

# Predicting Loan Approval of Bank Direct Marketing Data Using Ensemble Machine Learning Algorithms

Hossam Meshref, SMIEEE,  
Computer Science Department  
College of Computers and Information Technology, Taif University  
Kingdom of Saudi Arabia

Received: October 16, 2020. Revised: November 27, 2020. Accepted: December 2, 2020.  
Published: December 10, 2020.

**Abstract:-** The Bank Marketing data set at Kaggle is mostly used in predicting if bank clients will subscribe a long-term deposit. We believe that this data set could provide more useful information such as predicting whether a bank client could be approved for a loan. This is a critical choice that has to be made by decision makers at the bank. Building a prediction model for such high-stakes decision does not only require high model prediction accuracy, but also needs a reasonable prediction interpretation. In this research, different ensemble machine learning techniques have been deployed such as Bagging and Boosting. Our research results showed that the loan approval prediction model has an accuracy of 83.97%, which is approximately 25% better than most state-of-the-art other loan prediction models found in the literature. As well, the model interpretation efforts done in this research was able to explain a few critical cases that the bank decision makers may encounter; therefore, the high accuracy of the designed models was accompanied with a trust in prediction. We believe that the achieved model accuracy accompanied with the provided interpretation information are vitally needed for decision makers to understand how to maintain balance between security and reliability of their financial lending system, while providing fair credit opportunities to their clients.

**Keywords:-** loan prediction; machine learning; ensemble learning; decision trees; model interpretation

## I. INTRODUCTION

Various machine learning algorithms have been used effectively in many applications in banking. Some applications have focused on the banking

security area for fraud detection using reinforcement learning as well as deep learning techniques [1,2,3]. Among those techniques were Decision Tree (DT), deep learning, Artificial Neural Networks (ANN), which were used to detect frauds in internet banking transactions. Another application focused on the counterfeit of Banknotes using DT, ANN and Support Vector Machines (SVM) techniques [4,5]. The rules induced by the DT models were accurate enough to help in distinguishing between original and counterfeit Banknotes.

There were several applications that addressed the Kaggle Bank Marketing data set. It worth mentioning that the earliest work done on the Bank Marketing data was conducted by a group of researchers, who are considered the original creators of this data [6,7]. They tried to build a prediction model for the success of telemarketing calls that sell bank long-term deposits. Their original data had around 150 features, but they used feature engineering techniques to reduce them to 22 features. They have designed four different machine models: ANN, SVM, DT and logistic regression (LR). They have used two common performance evaluation metrics: the area of the receiver operating characteristic curve (AUC) and the area of the LIFT cumulative curve (ALIFT). Best results were obtained for the ANN model with an AUC of 79% and an ALIFT of 67.2%. Their work had valuable findings that were of importance, especially after the 2008 financial crisis, to help decision makers at the Portuguese retail bank to drive a successful long-term deposit marketing campaign for selected clients.

A few researchers tried to improve prediction accuracy obtained by different machine learning models that was due to a data imbalance problem [8,9]. They have used Synthetic Minority Oversampling Technique (SMOTE), which is a well

known approach that is capable of achieving balance in the analyzed data to improve prediction accuracy. Their target was to explore the possibilities of finding different strategies that could improve the bank marketing campaign. The use of SMOTE approach paid off as it achieved an accuracy of 90.81% for the Rotation Forest DT model. A few other research efforts have handled the Bank Marketing data set using different machine learning techniques, but eventually shared the same goal with the previous studies [10,11]. Those researchers used different classifiers such as MLP, DT, RF, and LR. For some studies, RF classifiers achieved an accuracy of 87% prediction ability, while others achieved 95.3% using the Radial Basis Function ANN.

Generally, in the reviewed literature, most of the applications focused on predicting whether or not the client will subscribe a long-term deposit. The major challenge that most researchers had to deal with is data imbalance, and most of them proposed different strategies to solve this problem to improve model prediction accuracy. However, we believe that this data set is rich and could be seen from a different angle to get more insight into other patterns hidden within the data. For instance, predicting whether or not bank clients could be approved for a loan by using the already given client loan observations provided in the data set. These predictions could help, for example, the decision maker to target and approach, by marketing, bank clients with no loans; instead of just marketing only for subscribing a long-term deposit. This approach is new and is expected to provide useful information to decision makers at the bank as they decide to grant loans to their clients.

