Unveiling the Power of Machine Learning: A Comparative Analysis in predicting Bank Term Deposit Subscriptions for Marketing data

Alexandra Espialidou
Faculty of Engineering, Environment and Computing
MSc Data Science
Coventry University
espialidoa@uni.coventry.ac.uk

Abstract - This project focuses on analyzing the direct marketing campaigns of a Portuguese banking institution. The dataset consists of client information and the target variable indicates whether the client subscribed to a term deposit ('yes') or not ('no'). The main objective of this paper is to employ classification algorithms, specifically Logistic Regression, Random Forest, and K-Nearest Neighbors, to predict the outcomes of unknown data. The study aims to assess the performance of these algorithms in classifying clients' subscription decisions and provide insights for improving marketing strategies in the banking sector.

Keywords – classification; marketing data; machine learning; logistic regression; Random Forest; K-nearest;

I. INTRODUCTION

THIS paper focuses on marketing data regarding the subscription of an individual to a short term deposit service from Portuguese banks. Direct marketing campaigns play a crucial role in the banking industry, where personalized targeting is key to achieving higher conversion rates and minimizing costs. 72% of financial services firms in the UK reported using Machine Learning applications in their everyday operations (Bank of England, 2022 [5]). By leveraging machine learning algorithms, banks can analyze vast amounts of client information and make accurate predictions about whether a client is likely to subscribe to a bank service or not. Portuguese banks, through direct marketing campaigns, reached out to potential clients via phone calls, provided information about their term deposit service and its benefits, and then reported their responses. In the end, a dataset was compiled consisting of clients' responses regarding their interest in subscribing to a term deposit, along with their corresponding personal data. This dataset provides a representative sample of targeted clients and a solid foundation for conducting machine learning research and classification.

The problem that this dataset explores is the possible subscription of clients, based on their personal characteristics to a term deposit service. On the basis of the campaign data, a correlation between the available personal data and the subscription can be found. By using 3 distinct machine learning models (Logistic Regression, Random Forest, K-Nearest Neighbour), this paper aims to explore their efficacy in predicting clients' decisions within the context of direct marketing campaigns.

The results were acquired with the use of HP 15s - eq2035nv (laptop), equipped with AMD Ryzen 5 4500U CPU with Radeon Graphics with 2.38 GHz and 8.00 GB RAM.

II. THE DATA SET

The dataset used for this paper was obtained from the UCI Machine Learning repository. It was created by Sérgio Moro (ISCTE-IUL), Paulo Cortez (Univ. Minho) and Paulo Rita (ISCTE-IUL) in 2012. The data set is named: "Bank_additional_full.csv" and consists of a total of 41,188 instances and 21 attributes with no missing values. Each instance in the dataset corresponds to the comprehensive information associated with a client, including various attributes relevant to the marketing campaign. This dataset serves as a valuable resource for analyzing and modeling the client data in order to gain insights and make predictions. The 21 columns describe the client and their characteristics. They consist of 10 numerical and 11 categorical attributes. The last column, consists of the client's answer to whether or not they will subscribe to a term deposit service. This column is binary, including 'yes' and 'no' answers. The previous 5 attributes are social and economic context attributes and they are all numerical. These attributes provide additional information about the macroeconomic environment and consumer behaviour during the marketing campaigns.

Within the data folder of the UCI archive, four distinct datasets were available for analysis.: 1 with 41,188 rows and 20 columns, ordered by date, 1 with 10% of the rows (4,119) and 20 columns, which were randomly selected from the bank-additional-full.csv dataset, 1 with 41,188 rows and 17 columns, ordered by date and lastly, 1 with 10 % of the examples of the previous dataset. For the purposes of this paper, the first dataset was selected because it contains the most data and enables a thorough exploration of the underlying patterns and relationships within the data.

TABLE 5 presents a detailed overview of the dataset variables, encompassing their descriptions, data types, and ranges as originally provided in the UCI repository.

III. DATA PREPARATION

The dataset from the UCI repository underwent preprocessing steps to ensure accurate and fair comparisons among the three structured algorithms This section examines the data analysis and data preparation that the dataset underwent. The dataset dealt with class balancing, bivariate analysis, categorical feature encoding, data standardization and Principal Component Analysis (PCA). The dataset obtained from the repository is devoid of any missing values in its attributes. However, it is worth noting that the columns: 'job', 'marital', 'education', 'default', 'housing', and 'loan', contain the value 'unknown'. Although the presence of these unknown values has the potential to impact the classification process, no modifications or imputations were made.

