Olateju Mary || 23029979

PREDICTING AND INTERPRETING THE IMPACT OF MARKETING CAMPAIGNS ON BANK'S PRODUCT ACQUISITION AND PERFORMANCE: A MACHINE LEARNING APPLICATION.

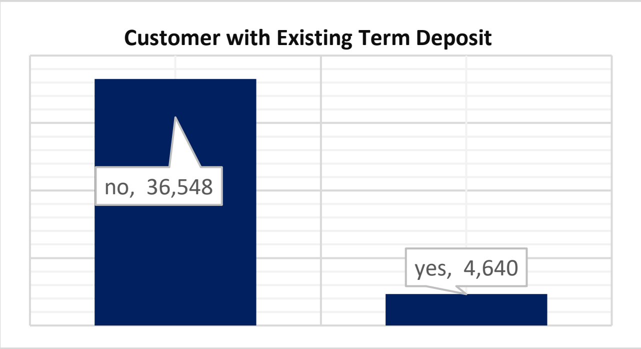
INTRODUCTION

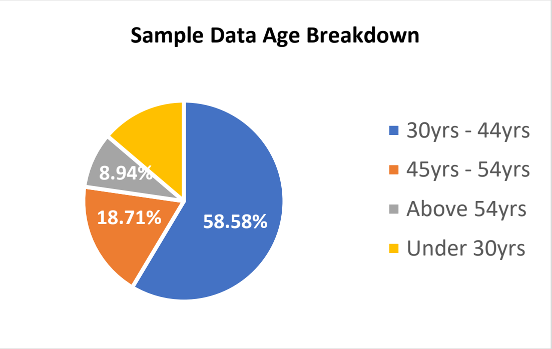
For retail banks, proper marketing campaign is important for success both in product acquisition and churn. With limited differentiation between banking products in the industry, quality marketing campaigns are essential to stand out from competitors. However, launching a successful campaign in a highly competitive market can be challenging. Customers sometimes feels choked with ads and are learning to ignore them. That's where personalized direct marketing comes in. This approach allows for customization of product properties on a per-customer basis, increasing efficiency and reducing costs. Retail banks have access to vast amounts of client transaction data, making personalization a natural extension of mass campaigns. As marketing continues to evolve, personalized direct marketing is becoming increasingly popular.

DATASET OVERVIEW

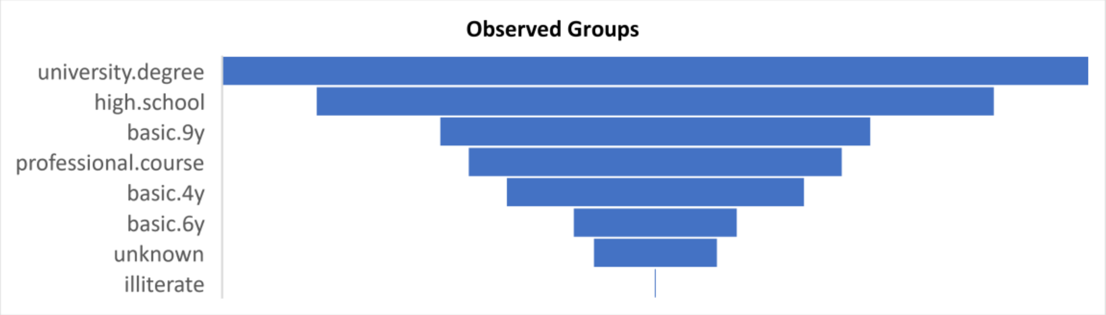
The data is related with direct marketing campaigns (phone calls) of a Portuguese banking institution (S. Moro et al. 2012). The data sample contacted was 41,188 customers, with a composition of 88.73% of clients without term-deposit and 11.27% of sample data has an existing product.

