**BUSINESS REPORT**

**BACKGROUND:**

N/LAB Enterprises is extending its business into the banking industry and intends to market a financial product known as the "N/LAB Platinum Deposit" that provides people with a competitive fixed term interest rate in exchange for a sizeable deposit (which cannot be withdrawn for a year). It is required but time-consuming to make cold calls to prospective customers in order to sell this new product, so it is crucial for the business to determine which individuals are most likely to respond favorably to their marketing initiatives.

N/LAB Enterprises has access to data from prior marketing calls for a similar product to aid with this process. This data contains details on the characteristics and demographics of the people who were called, as well as the features of the calls themselves. The data also includes a label indicating whether or not the individual subscribed to the product.

The goal of this project is to analyze this data and develop a predictive model that can be used to identify which new individuals are good prospects for the telemarketers to contact. The model will be evaluated and compared to alternative models, and business recommendations will be generated based on the results of the analysis. By using this model, N/LAB enterprises hopes to increase the efficiency and effectiveness of its marketing efforts and increase the number of successful subscriptions to the "N/LAB Platinum Deposit" product.

## **1.BUSINESS REQUIREMENTS DOCUMENT:**

1. **Objectives:** The primary objective of this project is to develop a predictive model that can be used to identify which new individuals are good prospects for the telemarketers to contact, in order to increase the efficiency and effectiveness of N/LAB enterprises' marketing efforts.
2. **Scope:** The scope of this project includes analyzing data from previous marketing calls and using machine learning techniques to build a predictive model based on this data. The model will be evaluated and compared to alternative models, and business recommendations will be generated based on the results of the analysis.
3. **Deliverables:** The final deliverables for this project will include a predictive model, a report detailing the analysis and results of the model, and business recommendations for N/LAB enterprises based on the results of the analysis.
4. **Data:** Data on the characteristics and demographics of the people who were contacted, as well as details about the calls themselves, will be included in the records from prior marketing calls used for this project. The data will also have a label stating whether or not the person subscribed to the product.
5. **Stakeholders:** The primary stakeholders for this project include the CEO and management team of N/LAB enterprises, as well as the telemarketers who will be using the model to identify good prospects for marketing efforts. Other stakeholders may include the customers who are being targeted, as well as any regulatory bodies or oversight organizations.
6. **Success Criteria:** The success of this project will be measured by the performance of the predictive model in identifying good prospects for the telemarketers to contact. This may be evaluated using metrics such as precision, recall. The model will also be evaluated based on its ability to improve the efficiency and effectiveness of the telemarketers' marketing efforts, as demonstrated by an increase in the number of successful subscriptions to the "N/LAB Platinum Deposit" product. In addition, the business recommendations generated as a result of the analysis should be feasible and practical for N/LAB enterprises to implement.

