	0.750	0.750	0.807	0.943	0.678	0.731
5,000	0.999	0.999	0.812	0.992	0.811	0.864
10,000	0.997	0.998	0.834	0.994	0.838	0.896
Low						
50	1.000	1.000	0.500	0.500	0.500	0.500
100	0.477	0.477	0.500	0.500	0.500	0.500
200	0.561	0.644	0.489	0.500	0.500	0.500
500	0.658	0.658	0.551	0.500	0.500	0.556
1,000	0.522	0.522	0.517	0.500	0.549	0.487
5,000	0.545	0.545	0.559	0.500	0.531	0.542
10,000	0.571	0.565	0.553	0.500	0.562	0.544

6.4 Youden Index-based Results

A zero value for Youden's index indicates that the model yields the same proportion of positive cases for both the positive and the negative groups; hence, the model is useless. A value of 1 no false positives or false negatives, i.e., the model is perfect. Upon analyzing the resulting values of the YI presented in Appendix B, it revealed the same patterns uncovered through the AUC. The most surprising finding is that the CART and the KNN delivered high classification performance only once.

Table 14: AUC, Ninetieth Decile Cut-off, 10 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	1.000	1.000	1.000	0.500	0.500	0.500
100	1.000	1.000	1.000	1.000	0.500	0.500
200	1.000	1.000	0.800	0.800	0.500	0.500
500	0.979	0.979	0.751	0.895	0.547	0.744
1,000	0.975	0.975	0.773	0.975	0.607	0.636
5,000	0.749	0.749	0.844	0.991	0.688	0.773
10,000	0.999	0.999	0.825	0.990	0.721	0.796
Low						
50	0.500	0.502	0.464	0.502	0.500	0.500
100	0.957	0.957	0.728	0.707	0.478	0.500
200	0.750	0.750	0.667	0.655	0.614	0.500
500	0.817	0.786	0.750	0.781	0.704	0.679
1,000	0.777	0.777	0.744	0.768	0.636	0.589
5,000	0.854	0.865	0.763	0.838	0.558	0.631
10,000	0.851	0.851	0.764	0.844	0.585	0.656

7. Real Data Application

Extending this analysis to real data is crucial to ensure the findings' generalizability. Real-world data often exhibits complexities, noise, and variations that may not be fully captured in simulated datasets. By applying these classification techniques to real data, one can assess their effectiveness in handling the intricacies and uncertainties in practical scenarios. The Pima Indian Diabetes dataset will be utilized. It is originally from the US National Institute of Diabetes and Digestive and Kidney Diseases. The dataset objective is to diagnostically predict whether an individual has diabetes based on specific diagnostic measurements included in the dataset. The data was retrieved from R software using the command "data(PimaIndiansDiabetes)" under the mlbench (version 2.1-3.1) package[1].

The data consists of 768 females who are 21 years or more of Pima Indian heritage and living near Phoenix, Arizona. The independent variables are 8, including the number of pregnancies the female has had, their BMI, age, insulin, glucose, and pressure levels, among others. Their descriptive statistics in terms of mean, standard deviation, minimum, and maximum are reported in Table 15.

In this dataset, the predictand indicates that the diabetes test result is either negative or positive. The regressand contains 65% of the observations belonging to the negative group, while 35% are classified as positive. This case can be compared to the case of the 0.75 cut point. Since the data contains 8 regressors, two scenarios will be considered: one with 3 predictors and the other with 5 covariates. The three variables that were chosen to be used as covariates are glucose (plasma

[1] In addition to the R command, the dataset is available through Kaggle (https://www.kaggle.com/uciml/pima-indians-diabetes-database) and is available via a CC0: Public Domain License.

glucose concentration), pressure (diastolic blood pressure), and triceps (triceps skin fold thickness). Regarding the 5-regressors case, the same three variables were employed in addition to mass (body mass index) and pedigree (diabetes pedigree function).

Table 15: Descriptive Statistics of the Diabetes Dataset

Variable \ Stat	Mean	S.D	Minimum	Maximum	Skewness	Kurtosis
Pregnant	3.845	3.370	0	17	0.902	0.159
Glucose	120.895	31.973	0	199	0.174	0.641
Pressure	69.105	19.356	0	122	-1.844	5.180
Triceps	20.536	15.952	0	99	0.109	-0.520
Insulin	79.799	115.244	0	846	2.272	7.214
Mass	31.993	7.884	0	67.1	-0.429	3.290
Pedigree	0.472	0.331	0.078	2.42	1.920	5.595
Age	33.241	11.760	21	81	1.130	0.643

Regarding the analysis process, the same procedures followed in the simulation study will be pursued regarding the data split into a 75% training set and a 25% testing set. Following, the six models will be trained using the training set. Subsequently, the fitted and trained models will be used to predict the outcome of the data points in the testing set.