There are a few other research articles that focused on the loan approval process using other data sets [12,13,14,15]. These data sets, also available at Kaggle, are based on peer to peer (P2P) lending companies such as *Dream House Finance* and *Lending Club*. These companies are different from banks and they represent a bridge between investors and borrowers with interest rates that are higher than the banks. The attributes in these data sets intersect with a few direct marketing data set attributes such as: *balance*, *default* status, *marital* status, and *education*. However, the P2P lending process is not stable for lenders and borrowers and it does not provide a compensation plan like the bank in case of the collapse of the lending platform [16,17]. That is why in this research we preferred to conduct our analysis on real bank collected data to guarantee that predicted loan approval decisions are more robust.

It worth mentioning that research articles which investigated the P2P lending process shared same ensemble machine learning techniques used in our research. As well, in a few cases, the results obtained in these research articles were comparable to our research results. In the results section, a comparison with the state of the art results of the abovementioned research is going to be discussed. On final comment one the available research in the literature is that the focus was on getting higher accuracies, but no prediction interpretations were provided. We believe that interpretation of the loan prediction decision is essential especially in a high-stakes decision like the bank loan approval.

This paper focuses on the Bank Marketing data set with the intention of achieving the following main objectives: (1) to understand the Bank Marketing data set from a new different perspective that has not been investigated before by analyzing loan approvals. (2) to build ensemble machine learning classifiers and to improve its accuracy to achieve the best prediction model. (3) to interpret, as much as possible, the designed classifiers using the rules induced by DT models to develop trust in the designed machine learning models.

This paper is structured as follows: section two lists the methods used in this paper. Section Three shows data pre-processing and feature selection. Section four covers results and discussion, and finally, section Five concludes the presented work.

## II. METHODS

A few ensemble machine learning techniques have been deployed such as Bagging, Boosting, LogitBoost, and Random Forests [18,19,20,21,22]. The DT machine learning model was mainly used during models' interpretations. In general, during analysis in this research, a 10-fold stratified cross validation method was implemented to evaluate the performance of the designed machine learning models. In the following section, the methods behind the used ensemble techniques are going to be explained.

### A. Bagging and Boosting

Bagging and Boosting are among the most popular techniques used in machine learning ensemble methods. Bagging creates and trains different classifiers on the training set samples of size N that are randomly sampled with replacement from the original data set of size N. Therefore, some training examples may be repeated in those training data samples that are going to be used for training the classifiers. Eventually, after training these classifiers, we can predict the class of a new

example by performing majority voting of the ensemble of solutions.

Mathematically, in our research, we have a data set  $S$  of size  $N$ , where  $S = \{(x_1, y_1), \dots, (x_N, y_N)\}$ . Using the bagging method, we constructed  $M$  classifiers  $h_m$  based on the bootstrap method and drew samples  $\{S_m\}$  with replacement. As we train the base learners  $h_m$  using the samples  $S_m$ , the majority voted output  $H(x)$  could be given by the following equation, where,  $I(h_m(x) = y)$  is an indicator function of the event  $h_m(x) = y$ .

$$H(x) = \operatorname{argmax}_{y \in Y} \sum_{m=1}^M I(h_m(x) = y) \quad (1)$$

On the other hand, the boosting method used in our research was designed differently. Boosting has been modified by many researchers leading to the Adaboost method. In essence, that method focuses on the weak learners of every iteration over the data set  $S = \{(x_i, y_i), i = 1 \dots n\}$  with  $n$  examples. During each iteration  $t$ , the algorithm tries to find a new classifier  $h_t: x \rightarrow y$  while minimizing an error  $\delta_t$  over the data set  $S_t$ . Throughout the iterations, the Adaboost method gives more weight to the weak learned examples, but on the other hand, gives less weight to the correctly classified ones. These weights are initialized before starting iterations to  $W_1(n) = \frac{1}{N}$ ,  $n \in \{1, 2, \dots, N\}$ . In each iteration, we train a base learner  $h_m$  using the data set  $S$  with distribution  $D_m$  and calculate the error  $\delta_m$  such that:

$$\delta_m = \sum_{i=1}^N W_m(i)(1 - y_{i,m}) \quad (2)$$

$$y_{i,m} = \begin{cases} 1, & D_m \text{ gives the correct label} \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

It worth mentioning that  $\delta_m \in (0, 0.5)$ , therefore, in case that  $\delta_m \geq 0.5$ , the weights  $W_m(i)$  are initialized to 1. Then, we calculate a parameter  $\alpha_m$  that will be used to calculate the new weight for the iteration  $m + 1$ :

$$\alpha_m = \frac{1 - \delta_m}{\delta_m} \quad (4)$$

$$W_{m+1}(i) = \frac{W_m(i)\alpha_m^{(1-y_{i,m})}}{\sum_{j=1}^N W_m(j)\alpha_m^{(1-y_{j,m})}} \quad (5)$$

