Predictive Classification Model for targeting potential clients for “N/LAB Platinum Deposit” telemarketers to contact

Executive summary

As N/LAB expands to banking sector, “N/LAB Platinum Deposit” requires a predictive classification model to support identifying potential customers for telemarketers to contact with more effectively. The data, coming from an analogous product, consists of demographic, call related features and result (Appendix A). Guiding by CRISP data mining process (Appendix B), this report starts with data understanding and assessing the quality of data, presenting a general idea and investigating correlations among all features. In section B, data exploration by utilising decision tree model identifies the feature importance. Feature selection and engineering are discussed for building up models in section C. Decision tree, random forest and naïve based models are carried out with K-fold cross validation as the evaluation strategy. As precision is chosen as the performance measurement, random forest is selected as the final model. Its justification and implementation are illustrated in section D and E. In section F, business insights are discussed for further business improvements.

Section A: Summarisation

Original data set contains 4000 thousand records with 15 input features and 1 output feature. There are 7 numeric features and 9 categorical features. Detailed descriptions for each variable are in Appendix A. As a result of testing data quality, no missing data and duplicated records exist. Figure 1 shows distributions of numerical inputs and there are some outliers (out of 3 standard deviation) existing. However, percentages are 0.875%,2.225%, 2.2%,1.95%,2.875%,1.975% outliers for ‘age’, ‘balance’,’duration’,’campaign’, ‘pdays’, ‘previous’ respectively which is acceptable. Although no missing data exist, 0.625%, 3.775%, 26.9%, 80.425% unknown data shows in ‘job’, ‘education’,’contact’,’poutcome’ respectively. 80.425% are unknown. However, aligning with 3215 ‘-1’ records in ‘pdays’, ’unknown’ in ‘poutcome’ suggests these customers are new which can be an important feature for modelling and also represent the actual business context. To investigate the correlation between input features and output features and within input features, categorical feature encoded to numerical type through python replace() method. Only ‘poutcome’ is projected to 3 dimensions which are ‘poutcome_0’,’poutcome_1’,’poutcome_2’ which indicating failure of success, other or not, unknown or not respectively for better separation new clients with repeated clients.