A) CLASS IMBALANCE:

Class imbalance is the problem where the training dataset has an uneven distribution of classes [3].

By running exploratory data analysis, it is evident that the dataset presents class imbalance as the class 'No' data values are more than the class 'Yes' values, as Fig. 1 below shows. Imbalanced datasets lead to poor performance and wrong classifications. This is because the majority of the machine learning algorithms were developed with the presumption that each class would have an equal number of examples [3].

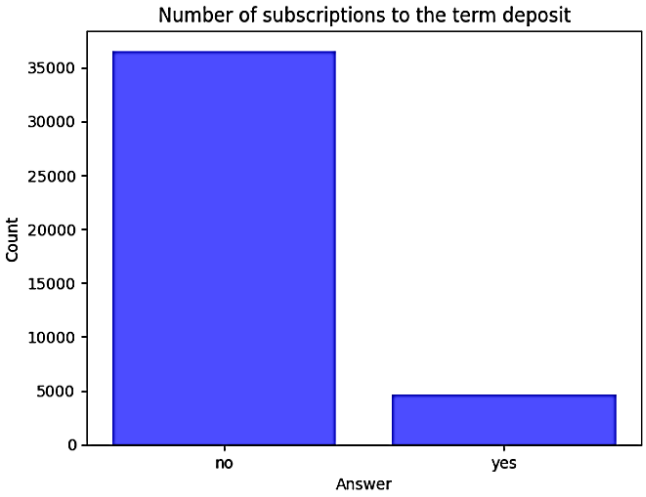

Fig. 1 The number of subscriptions to the term deposit service

From Fig 1 it can be seen that there is severe imbalance. To address this issue, the algorithm from the imbalanced learning library SMOTE (Synthetic Minority Over-sampling Technique) was used. SMOTE is a technique that generates synthetic samples for the minority class of the class imbalance, in order to balance the data. Synthetic examples of 'yes' values were generated in order to match the value of 'no' values. Balancing the dataset helped to mitigate the impact of skewed class distributions and enhance the model's ability to capture patterns from both classes.

B) BIVARIATE ANALYSIS:

In the bivariate analysis of the dataset, a focus was placed on the numerical attributes in order to explore the correlation between different attributes. It was observed that the social and economic context attributes exhibited high correlation, especially between emp.var.rate (employment variation rate) and euribor3m (euribor 3-month rate) with 97% and between euribor3m (euribor 3*-month rate) and nr.employed (number of employees) with 95% as seen in the correlation matrix shown in Fig. 2. PCA is applied to these attributes.

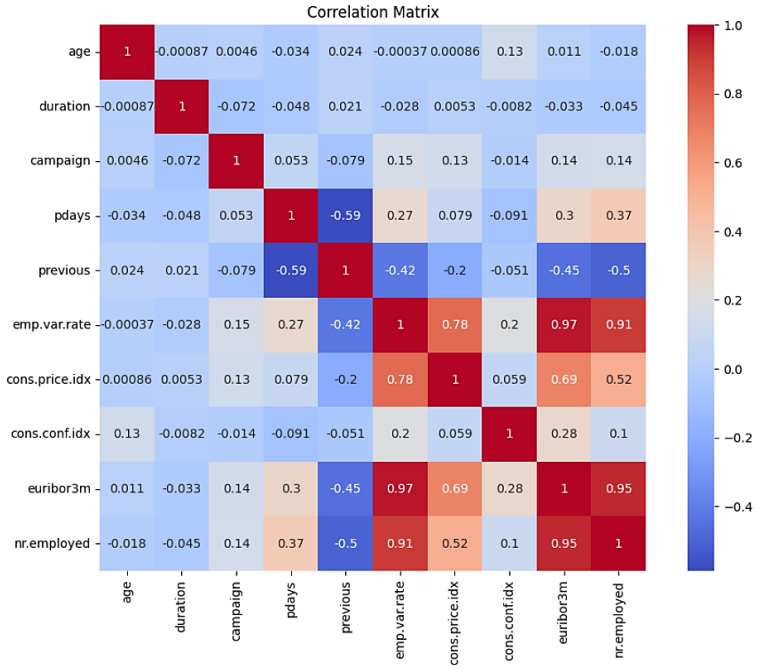

Fig. 2 The correlation matrix of numerical attributes

C) PCA:

Due to the strong correlation observed among the five social and economic features, a dimensionality reduction technique called Principal Component Analysis (PCA) was employed from the sklearn.decomposition library. PCA was used to transform the five correlated features into a single composite feature, preserving the most significant information while reducing the dimensionality of the dataset. This approach helps to mitigate multicollinearity and improve computational efficiency in subsequent analysis and modeling steps.

