Journal of Informatics and Web Engineering

Vol. 2 No. 2 (September 2023) eISSN: 2821-370X

Predicting Travel Insurance Purchases in an Insurance Firm through Machine Learning Methods after COVID-19

Shiuh Tong Lim[1], Joe Yee Yuan[1], Khai Wah Khaw[1*], XinYing Chew[2]

[1]School of Management, Universiti Sains Malaysia, 11800, Penang, Malaysia

[2]School of Computer Sciences, Universiti Sains Malaysia, 11800, Penang, Malaysia

**corresponding author: (khaiwah@usm.my; 0000-0003-2646-6477)*

Abstract – Travel insurance serves as a crucial financial safeguard, offering coverage against unforeseen expenses and losses incurred during travel. With the advent of the proliferation of insurance types and the amplified demand for Covid-related coverage, insurance companies face the imperative task of accurately predicting customers' likelihood to purchase insurance. This can assist the insurance providers in focusing on the most lucrative clients and boosting sales. By employing advanced machine learning techniques, this study aims to forecast the consumer segments most inclined to acquire travel insurance, allowing targeted strategies to be developed. A comprehensive analysis was carried out on a Kaggle dataset comprising prior clients of a travel insurance firm utilizing the K-Nearest Neighbors (KNN), Decision Tree Classifier (DT), Support Vector Machines (SVM), Naïve Bayes (NB), Logistic Regression (LR), and Random Forest (RF) models. Extensive data cleaning was done before model building. Performance evaluation was then based on accuracy, F1 score, and the Area Under Curve (AUC) with Receiver Operating Characteristics (ROC) curve. Inexplicably, KNN outperformed other models, achieving an accuracy of 0.81, precision of 0.82, recall of 0.82, F1 score of 0.80, and an AUC of 0.78. The findings of this study are a valuable guide for deploying machine learning algorithms in predicting travel insurance purchases, thus empowering insurance companies to target the most lucrative clientele and bolster revenue generation.

Keywords—Travel insurance, Machine learning techniques, Predictive modelling, Customer segmentation, Revenue optimization

Received: 17 June 2023; Accepted: 7 July 2023; Published: 16 September 2023

I. INTRODUCTION

Our daily lives are often filled with uncertainties, particularly when we embark on a journey. All too frequently, incidents like lost luggage, delayed flights, mishaps on public transit, foodborne infections, injuries, and health problems occur. To mitigate potential losses, individuals often turn to travel insurance. Travel insurance is specially designed to cover expenses and losses incurred due to unanticipated events during travel [1]. It acts as a financial buffer against the risk and mishaps that travellers may encounter while on vacation, especially after the COVID-19 pandemic. This is because the pandemic has become a significant change in all sectors [2]. The scope of coverage varied depending on the policy encompassing risks such as baggage loss, travel deposit loss, trip cancellation, medical costs including coverage for COVID-19, physical injuries, travel service provider bankruptcy, personal liability for a

Journal of Informatics and Web Engineering
https://doi.org/10.33093/jiwe.2023.2.2.4

Published by MMU Press. URL: https://journals.mmupress.com/jiwe

tort committed against citizens of the host countries, legal expenses, and even unforeseen situations typically not covered by standard policies, like terrorism and denied boarding [3].

In the current landscape, the COVID-19 pandemic has heightened the risks associated with travel, starting from the various modes of transportation. For instance, flight travel requires waiting in security lines and airport terminals full of regularly touched surfaces. Furthermore, maintaining social distancing becomes challenging during the flight itself. Additionally, travellers frequently experience last-minute flight disruptions due to travel restrictions by destination countries. Upon reaching their destinations, individuals must remain vigilant about COVID-19 precautions at every location they visit, including their accommodations, activities, and dining establishments. In the unfortunate event of falling ill, they can have trouble locating proper isolation facilities as well as accessing medical care that is not covered by their insurance [4].

The World Tourism Organization (2020) predicted that the tourism sector would contract due to a drop in travel demand brought on by the COVID-19 pandemic. However, a noteworthy trend is a significant rise in the adoption of travel insurance [6]. This idea was supported by an article highlighting that COVID-19's impact has directly led to a 20% to 30% increase in travel insurance premiums [7]. With a heightened awareness of potential health and other issues associated with travel, it is anticipated that the uptake rate of travel insurance will also witness an upsurge.

Travelling during the COVID-19 epidemic entails greatly increased risks and uncertainties, with the likelihood of travel plans being disrupted by unexpected border closures, flight cancellations, and the establishment of virus hotspots increasing. Consequently, prospective travellers and vacationers now place a high priority on travel insurance. Insurance companies have swiftly responded to the pandemic, launching specialized products tailored to the current conditions prevalent in the travel landscape. When the covid-19 outbreak has formally labelled a pandemic, several insurance companies refused to cover or suspended payments for such claims. Nonetheless, in response to evolving circumstances, numerous insurance, tour, and travel companies now offer dedicated travel insurance services, particularly in light of the ongoing COVID-19 pandemic.

Therefore, it is essential to understand and understand the variables that affect clients' decisions to get travel insurance so they can have immediate action depending on the always-changing circumstances. However, very few studies have been on this issue for the past years. Most of the paper discussed claiming intention instead of purchase intention. Besides, regarding insurance purchase intention, most of the paper focused on health insurance. Hence, in this work, the prediction of travel insurance purchases was conducted by implementing six supervised machine learning models on the dataset of previous travel insurance company customers. Furthermore, the primary objective of this study is to identify and recommend the most appropriate model for predicting travel insurance purchases. By doing so, valuable and actionable insights can be provided to the insurance company, enabling them to devise effective strategies to convert non-insurance buyers into purchasers. This strategic approach ultimately leads to increased revenue generated from travel insurance sales while assisting the company in targeting the most profitable customer segments.

II. LITERATURE REVIEW

A. Related Work

Purchase prediction is crucial in forecasting the items a customer will likely purchase during their next visit, primarily based on their past buying behaviour. While this area of research has long been of interest in consumer studies, the advent of customer analytics by marketing analysts has reinvigorated its significance in recent years [8] Decision-makers place a high value on the capacity to correctly predict purchases because it allows them to improve the customer experience, provide individualized recommendations, and ultimately boost revenue. Numerous studies in the literature have explored purchase prediction, with recent works focusing on developing frameworks that leverage users' previous session features and their real-time physical movements within a market environment [9]. These developments aim to hone and boost the precision of purchase prediction algorithms.