Finally, the performance of the models will be evaluated using performance metrics such as accuracy, sensitivity, specificity, precision, F1, AUC, and YI. These measures will be reported in Figure 1 and Figure 2 for the 3 and 5-regressors cases, respectively. Such figures portray the performance of the 6 methods across the four dimensions of evaluation measures stated earlier. Hence, the resulting shape should be a four-dimensional radar plot; each axis represents one performance measure, forming a closed polygon. To enumerate, the area under this resulting polygon can be interpreted as an overall performance score. A larger area would indicate better performance or higher proficiency across the four dimensions, while a smaller area may suggest weakness or lower performance. Additionally, the resulting values are depicted in Tables 16 and 17.

The resulting values refer to the logit regression model, followed by the probit regression model, being the best-predicting models in the two cases. This conclusion resembles the outcome of the simulation study.

Table 16: Performance Measures, 3 Xs, Diabetes Dataset

Model \ Measure	Acc	Sens	Spec	Prec	F1	AUC	YI	Index
Log	**0.792**	**0.606**	0.889	0.741	**0.667**	**0.747**	**0.495**	**1.338**
Prob	0.786	**0.606**	0.881	0.727	0.661	0.744	0.487	1.313
DA	0.76	0.5	0.897	0.717	0.589	0.698	0.397	1.124
SVM	0.766	0.485	**0.913**	**0.744**	0.587	0.699	0.398	1.141
CART	0.740	0.424	0.905	0.700	0.528	0.665	0.329	0.980
KNN	0.750	0.455	0.905	0.714	0.556	0.68	0.36	1.048