## 

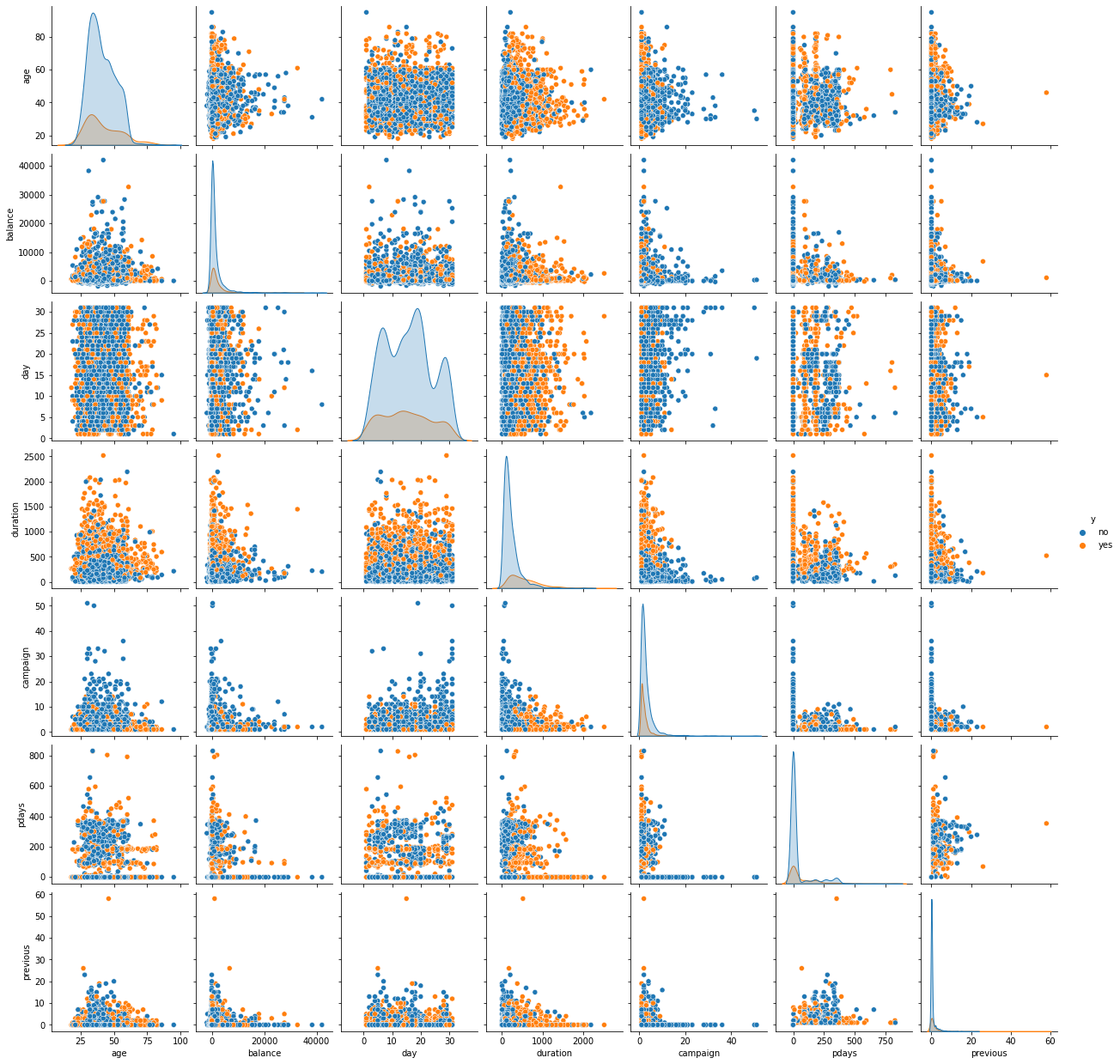
## **2.SECTION A: SUMMARISATION**

Here are some key highlights about the dataset in hand:

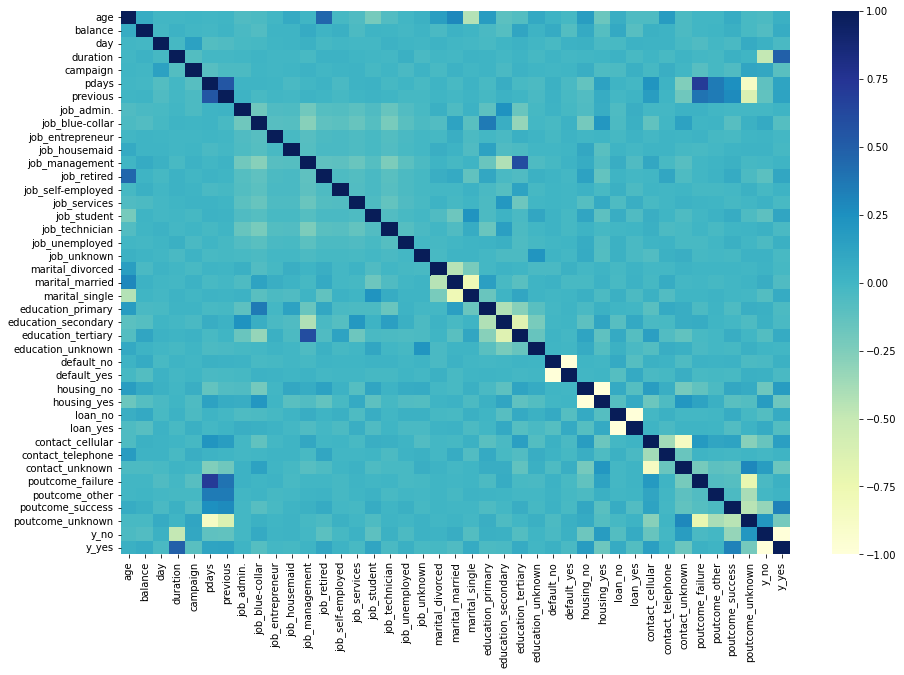
* ‘Default’ column is highly imbalanced (86.6%) as well as ‘poutcome’ (50.3%).
* The company has mainly called people with management (894), blue\_collar(824) as well as technician(645) jobs as shown by the categorical plot.
* The ‘balance’ column has 314 (7.8%) zeros and ‘previous’ column has 3210 (80.2%) zeros.
* Columns ‘balance’ and ‘pdays’ are highly skewed.
* No null values were found.
* The data is not linear.