Machine learning (ML) can be defined as a program or algorithm that has the capability to learn and improve its performance without explicit programming. Extensive data analysis, computer vision, speech recognition, and robotics are just a few of the complicated issues it has shown to be helpful in solving [10]. ML and data mining techniques have gained significant prominence in customer purchase prediction. The primary objective is to leverage these techniques to make accurate predictions based on the customer base analysis. Supervised learning and unsupervised learning are the two main types of ML. Algorithms used in supervised learning develop broad hypotheses from examples provided by external sources and make predictions about upcoming examples [11]. On the other hand, unsupervised learning refers to learning and organising information without providing an error signal to evaluate potential solutions. In this study, the dataset is labelled, and the values that need to be predicted are divided into "Purchase" and "Does Not Purchase”. This approach allows for developing models to effectively predict customer purchase behaviour based on the available labelled data.

A paper regarding health insurance purchase prediction was published in 2021. In the paper, four supervised ML models were constructed based on a dataset from a financial corporation, namely LR, DT, bagging, and RF models, to forecast the likelihood of customers purchasing health insurance. After model building, they were evaluated using the F1 score. LR model was the champion model after under-sampling, achieving an F1 score of 0.6974, followed by DT (0.6809), bagging (0.6128), and RF (0.5968) [12]. In addition, according to [13], six supervised ML algorithms were deployed to select the best model to prognosticate customers’ intention to purchase an insurance plan: LR, DT, RF, Extreme Gradient Boost (XGBoost), SVM, and Light Gradient-Boosting Machine (lightGBM). Homesite USA provided the dataset used. Unlike the previous paper, the performance of the models was evaluated using AUC in this work. The outcome displayed that lightGBM outperformed others with an AUC score of 0.962. The second runners were LR and SVM, with AUC scores of 0.957 and 0.955, respectively. RF and DT scored 0.89 and 0.85 in AUC, while the worst model was XGBoost, with only 0.64 [13]. Both papers consistently highlighted that Logistic Regression (LR) emerged as a notable model for forecasting customers' insurance purchase intention. The findings from these studies indicated that LR demonstrated strong predictive capabilities and exhibited promising performance in predicting customer behaviour related to insurance purchases.

In 2022, an ML-based insurance purchasing prediction, particularly in travel insurance paper, was presented [14]. The study aims to determine consumer interest in purchasing travel insurance based on age, employment type, graduation information, annual income, family size, chronic diseases, frequency of flying, and past travel experiences. The dataset utilized was collected from Kaggle through a survey conducted with a tourist group. To do so, ten ML algorithms were introduced, namely LR, KNN, Gaussian NB, Multinomial NB, DT, RF, Support Vector Clustering (SVC), XGBoost, Stochastic Gradient Descent (SGD), and Gradient Boosting Classifier (GBC). The accuracy of each model was studied to choose the optimal model for travel insurance purchase prediction. The finding revealed three models with the highest accuracy of 0.88: DT, RF, and SGD. Subsequently, XGB scored 0.86, followed by GBC (0.83), KNN (0.82), SVC (0.77), and Gaussian NB (0.76). LR was, surprisingly, the last, with an accuracy of 0.64, slightly lower than Multinomial NB (0.65). The outcome varied from the previous work, which showed LR as a remarkable model for insurance purchase prediction.

B. Machine Learning Algorithms for Binary Classification

Classification is one of the most often adopted jobs in data mining, with many applications. Predicting a categorical target variable from a set of input variables is the classification process. The use of six supervised learning algorithms for categorization is the main emphasis of this study. The algorithms are KNN, DT, SVM, NB, RF, and LR. These algorithms were selected based on their simplicity and popularity in addressing binary classification problems. Each algorithm employs distinct methods for extracting the relationship between variables, allowing for a comprehensive exploration of predictive modelling in this study.

KNN is a classification method that determines the class of an object based on the closest training examples in the feature space. It falls under instance-based learning or lazy learning, where the function is approximated locally, and computation is deferred until classification [15]. KNN is widely used in data mining and machine learning due to its simplicity and excellent performance [16]. It is known as Memory-Based Classification since the training examples must be stored in memory during runtime [17]. KNN performs well on large training sets and is robust to noisy data, especially when using weighted distances such as the Inverse Square [18]. It is considered a "Lazy Learner" as it does not learn during the training phase but rather stores the training dataset for real-time predictions. However, KNN is a noise-sensitive model, so its sensitivity towards noisy data, outliers and missing values is relatively high [19].

DT s a classifier that partitions the input space recursively based on attribute values [20]. It is a widely used method in data mining for establishing classification systems and developing prediction algorithms [21]. DT learning methods can be applied to classification and regression tasks [22]. In DT, internal nodes split the instance space into sub-spaces based on input attribute values, and each leaf is assigned to a class representing the most appropriate or frequent target value. Classification of instances involves traversing the tree from the root node, following the branches corresponding to attribute values [20]. The outputs of a decision tree are easily readable and interpretable without requiring statistical knowledge.

SVM is a supervised non-parametric statistical learning technique that does not make assumptions about the underlying data distribution [23]. SVM has shown successful applications in various classification and forecasting problems [24]. Compared to other classification algorithms, SVM has been reported to deliver higher accuracy [25]. It has been widely used in pattern recognition, regression estimation, dependency estimation, forecasting, and building intelligent machines [26]. One of the critical advantages of SVM is its flexibility in choosing the form of the threshold that separates different classes. With kernels, SVMs can handle non-linear relationships and non-monotonic variables without requiring specific adjustments for each variable [27].

NB is a probabilistic classifier introduced by Revered Thomas Bayes, which applies Bayes' theorem for classification based on simple probability and statistical methods [28]. It makes the naive assumption of conditional independence among predictors and utilizes prior, posterior, and class conditional probabilities [29]. NB classifiers are known for their scalability, as they require linear parameters relative to the number of features or predictors [30]. They are considered one of the fastest probabilistic classifiers, especially during the training phase. The parameter estimation for NB models often employs the maximum likelihood method, making it efficient in supervised learning settings without relying on Bayesian probability or methods [31]. Despite its naive design and oversimplified assumptions, NB classifiers have effectively performed in complex real-world situations. They have computational advantages and can handle numerical features by discretizing them or using the normal distribution for probability calculations [32].

LR also called the logistic or logit model, is a statistical method that examines the relationship between multiple independent variables and a categorical dependent variable. Its primary purpose is to estimate the probability of an event occurring by fitting the data to a logistic curve [33]. LR is widely utilized in modelling dichotomous outcomes and is particularly suitable for decision-making scenarios, making it a common choice in economic and financial analyses [34]. It is considered one of the simplest ML algorithms, offering ease of implementation and high training efficiency in some instances [35]. Logistic regression does not require substantial computational resources due to its characteristics. Moreover, logistic regression demonstrates reduced susceptibility to overfitting in low-dimensional datasets with adequate training examples. As a result, it is often used as a benchmark model for performance evaluation, allowing for quick and straightforward implementation rather than immediately resorting to complex models.