D) NUMERICAL REPRESENTATION:

The machine learning algorithms that were used in this paper mostly work with numerical data. To ensure compatibility, categorical data were converted to numerical data using python library scikit-learn utilizing *LabelEncoder* or *Pandas* in order to encode text into numbers. The following attributes were converted: *job, martial, education, default, housing, loan, contact, month, day_of_week, poutcome* along with the target value *'y'*. The target value, which was initially a string

representation of ‘yes’ and ‘no’ was converted to integer representations of ‘1’ and ‘0’, respectively.

E) DATA STANDARDIZATION:

After the encoding process, it becomes apparent that the newly created variables, representing the categorical features, are not on the same scale as the original numerical features. This discrepancy necessitates the need for data standardization. Data standardization is a technique used to transform the dataset into a common format, ensuring that all features are on a comparable scale. By standardizing the data, we prevent any individual feature from exerting a disproportionate influence on the learning algorithm, enabling fair and unbiased model training and evaluation. For instance, the variable '*pdays*' has a range of 0-999, while the variable '*job*' has a range of 0-11. This discrepancy in ranges implies that a value of 999 in '*pdays*' may appear to have a larger influence than a value of 11 in '*job*'. With the use of StandardScaler from the sklearn.preprocessing library, which works by scaling each feature to a given range on the train set, the dataset is scaled.

The processed dataset, after all the necessary transformations and preprocessing steps, consists of 41,188 rows and 17 columns. This refined dataset is now ready to be utilized for further analysis and modeling tasks.

IV. MACHINE LEARNING CLASSIFICATION TECHNIQUES

A) LOGISTIC REGRESSION

Logistic Regression is a supervised statistical method used for binary classification tasks. It employs a logistic function, called sigmoid function, to model the relationship between the input features and the binary target outcome. In the context of this paper, the target variable represents the likelihood of a client subscribing to a term deposit. Logistic regression maps the given features to either 0 or 1, corresponding to the probabilities of the two possible outcomes. The threshold value, typically set at 0.50, serves as a classification boundary. If the predicted probability is above the threshold, the instance is classified into one class, otherwise, it is assigned to the other class.

The sigmoid function,

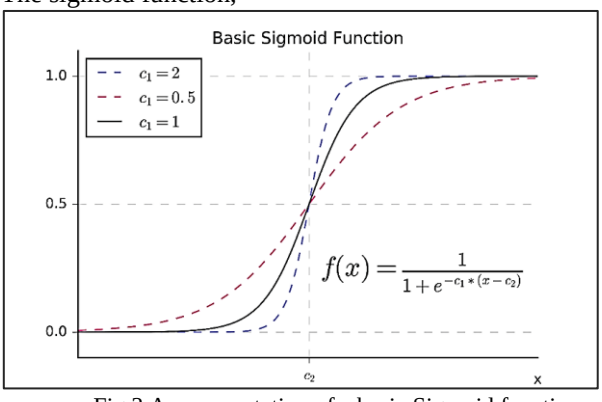

Fig.3 A representation of a basic Sigmoid function

(*f(x) = 1 / (1 + e^(-x))*), is an S-shaped mathematical function that maps input values to a range between 0 and 1. The sigmoid function facilitates the transformation of the linear combination of features into predicted probabilities, enabling the interpretation of results and informed classification decisions based on these probabilities.

B) RANDOM FOREST

Random Forest is a supervised ensemble machine learning algorithm used for classification and regression tasks. It combines multiple decision trees, each trained on a different random subset of the dataset in order to make predictions. The algorithm aggregates the predictions from individual trees to make the final decision. In classification, the class with the majority vote is selected, while in regression, the predictions are averaged. The random selection of subsets helps mitigate overfitting and improves the model's ability to generalize to unseen data. Random Forest is a robust and versatile ensemble algorithm that can handle diverse datasets by randomly selecting subsets, which reduces overfitting and improves generalization.