The AdaBoost algorithm stops the ensemble construction process, presumably, at an iteration  $T$ , resulting in an ensemble learner of the classifiers  $h_1$  to  $h_T$ . The Final classifier output could be given by:

$$H(x) = \operatorname{argmax}_{y \in Y} \sum_{m=1}^M \log \frac{1 - \delta_m}{\delta_m} I(h_m(x) = y) \quad (6)$$

## B. Logistic Regression and LogitBoost

LogitBoost is a Boosting algorithm that applies LR for its created base learners, while fitting its generalized additive model to the training data. In this research we are using an ordinal class variable,  $y$ , with two values: yes for a client who has a loan and no for a client who does not have a loan. In LR, for  $n$  independent variables  $x_1, x_2, \dots, x_n$ , we find the probability  $p$  that a client will be granted a loan. Then, we apply logistic transformation,  $\operatorname{logist}(y)$ , that will find the linear relationship between the log odds and the  $n$  independent variables, or  $x$  predictors, by having:

$$\operatorname{logist}(y) = \ln \frac{p}{1-p} \quad (7), \text{ therefore,}$$

$$\operatorname{logist}(y) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_n x_n \quad (8)$$

In order to perform prediction, we needed to estimate the coefficients:  $\alpha_0, \alpha_1, \dots, \alpha_n$  using a statistical method such as the maximum likelihood estimation. For the entire sample data, the maximum likelihood function could be given by:

$$L(\alpha|y) = \prod_{i=1}^n p_i^{y_i} (1 - p_i)^{1-y_i} \quad (9)$$

Recalling the earlier explanation in this section for the theory behind Boosting, LogitBoost is basically seeking an additive LR model for its base learners. At iteration  $M$ , the created additive function is:

$$F(x) = \sum_{m=1}^M f_m(x) \quad (10)$$

As the trend in boosting, iterations start with  $W_i = \frac{1}{N}$ ,  $n \in \{1, 2, \dots, N\}$ ,  $F(x) = 0$ ,  $p(x_i) = 0.5$ . We calculate the weight  $w_i$  and the working response  $z_i$  as follows:

$$w_i = p(x_i)(1 - p(x_i)) \quad (11)$$

$$z_i = \frac{y_i - p(x_i)}{p(x_i)(1 - p(x_i))} \quad (12)$$

Then, we repeat for  $t$  iterations,  $t = 1, 2, \dots, T$ , while fitting  $f_t(x)$  by a weighted least-square regression from  $z_i$  to  $x_i$  using the calculated weights  $w_i$ 's. As we update  $F(x)$  iteratively, we end up at iteration  $T$  with our output classifier:

$$\operatorname{sign}[F(x)] = \operatorname{sign}[\sum_{i=1}^T f_i(x)] \quad (13)$$

### C. Decision Trees and Random Forests

One major advantage of DT, unlike most other machine learning models, that it is transparent as you can follow its hierarchical structure to understand how a prediction decision took place. In DT, *Entropy* measures disorder in the data, and can give an indication of how untidy the data is. For that reason, it is used as an algorithm to tidy the data by separating it and grouping the samples in the classes they belong to. A data set could be considered ordered, or tidy, when all the data items in it share the same label and is considered untidy if it has a blend of items with different labels. The DT algorithm utilizes the *Entropy* equation to ensure that each sub data group is in order and carries a similar label. Next, the information gain, *Gain*, is computed, and the best attribute with the highest gain is utilized as a node for further branching down the tree. Given a node, attribute, split argument  $\vec{S}$  by a certain value  $i$ , *Entropy* could be calculated using the following equations:

$$Entropy(\vec{S}) = -\sum_{j=1}^n \frac{|S_{ij}|}{|\vec{S}|} \log \left( \frac{|S_{ij}|}{|\vec{S}|} \right) \quad (14)$$

And the overall Gain is calculated for an attribute  $j$  as:

$$Gain(\vec{S}, j) = Entropy(\vec{S}) - Entropy(j/\vec{S}) \quad (15)$$

The DT classifier that we have discussed so far is the basic building block of the RF classifier. It applies the Bagging method discussed earlier to a group of tree learners over a data set  $D_n$ . In principle, the Random Forest classifier consists of  $M$  randomized DT's, where the predicted value of a new query sample  $p$  is denoted by  $m_n(p; S_j, D_n)$ .  $S_1$  to  $S_m$  represent the features for each data sample of the data set  $D_n$ . Since we are building a classification model in this research, the final RF model could be given by a majority voting among the  $M$  DT's:

$$m_n(p; S_1, \dots, S_m, D_n) = \begin{cases} 1, & \frac{1}{M} \sum_{j=1}^M m_n(p; S_j, D_n) \geq 0.5 \\ 0, & \text{otherwise} \end{cases} \quad (16)$$