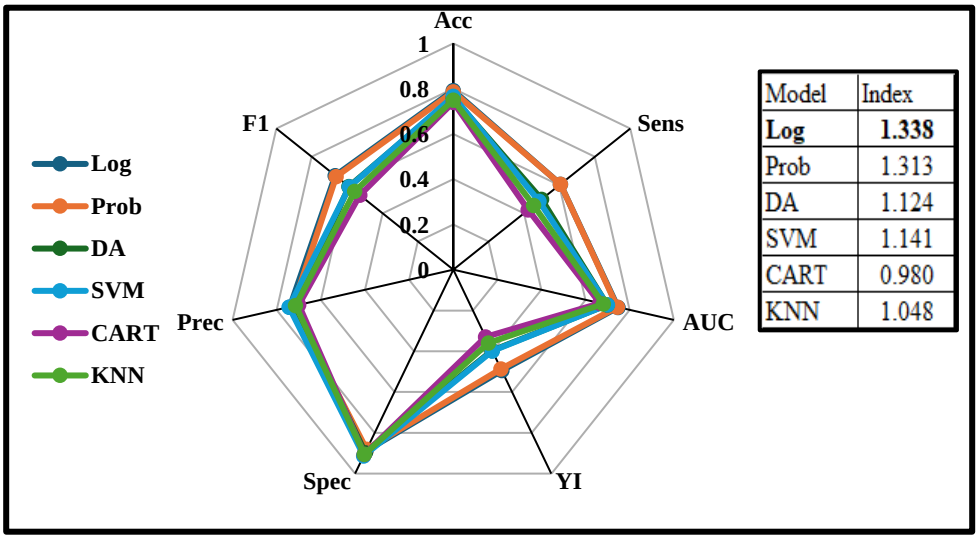

Figure 1: Performance Measures, 3 Xs, Diabetes Dataset

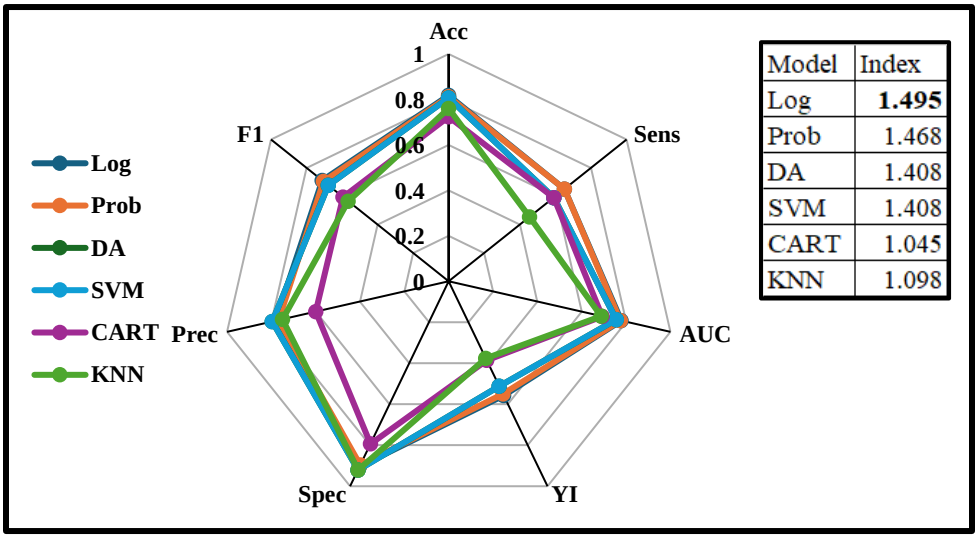

Figure 2: Performance Measures, 5 Xs, Diabetes Dataset

Table 17: Performance Measures, 5 Xs, Diabetes Dataset

Model \ Measure	Acc	Sens	Spec	Prec	F1	AUC	YI	Index
Log	**0.818**	**0.652**	0.905	0.782	**0.711**	**0.778**	**0.557**	**1.495**
Prob	0.813	**0.652**	0.897	0.768	0.705	0.774	0.549	1.468
DA	0.807	0.591	**0.921**	**0.796**	0.678	0.756	0.512	1.408
SVM	0.807	0.591	**0.921**	**0.796**	0.678	0.756	0.512	1.408
CART	0.724	0.591	0.794	0.600	0.595	0.692	0.385	1.045
KNN	0.760	0.455	**0.921**	0.750	0.566	0.688	0.376	1.098

8. Discussion and conclusion

This section allows the researcher to summarize and discuss the study's findings. Overall, the well-grounded, theory-based models of the logit regression as well as the probit regression resulted in fair -(AUC≥ 0.7)- to perfect -(AUC=1)- classification performance in about 52% and 53% of the

whole set of 84 simulated scenarios, respectively. The DA followed them by 46%, then the SVM, the KNN, and the CART in descending order. Likewise, the two models yielded the most accurate predictions in 78.5% and 83.6% of the simulated datasets. Then, the SVM, the DA, the KNN, and the CART followed. At the *high* correlation level with the 90th decile cut-point, the SVM algorithm becomes the best at the sample sizes of 1,000, 5,000, and 10,000. As the number of regressors increases from 3 through 10, the SVM classifier's performance improves at the *high* level of correlation. As the distribution of the two groups of the response variable becomes more imbalanced, the SVM -along with the logit and the probit models- was the only model that did not suffer.

In order to conclude the results of the study, Table 18 provides insights regarding the best-performing model in each scenario according to the number of regressors, the sample size, and the cut-off point. These findings are based on the most critical performance measures in the study, for instance, the accuracy, sensitivity, and the AUC. The boldface cells refer to the cases where the same model/s excelled according to the three performance criteria.

Further, Fig. 3 portrays the performance of the 6 methods across the four dimensions of evaluation measures stated earlier. Hence, the resulting shape should be a four-dimensional radar plot comprising an overall index similar to the one calculated and illustrated in Section 7, where each axis represents one performance measure, forming a closed polygon.