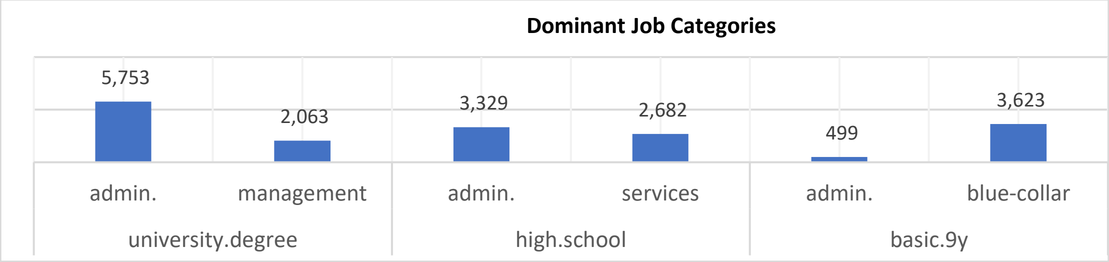
- 20 input features/variables
- One binary target variable "y" indicating if the customer subscribed to a term deposit or not.





Largest Groups/Dominant Job Categories:





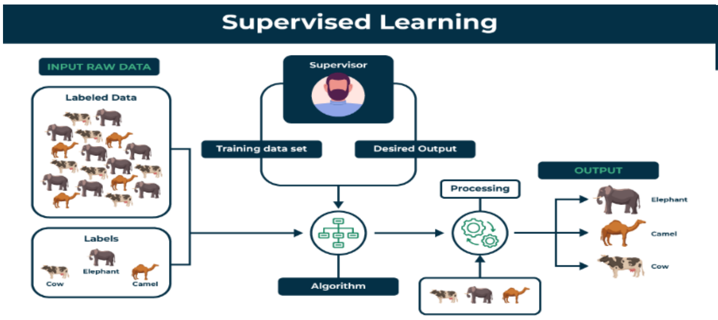
- **Potential Further Analysis:**
  - Analysing the relationship between education levels and job categories using statistical methods like chi-square tests or logistic regression could provide insights into the association between these variables.
  - Investigating potential differences in job categories based on other demographic factors, such as age or gender, could add additional context to the analysis.

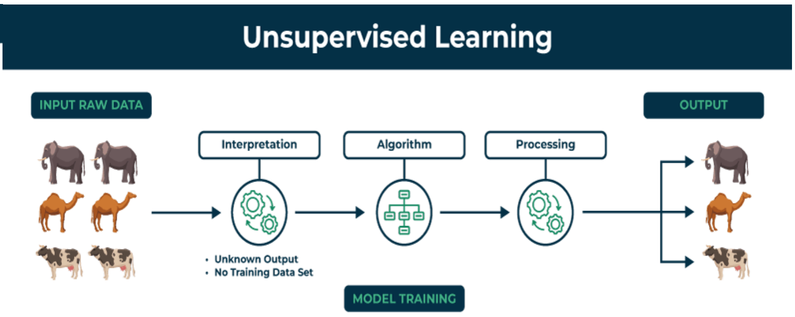
PROBLEM DEFINITION

A functional predictive model for this binary classification task is important to ensure success in the bank's marketing efforts, resource allocation and increased term-deposit acquisition rate. In the competitive banking sector, the effectiveness of marketing campaigns is crucial for product acquisition and overall performance. However, predicting and interpreting the impact of these campaigns, particularly direct marketing campaigns via phone calls, poses a significant challenge. This is due to the complex nature of customer behaviour, influenced by a different factor that are often difficult to quantify and track. The Portuguese banking institution, as studied by S. Moro et al. (2012), has been actively conducting direct marketing campaigns. Despite the volume of data generated from these campaigns, the bank faces difficulties in accurately predicting the success rate of these campaigns and understanding the underlying factors that contribute to the acquisition of banking products by customers.

The problem, therefore, is developing a machine learning model that can not only predict the outcome of these marketing campaigns but also provide interpretable results. This would enable the bank to optimize their marketing strategies, improve customer targeting, and enhance product acquisition and performance.

Supervised learning is a type of machine learning algorithm that learns from labelled data. Labelled data is data that has been tagged with a correct answer or classification. Unsupervised learning is a type of machine learning that learns from unlabelled data (Julianna Delua et al, 2021). This means that the data does not have any pre-existing labels or categories.





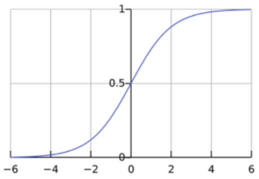
In this case, the target variable is the binary variable "y - has the client subscribed a term deposit?" with values "yes" or "no".