### D. Model Evaluation Metrics and Interpretation Approach

Table 1. Confusion Matrix Parameters

		Actual	
Predicted	Positive	TP	FP
	Negative	FN	TN

In our research, confusion matrix was used for model evaluation, because it gives a summary of the performance of each machine learning model [23,24]. Table 1 shows a few parameters such as True Positive values (TP), True Negative values (TN), False Positive values (FP), and False Negative values (FN). Based on these parameters, model accuracy has been calculated using the following evaluation metrics:

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad (17)$$

$$Precision = \frac{TP}{TP + FP} \quad (18)$$

$$Recall = \frac{TP}{TP + FN} \quad (19)$$

$$F_1Score = 2 \left( \frac{Precision \times Recall}{Precision + Recall} \right) \quad (20)$$

For model interpretation, using black-box models versus using white-box models has been considered an important debate, especially for certain high-stakes decisions [25,26,27]. In our case, for the Bank Marketing data, deciding whether or not to approve a loan could be considered a high-stakes decision that would commit the client for regular deposits and many other financial benefits to the bank. DT is considered to be a white-box model, and it is well known for its ability to interpret predictions of most black-box machine learning models [28,29]. Therefore, the DT model induced rules could be used to explain to the decision makers at the bank why certain loan decisions are made; given that its performance is comparable to the other deployed black-box models. Eventually, we are not only seeking high models' accuracy, but also we need to balance that with a reasonable model interpretation, while avoiding over-fitting.

## III. DATA PRE-PROCESSING

Searching available Bank Marketing data sets, it was found that the Bank Marketing data set at Kaggle is one of the frequently analyzed data sets using machine learning techniques. In essence, there were two data sets at Kaggle related to bank marketing direct campaigns. The full data set has around 45,211 data instances with 20 features, out of which 10 were numeric features and 10 were categorical. The full data set was designed to be used for predicting whether or not the bank client will subscribe a long-term deposit. In this research, the focus is on the client's data that is believed to influence decision maker's choice of whether to

approve a loan or not. As shown in table 2, only a subset of the features was used: *age*, *job*, *marital*, *education*, *default*, *balance*, *loan*, and *housing*. By conducting an initial investigation of the data set, there were no missing values found in all the chosen attributes.

Table 2. Feature Descriptions for the Bank Marketing Data

Feature Name	Description	Type
age	Age in years	Numeric
Job	Type of occupation	Categorical
marital	Marital status	Categorical
education	Educational level	Categorical
default	Has credit in default?	Categorical
balance	Bank account balance	Numeric
housing	Has housing loan?	Categorical
loan	Has a personal loan?	Categorical

The main analysis tools that have been used in this research are the Python language and the Waikato Environment for Knowledge Analysis (WEKA) machine learning workbench [30]. To perform feature selection, further analysis was conducted to understand feature importance of the Bank Marketing data set. It was necessary to perform feature importance analysis using ensemble method of DT. This method compares the performance of the designed machine learning model with and without a specific feature out of the chosen seven features. As shown in the in Fig. 1, feature importance values ranged between 0.004 for the *housing* feature to 0.71 for the *balance* feature, which is considered for the designed DT model to have the most predictive value.

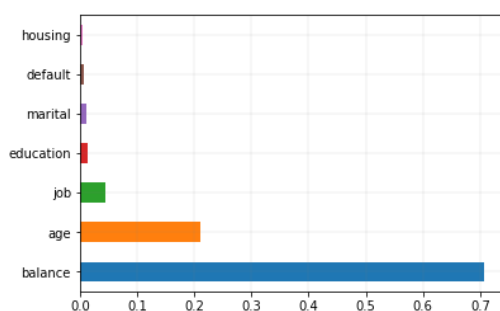


Fig.1. Significant Features

As we analyzed the Bank Marketing data set, there were two numeric attributes: *age* and *balance*, see Fig.2 and Fig.4. In our initial steps building the DT interpretation models, it was noticed that there were a few feature nodes repeated down the trees. Certainly, there would be no problem if the node is

repeated but in a different branch; however, for example, the *balance* attribute repetition was within the same branch leading to confusion during deduced rules' interpretation. Nevertheless, this is an expected behavior during tree construction when some features are numeric.