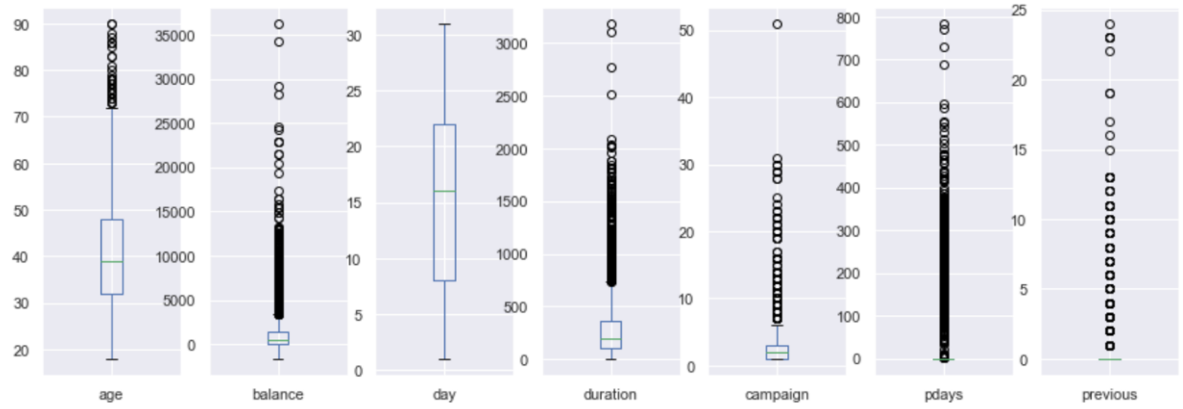


Figure 1 Box plot for numerical input variables

Figure 2 displays the histogram for all variables, numerical values are binned for displaying with bins equal to 10. ‘poutcome’, ‘pdays’ and ‘previous’ show concentration distribution since there are 80.38% (using ‘-1’ in ‘pdays as index) new clients. ‘Default’ distributes mainly on ‘no’ (1.75%). Target variable y contains 21.23% of ‘yes’ and 78.78% for ‘no’. Figure 3 represents correlation coefficients among all variables. Dark color indicates a high correlation which means a stronger linear relationship between two variables. The last row is the correlation coefficient between all variables and 'y'. 'duration' shows a moderate uphill relationship with 'y' followed by ''poutcome_1' and 'housing'. The high correlations among poutcome variables, 'pdays' and 'previous' are mainly caused by new clients. If a client is not exist before, data records '-1'for 'pdays', '0' for 'previous', and '0','0','1' for'poutcome_0','poutcome_1','poutcome_2'.