Some suitable supervised learning algorithms for this binary classification problem include:

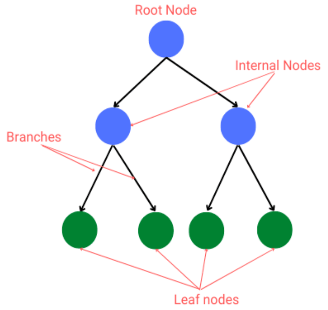
- Logistic Regression: Logistic Regression is a data analysis technique that uses mathematics to find the relationships between two data factors. It then uses this relationship to predict the value of one of those factors based on the other (AWS). To calculate logistics regression:

$$f(x) = \frac{1}{1 + e^{-x}}$$

If logistics regression equation is plotted, the result is the S-curve as shown below:



- Decision Trees and Random Forests: Decision Trees are supervised machine-learning algorithms for classification and regression problems. A decision tree builds its model in a flowchart-like tree structure, where decisions are made from a bunch of "if-then-else" statements (IBM). Random Forests are an ensemble learning method that combines multiple decision trees, reducing overfitting and improving predictive performance.

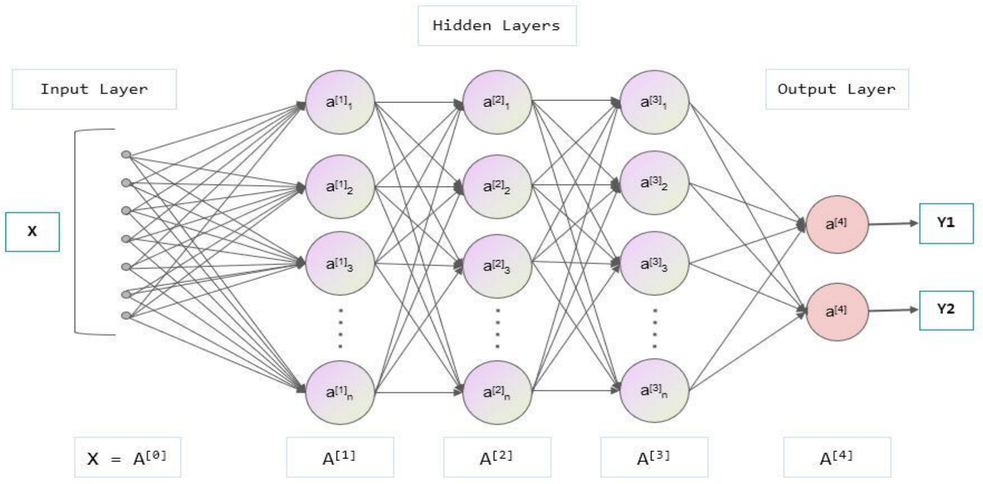


Structure of a decision tree:

- **Root node:** first node in the tree. This node has no incoming branches.
- **Internal nodes:** these are the decision nodes.
- **Branches:** they are arrows connecting nodes. They represent the decision-making steps.
- **Leaf Nodes:** these nodes represent all the possible outcomes from a dataset.

Decision trees allow continuous data splitting based on given parameters until a final decision is reached. They

- Neural Networks: A neural network is a method in artificial intelligence that teaches computers to process data in a way that is inspired by the human brain. It is a type of machine learning process, called deep learning, that uses interconnected nodes or neurons in a layered structure that resembles the human brain (AWS).



Input Layer: Information from the outside world enters the artificial neural network from the input layer. Input nodes process the data, analyse, or categorize it, and pass it on to the next layer.

Hidden Layer: Hidden layers take their input from the input layer or other hidden layers. Artificial neural networks can have many hidden layers. Each hidden layer analyses the output from the previous layer, processes it further, and passes it on to the next layer.

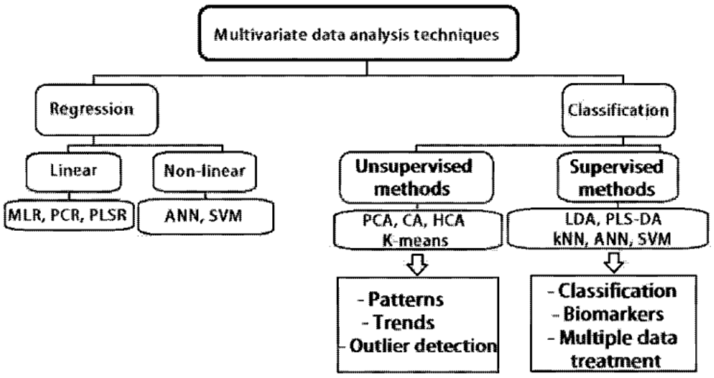
Output Layer: The output layer gives the result of all the data processing by the artificial neural network. It can have single or multiple nodes. For instance, if we have a binary (yes/no) classification problem, the output layer will have one output node, which will give the result as 1 or 0. However, if we have a multi-class classification problem, the output layer might consist of more than one output node.