To overcome that aforementioned node repetition problem, data transformation was performed on those two numeric attributes: *age* and *balance*, by converting them into categorical variables. The choice of the numbers of categories was a challenge, especially for the *balance* attribute. Starting from the number of bins=5 to number of bins=50, many trials have been made, as we studied the *balance* attribute distribution, to carefully choose the suitable number of bins. A choice of number of bins=6 seemed to be a reasonable choice to be able to use a suitable and limited number of categories during our DT machine learning model design. Fig. 3 and Fig. 5 show the new distributions after attribute transformation from numeric to categorical.

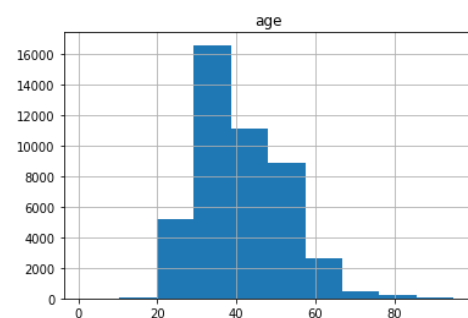


Fig.2. Age Attribute.

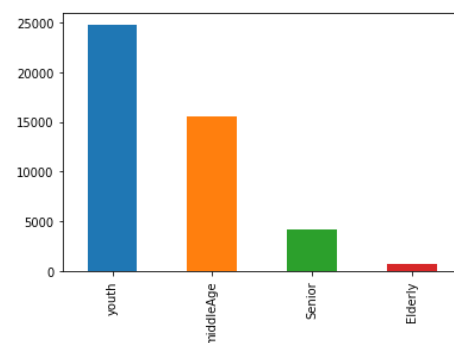


Fig.3. Age Categories.

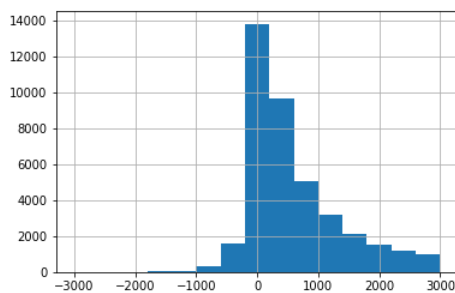


Fig.4. Balance Attribute.  
(Range: -3000 to 3000, 15 bins)

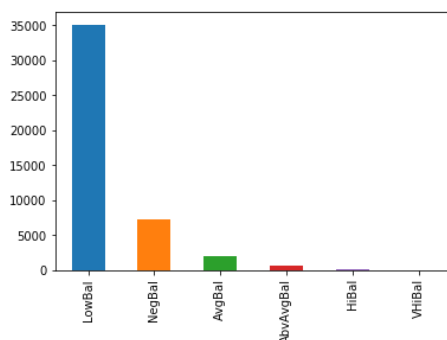


Fig.5. Balance Categories.

#### IV. RESULTS AND DISCUSSION

Recalling Fig. 1, by using feature importance analysis, the *balance* attribute had the highest rank, while the *default* attribute had the lowest rank. To settle on the best set of attributes, detailed analysis

had to be conducted by comparing the effect of each feature set on different machine learning models' performances.

For loan approval prediction, table 3 shows models' performances for different ensemble machine learning techniques such as AdaBoost, LogitBoost, Bagging, and Random Forest. As an overall comment on table 3, all the deployed ensemble methods had almost comparable performances; however, Boosting techniques performed slightly better than the Bagging techniques. The best performance was achieved by the LogitBoost model yielding approximately 84% accuracy and that performance is achieved when all the seven data set features were used. However, most deployed ensemble models performed better when the number of features was reduced to the *balance*, *age*, and *job* attributes. Despite the fact that ensemble techniques are renowned for its high performance, the nature of the collected features' observations in that subset of the Bank Marketing Data does not support accuracy higher than 84%.