RF algorithm, initially proposed by Leo Breiman and Adele Cutler, combines the "Bootstrap aggregating" method and the "random subspace method" to construct a set of decision trees for classification [36]. The algorithm randomly selects features from the sample dataset and subsets of the sample data as the training set. RF operates by dividing the dataset into training and test sets. Multiple samples are randomly selected from the training set, and a decision tree is built for each sample. The best division is used to split each selection into two branches. This process is repeated, and the final prediction is determined by aggregating the votes from each tree and selecting the prediction with the most votes [37]. Meanwhile, Random Forest is well-suited for modelling high-dimensional data as it can handle missing values and various data types such as continuous, categorical, and binary [38]. Using ensemble strategies and random sampling enables Random Forest to mitigate overfitting issues and achieve accurate predictions and better generalizations [39]. Table 1 depicts the comparison of various machine learning classifiers.

III. RESEARCH METHODOLOGY

In this paper, the framework consisted of four major stages: Data Retrieval, Data Pre-processing, Data Modelling, and Model Evaluation, as shown in Figure 1. The framework was carried out in Python.

Table 1. Comparison of the Classifiers.

Models	Applicable on Classification or Regression	Perform Well with Large Dataset	Simple Interpretation and Operation	Noise Sensitive	Ensembled
KNN	Both	Yes	Yes	Yes	No
DT	Classification	No	Yes	Yes	No
SVM	Both	No	No	Yes	No
NB	Classification	Yes	Yes	No	No
LR	Both	Yes	Yes	Yes	No
RF	Both	Yes	No	No	Yes

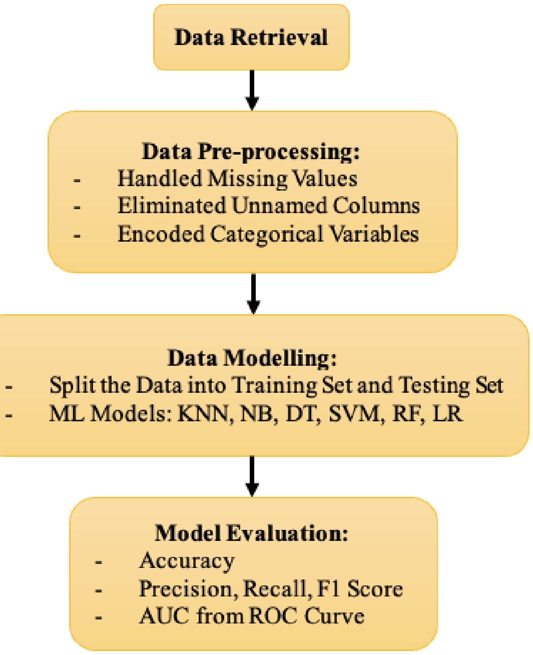
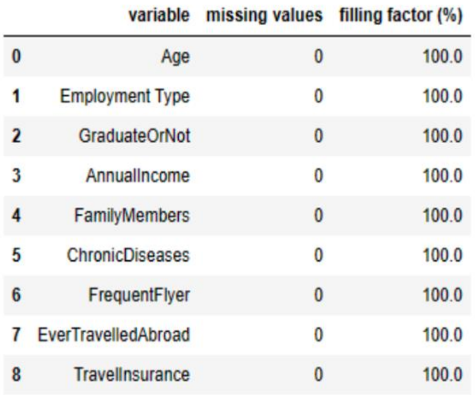

Data Retrieval
Data Pre-processing:
- Handled Missing Values
- Eliminated Unnamed Columns
- Encoded Categorical Variables
Data Modelling:
- Split the Data into Training Set and Testing Set
- ML Models: KNN, NB, DT, SVM, RF, LR
Model Evaluation:
- Accuracy
- Precision, Recall, F1 Score
- AUC from ROC Curve

Figure 1. Research Framework.

A. Data Retrieval

In this stage, the dataset employed, Travel Insurance Prediction Data, was downloaded from Kaggle. The dataset consisted of nine features and 1986 rows/observations. The independent features were age, employment type, graduation state, annual income, family members, chronic disease, flying frequency, and travel experiences as depicted in Figure 2.



	variable	missing values	filling factor (%)
0	Age	0	100.0
1	Employment Type	0	100.0
2	GraduateOrNot	0	100.0
3	AnnualIncome	0	100.0
4	FamilyMembers	0	100.0
5	ChronicDiseases	0	100.0
6	FrequentFlyer	0	100.0
7	EverTravelledAbroad	0	100.0
8	TravelInsurance	0	100.0

Figure 2. Features in the Dataset.

B. Data Pre-processing

Several steps were conducted in this stage to obtain a cleaned dataset for ML model building. Firstly, the unnamed columns were removed as they contained unclear information. This step aimed to optimize memory usage and enhance data analysis. Next, the dataset was checked for missing values. Fortunately, the retrieved dataset had a 100% filling factor, indicating no missing values.

Furthermore, the columns 'Employment Type', 'GraduateOrNot', 'FrequentFlyer', and 'EverTravelledAbroad' contained textual data. However, machine learning algorithms require numerical input and output. Therefore, these categorical variables must be encoded as integer values before being utilized for model training and testing. The LabelEncoder package from the Scikit-learn pre-processing library was employed to encode the categorical features in this project. All categorical values were transformed into a range between 0 and *n*-1, where *n* represents the total number of classes.

C. Data Modelling

Before building models using the ML algorithms, the cleaned dataset was split into a training set and a test set with a ratio of 75:25. In the modelling phase; various modelling techniques were applied, including model selection and parameter fine-tuning. This study built six classification models, namely KNN, DT, SVM, NB, LG, and RF. An object was created for each mode to store the algorithm along with its specified parameters. This object also contained the information extracted by the algorithm from the training data. Subsequently, the classification models were built on the training set using the fit method, which takes NumPy arrays - X_train for the features and y_train for the target variable, as arguments. Once the model is trained, predictions can be made by calling the prediction method on the object using the features from the test set (X_test). The ML algorithms were called out from the Scikitlearn library, and the processes are shown in Figure 4 to Figure 9.