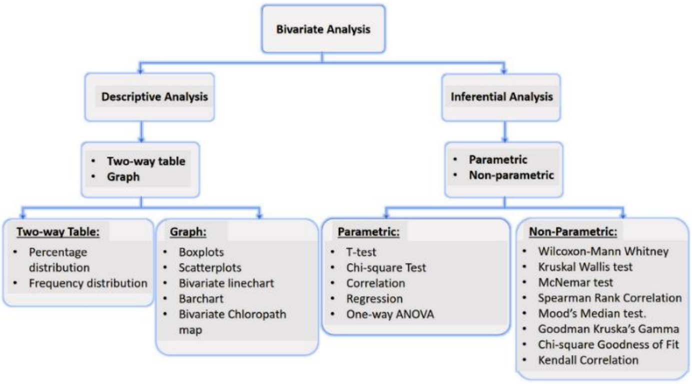
All these algorithms fall under the category of supervised learning because they require a labelled dataset with known target values (subscription: "yes" or "no") to learn the mapping function from the input features to the target variable.

BIVARIATE ANALYSIS VS MULTIVARIATE ANALYSIS

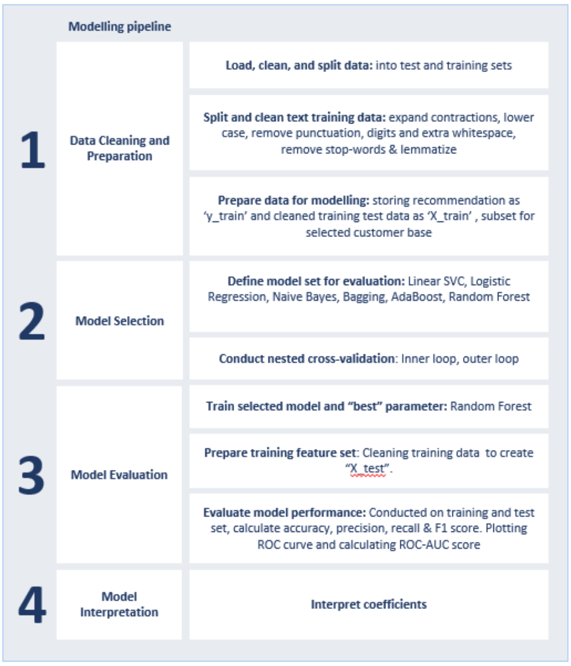
Bivariate analysis involves analysing the relationship between two variables, typically the target variable and one predictor variable at a time. While this analysis can provide insights into the individual relationships between each predictor and the target, it fails to capture the combined effects and interactions among multiple predictors, which is often the case in real-world scenarios.

Multivariate analysis factors in relationships between multiple independent variables and the target variable simultaneously. It considers the potential interactions and combined effects of the predictors, which can lead to more accurate and reliable predictions. Multivariate offers a wide range of benefits for the case study which include Capturing complex relationships, handling confounding variables, feature selection and importance, improved prediction accuracy.