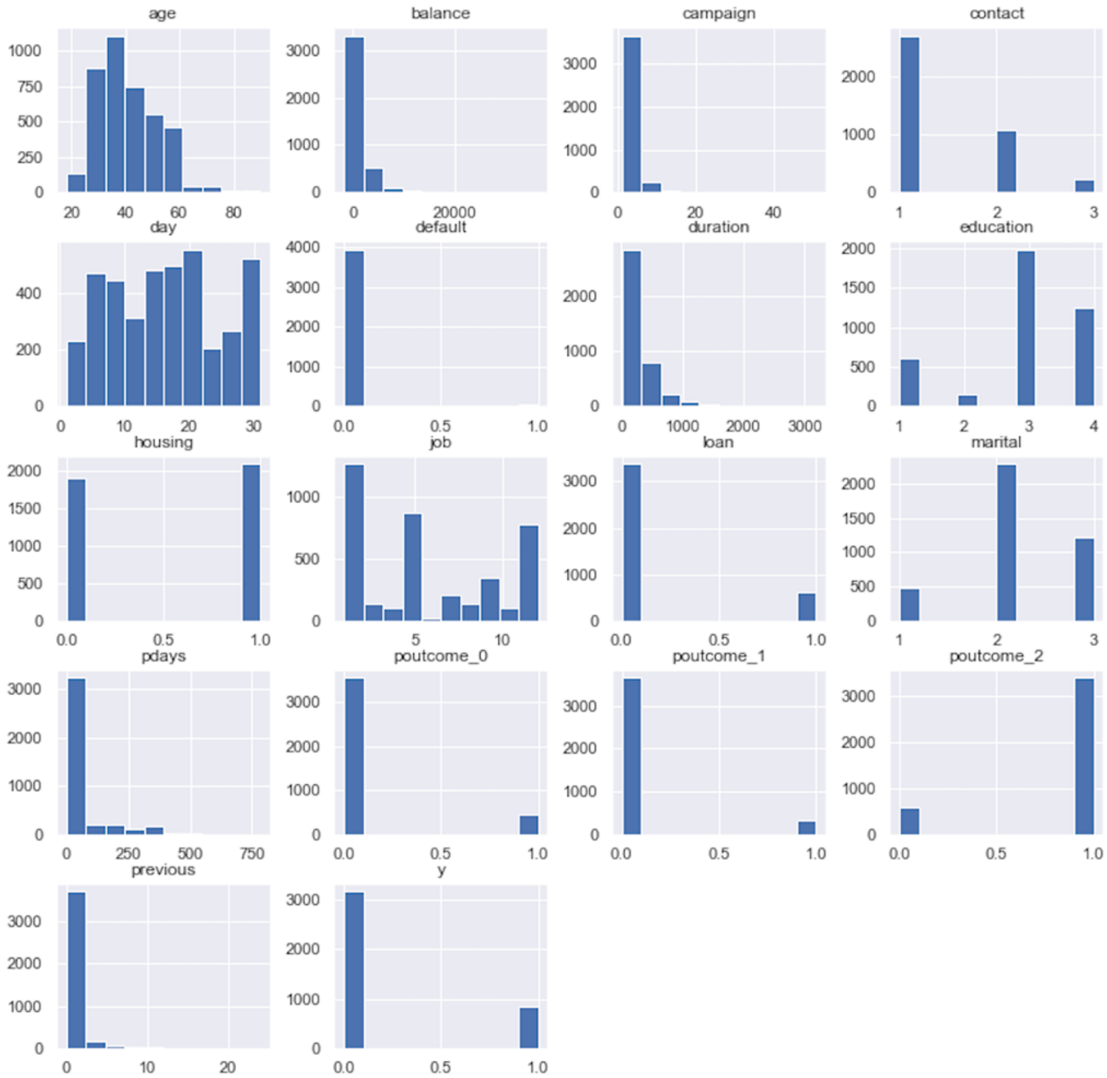
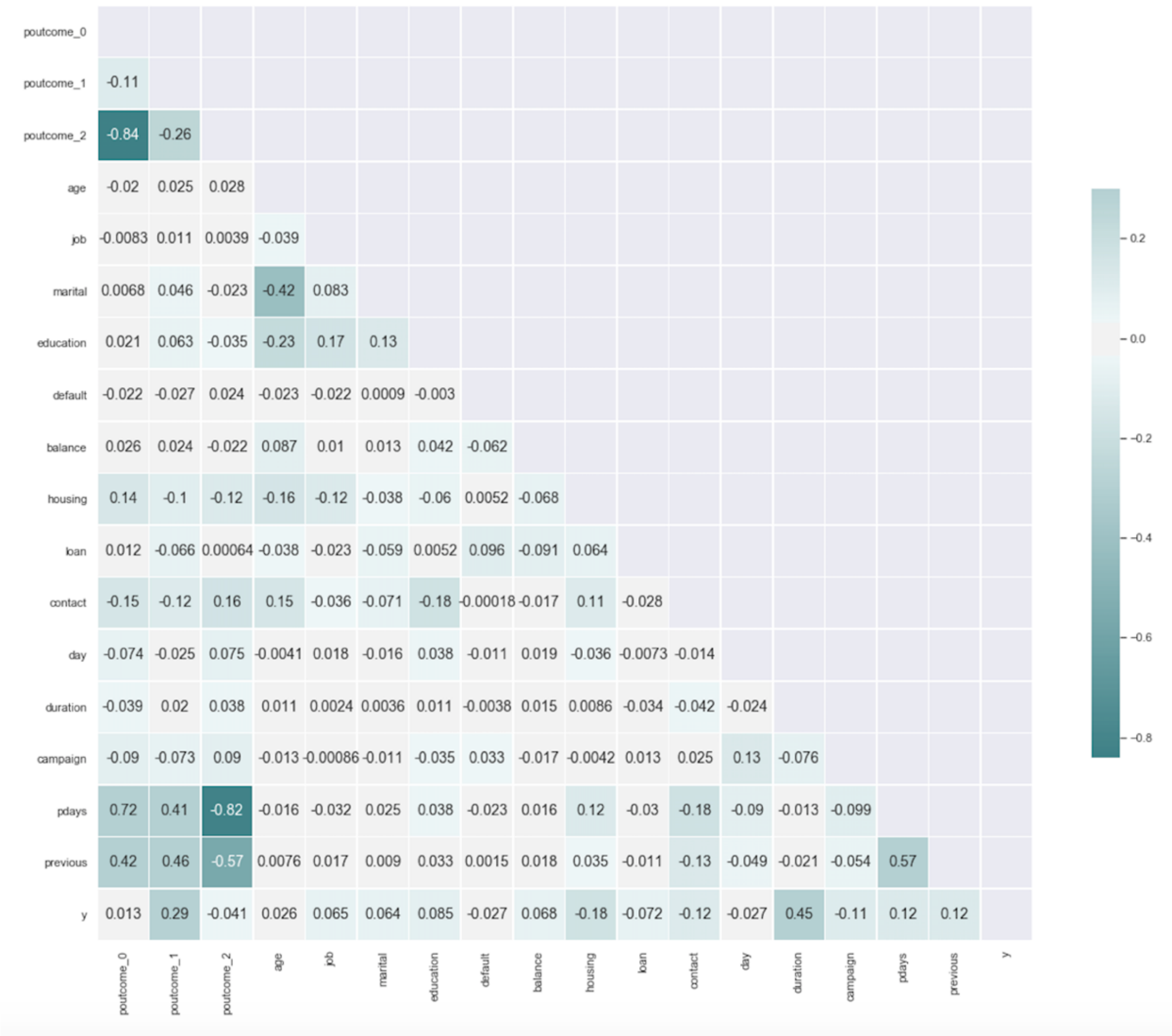