We compared our obtained results with the state-of-the-art results found in the literature, see table 4. Our best classification results, average of approximately 84%, performed almost 25% better than other studies in most of the ensemble techniques used. The performance of Bagging techniques models in our study was so close to study of Papouškova and Hajek corresponding results, however in that study, the performance of

Table 3. Personal Loan Prediction Models' Performance with Different Selected Features

No.	Features	Performance	AdaBoost	LogitBoost	Bagging	Random Forests
1	All Features	Accuracy	83.98	83.96	83.94	83.81
		F1-Score	76.7	76.7	76.7	77
2	All, Except default	Accuracy	83.98	83.98	83.95	83.89
		F1-Score	76.7	76.7	76.7	76.8
3	All, Except default, marital& education	Accuracy	83.98	83.98	83.97	83.92
		F1-Score	76.7	76.7	76.7	76.7
4	age, job, balance	Accuracy	83.98	83.98	83.98	83.96
		F1-Score	76.7	76.7	76.7	76.7

Table 4. Comparing Loan Prediction Models' Performance with State-of-the-Art Results

No.	Research Study	Performance	AdaBoost	LR	MLP	LogitBoost	Bagging	RF
1	Our best Classification Results Loan Approval Prediction	Accuracy	83.98	83.94	83.84	83.96	83.94	83.81
		F1-Score	76.7	76.7	76.7	76.7	76.7	77
2	Papouškova and Hajek (2019) Loan Approval Prediction	Accuracy	66.9	69.77	N/A	68.22	82.39	87.97
		AUC	73.05	76.80	N/A	75.48	90.04	95.30
3	Song et al. (2020) Loan Approval Prediction	Accuracy	75.62	56.30	72.45	N/A	N/A	77.01
		AUC	52.24	56	48.92	N/A	N/A	57.95
4	Moscato et al (2020) Loan Approval Prediction	Accuracy	N/A	77	77.1	N/A	N/A	77.3
		AUC	N/A	68.5	70.4	N/A	N/A	72

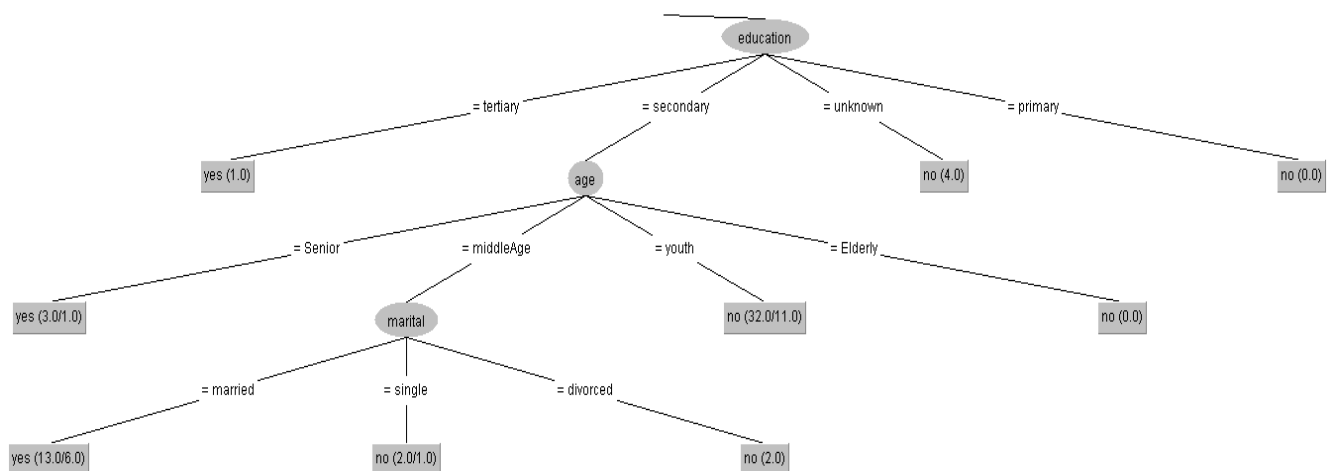


Fig.6. Sub-Tree Interpretation for the Personal Loan Approval Prediction Model.

their RF model performed better than ours. However, further interpretation analysis needed to be implemented as bank decision makers could be concerned with understanding the logic behind the loan approval models' predictions.

The interpretation given by the DT model showed some promising decision rules. The generated tree size was huge and hard to be put in one graph. Therefore, a smaller reduced-size sub-trees were chosen to show some possible model interpretations. The chosen sub-tree is branching out of previous leaves' splits at *balance*=NegBal, *default*=yes, and *job*=administrator. It shows a critical situation, where a client who works as an administrator has a negative balance and his credit is at default. A few promising rule chains have been generated stating that a personal loan could be granted to: clients whose education is tertiary, clients who are middle-age, married and have a secondary degree, but on the other hand, denied clients whose education status is unknown, see figure 6. It was also noticed that node splitting on the *default* feature appeared early at the personal loan prediction model. That is a good interpretation clue to the decision maker as it gives credit default an early attention during the decision making process.