Random Forest

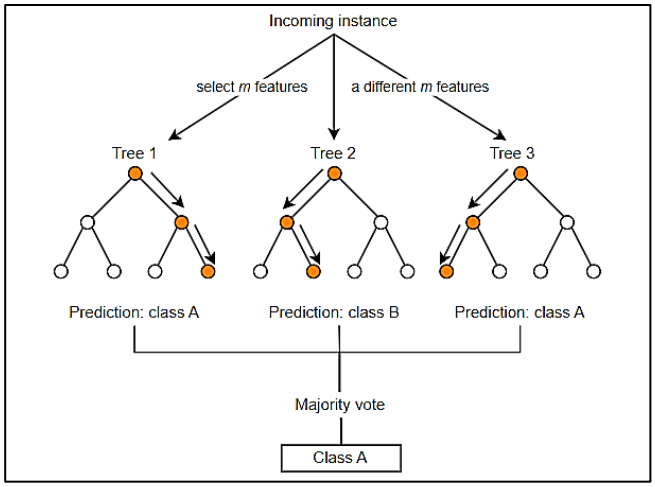

Fig. 4 A representation of Random Forest

C) K- NEAREST NEIGHBOUR

K-Nearest Neighbors (KNN) is a supervised machine learning algorithm used for both classification and regression tasks. It operates on the principle of similarity, where it finds the k number of data points in the training set that are closest to the input data point. The value of k, which is a hyperparameter, determines the number of neighbors to consider. K can be any integer but the most common used ones are 3 and 5. This method locates the k- nearest examples of the data points in the training set with the input data points as shown in fig. 5.

K-Nearest Neighbour

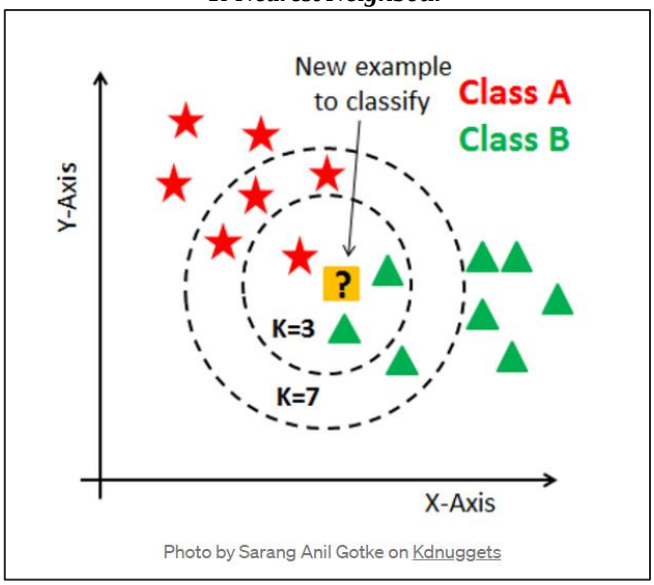

Fig. 5 A representation of K- Nearest Neighbours

Source: https://ai.plainenglish.io/introduction-to-k-nearest-neighbors-knn-algorithm-e8617a448fa8

The similarity between points is usually calculated with metrics such as the Euclidean distance, Manhattan distance etc. The most common one is the Euclidean distance. The class that the majority of the k nearest neighbours belong to, is the class that is assigned to this input data point. In order to select the optimal k, the *Hyperparameter tuning* is used. This process aims to test the training dataset with various parameters and find the best combination of hyperparameter values that lead to the highest model performance.

V. RESULTS AND MODEL EVALUATION

The following table presents the results of the three classification algorithms that were used.

TABLE 1: CLASSIFICATION RESULTS

Classification algorithm	Accuracy	% of accuracy
Logistic Regression	0.8514202476329206	85.14%
KNN	0.8770737233956462	87.70%
Random Forest	0.9071781176661002	90.71%

After extensive analysis and evaluation of all three of the classification algorithms, it is evident that Random Forest outperformed the remaining two methods and achieved the highest accuracy with 90.71%. Despite this distinction, all three methods showcased promising results in predicting the likelihood of clients subscribing to a term deposit service.

TABLES 2,3 and *4* provide an evaluation of the machine learning models' performance. The metrics in the classification report are precision, recall, f1-score, support, accuracy, Macro Avg and Weighted Avg. Precision, or Positive Predicted Value, calculates the number of True positives out of all the predicted instances. Recall, also known as sensitivity, measures the proportion of True positives out of all the actual positive instances. F1-score is a mean of precision and recall and what actually makes the accuracy score. Accuracy measures the true positives and true negatives out of all the predicted instances, giving the accuracy of the model. Macro avg calculates the average of precision, recall, and F1-score for each class and weighted avg gives the weighted average of precision, recall, and f1-score for each class. Support indicates the number of instances in class 0 and class 1, showing the distribution across the different classes.