With emphasis to the column ‘y’, the data shows the following:

* Upon chi square analysis we found out that ‘job’ (Chi-square statistic: 133.451, p-value: 0.000) and ‘poutcome’ (Chi-square statistic: 443.821, p-value: 0.000) columns have a significant relationship with the column ‘y’. ‘poutcome’ represents the results of trying to sell another campaign to the same person thus it makes sense why it is has a significant relationship with ‘y’.
* Upon examining the pairplot with hue set as ‘y’, we find that the dataset is excessively overlapping. We can notice that some variables contain more information than other variables as they are less overlapped.



* Upon examining the heatmap we can find the correlation between the yes category of the ‘y’ column and other columns categories. Retired people and students are the most likely to say yes followed by ‘poutcome\_success’. If the company was able to sell something on a previous campaign successfully, they are more likely to purchase this one too.

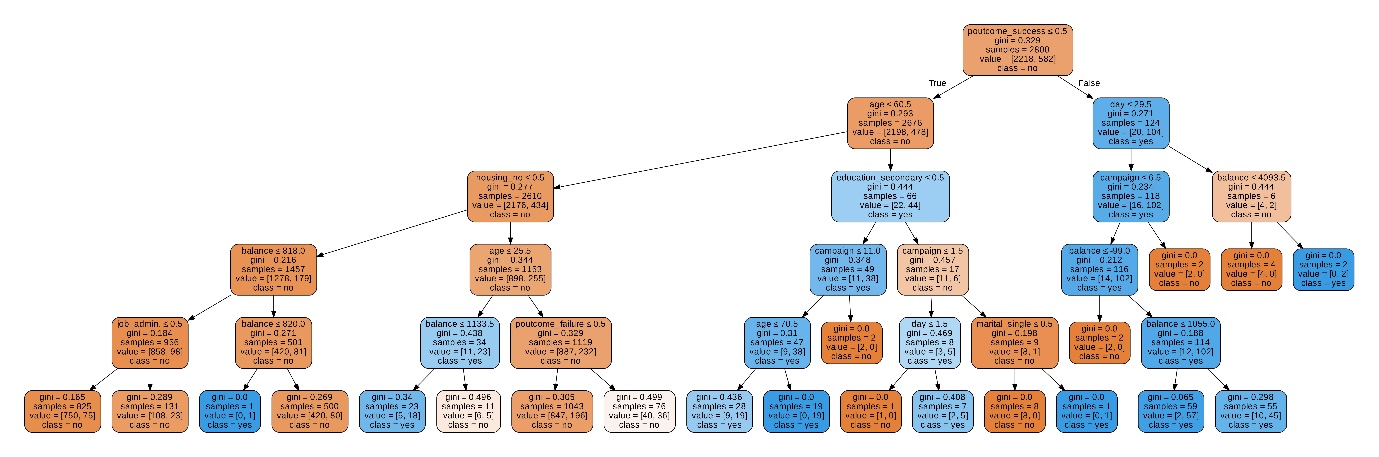


## **3.SECTION B: EXPLORATION**

On application of decision tree, we can find out that the following variables appear to be the most important:

|  |  |
| --- | --- |
| Feature | Importance |
| balance | 0.20527662410881897 |
| age | 0.18053538576765624 |
| day | 0.12506618152792318 |
| Poutcome\_success | 0.11200123636180222 |
| campaign | 0.05346788388723598 |
| Education\_tertiary | 0.025269591736357677 |
| previous | 0.022131817745396378 |
| Poutcome\_failure | 0.02077764870164234 |
| Contact\_cellular | 0.02010771081515456 |
| Job\_technician | 0.019601834636358145 |

One thing to note here is that the data has been one hot encoded before this analysis. The data has been pruned and the maximum depth has been set to 5. ‘balance’ has a lot of zeros which justifies the first position in the important features list. Having a lot of zeros makes classification easy for the model.

Here is a visual representation of the decision tree generated by the model: 

In the previous analysis we figured out that ‘poutcome\_success’0 had the highest correlation ‘y’ and based on the decision tree analysis we notice that ‘poutcome\_success’(gini = 0.329) followed by ‘age’ (0.293) have high gini index making it an efficient way to divide the data. Thus, we can combine these variables to make a good prediction and identify useful subpopulations.