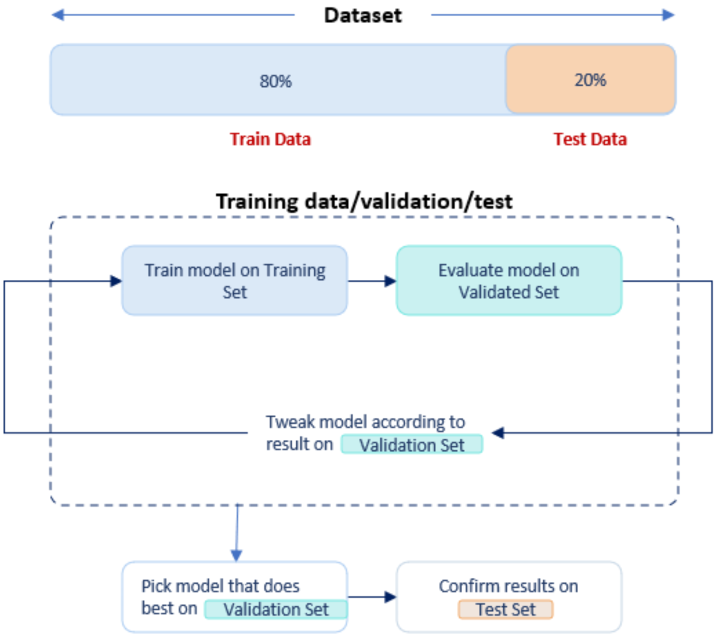


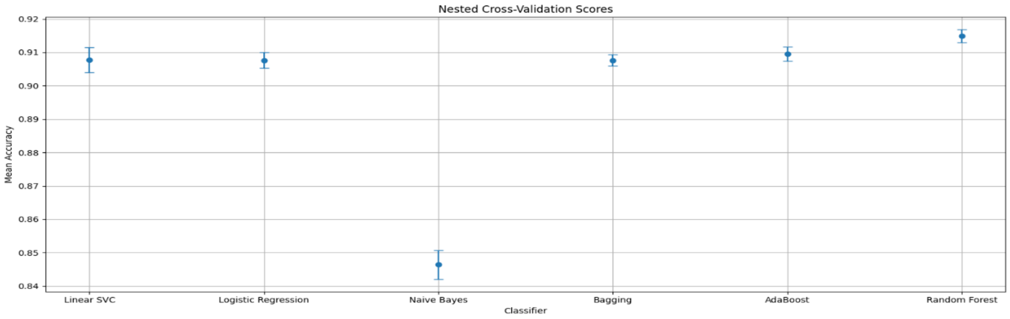
EVALUATION AND JUSTIFICATION



Modelling pipeline

1	Data Cleaning and Preparation	**Load, clean, and split data:** into test and training sets
		Split and clean text training data: expand contractions, lower case, remove punctuation, digits and extra whitespace, remove stop-words & lemmatize
		Prepare data for modelling: storing recommendation as 'y_train' and cleaned training test data as 'X_train' , subset for selected customer base
2	Model Selection	**Define model set for evaluation:** Linear SVC, Logistic Regression, Naive Bayes, Bagging, AdaBoost, Random Forest
		Conduct nested cross-validation: Inner loop, outer loop
3	Model Evaluation	**Train selected model and "best" parameter:** Random Forest
		Prepare training feature set: Cleaning training data to create "X_test".
		Evaluate model performance: Conducted on training and test set, calculate accuracy, precision, recall & F1 score. Plotting ROC curve and calculating ROC-AUC score
4	Model Interpretation	**Interpret coefficients**


Dataset
80%
20%
Train Data
Test Data
Training data/validation/test
Train model on Training Set
Evaluate model on Validated Set
Tweak model according to result on Validation Set
Pick model that does best on Validation Set
Confirm results on Test Set



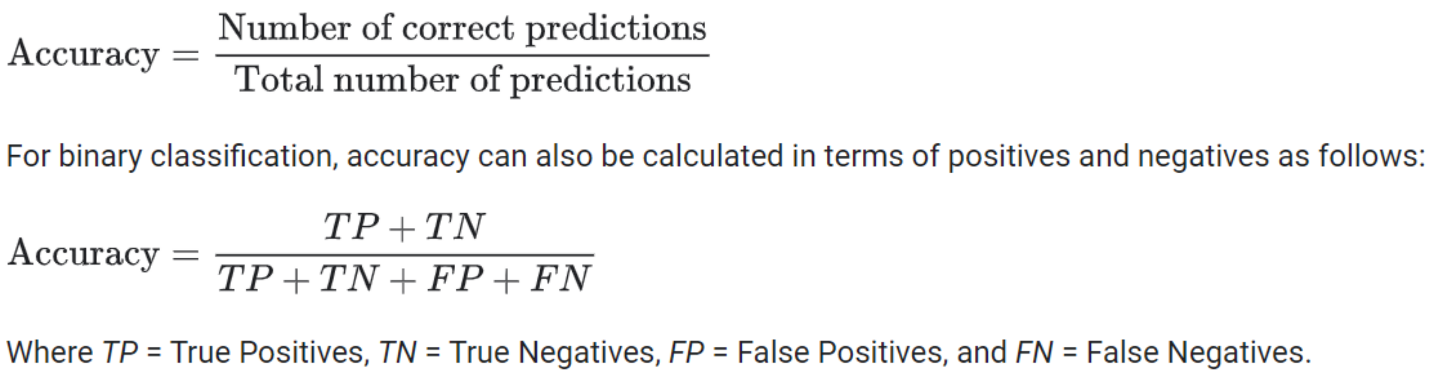
Linear SVC: Mean Outer CV Score: 0.9076 +/- 0.0037
Logistic Regression: Mean Outer CV Score: 0.9076 +/- 0.0024
Naive Bayes: Mean Outer CV Score: 0.8464 +/- 0.0043
Bagging: Mean Outer CV Score: 0.9076 +/- 0.0017
AdaBoost: Mean Outer CV Score: 0.9095 +/- 0.0021
Random Forest: Mean Outer CV Score: 0.9148 +/- 0.0020