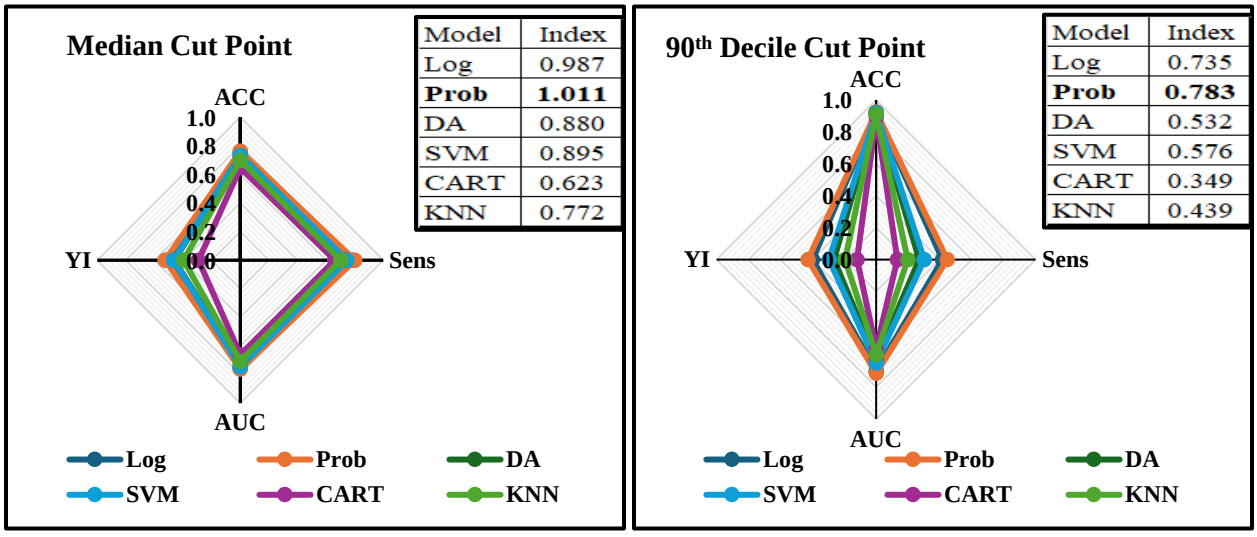

Figure 3: Average Measures, all Scenarios

Table 18: Results Summary Across All Scenarios

Xs	N	0.5			0.9		
		Acc	**Sen**	**AUC**	**Acc**	**Sen**	**AUC**
3	**50**	**Prob**			Log-Prob-DA	Log-Prob	
	100	**Log-Prob**			**Log-Prob**		
	200	Log-Prob	Log	Log-Prob	KNN	Prob	KNN
	500	Prob	Log	Prob	**Log-Prob**		
	1,000	Prob	Log	Prob	Log	Log-Prob	
	5,000	Log-Prob	SVM	Log-Prob	Log-Prob	Log	
	10,000	DA	Log	Log-Prob	Log-Prob	Log	
5	**50**	**Prob**			**Log**		
	100	**Log-Prob**			KNN	All but CART	KNN
	200	Prob			**Prob**		
	500	Log	Prob	Log	**Log-Prob**		
	1,000	Log			**SVM**		
	5,000	Log-Prob	Log	Prob	**Log-Prob**		
	10,000	Log-Prob	Prob	Log-Prob	Log-Prob	Log	
10	**50**	Prob	SVM		SVM	Prob	
	100	**Prob**			**Log-Prob**		
	200	**Prob**			**Prob**		
	500	**DA-SVM**		DA	Log-Prob	Prob	Log-Prob
	1,000	Log-Prob	SVM	Log-Prob	Prob	Log-Prob	
	5,000	Log-Prob	Prob	Log-Prob	Log-Prob	Prob	Prob
	10,000	Log-Prob	SVM	Log-Prob-SVM	**SVM**		

Generally, the performance of the KNN algorithm is always superior to its CART counterpart. In the most imbalanced dependent variable distribution scenario, the KNN was revealed to be the best-discriminating classifier among all others under the high correlation level coupled with 3X's and 5X's at sample sizes of 200 and 100, respectively. At the low correlation level, all models' performance significantly improves as the number of regressors rises. In the high correlation case, under all regressand distributions, the CART performance deteriorates as the number of predictors grows. By contrast, its performance is enhanced at the low correlation level with the rising number of predictors. In the high correlation situation, under all dependent variable distributions, the KNN performance worsens as the number of predictors upsurges. By way of contrast, at the low correlation level, its performance improves with increasing covariates.

Along with the logit and the probit, the DA was the sole significantly classifying model at the sample size 50 with low correlation, 3X's, and 0.9 cut-point. At the cut-point of the ninetieth decile, the DA was the only model that attained a fair classification power among all other models at the 50-sample size, 5-covariates, and high correlation level, with an AUC value larger than 0.7.

In most scenarios, the SVM resulted in a classification performance similar to the DA. However, there were some anomalies where one excelled, including:

- At the 0.9 cut-off, 3 X's, large sample sizes of 500 up to 10,000, and a high correlation level, the SVM's classification performance was superior to the DA.
- The DA performed better than the SVM with an AUC of 1 at the 0.9 cut-off, 10 X's, sample size of 50, and high level of correlation.
- The SVM performance was way superior to the DA at the 0.9 cut-off, 10 X's, large sample sizes of 500 up to 10,000, and high level of correlation. Similarly, in the low correlation case with sample sizes of 5,000 and 10,000.

Referring to the research questions, upon conducting this study, it was revealed that analyzing the same dataset by the selected models did not result in the same classification performance. The logistic and the probit regressions were the best-fit models in most simulated scenarios. Further, the application of ML algorithms did not enhance the predictive accuracy at the expense of classic statistical models, as claimed by other studies. Instead, the traditional statistical models exhibited better performance than classification models. Moreover, the probit model proved to be as excellent as the logit model, if not better than it. Hence, it would be recommended that future studies include it compared to the various ML techniques besides the logistic model. However, machine learning techniques, with their ability to automatically learn from data and discern complex patterns, have proven invaluable in many big data applications. For instance, deep learning models have achieved unprecedented accuracy in image recognition tasks by learning hierarchical representations directly from pixel data, thereby bypassing the need for manual feature engineering, which is often required in traditional statistical approaches.

After conducting a simulation to compare statistical models with ML algorithms, several possible future avenues and directions of work can be explored, such as expanding the study by including other ML algorithms, such as neural networks, naive Bayes, or random forests. Further, a thorough hyperparameter tuning for each model is performed to optimize their performance. Utilize techniques like random search or Bayesian optimization. Moreover, investigate the interpretability and explainability of the models on real data sets; hence, the regressors are meaningful. While machine learning models can be alternatives to traditional statistical models, they are sometimes seen as "black boxes." Furthermore, it is essential to depart from the normality assumption of the regressors in order to check the performance of the six models. Hence, data should be generated using other distributions. By pursuing these potential future works, researchers can further enhance the understanding of the evaluated models, explore their performance in real-world scenarios, and uncover opportunities for improving their accuracy, efficiency, and interpretability.