TABLE 3 CLASSIFICATION REPORT FOR LOGISTIC REGRESSION

	precision	recall	f1-score	support
0	0.98	0.85	0.91	10968
1	0.42	0.87	0.57	1389
accuracy			0.85	12357
macro avg	0.70	0.86	0.74	12357
weighted avg	0.92	0.85	0.87	12357

Accuracy: 0.8514202476329206

Based on the classification report for Logistic Regression, the following key observations can be made:
For precision, for class 0, 98% of the instances predicted as class 0, were actually class 0 and for class 1, only 42% of the instances predicted as class 1, were actually class 1. Regarding recall, for class 0, 85% out of all the class 0 instances were correctly predicted as class 0, while for class 1, 87% of the class 1 instances were correctly predicted as class 1. Overall, the model demonstrates high precision for class 0, but suggests a high number of false positives for class 1. Since the recall is reasonably high for both classes, the model captures a substantial proportion of instances for both class 0 and class 1.

TABLE 3 CLASSIFICATION REPORT FOR KNN

	precision	recall	f1-score	support
0	0.93	0.93	0.93	10968
1	0.46	0.49	0.47	1389
accuracy			0.88	12357
macro avg	0.70	0.71	0.70	12357
weighted avg	0.88	0.88	0.88	12357

Accuracy: 0.8770737233956462

Moreover, based on the classification report for KNN, the following key observations can be made:
For precision, for class 0, 93% of the instances predicted as class 0, were actually class 0 and for class 1, only 46% of the instances predicted as class 1, were actually class 1. Regarding recall, for class 0, 93% out of all the class 0 instances were correctly predicted as class 0, while for class 1, 49% of the class 1 instances were correctly predicted as class 1. This suggests that the model may struggle to accurately identify instances belonging to class 1, potentially resulting in a higher number of false positives for this class.

TABLE 4 CLASSIFICATION REPORT FOR RANDOM FOREST

	precision	recall	f1-score	support
0	0.95	0.94	0.95	10968
1	0.58	0.64	0.61	1389
accuracy			0.91	12357
macro avg	0.77	0.79	0.78	12357
weighted avg	0.91	0.91	0.91	12357

Accuracy: 0.9071781176661002

Finally, based on the classification report for Random Forest, the following key observations can be made:
For precision, for class 0, 95% of the instances predicted as class 0, were actually class 0 and for class 1, only 58% of the instances predicted as class 1, were actually class 1. Regarding recall, for class 0, 94% out of all the class 0 instances were correctly predicted as class 0, while for class 1, 64% of the class 1 instances were correctly predicted as class 1. Considering all factors, the model indicates high precision for class 0, and a reasonably high recall for both classes, even though it is higher for class 0. However, there is room for improvement in achieving a better balance for class 1.

As this paper shows, bank marketing data are indeed well-suited for machine learning experiments due to their often-inherent structure and abundance of data. Data relatable to marketing and banking often contain a wide range of variables and features that capture various aspects of customer behaviour, demographics, and financial patterns. This rich set of information offers valuable insights and potential predictive power when leveraged by machine learning algorithms.

Overall, this study contributes to the understanding of classification techniques in the context of the bank marketing dataset and serves as a stepping stone for future research and practical applications in the field of machine learning and data analysis. With the rapid advancement of machine learning, by understanding the strengths and weaknesses of different classification techniques, and by analyzing credit defaults data, banks and financial institutions can make more informed decisions, manage their portfolios effectively and the optimization of marketing techniques and improvision of customer targeting can be accomplished.