To build a robust K-Nearest Neighbors (KNN) model, the value of *k* (the number of nearest neighbours) must be fine-tuned. To determine the optimal value of *k*, a range of values from 1 to 25 was considered, with a step size of 5. The results of different *k* values are displayed in Figure 3. The optimal *k* value was 10; thus, the parameter from KNeighborsClassifier, n_neighbors, was set as 10.

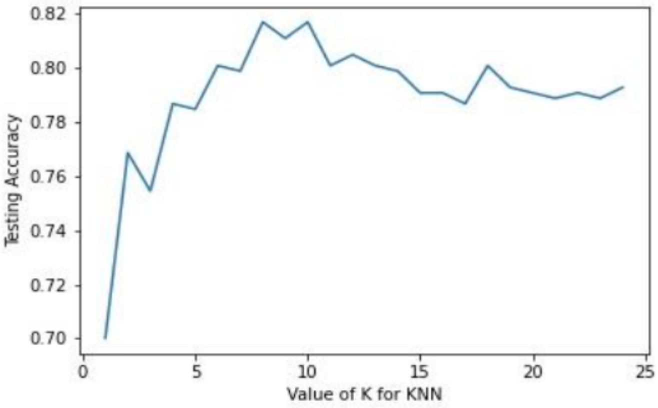


Figure 3. Graph of KNN Testing Accuracy against Value of *k*.

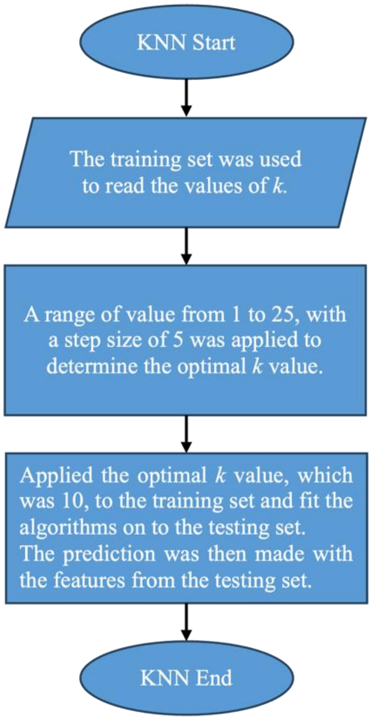

KNN Start
The training set was used to read the values of *k*.
A range of value from 1 to 25, with a step size of 5 was applied to determine the optimal *k* value.
Applied the optimal *k* value, which was 10, to the training set and fit the algorithms on to the testing set. The prediction was then made with the features from the testing set.
KNN End

Figure 4. Flowchart of KNN.

DT classifier was built using the DecisionTreeClassifier(). In this case, since no specific parameters were specified, the default parameters of the function were utilized. By default, the DecisionTreeClassifier() function uses the Gini impurity as the criterion to measure the quality of each split (criterion='gini'). The Gini impurity measures how often a randomly selected element from a given class would be incorrectly labelled if it was randomly labelled according to the distribution of labels in the subset. The minimum number of samples required to split an internal node was also set to 2 (min_samples_split=2) by default. This meant an internal node would only be split if it contained at least 2 samples. Adjusting this parameter can affect the depth and complexity of the resulting decision tree.

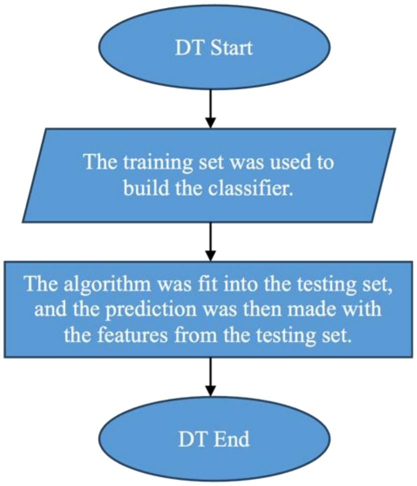

DT Start
The training set was used to build the classifier.
The algorithm was fit into the testing set, and the prediction was then made with the features from the testing set.
DT End

Figure 5. Flowchart of DT.

Subsequently, the SVM classifier was built using the SVC () function. The kernel type used was Radial Basis Function (RBF), and the probability estimates were enabled. They were specified as kernel='rbf' and probability=True in the function's parameters.

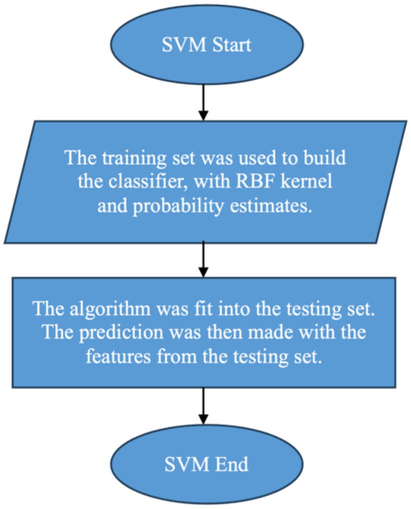

SVM Start
The training set was used to build the classifier, with RBF kernel and probability estimates.
The algorithm was fit into the testing set. The prediction was then made with the features from the testing set.
SVM End

Figure 6. Flowchart of SVM.

The NB classifier was built using the GaussianNB() function. This function did not specify specific parameters, so the default parameters were used. By default, the prior probabilities are adjusted according to the data (priors=None), which estimates the class prior probabilities from the training data. Additionally, the portion of the largest variance of all features is set to 1 × 10^ (-9) (var_smoothing=1e-09). This parameter helps avoid issues when dealing with features with zero variance by adding a small value.

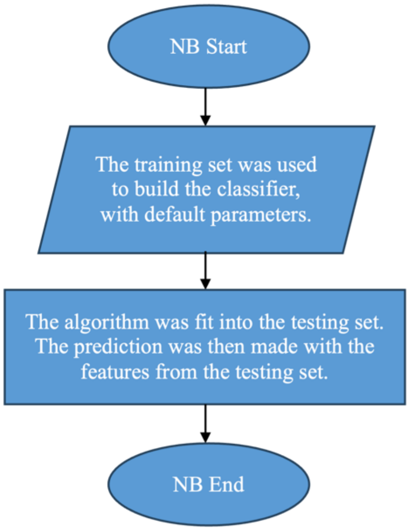

NB Start
The training set was used to build the classifier, with default parameters.
The algorithm was fit into the testing set. The prediction was then made with the features from the testing set.
NB End

Figure 7. Flowchart of NB.

To implement the LR, the LogisticRegression model was used. LR made certain assumptions about the input data to perform effectively. One of these assumptions is that the independent variables should have little to no multicollinearity.