References

Alsoruji, A. H., Binhimd, S., & Abd Elaal, M. K. (2018). A Comparison of Univariate Probit and Logit Models Using Simulation. *Applied Mathematical Sciences*, *12* (4), 185-204. https://doi.org/10.12988/ams.2018.818

Amemiya, T. (1981). Qualitative Response Models: A survey. *Journal of Economic Literature*, *19* (4), 1483-1536. https://www.jstor.org/stable/2724565

Arthur, S. (1959). Some Studies in Machine Learning Using the Game of Checkers. IBM Journal of Research and Development, 3 (3), 210–229. https://doi.org/10.1147/rd.441.0206

Awad, M., & Khanna, R. (2015). Support Vector Machines for Classification. In *Efficient Learning Machines* (pp. 39-66). Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4302-5990-9_3

Bichler, M., & Kiss, C. (2004). A Comparison of Logistic Regression, K-Nearest Neighbor, And Decision Tree Induction for Campaign Management. *AMCIS 2004 Proceedings*, 230. https://aisel.aisnet.org/amcis2004/230

Cakmakyapan, S., & Goktas, A. (2013). A Comparison of Binary Logit and Probit Models with A Simulation Study. *Journal of Social and Economic statistics*, *2* (1), 1-17.

Côté, M., Osseni, M. A., Brassard, D., Carbonneau, É., Robitaille, J., Vohl, M. C., ... & Lamarche, B. (2022). Are Machine Learning Algorithms More Accurate in Predicting Vegetable and Fruit Consumption Than Traditional Statistical Models? An exploratory analysis. *Frontiers in Nutrition*, *9*, 740898. https://doi.org/10.3389/fnut.2022.740898

Faisal, M., Scally, A., Howes, R., Beatson, K., Richardson, D., & Mohammed, M. A. (2020). A Comparison of Logistic Regression Models with Alternative Machine Learning Methods to Predict the Risk of In-Hospital Mortality in Emergency Medical Admissions Via External Validation. *Health Informatics Journal*, *26* (1), 34-44. https://doi.org/10.1177/1460458219881360

Farhat, A., & Cheok, K. C. (2021). Classifying Driver Attention Level Using Logistic Regression and Support Vector Machine. *EPiC Series in Computing*, *75*, 32-40. https://doi.org/10.29007/gr9w

Fawcett, T. (2006). An Introduction to ROC Analysis. *Pattern Recognition Letters*, *27* (8), 861-874. https://doi.org/10.1016/j.patrec.2005.10.010

Greene, W. H. (2012). *Econometric Analysis*. Upper Saddle River, N.J: Prentice Hall.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. New York, USA: Springer New York Inc. https://doi.org/10.1007/978-0-387-21606-5

Hu, X., Zhang, X., & Lovrich, N. (2021). Public Perceptions of Police Behavior During Traffic Stops: Logistic Regression and Machine Learning Approaches Compared. *Journal of Computational Social Science*, *4*, 355-380. https://doi.org/10.1007/s42001-020-00079-4

Itoo, F., & Singh, S. (2021). Comparison and Analysis of Logistic Regression, Naïve Bayes And KNN Machine Learning Algorithms for Credit Card Fraud Detection. *International Journal of Information Technology*, *13* (4), 1503-1511. https://doi.org/10.1007/s41870-020-00430-y

Kurt, I., Ture, M., & Kurum, A. T. (2008). Comparing Performances of Logistic Regression, Classification and Regression Tree, And Neural Networks for Predicting Coronary Artery Disease. *Expert Systems with Applications*, *34* (1), 366-374. https://doi.org/10.1016/j.eswa.2006.09.004

Liew, B. X., Kovacs, F. M., Rügamer, D., & Royuela, A. (2022). Machine Learning Versus Logistic Regression for Prognostic Modelling in Individuals with Non-Specific Neck Pain. *European Spine Journal*, *31* (8), 2082-2091. https://doi.org/10.1007/s00586-022-07188-w

Liu, Y. Y., Yang, M., Ramsay, M., Li, X. S., & Coid, J. W. (2011). A Comparison of Logistic Regression, Classification and Regression Tree, And Neural Networks Models in Predicting Violent Re-Offending. *Journal of Quantitative Criminology*, *27* (4), 547-573. https://doi.org/10.1007/s10940-011-9137-7

Lynam, A. L., Dennis, J. M., Owen, K. R., Oram, R. A., Jones, A. G., Shields, B. M., & Ferrat, L. A. (2020). Logistic Regression Has Similar Performance to Optimised Machine Learning Algorithms in A Clinical Setting: Application to The Discrimination Between Type 1 And Type 2 Diabetes in Young Adults. *Diagnostic and Prognostic Research*, *4* (1), 1-10. https://doi.org/10.1186/s41512-020-00075-2

Maddala, G. S. (1986). *Limited-dependent and Qualitative Variables in Econometrics*. Cambridge University Press.

Meyer, D., & Wien, F. T. (2021). Support Vector Machines. *R News*, *1* (3), 23-26.