Figure 2 Histogram for all variables

To highlight the key points, the data is a qualified with acceptable outliers and the high volume unknown data in 'poutcome' result from 80.38% new clients which is an important information representing reality. Projecting it into 3 dimensions assists the analysis on new and repeated clients. Due to this reason, correlations among poutcome_0,1,2, 'pdays' and 'previous' are high. Hence, the data inside each of them still can be informative. Default value for y is 21.23% of 'yes' and 78.78% of 'no' representing an unbalance data structure with 'duration' holds the highest correlation. 'Default' is highly concentrated on 'yes' hence may not be influential in the analysis.

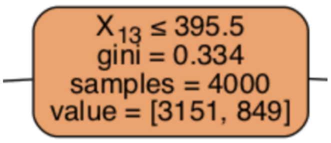


	poutcome_0	poutcome_1	poutcome_2	age	job	marital	education	default	balance	housing	loan	contact	day	duration	campaign	pdays	previous	y
poutcome_0																		
poutcome_1	-0.11																	
poutcome_2	-0.84	-0.26																
age	-0.02	0.025	0.028															
job	-0.0083	0.011	0.0039	-0.039														
marital	0.0068	0.046	-0.023	-0.42	0.083													
education	0.021	0.063	-0.035	-0.23	0.17	0.13												
default	-0.022	-0.027	0.024	-0.023	-0.022	0.0009	-0.003											
balance	0.026	0.024	-0.022	0.087	0.01	0.013	0.042	-0.062										
housing	0.14	-0.1	-0.12	-0.16	-0.12	-0.038	-0.06	0.0052	-0.068									
loan	0.012	-0.066	0.00064	-0.038	-0.023	-0.059	0.0052	0.096	-0.091	0.064								
contact	-0.15	-0.12	0.16	0.15	-0.036	-0.071	-0.18	-0.00018	-0.017	0.11	-0.028							
day	-0.074	-0.025	0.075	-0.0041	0.018	-0.016	0.038	-0.011	0.019	-0.036	-0.0073	-0.014						
duration	-0.039	0.02	0.038	0.011	0.0024	0.0036	0.011	-0.0038	0.015	0.0086	-0.034	-0.042	-0.024					
campaign	-0.09	-0.073	0.09	-0.013	-0.00086	-0.011	-0.035	0.033	-0.017	-0.0042	0.013	0.025	0.13	-0.076				
pdays	0.72	0.41	-0.82	-0.016	-0.032	0.025	0.038	-0.023	0.016	0.12	-0.03	-0.18	-0.09	-0.013	-0.099			
previous	0.42	0.46	-0.57	0.0076	0.017	0.009	0.033	0.0015	0.018	0.035	-0.011	-0.13	-0.049	-0.021	-0.054	0.57		
y	0.013	0.29	-0.041	0.026	0.065	0.064	0.085	-0.027	0.068	-0.18	-0.072	-0.12	-0.027	0.45	-0.11	0.12	0.12	

Figure 3 Correlations among all variables

Section B: Exploration

The decision tree is implemented through python sklearn, setting parameters as defaulted value. Since the aim of the decision tree to examine feature importance, the decision tree is expected to explore as much as possible ignoring over-fitting. Considering the large size of decision tree, the whole image is visualized in 'evaluation code.ipynb' section B. Figure 4 shows the first node in decision tree representing the most important feature to X13:'duration' with a gini coefficient 0.334.