Random Forest appears to be the most effective classifier for this dataset. Random Forests train multiple decision trees on random subsets of data and features to reduce overfitting and improve generalization.

MODEL EVALUATION

The performance of the chosen random forest model can be comprehensively evaluated using several metrics:

Classification Accuracy: Percentage of correct classifications. The model correctly classifies 91% of the customers as either subscribing or not subscribing to term-deposit. While this is a reasonably good overall accuracy, it's important to look at the other metrics to understand the model's performance in more depth.



$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

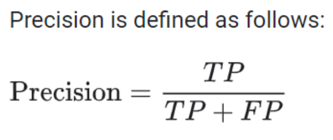
$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

Where *TP* = True Positives, *TN* = True Negatives, *FP* = False Positives, and *FN* = False Negatives.

Precision, Recall, and F1-score: These metrics provide a more comprehensive evaluation by considering true positives (correctly predicted high-chance), false positives, and false negatives. Precision measures the proportion of true positives among all instances predicted as positive, while recall measures the proportion of true positives correctly identified by the model. The F1-score is the harmonic mean of precision and recall, providing a balanced measure of both. The AUC-ROC is a threshold-independent metric that measures the model's ability to distinguish between the two classes (high and low risk).

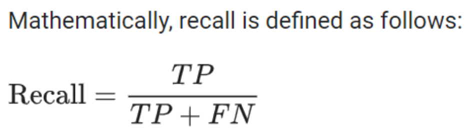
ROC AUC is a more robust and comprehensive metric than accuracy, precision, recall, or F1-score alone, as it evaluates the model's performance across different classification thresholds and considers the trade-off between true positive rate and false positive rate. With an F1-score of 0.5692 and an ROC AUC of 0.7377, the ROC AUC metric can be considered the better evaluation tool for the following reasons because though the F1-score is relatively low (0.5692), which indicates that the model may not be performing well in terms of both precision and recall, The ROC AUC of 0.7377 suggests that the model has a reasonable ability to distinguish between positive and negative instances, even though its precision and recall may not be optimal.

Therefore, the ROC AUC metric can be considered the better evaluation tool for assessing the model's performance, as it provides a more reliable and comprehensive measure of the model's ability to discriminate between positive and negative instances, even when the other metrics are not particularly high.



Precision is defined as follows:

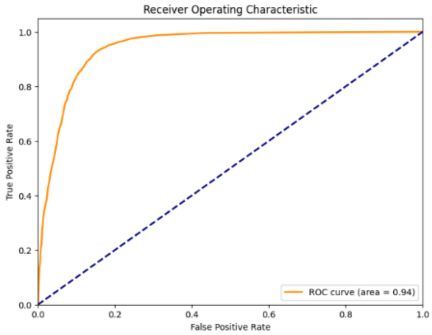
$$\text{Precision} = \frac{TP}{TP+FP}$$

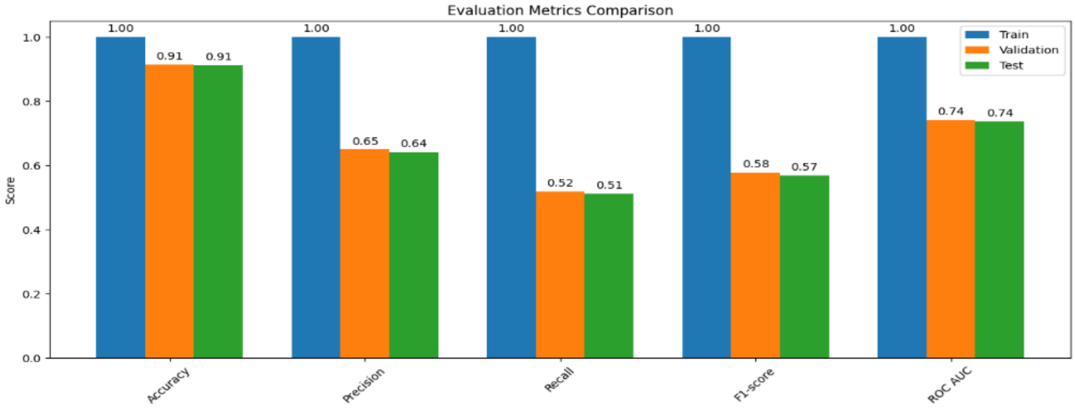


Mathematically, recall is defined as follows:

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$\text{F1 Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$





The high accuracy on the train set and the discrepancy with the validation and test set performance suggest potential overfitting, which could be mitigated through techniques like regularization or early stopping. The high variance observed in the model's performance on the training set (perfect scores of 1.0) compared to the validation and test sets (lower scores) is due to overfitting. By using random forests, the ensemble learning, bagging, and feature randomness techniques help to reduce this high variance and overfitting, resulting in more stable and reliable performance on the unseen validation and test data, as indicated by the reasonable ROC AUC scores.

Deploying a Model			
1	Model	Training	training dataset with Random Forest model
		Evaluation	Evaluate the performance of your model using a validation dataset
		Optimization	Tune the hyperparameters of the model to optimize its performance.
		Serialization	Once model is trained and optimized, then serialize it (i.e., convert into a format that can be stored and loaded). This is typically done using libraries like `pickle` in Python.
2	Deployment		Deploy your model to a server or a cloud-based platform. There are several options for this, including but not limited to:
		Local Server	You can deploy your model on a local server using a web service framework like Flask or Django in Python. The model would run on this server and provide predictions through API endpoints.
		Cloud Platforms	Deploy your model on cloud platforms like AWS, Google Cloud, or Azure. These platforms provide services to host the model, scale it, and connect it with other cloud services.
		Machine Learning Platforms	Machine learning platforms like Google's AI Platform. These platforms provide end-to-end solutions for deploying machine learning models.
3	Create an API		Create an API that will accept input data, run the data through the model, and return the model's predictions.
4	Testing		Once model is deployed and API is set up, thoroughly test it to ensure it works as expected.
5	Monitoring and updating		After deployment, continuously monitor the model's performance and update or retrain it as needed.

By deploying an accurate predictive model, the bank can optimize its marketing campaigns, target customer segments with a high propensity to subscribe, and potentially increase revenue. Further improvements to the model's performance could lead to even more significant benefits for the bank's marketing strategies.

LESS SUITABLE ALGORITHM

Naive Bayes classifier has the lowest mean score of 0.8464. While Naive Bayes is simple and computationally efficient, it makes strong independence assumptions between features, which might not hold true in many real-world datasets. Deep Neural network on the other hand, although extremely powerful will be overly complex for this structured tabular dataset with no unstructured data like text or images. Deep Neural network also require careful tuning of many hyperparameters and substantial training data to avoid overfitting.

It's important to note that these rankings are based solely on the mean CV scores and their standard deviations. Other factors such as the nature of the data, the interpretability of the model, and the computational efficiency might also influence which model is considered "best" or "least effective" in each situation. For instance, even though Naive Bayes is the least effective here in terms of CV score, it might be preferred in situations where computational efficiency is a priority, as it is relatively simple and fast to train. Similarly, Random Forest, despite having the highest CV score, is a more complex model which might not be as easily interpretable as some of the other models.

The most prefer and efficient model I would adopt is the ensemble tree methods which strike a good balance. Based on the mean outer cross-validation scores, the Random Forest classifier has the highest mean score of 0.9148, indicating the best average performance across different folds of the cross-validation.