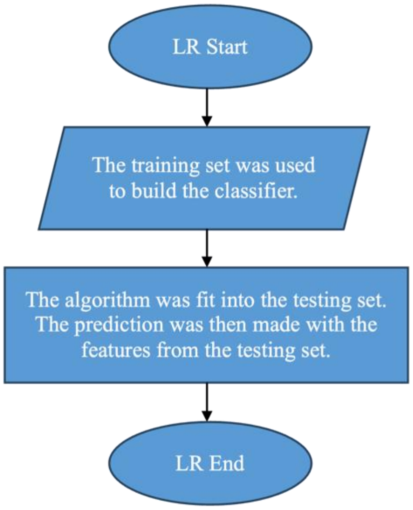

LR Start
The training set was used to build the classifier.
The algorithm was fit into the testing set. The prediction was then made with the features from the testing set.
LR End

Figure 8. Flowchart of LR.

When constructing the RF model, RandomForestClassifier was introduced. Since RF is known to be less prone to overfitting compared to other models, RF often requires less hyperparameter tuning. The two important parameters influencing the prediction performance are n_estimators and max_features. The n_estimators parameter represents the number of decision trees in the forest. It does not have a local optimum, meaning that increasing the number of estimators typically leads to improved performance. In this study, a value of 1000 was used for n_estimators. Meanwhile, the max_features parameter determines the maximum number of features the algorithm considers when looking for the best split at each node of the trees. By limiting the number of features, Random Forests can reduce the risk of overfitting and increase the diversity among the trees. In this study, a value of 3 was used for max_features.

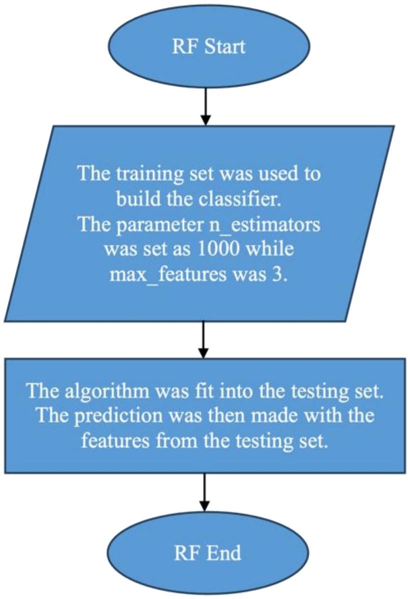
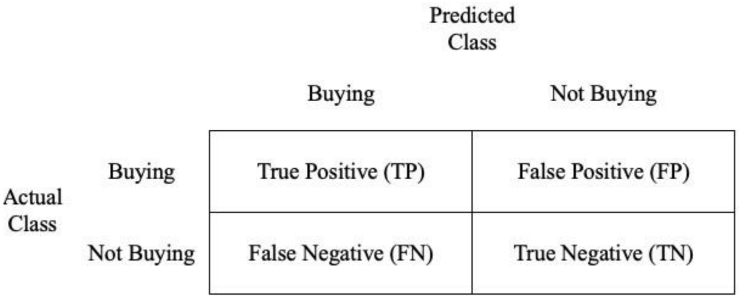

RF Start
The training set was used to build the classifier. The parameter n_estimators was set as 1000 while max_features was 3.
The algorithm was fit into the testing set. The prediction was then made with the features from the testing set.
RF End

Figure 9. Flowchart of RF.

D. Model Evaluation

This section describes the evaluation metrics used to compare the performance of the classifiers. Three evaluation metrics were utilized in this work: Accuracy, Confusion Matrix, and AUC-ROC. These metrics are frequently employed in classification issues and offer insightful data regarding the effectiveness of the algorithms. To assess and compare the performance of the classifiers, a comparison analysis may be done by looking at these indicators.



		Predicted Class	
		Buying	Not Buying
Actual Class	Buying	True Positive (TP)	False Positive (FP)
	Not Buying	False Negative (FN)	True Negative (TN)

Figure 10. Confusion Matrix.

One can define "accuracy" as the percentage of correct predictions for the test data, calculated by dividing the correct predictions (true positives, TP and true negatives, TN) by the total number of predictions, as shown in Equation (1). Although accuracy is a useful indicator that is easily interpretable, it has several drawbacks because it does not take into consideration class imbalances or the different costs of false positives (FP) and false negatives (FN). Therefore, other performance metrics were applied in this work to provide a more comprehensive and detailed assessment of the results.

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)} \tag{1}$$

In addition, precision, recall, and F1 score can be determined from the confusion matrix. The F1 score acts as a metric to assess the accuracy of a model on a given dataset. It is especially relevant to the assessment of a binary classification system, where examples are categorized as either “positive” or “negative” [40]. The model’s precision and recall are combined in the F1 score, presenting a harmonic mean of these two measures, as denoted in Equation (2). Meanwhile, precision is the proportion of genuine positive examples among those classified as positive by the model. The corresponding formula is shown in Equation (3). Recall, also known as sensitivity, reflects the fraction of correctly classified positive examples out of all positive ones. To be plainer, the formula of recall is displayed in Equation (4).

$$F1\ Score = \frac{2 \times (Precision \times Recall)}{(Precision + Recall)} \tag{2}$$

$$Precision = \frac{TP}{TP+FP} \tag{3}$$

$$Recall = \frac{TP}{TP+FN} \tag{4}$$

The ROC curve (see Figure 11) is an informative tool in binary classification tasks, and the AUC is a metric that quantifies the classifier’s ability to differentiate between classes. It serves as a summary of the ROC curve. For choosing classifiers based on performance, a ROC graph offers a visual presentation and organizational method [41]. It is a probability curve that plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold values, effectively separating the 'signal' from the 'noise'. A higher AUC signifies better model performance in distinguishing between positive and negative classes. When AUC = 1, the classifier can perfectly discern between all positive and negative data points. Conversely, if the AUC were 0, the classifier would incorrectly predict all negatives as positives and vice versa [42].

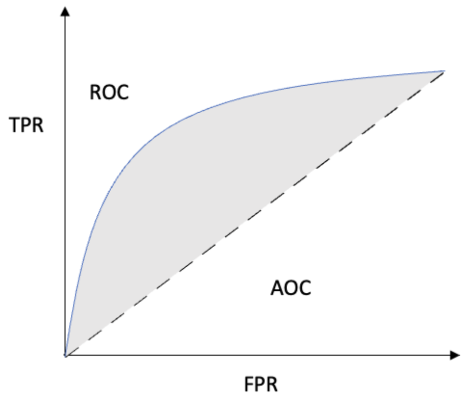


Figure 11. AUC – ROC Curve.