If the previous outcome was a failure and the age of the person is <= 60.5, it is likely going to predict ‘y’ as no.

We can also observe that ‘campaign’ and ‘education’ has a good significance in further dividing the data.

A combination of ‘poutcome\_success’ , ‘day’ and ‘campaign’ is also a good division method but we must take into consideration that there are less samples for the ‘day’ and ‘campaign’ feature. Nevertheless, we can infer that if the previous outcome was a success and day of the month the individual was last contacted is ≤ 29.5 along with number of contacts performed during this campaign and for this client is ≤ 6.5, the likely prediction of ‘y’ is yes.

In this decision tree, we found a select number of variables useful for our prediction. ‘Duration’ column had to be removed as it doesn’t add much value in theoretically but was affecting the prediction too much.

## **4.SECTION C: MODEL EVALUATION**

**Models:**

* Four models were selected to test the data: **logistic regression, random forest, Naive Bayes and KNN**. These models were chosen because they are widely used in classification tasks and can handle large datasets.
* For the logistic regression model, max\_iter parameter was used which controls the maximum number of iterations that an algorithm will run for. It is used to control the runtime of the algorithm and ensure that it does not run indefinitely.
* For the random forest model, default parameters were used.
* For the Naive Bayes model, the default parameters were used. These include the assumption that all features are independent, which simplifies the model and allows it to make predictions based on fewer data points.
* KNN model was chosen to test the data because it is a simple and effective method for classification tasks. The parameterization chosen for the KNN model was the number of nearest neighbors to consider, which was varied from 1 to 40. The reason for choosing this range of values is to determine the optimal number of nearest neighbors for the KNN model, as a larger number of nearest neighbors may lead to a smoother decision boundary and a lower variance, but also a higher bias. By testing a range of values, we can identify the optimal number of nearest neighbors that strikes a balance between bias and variance.

**Metrics:**

* **Precision:** Precision measures the accuracy of the positive predictions made by the model. A model with high precision has a low false positive rate, which means that it is less likely to predict that an example belongs to the positive class when it actually belongs to the negative class.
* **Recall:** Recall measures the completeness of the positive predictions made by the model. A model with high recall has a low false negative rate, which means that it is more likely to correctly predict that an example belongs to the positive class.
* **F1 Score:** The F1 score is a metric that combines precision and recall. It is defined as the harmonic mean of precision and recall, with a higher value indicating a better balance between precision and recall. The F1 score is calculated as follows:

F1 = 2 \* (precision \* recall) / (precision + recall)

* **Accuracy:** The accuracy of a model is a measure of how frequently the model predicts correctly. It is calculated by dividing the number of correct predictions by the total number of predictions made by the model. It's also worth noting that the randomness of the training and evaluation processes can have an impact on model accuracy. The number in parentheses after the accuracy represents the model's standard deviation over multiple runs. This can give you an idea of how stable the model's accuracy is. A higher standard deviation may indicate that the model is more sensitive to the specific training and evaluation data used, and that it will not generalise well to new data. Kfold cross validation is used to get an accurate number.

We chose these metrics because when dealing with **imbalanced classes**, these metrics can be more informative than accuracy as they take into account the number of true positives, false positives, and false negatives. Also, depending on the problem and context, one type of error might be prioritized over the other, hence precision can be prioritized. Additionally, these metrics are useful for comparing multiple models and getting a balanced view of the performance of a model.

**Results and Interpretation:**

1. **Random forest:**

In summary, the random forest model has a precision of 0.55, which means that it is correct about 55% of the time when it predicts that an example belongs to the positive class. The model has a recall of 0.34, which means that it is able to identify 34% of the positive examples in the data. The F1 score of the model is 0.42, which indicates that it has a balanced performance in terms of precision and recall. The accuracy of the model is 81.65%, with a standard deviation of 0.83%.

1. **Logistic Regression:**

In summary, the logistic regression model has a precision of 0.83, which means that it is correct about 83% of the time when it predicts that an example belongs to the positive class. The model has a recall of 0.21, which means that it is able to identify 21% of the positive examples in the data. The F1 score of the model is 0.34, which indicates that it has a lower performance in terms of recall compared to precision. The accuracy of the model is 81.92%, with a standard deviation of 0.71%.

1. **Naive Bayes Classifier:**

In summary, the Naive Bayes Classifier model has a precision of 0.42, which means that it is correct about 42% of the time when it predicts that an example belongs to the positive class. The model has a recall of 0.62, which means that it is able to identify 62% of the positive examples in the data. The F1 score of the model is 0.50, which indicates that it has a balanced performance in terms of precision and recall. The accuracy of the model is 76.22%, with a standard deviation of 1.78%.