TABLE *5*

No	***Column***	***Description***	***Type***	***Range***
1	*Age*	*Age of the client*	*Numeric*	*17 to 98*
2	*Job*	*Type of job*	*Categorical*	*'housemaid', 'services', 'admin.', 'blue-collar', 'technician', 'retired', 'management', 'unemployed', 'self-employed', 'unknown', 'entrepreneur', 'student'*
3	*Marital*	*Marital status*	*Categorical*	*married', 'single', 'divorced', 'unknown'*
4	*Education*	*Education level*	*Categorical*	*basic.4y', 'high.school', 'basic.6y', 'basic.9y', 'professional.course', 'unknown', 'university.degree', 'illiterate'*
5	*Default*	*Defaulted credit categorization*	*Categorical*	*Yes, no, unknown*
6	*Housing*	*Housing loan classification*	*Categorical*	*Yes, no, unknown*
7	*Loan*	*Personal loan classification*	*Categorical*	*Yes, no, unknown*
8	*Contact*	*Contact communication type*	*Categorical*	*Telephone, cellular*
9	*Month*	*Month that the last contact was made*	*Categorical*	*March to December*
10	*Day_of_week*	*Day of the week that the last contact was made*	*Categorical*	*Monday to Friday*
11	*Duration*	*Duration of the last contact*	*Numeric*	*0 to 4918*
12	*Campaign*	*Number of contacts to this client (including the last one)*	*Numeric*	*1 to 56*
13	*Pdays*	*Number of days that passed by after the client was last contacted from a previous campaign*	*Numeric*	*0 to 999*
14	*Previous*	*Number of contacts performed before this campaign*	*Numeric*	*0 to 7*
15	*Poutcome*	*Outcome of the previous marketing campaign*	*categorical*	'nonexistent', 'failure', 'success'
16	*Emp.var.rate*	*employment variation rate - quarterly indicator*	*Numeric*	*-3.4 to 1.4*
17	*Cons.price.idx*	*consumer price index - monthly indicator*	*Numeric*	*92.201 to 94.767*
18	*Cons.conf.idx*	*consumer confidence index - monthly indicator*	*Numeric*	*-50.8 to -26.9*
19	*Euribor3m*	*- daily indicator*	*Numeric*	*0.634 to 5.045*
20	*Nr.employed*	*number of employees - quarterly indicator*	*Numeric*	*4963.6 to 5228.1*
21	*y*	*Term Deposit Subscription*	*Binary*	*Yes, no*

VI. APPENDIX

This link contains the dataset, the python code used for the pre-processing of the data set and for the application of the classification techniques:

https://github.com/AlexEspialidou/7072CEM-CW

VII. REFERENCES

[1] Barnes-Batterbee, R. (2022). *Recognizing and Classfying Human Activity Using Sensors*.

[2] Bertulis, A. (2022). *Prediction of Online Shoppers Purchasing Intentions using Support Vector Machines, K-Nearest Neighbor, Random Forest, Adaptive Boosting and Extremely Randomized Trees Algorithms*.

[3] Brownlee, J. (2020, January 16). *SMOTE for Imbalanced Classification with Python*. Machine Learning Mastery. https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/

[4] Bhaidkar, Y. (2021, June 28). *Prediction of term deposit subscription for Portuguese banking institution Using machine Learning….* Medium. https://yashwantbhaidkar.medium.com/prediction-of-term-deposit-subscription-for-portuguese-banking-institution-using-machine-learning-ff0a0e7609fe

[5] Blake, K., Gharbawi, M., Thew, O., Visavadia (Bank), S., Gosland, L., & Mueller (FCA), H. (2022, October 11). *Machine learning in UK financial services*. Www.bankofengland.co.uk. https://www.bankofengland.co.uk/report/2022/machine-learning-in-uk-financial-services

[6] Brownlee, J. (2019, December 22). *A Gentle Introduction to Imbalanced Classification*. Machine Learning Mastery. https://machinelearningmastery.com/what-is-imbalanced-classification/

[7] Cislo, P. (2022). *Classification of Audio Features Using Machine Learning Algorithms*.

[8] E R, S. (2021, June 17). *Random Forest | Introduction to Random Forest Algorithm*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/

[9] Moro, S., Cortez, P., & Rita, P. (2014). *UCI Machine Learning Repository*. Archive.ics.uci.edu. https://archive.ics.uci.edu/dataset/222/bank+marketing

[10] Negrini de Araujo, J. (2022). *Application of machine learning techniques for stock price direction forecasting*.

VIII. FIGURES

Fig. 3: Leibovich-Raveh, T. (2008). *Figure 2: A Basic sigmoid function with two parameters (c1 and c2) as...* ResearchGate. https://www.researchgate.net/figure/A-Basic-sigmoid-function-with-two-parameters-c1-and-c2-as-commonly-used-for-subitizing_fig2_325868989

Fig. 4: Wood, T. (2019, May 17). *Random Forests*. DeepAI. https://deepai.org/machine-learning-glossary-and-terms/random-forest

Fig. 5: Shah, R. (2021, March 5). *Introduction to k-Nearest Neighbors (kNN) Algorithm*. Medium. https://ai.plainenglish.io/introduction-to-k-nearest-neighbors-knn-algorithm-e8617a448fa8