IV. RESULTS AND DISCUSSIONS

In this section, the prediction results obtained by various algorithms were presented. The six algorithms' performance metrics, including accuracy, precision, recall, F1 score, and AUC from the ROC curve, are illustrated in the figures and tables below. Moreover, the result of the champion model is highlighted.

A. Prediction Results

The accuracy ratings of the six classifiers employed in the investigation are shown in Table 2. Five of the classifiers achieved accuracy scores ranging from 75% to 82%. KNN achieved the highest accuracy score of 0.8169, followed closely by NB, RF, LR, and DTC, which obtained accuracy scores of 0.7928, 0.7847, 0.7586, and 0.7525, respectively. In contrast, SVM had the lowest accuracy score of only 0.66.

The KNN algorithm in this study achieved an accuracy of 0.8169, almost identical to the accuracy reported in the previous work (0.82) [14]. However, there are variations in the performance of other algorithms compared to the previous work. The DT and RF algorithms did not perform as well as the previous study, which achieved an accuracy of 0.88. In contrast, the SVM exhibited poorer performance with an accuracy of 0.66, which differs from the previous work, where the SVM achieved an accuracy of 0.77. Furthermore, the previous work identified LR as the worst-performing model with an accuracy of 0.64, whereas in this study, LR showed improved performance with an accuracy of 0.7586.

Table 2. Accuracy of the Classifiers.

Classifier	Accuracy
KNN	0.8169
DT	0.7525
SVM	0.66
NB	0.7928
LR	0.7586
RF	0.7847

The findings from the travel insurance purchase dataset are shown in Table 3. It is evident that the dataset does not exhibit a class imbalance issue as the classes are relatively balanced, with 66% for the "does not purchase" class and 34% for the "purchase" class, totalling 328 and 169 instances, respectively. Regarding precision, the "does not purchase" class of the DTC has the highest precision of 0.82. On the other hand, SVM completely disregarded this class to attain a recall of 1, which led to a recall of 0. The KNN algorithm yielded the best recall and F1-score for the "does not purchase" class, with values of 0.95 and 0.87, respectively. Additionally, results were typically poor for the "purchase" class, especially for recall and F1-score. The KNN algorithm demonstrated relatively better results for this class. It achieved the highest precision and F1-score of 0.86 and 0.67, respectively. The DT and RF algorithms attained the maximum recall score of 0.62 for the "purchase" class. SVM fared poorly, on the other hand, with the lowest precision, recall, and F1-score of 0.44, 0.66, and 0.52, respectively. The KNN algorithm produced the best results since it effectively addressed both classes and achieved the highest precision, recall, and F1 score.

Table 3. Findings from the Travel Insurance Purchase Dataset.

Classifier	Class	Precision	Recall	F1 Score
KNN	Does Not Purchase	0.80	0.95	0.87
	Purchase	0.86	0.55	0.67
	Average	0.82	0.82	0.80
DT	Does Not Purchase	0.81	0.64	0.81
	Purchase	0.64	0.62	0.63
	Average	0.75	0.63	0.75
SVM	Does Not Purchase	0.66	1.00	0.800
	Purchase	0.00	0.00	0.00

	Average	0.44	0.66	0.52
NB	Does Not Purchase	0.79	0.94	0.86
	Purchase	0.81	0.51	0.63
	Average	0.80	0.79	0.78
LR	Does Not Purchase	0.77	0.91	0.83
	Purchase	0.72	0.47	0.57
	Average	0.75	0.76	0.74
RF	Does Not Purchase	0.82	0.88	0.85
	Purchase	0.73	0.62	0.67
	Average	0.79	0.79	0.79

Furthermore, Figure 12 provides a detailed visualization of the ROC curves for each algorithm, enabling a thorough evaluation of their performance. The performance of all other algorithms is surpassed by the red line, which represents the Naive Bayes (NB) method, which has a superior ROC AUC value of 0.791 across all thresholds. The NB, K-Nearest Neighbors (KNN), and Random Forest (RF) algorithms achieved AUC values of 0.791, 0.784, and 0.783, respectively. This implies a 79.10% chance for NB, 78.40% chance for KNN, and 78.30% chance for RF to successfully distinguish between the "Purchase" and "Does Not Purchase" classes. As a result, NB, KNN, and RF are regarded as skilled classifiers since they can discriminate between positive and negative classes with respectable discrimination skills.

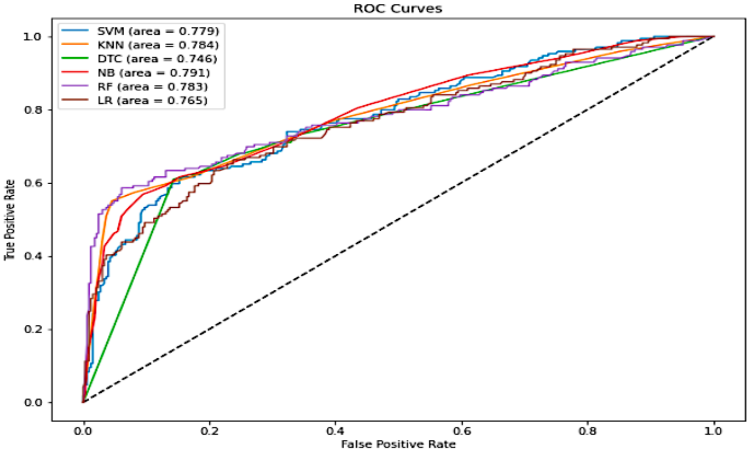


Figure 12. ROC AUCs of Each Classifier.

B. Champion Model

The performance results for the four classifiers are summarised in Table 4, which is best viewed horizontally. An average score of the six-performance metrics was calculated for each classifier to select the champion model. The best results for each metric have been bolded in the table.

Table 4. Summary of the Performance Outcomes of Each Classifier.

Classifier	Accuracy	Precision	Recall	F1 Score	ROC AUC	Average
KNN	**0.81**	**0.82**	**0.82**	**0.80**	0.784	**0.81**
DT	0.75	0.75	0.75	0.75	0.746	0.75
SVM	0.66	0.44	0.66	0.52	0.779	0.61
NB	0.79	0.80	0.79	0.78	**0.791**	0.79
LR	0.75	0.75	0.76	0.74	0.765	0.75
RF	0.78	0.79	0.79	0.79	0.783	0.79

KNN, which received an average score of 0.81, is clearly the best-performing model in this study based on the average scores. NB and RF, both of which received an average score of 0.79, are close behind KNN. The DT and LR models achieved slightly lower average scores of 0.75. On the other hand, SVM performed the poorest, with an average score of only 0.61. As a result, it can be said that KNN is the study's top model, outperforming the other classifiers in terms of performance.