1. **KNN:**

In summary, the KNN model has a precision of 0.65, which means that it is correct about 65% of the time when it predicts that an example belongs to the positive class. The model has a recall of 0.26, which means that it is able to identify 26% of the positive examples in the data. The F1 score of the model is 0.37, which indicates that it has a lower performance in terms of precision compared to recall. The accuracy of the model is 79.15%, with a standard deviation of 0.80%.

## **5.SECTION D: FINAL ASSESSMENT**

It appears that the logistic regression model has the highest precision of 0.83, which means that it is correct about 83% of the time when it predicts that an example belongs to the positive class.

However, it has the lowest recall of 0.21, which means that it is able to identify only 21% of the positive examples in the data. This suggests that the logistic regression model is good at identifying true negatives, but not good at identifying true positives. But I want to rule this out as the data is not linear.

On the other hand, the Naive Bayes classifier model has the highest recall of 0.62, which means that it is able to identify 62% of the positive examples in the data. However, its precision is only 0.42, which means that it is correct about 42% of the time when it predicts that an example belongs to the positive class. This suggests that the Naive Bayes classifier model is good at identifying true positives, but not good at identifying true negatives.

Random Forest has a balanced performance in terms of precision and recall, with precision of 0.55 and recall of 0.34, while KNN model has a lower performance in terms of recall compared to precision with precision of 0.65 and recall of 0.26.

Keeping in mind the goal of the business and the performance metrics, **Random Forest is my WINNER classifier.**

## **6.SECTION E: MODEL IMPLEMENTATION**

To use the model code/files to process a new test set, the recipient should:

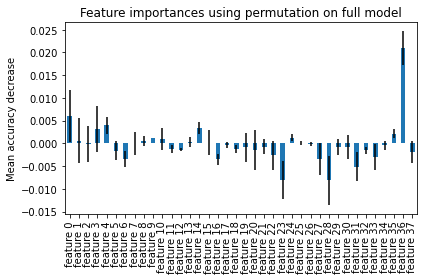
1. Load the dataset using pandas' read\_csv function.
2. Use the get\_dummies function to one-hot encode any categorical features in the dataset.
3. Split the dataset into features (X) and target (y) variables.
4. Standardize the features using the StandardScaler function.
5. Split the standardized features and target variables into training and testing sets using the train\_test\_split function.
6. Fit the model to the training data and make predictions on the testing data using the predict function.

To make new predictions from the model, the recipient can use the model's predict function on a new set of features. It is important to ensure that the new set of features is standardized in the same way as the training set was.

## 

## **7.SECTION F: BUSINESS CASE RECOMMENDATIONS**

Based on the feature importances plot generated of the random forest classifier model, the company has to focus of people who they could previously successfully sell a product (poutcome\_success), the age group that they belonged to (age) and students (job\_students). It is shown below:



Since the goal of the business is to reduce the costs by cutting down the calls done to non-prospective customers and making sure that they are making calls that translate into a sale of N/LAB Platinum Deposit, it is recommended to focus the calls on ages people and young people. People in their middle ages are reluctant to a similar product and thus can be avoided.

If someone had bought a product in a different campaign, they are more likely to buy this too and hence they should be narrowed down as the most potential buyers. The company should not consider housemaids(job\_housemaid) and the ones in ‘job\_services’ as they aren’t likely to buy the product.

## **8.LIMITATIONS AND FURTHER STEPS**

This project ignores the imbalances in the data owing to future collection of more data. Hyperparameter tuning can be done to increase the efficiency of the model.

However, some examples of algorithms that can be effective on imbalanced data include:

**Oversampling:** This approach involves generating additional synthetic examples of the minority class to balance the dataset. One popular oversampling algorithm is SMOTE (Synthetic Minority Oversampling Technique).

**Undersampling:** This approach involves removing examples from the majority class to balance the dataset. One popular undersampling algorithm is Tomek links.

**Ensemble methods:** Some ensemble methods, such as bagging and boosting, can be effective at handling imbalanced data because they train multiple classifiers and combine their predictions.

## **9.REFERENCES**

<https://klib.readthedocs.io/en/latest/>

<https://github.com/krishnaik06/K-Nearest-Neighour/blob/master/K%20Nearest%20Neighbors%20with%20Python.ipynb>

<https://scikit-learn.org/stable/index.html>