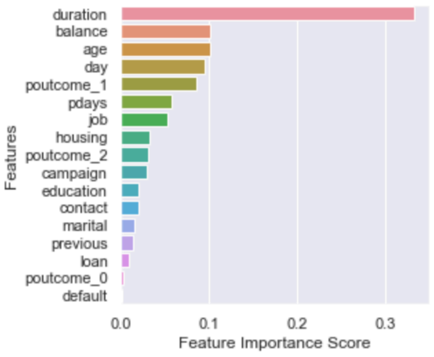


Figure 4 First node in decision tree (1)

Figure 5 Feature importance Score for decision tree (1)

Figure 5 displays feature importance by ranking Gini importance (mean decrease in impurity) of each input variables. ‘duration’, ’balance’, ’age’ ‘day’ are the top four important feature while ‘loan’, ‘poutcome_0’, ‘default’ as the least important features. Since the high rank in importance of ‘balance’, ‘age’ and ‘day’, figure 6 plots the different distribution between ‘yes’ and ‘no’ in ‘y’ of ‘age’, ‘balance’ and ‘day’. Except the size difference, their distributions between ‘yes’ and ‘no’ are similar. The high rank of ‘age’ and ‘balance’ may result from their right skewed distribution. As a result, ‘age’,’balance’ and ‘days’ are cut with group interval 10, 5000, and 5. ‘duration’ is not cut since it will not be included in the predictive model.

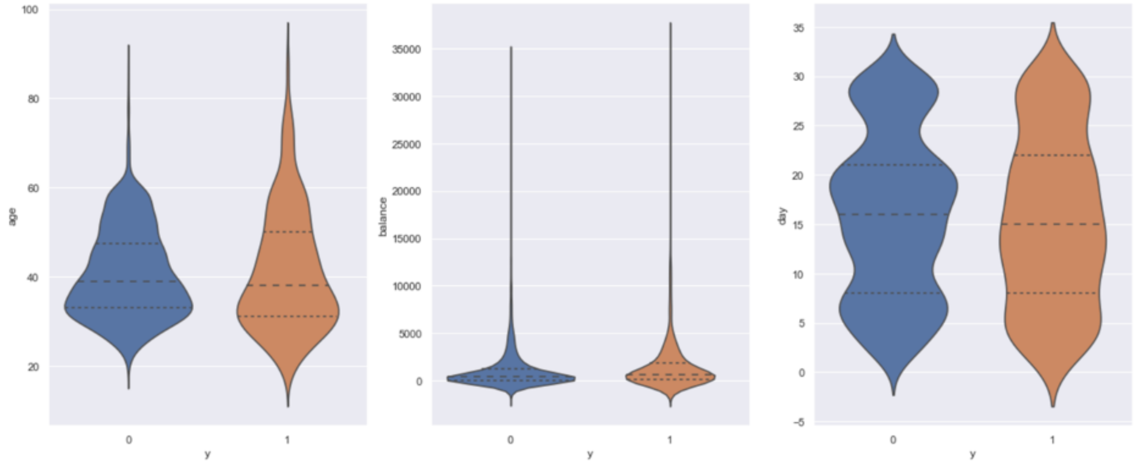
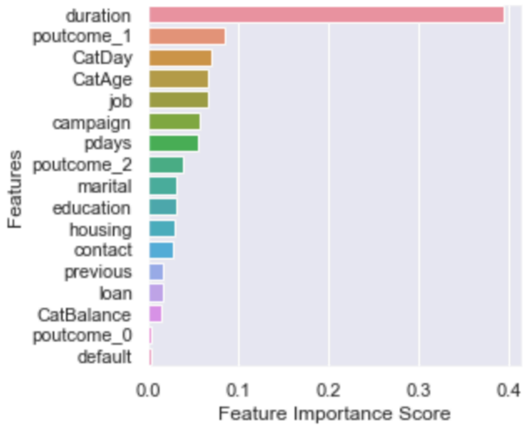