V. CONCLUSION AND FUTURE WORK

In conclusion, this study has practical implications for insurance companies, as it highlights the potential of supervised learning algorithms in predicting travel insurance purchases. Among the six classifiers examined, KNN emerged as the top-performing model, achieving an impressive accuracy of 81%, the highest among all classifiers. KNN also exhibited superior precision, with a score of 0.82, suggesting its effectiveness in accurately identifying potential customers for travel insurance. Unexpectedly, SVM fared poorly in this investigation, scoring only 66% accuracy. Contrary to earlier studies that praised SVM as a top-performing model, this work suggests that the efficacy of SVM may vary depending on the dataset and problem domain. KNN achieved the greatest F1-score of 80%, demonstrating its ability to balance precision and recall successfully. It is important to remember that prior research has revealed a range of outcomes, with RF occasionally getting the highest F1 score. Thus, specific characteristics of the dataset and the particular problem should be considered when selecting the optimum model. Additionally, NB outperformed KNN regarding the ROC AUC, achieving a score of 0.791 compared to KNN's 0.784. This highlights the importance of considering multiple evaluation metrics when selecting the most suitable classifier.

Even though KNN showed the best overall performance, selecting a classifier requires careful consideration of the particular needs and priorities of the work. To learn more about these classifiers' strengths and weaknesses, future research should examine how well they function across other datasets and problem domains. For predicting the acquisition of travel insurance, this study offers helpful insights into using supervised learning algorithms. The numerical results show the models' potential for precisely identifying customers likely to buy travel insurance, which has ramifications for insurance companies on a practical level.

Nonetheless., future work should address the shortcomings that happened in this work. In order to fully explore the algorithms' potential, additional data, thorough hyper-parameter tuning, and more training iterations would be possible on a more potent computational machine with the potential for parallelization. Secondly, exploring the use of additional data features, such as click information, could improve predictions. Performance could be enhanced in addition to algorithms exploration by improving the data foundation with the latest trend. Additionally, incorporating the latest research findings and relevant variables would further improve the model.

ACKNOWLEDGEMENT

The research for this paper was not funded.

REFERENCES

[1] D. A. Hamzah, A. A. Kalambe, L. S. Goklas, and N. G. Alkhayyat, "Predicting travel insurance policy claim using logistic regression," *Applied Quantitative Analysis*, vol. 1, no. 1, 2021.

[2] S. K. C. R. Wickramasinghe and K. ABD Razak, "The Impact Of The Telecommunication Industry As A Moderator on Poverty Alleviation and Educational Programmes To Achieve Sustainable Development Goals In Developing Countries," *Journal of Informatics and Web Engineering*, vol. 2, no. 1, pp. 25–37, Mar. 2023, doi: 10.33093/jiwe.2023.2.1.3.

[3] A. A. Hasan and N. C. Abdullah, "Compulsory Travel Insurance in Malaysia: Some Regulatory Considerations," *Procedia Soc Behav Sci*, vol. 172, pp. 344–351, Jan. 2015, doi: 10.1016/j.sbspro.2015.01.375.

[4] K. Herbst *et al.*, "Protocol: Leveraging a demographic and health surveillance system for Covid-19 Surveillance in rural KwaZulu-Natal," *Wellcome Open Res*, vol. 5, 2020, doi: 10.12688/wellcomeopenres.15949.1.

[5] World Tourism Organization, "IMPACT ASSESSMENT OF THE COVID-19 OUTBREAK ON INTERNATIONAL TOURISM," *The World Tourism Organization*, 2020.

[6] M. Diakonidze, "Tourism Insurance Market, Risks and Prospects: The Case Study," *Journal of corporate governance, insurance and risk management*, vol. 8, Jun. 2021, doi: 10.51410/jcgirm.8.1.5.

[7] S. L. Kang, "Insurers see rising demand for travel insurance as more countries make it mandatory," *The Edge Malaysia*, Nov. 09, 2021.

[8] A. Martínez, C. Schmuck, S. Pereverzyev, C. Pirker, and M. Haltmeier, "A machine learning framework for customer purchase prediction in the non-contractual setting," *Eur J Oper Res*, vol. 281, no. 3, pp. 588–596, Mar. 2020, doi: 10.1016/j.ejor.2018.04.034.

[9] R. Esmeli, M. Bader-El-Den, and H. Abdullahi, "Towards early purchase intention prediction in online session based retailing systems," *Electronic Markets*, vol. 31, no. 3, pp. 697–715, Sep. 2021, doi: 10.1007/s12525-020-00448-x.

[10] N. Amruthnath and T. Gupta, "A Research Study on Unsupervised Machine Learning Algorithms for Early Fault Detection in Predictive Maintenance," in *5th International Conference on Industrial Engineering and Applications355978-1-5386-5748-5/18/$31.00 ©2018 IEEE*, 2018, pp. 355–361.

[11] F.Y. Osisanwo, J.E.T. Akinsola, O. Awodele, J.O. Hinmikaiye, O. Olakanmi, and J. Akinjobi, "Supervised Machine Learning Algorithms: Classification and Comparison," *International Journal of Computer Trends and Technology*, vol. 48, 2017, pp. 128-138.

[12] Y. Liu, Y. Liu, K. Bo, Q. Yi, Z. Wang, Y. Sun, J. Xu, X. Zhang, R. Xu, "Predict Health Insurance Purchase with Machine Learning Techniques," 2021. [Online]. Available: https://ssrn.com/abstract=3968385

[13] R. Jaiswal, "Prognosticating Customers' Intention To Purchase An Insurance Plan With Machine Learning," in *Fostering Resilient Business Ecosystems and Economic Growth: Towards the Next Normal*, A. Gawande and A. Kumar, Eds., India: Research and Publication Cell , Apr. 2022, pp. 292–296.

[14] M. A. Rubi, M. Hasan Imam Bijoy, S. Chowdhury, and M. K. Islam, "Machine Learning Prediction of Consumer Travel Insurance Purchase Behavior," in *2022 13th International Conference on Computing Communication and Networking Technologies, ICCCNT 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICCCNT54827.2022.9984470.

[15] S. B. Imandoust and M. Bolandraftar, "Application of K-nearest neighbor (KNN) approach for predicting economic events theoretical background," *Int J Eng Res Appl*, vol. 3, pp. 605–610, Jun. 2013.

[16] S. Zhang, X. Li, M. Zong, X. Zhu, and D. Cheng, "Learning k for kNN Classification," *ACM Trans Intell Syst Technol*, vol. 8, pp. 1–19, Jun. 2017, doi: 10.1145/2990508.