CONCLUSION

Developing an accurate predictive model for the bank's direct marketing campaigns is important for optimizing customer targeting, improving product acquisition rates, and driving revenue growth. This project demonstrates the effectiveness of applying supervised machine learning techniques, specifically ensemble tree methods like Random Forest, to this binary classification problem.

By deploying the trained and optimized Random Forest model, the bank can leverage its predictive capabilities to identify customers with a high propensity to subscribe to term deposits, enabling more targeted and effective marketing campaigns. This data-driven approach not only enhances the efficiency of marketing efforts but also leads to cost savings by reducing unnecessary outreach to customers with a low likelihood of subscribing.

However, it is worthy of note that model performance can degrade over time due to changes in customer behaviour, market dynamics, or the introduction of new products and services. Consequently, continuous monitoring, evaluation, and periodic retraining of the model are important to ensure its ongoing effectiveness and relevance.

Furthermore, integrating additional data sources, such as customer feedback, market trends, and competitive intelligence, could further improve the model's predictive power and provide deeper insights into customer preferences and behaviours.

Overall, this project demonstrates the value of leveraging machine learning techniques and data-driven approaches in the banking industry, particularly for direct marketing campaigns. By embracing these technologies and continuously refining and updating the predictive models, the bank can gain a competitive edge, enhance customer satisfaction, and drive sustainable growth in an increasingly data-driven and customer-centric landscape.

Total_Word_Count: 2121

REFERENCE

Zatonatska Tetiana, Hubska, Maryna, and Shpyrko Viktor, (2022) Marketing Strategies in the Banking Services Sector with the Help of Data Science. *Digitales Archiv* [online]. 2227-6718 (2), pp. 121-127.

Sruthi, E.R. (2024) Understand Random Forest Algorithms with Examples. *Analytics Vidhya* [online].

Linwei Hu, Jie Chen, Joel Vaughan, Soroush Aramideh, , Hanyu Yang, , Kelly Wang, , Agus Sudjianto, and Vijayan N. Nair, (2021) Supervised Machine Learning Techniques: An Overview with Applications to Banking. *International Statistical Review* [online]. 89 (3), pp. 573-604.

Pedro Guerra, and Mauro Castelli, (2021) Machine Learning Applied to Banking Supervision a Literature Review. [online].

Roshan Kumari, and Saurabh Kr. Srivastava, () Machine Learning: A Review on Binary Classification. *International Journal of Computer Applications* [online]. 160 (7), pp. 11-15.

Kaitlin Kirasich, Trace Smith, and Bivin Sadler, (2018) Random Forest Vs Logistic Regression: Binary Classification For Heterogeneous Datasets. *Smu Scholar* [online]. 1 (3)

Xiaonan Zou, , Yong Hu, , Zhewen Tian, and Kaiyuan Shen, (2019) Logistic Regression Model Optimization and Case Analysis. *Ieee* [online].

Mohamed Alloghani, , Dhiya Al-jumeily, , Jamila Mustafina, , Abir Hussain, and Ahmed J. Aljaaf, () A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms For Data Science. *Unsupervised and Semi-supervised Learning* [online]., pp. 3-21.

Ciro Donalek, (2011) Supervised And Unsupervised Learning. [online].

Frank Harrell Jr, E. (2015) *Regression Modeling Strategies: Binary Logistic Regression*. 2nd ed. : Springer Series in Statistics.

Jaime Lynn Speiser, , Michael E. Miller, , Janet Tooze, and Edward Ip, (2019) A Comparison of Random Forest Variable Selection Methods For Classification Prediction Modeling. *Science Direct* [online]. 134 (15), pp. 93-101.

Leo Breiman, (2001) Random Forests. *Machine Learning* [online]. 45 (4), pp. 5-32.

Gerard Biau, (2012) Analysis of a Random Forests Model. *Journal of Machine Learning Research* [online].

Michael Nielsen, (2006) *Neural Networks and Deep Learning*. : .

Rene Choi, Y., Aaron Coyner, S., Jayashree Kalpathy-cramer, , Michael Chiang, F. and Peter Campbell, (2020) Introduction to Machine Learning, Neural Networks, and Deep Learning. *Translational Vision Science & Technology* [online]. 9 (14)

Cristinel Constantin, (2012) A Comparison Between Multivariate and Bivariate Analysis Used in Marketing Research. *Bulletin of The transilvania university of Braşov* [online]. 5 (54)