Musa, A. B. (2013). Comparative Study on Classification Performance Between Support Vector Machine and Logistic Regression. *International Journal of Machine Learning and Cybernetics*, *4* (1), 13-24. https://doi.org/10.1007/s13042-012-0068-x

Park, H. M. (2015). *Regression Models for Ordinal and Nominal Dependent Variables Using SAS, Stata, LIMDEP, and SPSS.* Working Paper. The University Information Technology Services (UITS) Center for Statistical and Mathematical Computing, Indiana University. https://hdl.handle.net/2022/19741

Pohar, M., Blas, M., & Turk, S. (2004). Comparison of Logistic Regression and Linear Discriminant Analysis: A Simulation Study. *Metodoloski zvezki*, *1* (1), 143-161.

Powers, D., & Xie, Y. (2008). *Statistical Methods for Categorical Data Analysis*. Emerald Group Publishing.

Prempeh, E. A. (2009). *Comparative Study of The Logistic Regression Analysis and The Discriminant Analysis* (Doctoral dissertation, University of Cape Coast).

Press, S. J., & Wilson, S. (1978). Choosing Between Logistic Regression and Discriminant Analysis. *Journal of the American Statistical Association*, *73* (364), 699-705. https://doi.org/10.1080/01621459.1978.10480080

Scholz, M., & Wimmer, T. (2021). A comparison of classification methods across different data complexity scenarios and datasets. Expert Systems with Applications, 168, 114217. https://doi.org/10.1016/j.eswa.2020.114217

Settouti, N., Bechar, M. E. A., & Chikh, M. A. (2016). Statistical Comparisons of The Top 10 Algorithms in Data Mining for Classification Task. *International Journal of Interactive Multimedia and Artificial Intelligence*, *4* (1), 46-51. http://doi.org/10.9781/ijimai.2016.419

Timofeev, R. (2004). Classification and Regression Trees (CART) Theory and Applications. *Humboldt University, Berlin*, 1-40.

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., ... & Steinberg, D. (2008). Top 10 Algorithms in Data Mining. *Knowledge and Information Systems*, *14* (1), 1-37. https://doi.org/10.1007/s10115-007-0114-2

Appendix A: Variance-Covariance Matrices

A.1 The 3-Regressors Matrices

Given below are the resulting 4×4 matrices to be utilized in the cases involving 3 regressors along with one regressand. The matrices are provided in the order of moderate to high -Σ_{M-H}, and low -Σ_L, respectively.

$$\begin{bmatrix} 1.000 & 0.605 & -0.623 & 0.719 \\ 0.605 & 1.000 & -0.119 & 0.184 \\ -0.623 & -0.119 & 1.000 & -0.182 \\ 0.719 & 0.184 & -0.182 & 1.000 \end{bmatrix} \quad \begin{bmatrix} 1.000 & 0.349 & -0.291 & 0.207 \\ 0.349 & 1.000 & -0.060 & -0.028 \\ -0.291 & -0.060 & 1.000 & -0.064 \\ 0.207 & -0.028 & -0.064 & 1.000 \end{bmatrix}$$

A.2 The 5-Regressors Matrices

Given below are the resulting 6×6 matrices to be utilized in the cases involving 5 regressors along with one regressand. The matrices are provided in the order of moderate to high -Σ_{M-H}, and low -Σ_L, respectively.

$$\begin{bmatrix} 1.000 & 0.691 & -0.397 & -0.413 & 0.593 & 0.595 \\ 0.691 & 1.000 & 0.018 & 0.170 & 0.233 & 0.171 \\ -0.397 & 0.018 & 1.000 & -0.218 & 0.232 & -0.263 \\ -0.413 & 0.170 & -0.218 & 1.000 & -0.281 & -0.294 \\ 0.593 & 0.233 & 0.232 & -0.281 & 1.000 & 0.287 \\ 0.595 & 0.171 & -0.263 & -0.294 & 0.287 & 1.000 \end{bmatrix} \quad \begin{bmatrix} 1.000 & 0.254 & 0.313 & -0.237 & -0.234 & 0.309 \\ 0.254 & 1.000 & -0.049 & -0.091 & 0.002 & 0.091 \\ 0.313 & -0.049 & 1.000 & -0.049 & -0.034 & -0.099 \\ -0.237 & -0.091 & -0.049 & 1.000 & -0.015 & 0.089 \\ -0.234 & 0.002 & -0.034 & -0.015 & 1.000 & -0.060 \\ 0.309 & 0.091 & -0.099 & 0.089 & -0.060 & 1.000 \end{bmatrix}$$

A.3 The 10-Regressors Matrices

Given below are the resulting 11×11 matrices to be utilized in the cases involving 10 regressors along with one regressand. The matrices are provided in the order of moderate to high -Σ_{M-H}, and low -Σ_{L}, respectively.