[17] M. Shouman, T. Turner, and R. Stocker, "Applying k-Nearest Neighbour in Diagnosing Heart Disease Patients," *International Journal of Information and Education Technology*, vol. 2, no. 3, pp. 220–223, 2012.

[18] V. M. Sreeja and K. Umamaheswari, "Type 2 Diabetic Prediction Using Machine Learning Algorithm," *American Scientific Research Journal for Engineering*, vol. 45, no. 1, pp. 299–307, 2018, [Online]. Available: http://asrjetsjournal.org/

[19] Y. Lim, K.-W. Ng, P. Naveen, and S.-C. Haw, "Emotion Recognition by Facial Expression and Voice: Review and Analysis," *Journal of Informatics and Web Engineering*, vol. 1, no. 2, pp. 45–54, Sep. 2022, doi: 10.33093/jiwe.2022.1.2.4.

[20] A. Mashat, M. Fouad, P. Yu, and T. Gharib, "A Decision Tree Classification Model for University Admission System," *Journal of Advanced Computer Science and Applications(IJACSA)*, vol. 3, Jun. 2012, doi: 10.14569/IJACSA.2012.031003.

[21] Y. Y. Song and Y. Lu, "Decision tree methods: applications for classification and prediction," *Shanghai Arch Psychiatry*, vol. 27, no. 2, pp. 130–135, Apr. 2015, doi: 10.11919/j.issn.1002-0829.215044.

[22] I. H. Sarker, “Machine Learning: Algorithms, Real-World Applications and Research Directions,” *SN Computer Science*, vol. 2, no. 3. Springer, May 01, 2021. doi: 10.1007/s42979-021-00592-x.

[23] G. Mountrakis, J. Im, and C. Ogole, “Support vector machines in remote sensing: A review,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 3. pp. 247–259, May 2011. doi: 10.1016/j.isprsjprs.2010.11.001.

[24] J. Nayak, B. Naik, and Prof. Dr. H. Behera, “A Comprehensive Survey on Support Vector Machine in Data Mining Tasks: Applications & Challenges,” *International Journal of Database Theory and Application*, vol. 8, pp. 169–186, Jun. 2015, doi: 10.14257/ijdta.2015.8.1.18.

[25] D. Srivastava and L. Bhambhu, “Data classification using support vector machine,” *J Theor Appl Inf Technol*, vol. 12, pp. 1–7, Jun. 2010.

[26] Q. Li, Q. Meng, J. Cai, H. Yoshino, and A. Mochida, “Applying support vector machine to predict hourly cooling load in the building,” *Appl Energy*, vol. 86, no. 10, pp. 2249–2256, 2009, doi: 10.1016/j.apenergy.2008.11.035.

[27] L. Auria, R. A. M. Berlin, and R. A. Moro, “Support Vector Machines (SVM) as a Technique for Solvency Analysis,” 2008.

[28] A. P. Wibawa *et al.*, “Naïve Bayes Classifier for Journal Quartile Classification,” *International Journal of Recent Contributions from Engineering, Science & IT (iJES)*, vol. 7, no. 2, p. 91, Jun. 2019, doi: 10.3991/ijes.v7i2.10659.

[29] A. Jamain and D. J. Hand, “The Naive Bayes Mystery: A classification detective story,” *Pattern Recognit Lett*, vol. 26, no. 11, pp. 1752–1760, Aug. 2005, doi: 10.1016/j.patrec.2005.02.001.

[30] U. Dulhare, “Prediction system for heart disease using Naive Bayes and particle swarm optimization,” *Biomedical Research*, vol. 29, Jun. 2018, doi: 10.4066/biomedicalresearch.29-18-620.

[31] S. Jyothi and P. Bhargavi, “Applying Naive Bayes Data Mining Technique for Classification of Agricultural Land Soils,” *57. P.Bhargavi , S. Jyothi*, vol. 9, Jun. 2009.

[32] S. Kotsiantis, “Supervised Machine Learning: A Review of Classification Techniques,” *Informatica (Ljubljana)*, vol. 31, Jun. 2007.

[33] H. A. Park, “An introduction to logistic regression: from basic concepts to interpretation with particular attention to nursing domain,” *J Korean Acad Nurs*, vol. 43, no. 2, p. 154—164, Apr. 2013, doi: 10.4040/jkan.2013.43.2.154.

[34] A. Strzelecka, A. Kurdyś-Kujawska, and D. Zawadzka, “Application of logistic regression models to assess household financial decisions regarding debt,” *Procedia Comput Sci*, vol. 176, pp. 3418–3427, Jun. 2020, doi: 10.1016/j.procs.2020.09.055.

[35] C. Y. J. Peng, K. L. Lee, and G. M. Ingersoll, “An introduction to logistic regression analysis and reporting,” *Journal of Educational Research*, vol. 96, no. 1, pp. 3–14, 2002, doi: 10.1080/00220670209598786.

[36] Q. Ren, H. Cheng, and H. Han, “Research on machine learning framework based on random forest algorithm,” in *AIP Conference Proceedings*, Jun. 2017, p. 80020. doi: 10.1063/1.4977376.

[37] F. Tarsha-Kurdi, W. Amakhchan, and Z. Gharineiat, “Random Forest Machine Learning Technique for Automatic Vegetation Detection and Modelling in LiDAR Data Mini Review Int J Environ Sci Nat Res,” *Journal of Environmental Science and Natural Resources*, vol. 28, Jun. 2021, doi: 10.19080/IJESNR.2021.28.556234.

[38] J. Ali, R. Khan, N. Ahmad, and I. Maqsood, “Random Forests and Decision Trees,” 2012. [Online]. Available: www.IJCSI.org

[39] M. Aria, C. Cuccurullo, and A. Gnasso, “A comparison among interpretative proposals for Random Forests,” *Machine Learning with Applications*, vol. 6, p. 100094, Dec. 2021, doi: 10.1016/j.mlwa.2021.100094.

[40] T. Wood, “What is F-score?,” *DeepAI*. https://deepai.org/machine-learning-glossary-and-terms/f-score (accessed Jun. 17, 2023).

[41] T. Fawcett, “An introduction to ROC analysis,” *Pattern Recognit Lett*, vol. 27, no. 8, pp. 861–874, Jun. 2006, doi: 10.1016/j.patrec.2005.10.010.

[42] S. Narkhede, “Understanding AUC - ROC Curve,” *Towards Data Science*, Jun. 27, 2018. https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5 (accessed Jun. 17, 2023).