Figure 6 Distribution between ‘yes’ and ‘no’ in ‘y’ of ‘age’, ‘balance’, ‘day’



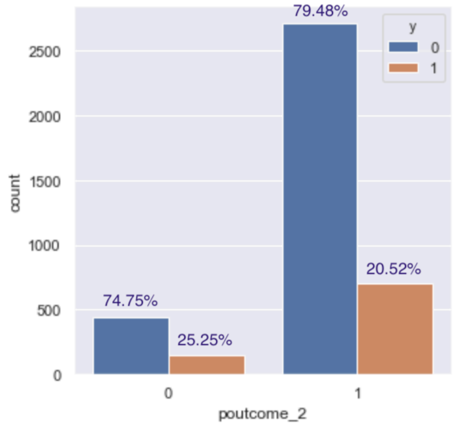


Figure 7 Feature importance score for decision tree (2)   Figure 8 Count plot ‘poutcome_2’ with ‘y’

After grouping three numerical input, another decision tree model is built and new feature importance is illustrated in figure 7. ‘duration’,’poutcome_1’,’CatBalance’ becomes the top three important features while ‘loan’,’poutcome_0’ and ‘default’ are the least. Since ‘duration’ can be known only after the call, it is dropped for the predictive model. Based on the insights from section A and B, ‘default’ is deleted as the weak correlation between ‘default’ and ‘y’ and its zero Gini importance. Although high correlations exist among poutcome_0,1,2, ‘pdays’ and ‘previous’ from new clients, those variables can be informative for existing customers. ‘poutcome_2’ indicates the customer is new or not and ‘poutcome_0’ is an indication of the result of the previous campaign for existing clients. Figure 8 shows that previously contacted clients have a 4.73% more purchasing rate than new clients. ‘poutcome_1’ displays 0.29 correlation with ‘y’ and rank high in feature importance. However, ‘poutcome_1’ actually represent not knowing the result of a previous campaign for existing customers. To eliminate the influence, ‘poutcome_1’ is deleted. As a result of feature engineering and feature selection, ’poutcome_0’ and ‘poutcome_2’ are generated while ‘duration’, ‘default’,’poutcome_1’ are dropped. ‘age’, ‘balance’ and ‘day’ are cut into groups.

Section C: Model Evaluation

After data understanding and preparation, there are 4,000 records with 14 input variables and 1 output variable. Decision tree (DT), random forest (RF) and Naïve based (NB) models are implemented for classification. Since models are derived for identifying potential clients of "N/LAB Platinum Deposit" for telemarketers to contact with, as mentioned by CEO the biggest cost come from ineffective calls which means to avoid false alarms and gain more true alarms. In addition to accuracy, precision (true positive/ false positive + true positive) is more stressed for evaluating model performance. Area under the receiver operating characteristic (AUC) is also examined as it shows how well the model classify classes. K-folds cross validation is selected as the evaluation strategy to have a comprehensive learning and testing. Data set is split into 10 folds and shuffle before splitting. To benchmark, a point model is deployed with precision 0.2105.

- Decision Tree Model (DT)

Decision tree is selected as it is easy to implement and its ability to deal with non-linear relationship. Especially in a business context, the visualisation of decision tree can be useful in interpreting and generating business insights. Important parameters for decision tree model are max_depth, min_sample_splits, min_sample_leafs. Max_depth decides the depth of tree and deeper tree may result overfitting. Min_samples_split means the minimum number samples to split and higher number constrain tree to consider more samples at each node. Min_samples_splits define the minimum samples for a leaf node (external node). Max_depth is tested from 1 to 32. min_samples_splits tests 40 to 400 with interval 40 and min_sample_leafs test range from 40 to 160 with gap 40. All tests are in python3 using 'for' loops. Figure 9 represents the accuracy and precision change. To optimise the DT performance, max_depth sets to 3, min_sample_splits and min_sample_leafs set to 80.