$$\begin{bmatrix}
1.000 & 0.298 & -0.445 & -0.364 & -0.495 & 0.497 & 0.508 & 0.544 & 0.477 & 0.516 & 0.463 \\
0.298 & 1.000 & -0.364 & 0.120 & -0.099 & 0.086 & 0.146 & 0.282 & -0.353 & -0.313 & 0.360 \\
-0.445 & -0.364 & 1.000 & 0.240 & 0.342 & 0.145 & 0.221 & -0.067 & 0.292 & -0.278 & -0.276 \\
-0.364 & 0.120 & 0.240 & 1.000 & -0.315 & 0.224 & -0.125 & -0.343 & 0.120 & -0.324 & 0.335 \\
-0.495 & -0.099 & 0.342 & -0.315 & 1.000 & -0.356 & -0.078 & -0.292 & -0.374 & -0.296 & -0.316 \\
0.497 & 0.086 & 0.145 & 0.224 & -0.356 & 1.000 & -0.114 & 0.224 & 0.405 & -0.236 & -0.067 \\
0.508 & 0.146 & 0.221 & -0.125 & -0.078 & -0.114 & 1.000 & 0.267 & 0.255 & 0.284 & 0.411 \\
0.544 & 0.282 & -0.067 & -0.343 & -0.292 & 0.224 & 0.267 & 1.000 & -0.248 & 0.385 & 0.191 \\
0.477 & -0.353 & 0.292 & 0.120 & -0.374 & 0.405 & 0.255 & -0.248 & 1.000 & 0.267 & -0.049 \\
0.516 & -0.313 & -0.278 & -0.324 & -0.296 & -0.236 & 0.284 & 0.385 & 0.267 & 1.000 & 0.083 \\
0.463 & 0.360 & -0.276 & 0.335 & -0.316 & -0.067 & 0.411 & 0.191 & -0.049 & 0.083 & 1.000
\end{bmatrix}$$

$$\begin{bmatrix}
1.000 & 0.312 & 0.374 & 0.239 & 0.213 & -0.247 & -0.217 & -0.289 & -0.231 & 0.302 & 0.304 \\
0.312 & 1.000 & -0.089 & 0.063 & -0.062 & -0.007 & -0.021 & -0.052 & -0.089 & -0.069 & -0.065 \\
0.374 & -0.089 & 1.000 & -0.085 & -0.075 & 0.021 & 0.042 & 0.015 & -0.026 & -0.087 & 0.027 \\
0.239 & 0.063 & -0.085 & 1.000 & 0.088 & 0.079 & -0.054 & -0.051 & 0.069 & -0.083 & -0.077 \\
0.213 & -0.062 & -0.075 & 0.088 & 1.000 & 0.062 & -0.015 & 0 & 0.016 & -0.029 & 0.017 \\
-0.247 & -0.007 & 0.021 & 0.079 & 0.062 & 1.000 & -0.017 & -0.031 & 0.076 & 0.007 & 0.049 \\
-0.217 & -0.021 & 0.042 & -0.054 & -0.015 & -0.017 & 1.000 & -0.05 & -0.068 & -0.019 & 0.058 \\
-0.289 & -0.052 & 0.015 & -0.051 & 0 & -0.031 & -0.05 & 1.000 & -0.046 & 0.051 & -0.079 \\
-0.231 & -0.089 & -0.026 & 0.069 & 0.016 & 0.076 & -0.068 & -0.046 & 1.000 & 0.052 & -0.044 \\
0.302 & -0.069 & -0.087 & -0.083 & -0.029 & 0.007 & -0.019 & 0.051 & 0.052 & 1.000 & 0.021 \\
0.304 & -0.065 & 0.027 & -0.077 & 0.017 & 0.049 & 0.058 & -0.079 & -0.044 & 0.021 & 1.000
\end{bmatrix}$$

Appendix B: YI Measure Results

In this Appendix, the tables of theYI of the six methods as a result of the simulation process are presented.

Table B1: YI, Median Cut-off, 3 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.839	0.847	0.797	0.803	0.474	0.749
100	0.920	0.920	0.880	0.883	0.598	0.849
200	0.882	0.889	0.800	0.784	0.629	0.772
500	0.900	0.900	0.863	0.864	0.677	0.807
1,000	0.877	0.877	0.848	0.858	0.758	0.853
5,000	0.889	0.889	0.876	0.877	0.820	0.858
10,000	0.872	0.872	0.869	0.868	0.837	0.856
Low						
50	0.457	0.457	0.236	0.229	0.133	0.164
100	0.497	0.497	0.313	0.305	0.214	0.121
200	0.320	0.320	0.160	0.161	0.320	0.280
500	0.468	0.484	0.403	0.387	0.226	0.210
1,000	0.320	0.336	0.248	0.224	0.280	0.144
5,000	0.286	0.288	0.280	0.278	0.234	0.240
10,000	0.330	0.331	0.326	0.325	0.304	0.252