## V. CONCLUSION AND FUTURE WORK

In this paper, a new different insight into the Kaggle Bank Marketing data was proposed. The Bank Marketing data set has been analyzed and loan approvals prediction models have been designed. Our best classification results had an average of approximately 84%. The used Boosting algorithms performed better than other studies by

approximately 25%, while the Bagging algorithms had a comparable performance with the other studies.

The designed DT model, as a white-box model, was informative as it interpreted, for example, one of the critical cases that a decision maker at the bank may encounter, which renders it advantageous over other black-box models, especially for this high-stakes loan decision. The provided interpretation information is vitally needed for bank decision makers to understand how to maintain balance between security and reliability of their financial lending system, while providing fair credit opportunities to their clients.

In future work, perhaps researchers can try different techniques to improve the performance of the loan prediction models using Deep Learning for example. Other researchers can work on designing the housing mortgage loan prediction models by analyzing the data using the *housing* attribute as a class label.

In general, given the efforts done so far, we believe that the objectives of this research have been achieved. It is anticipated that the use of the previous findings will be useful to the machine learning community in general, and to the bank marketing decision maker in particular. As well, the introduced analysis approach could open new directions by showing the process of leveraging the information gained from different investigated data sets that are being analyzed every day. In that sense, a data that is collected for a specific purpose could be useful to some extent for another constructive purpose.

*References:*

- [1] A. E. Bouchti, A. Chakroun, H. Abbar and C. Okar, "Fraud detection in banking using deep reinforcement learning," 2017 Seventh International Conference on Innovative Computing Technology (INTECH), Luton, 2017, pp. 58-63, doi: 10.1109/INTECH.2017.8102446.
- [2] A. Mead, T. Lewris, S. Prasanth, S. Adams, P. Alonzi and P. Beling, "Detecting fraud in adversarial environments: A reinforcement learning approach," 2018 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, 2018, pp. 118-122, doi: 10.1109/SIEDS.2018.8374720.
- [3] S. Kataria and M. T. Nafis, "Internet Banking Fraud Detection Using Deep Learning Based on Decision Tree and Multilayer Perceptron," 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2019, pp. 1298-1302.
- [4] C. Kumar and A. K. Dudyala, "Bank note authentication using decision tree rules and machine learning techniques," 2015 International Conference on Advances in Computer Engineering and Applications, Ghaziabad, 2015, pp. 310-314, doi: 10.1109/ICACEA.2015.7164721.
- [5] M. Singh, P. Ozarde and K. Abhiram, "Image Processing Based Detection of Counterfeit Indian Bank Notes," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bangalore, 2018, pp. 1-5, doi: 10.1109/ICCCNT.2018.8493763.
- [6] S. Moro, R. Laureano and P. Cortez, "Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology," In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121, Guimaraes, Portugal, October, 2011. EUROSIS.
- [7] S. Moro, P. Cortez and P. Rita, "A Data-Driven Approach to Predict the Success of Bank Telemarketing," Decision Support Systems, Elsevier, vol.(62), pp. 22-31, June 2014.
- [8] P. Ruangthong and S. Jaiyen, "Bank direct marketing analysis of asymmetric information based on machine learning," 2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE), Songkhla, 2015, pp. 93-96, doi: 10.1109/JCSSE.2015.7219777.
- [9] M. S. Islam, M. Arifuzzaman and M. S. Islam, "SMOTE Approach for Predicting the Success of Bank Telemarketing," 2019 4th Technology Innovation Management and Engineering Science International Conference (TIMES-iCON), Bangkok, Thailand, 2019, pp. 1-5, doi: 10.1109/TIMES-iCON47539.2019.9024630.
- [10] J. Asare-Frempong and M. Jayabalan, "Predicting customer response to bank direct telemarketing campaign," 2017 International Conference on Engineering Technology and Technopreneurship (ICE2T), Kuala Lumpur, 2017, pp. 1-4, doi: 10.1109/ICE2T.2017.8215961.
- [11] Annisa Nurul Puteri, Dewiani, Zulkifli Tahir, "Comparison of Potential Telemarketing Customers Predictions with a Data Mining Approach using the MLPNN and RBFNN Methods", Information and Communications Technology (ICOIACT) 2019 International Conference on, pp. 383-387, 2019.
- [12] Vincenzo Moscato, Antonio Picariello, Giancarlo Sperli, "A benchmark of machine learning approaches for credit score prediction, Expert Systems with Applications," Volume 165, 2021, 113986, ISSN 0957-4174, <https://doi.org/10.1016/j.eswa.2020.113986>.
- [13] Monika Papouskova, Petr Hajek, "Two-stage consumer credit risk modelling using heterogeneous ensemble learning," Decision Support Systems, Volume 118, 2019, Pages 33-45, ISSN 0167-9236, <https://doi.org/10.1016/j.dss.2019.01.002>.
- [14] Lin Ma, Xi Zhao, Zhili Zhou, Yuanyuan Liu, "A new aspect on P2P online lending default prediction using meta-level phone usage data in China, Decision Support Systems," Volume 111, 2018, Pages 60-71, ISSN 0167-9236, <https://doi.org/10.1016/j.dss.2018.05.001>.
- [15] Yu Song, Yuyan Wang, Xin Ye, Dujuan Wang, Yunqiang Yin, Yanzhang Wang, Multi-view ensemble learning based on distance-to-model and adaptive clustering for imbalanced credit risk assessment in P2P lending, Information Sciences, Volume 525, 2020, Pages 182-204, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2020.03.027>.
- [16] Martina Pokorná, Miroslav Sponer, Social Lending and Its Risks, Procedia - Social and Behavioral Sciences, Volume 220, 2016, Pages 330-337, ISSN 1877-0428, <https://doi.org/10.1016/j.sbspro.2016.05.506>.
- [17] Milad Malekipirbazari, VuralAksakalli, Risk assessment in social lending via random