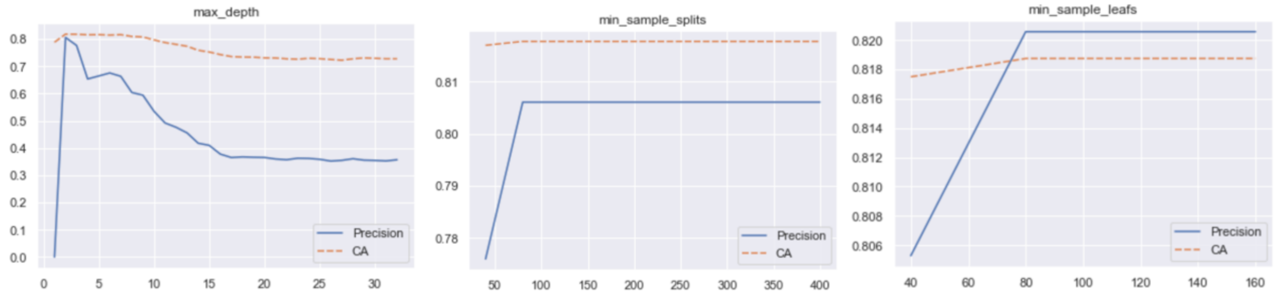


Figure 9 Parameter tuning performance for decision tree

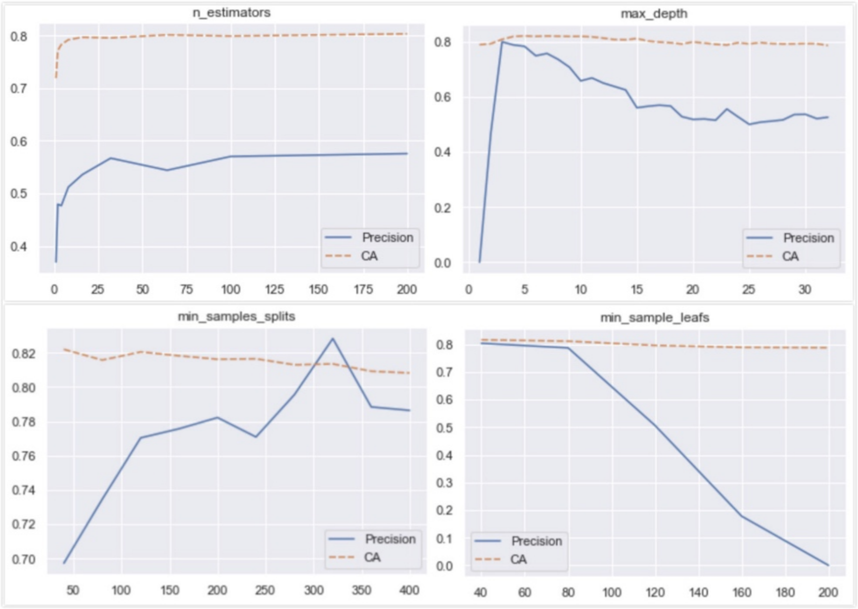


Figure 10 Parameter tuning performance for random forest

- Random Forest Model (RF)

As random forest builds multiple decision trees and voting to make predictions, it considers more accurate and robust. As a result of considering the average of predictions, it can also deal with overfitting problem. Similar as DT, RF can also handle with non-linear model. Although RF can not be visualised to interpret, the feature importance can be utilised to generate business insights. Parameter tuning is similar to decision tree except 'N_estimators' is added. N_estimator means how many decision tree in the random forest. Higher figure increases the possibility of accuracy but requires more time to build model. It is tested with 1, 2, 4, 8, 16, 32, 64, 100, 200 to find a proper quantity. max_depth, min_sample_splits and min_sample_leafs are tested with the same test parameters as DT. Figure 10 displays the parameter tuning performances.