Table B2: YI, Median Cut-off, 5 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.793	0.820	0.810	0.827	0.389	0.671
100	0.942	0.942	0.869	0.893	0.355	0.672
200	0.621	0.953	0.913	0.950	0.512	0.789
500	0.998	0.998	0.969	0.964	0.583	0.868
1,000	0.989	0.547	0.983	0.980	0.677	0.832
5,000	1.000	1.000	0.973	0.994	0.782	0.901
10,000	0.999	0.999	0.986	0.995	0.814	0.908
Low						
50	0.496	0.497	0.283	0.274	0.116	0.199
100	0.421	0.421	0.402	0.326	0.391	0.109
200	0.250	0.249	0.171	0.208	0.167	0.125
500	0.339	0.323	0.243	0.243	0.306	0.242
1,000	0.440	0.440	0.392	0.392	0.296	0.375
5,000	0.437	0.440	0.430	0.438	0.381	0.352
10,000	0.388	0.387	0.379	0.383	0.325	0.340

Table B3: YI, Median Cut-off, 10 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.698	0.726	0.737	0.772	0.287	0.590
100	0.907	0.957	0.873	0.861	0.237	0.824
200	0.326	0.922	0.851	0.901	0.476	0.600
500	0.707	0.707	0.958	0.956	0.399	0.793
1,000	0.992	0.992	0.939	0.953	0.537	0.791
5,000	0.998	0.998	0.969	0.995	0.600	0.851
10,000	0.999	0.999	0.979	0.993	0.660	0.854
Low						
50	0.553	0.576	0.611	0.594	0.333	0.372
100	0.817	0.806	0.676	0.725	0.083	0.561
200	0.920	0.880	0.840	0.840	0.240	0.440
500	0.790	0.790	0.710	0.710	0.290	0.435
1,000	0.800	0.800	0.776	0.776	0.360	0.592
5,000	0.776	0.774	0.766	0.763	0.424	0.656
10,000	0.802	0.802	0.795	0.802	0.480	0.714

Table B4: YI, Ninetieth Decile Cut-off, 3 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	1.000	1.000	0.500	0.500	0.000	0.000
100	0.957	0.957	0.000	0.000	-0.043	0.000
200	0.978	0.978	0.800	0.778	0.121	0.978
500	0.991	0.991	0.583	0.750	0.461	0.500
1,000	0.954	0.954	0.818	0.909	0.557	0.863
5,000	0.833	0.832	0.645	0.823	0.612	0.783
10,000	0.876	0.860	0.592	0.817	0.797	0.789
Low						
50	0.900	0.900	0.500	0.000	0.000	0.000
100	0.000	0.000	0.000	0.000	0.000	0.000
200	0.000	0.178	0.200	0.000	0.000	0.200
500	0.316	0.316	0.111	0.000	0.000	0.222
1,000	0.316	0.311	0.079	0.000	-0.009	0.079
5,000	0.059	0.027	0.024	0.000	0.000	0.067
10,000	0.082	0.082	0.034	0.000	0.000	0.059

Table B5: YI, Ninetieth Decile Cut-off, 5 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	0.002	0.002	0.501	0.002	0.000	0.000
100	0.957	0.957	0.913	0.913	0.060	0.957
200	0.978	1.000	0.600	0.756	-0.021	0.200
500	0.967	0.968	0.625	0.792	0.504	0.553
1,000	0.500	0.500	0.614	0.887	0.356	0.462
5,000	0.999	0.999	0.625	0.985	0.622	0.728
10,000	0.994	0.995	0.668	0.989	0.676	0.793
Low						
50	1.000	1.000	0.000	0.000	0.000	0.000
100	-0.045	-0.045	0.000	0.000	0.000	0.000
200	0.121	0.288	-0.023	0.000	0.000	0.000
500	0.316	0.316	0.102	0.000	0.000	0.111
1,000	0.043	0.043	0.035	0.000	0.098	-0.026
5,000	0.090	0.090	0.118	0.000	0.063	0.085
10,000	0.142	0.130	0.106	0.000	0.123	0.088

Table B6: YI, Ninetieth Decile Cut-off, 10 Regressors

Row Labels	Log	Prob	DA	SVM	CART	KNN
High						
50	1.000	1.000	1.000	0.000	0.000	0.000
100	1.000	1.000	1.000	1.000	0.000	0.000
200	1.000	1.000	0.600	0.600	0.000	0.000
500	0.957	0.957	0.503	0.790	0.093	0.489
1,000	0.950	0.950	0.545	0.950	0.214	0.273
5,000	0.497	0.498	0.687	0.982	0.376	0.545
10,000	0.999	0.999	0.650	0.979	0.441	0.592
Low						
50	0.000	0.004	-0.071	0.004	0.000	0.000
100	0.913	0.913	0.457	0.413	-0.043	0.000
200	0.500	0.500	0.333	0.311	0.228	0.000
500	0.634	0.571	0.500	0.562	0.408	0.357
1,000	0.554	0.554	0.487	0.537	0.271	0.179
5,000	0.708	0.730	0.526	0.677	0.116	0.261
10,000	0.703	0.702	0.529	0.688	0.171	0.313