- forests, *Expert Systems with Applications*, Volume 42, Issue 10, 2015, Pages 4621-4631, ISSN0957-4174,  
<https://doi.org/10.1016/j.eswa.2015.02.001>.
- [18] Zekic-Susac, Marijana&Sarlija, Natasa & Has, Adela & Bilandzic, Ana. Predicting company growth using logistic regression and neural networks. *Croatian Operational Research Review*. 7. 229-248. 10.17535/corr.2016.0016, 2016.
- [19] L.I. Kuncheva, M. Skurichina, R.P.W. Duin, An experimental study on diversity for bagging and boosting with linear classifiers, *Information Fusion*, Volume 3, Issue 4, 2002, Pages 245-258.
- [20] Wang, Boyu & Pineau, Joelle. (2016). Online Bagging and Boosting for Imbalanced Data Streams. *IEEE Transactions on Knowledge and Data Engineering*. PP. 1-1. 10.1109/TKDE.2016.2609424.
- [21] Friedman, Jerome & Hastie, Trevor & Tibshirani, Robert. (2000). Additive Logistic Regression: A Statistical View of Boosting. *The Annals of Statistics*. 28. 337-407. 10.1214/aos/1016218223.
- [22] Biau, G., Scornet, E. A random forest guided tour. *TEST* 25, 197–227 2016.
- [23] Witten, I.H., Frank, E. and Hall, M.A., *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publications, 2017.
- [24] Willi Richert, and Luis Pedro Coelho, *Building Machine Learning Systems with Python*. Packt Publishing, 2013.
- [25] O. Loyola-González, "Black-Box vs. White-Box: Understanding Their Advantages and Weaknesses From a Practical Point of View," in *IEEE Access*, vol. 7, pp. 154096-154113, 2019, doi: 10.1109/ACCESS.2019.2949286.
- [26] Rudin, C, " Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, pp. 206–215, 2019, <https://doi.org/10.1038/s42256-019-0048-x>.
- [27] Freitas, A., "Comprehensible classification models: a position paper," *ACM SIGKDD Explorations Newsletter*, vol. 15, pp. 1–10, 2014, doi: 10.1145/2594473.2594475.
- [28] T. Ueno and Q. Zhao, "Interpretation of Deep Neural Networks Based on Decision Trees," 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech), Athens, 2018, pp. 256-261, doi:10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00052.
- [29] N. Schaaf, M. Huber and J. Maucher, "Enhancing Decision Tree Based Interpretation of Deep Neural Networks through L1-Orthogonal Regularization," 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), Boca Raton, FL, USA, 2019, pp. 42-49, doi: 10.1109/ICMLA.2019.00016.
- [30] WEKA, "The Workbench for Machine Learning". Available online at: <http://www.cs.waikato.ac.nz/ml/weka>. [Accessed: 020-Oct-2020].

## **Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)**

This article is published under the terms of the Creative Commons Attribution License 4.0

[https://creativecommons.org/licenses/by/4.0/deed.en\\_US](https://creativecommons.org/licenses/by/4.0/deed.en_US)