- Naïve based Model (NB)

NB is selected as the fast computation with low cost. In addition, it works well with discrete variables. Although there are some correlations exist among variables, they are weak to break the assumption of independence holds.

Figure 11 displays the classification accuracy and precision rate for point model, decision tree, random forest and naïve based model. Figure 12 plots the confusion matrices for visualisation and better understanding. All three models perform well on classification accuracy and AUC but since the main measurement is precision where Naïve base performs poorly while DT and RF both obtain high rates. RF perform slightly higher rate while DT alarm in a larger size.

	Accuracy	Precision	AUC
Decision Tree (DT)	0.8188	0.8206	0.6822
Random Forest (RF)	0.8118	0.8579	0.7238
Naïve Based (NB)	0.744	0.3943	0.6822

Figure 11 Model performance comparison

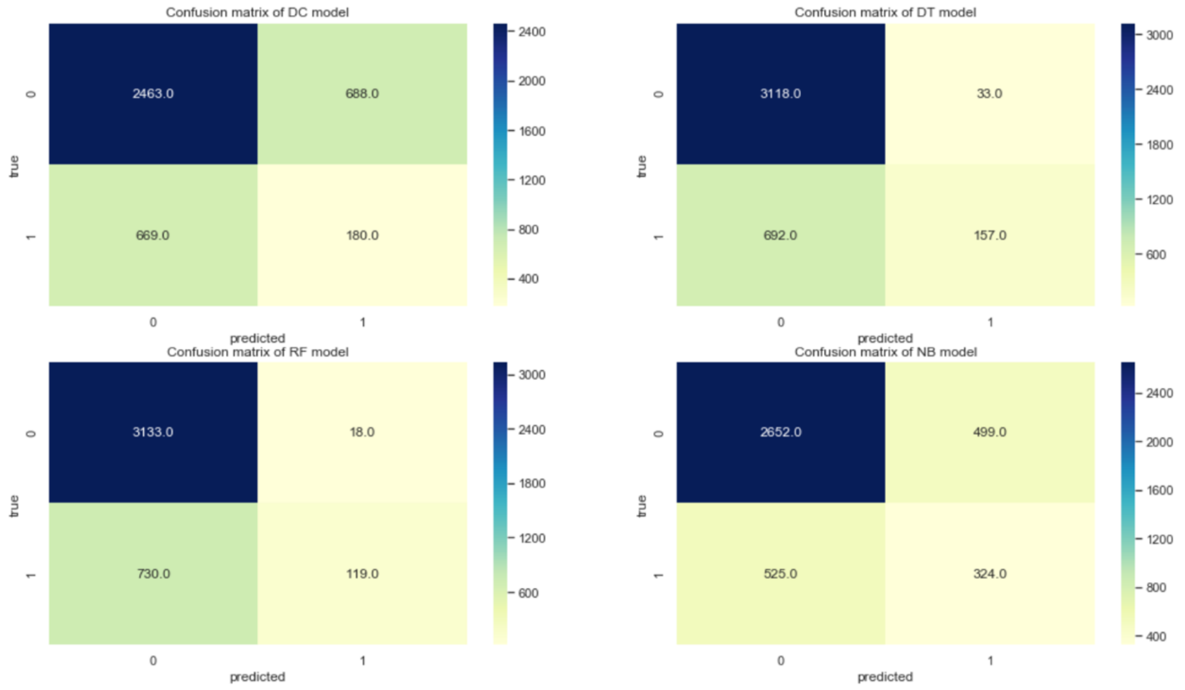


Figure 12 Confusion matrices for point, DT, RF, NB models

Section D: Final Assessment

For final model selecting, DT is visualised in figure 13. Only considers ‘poutcome_2’, ‘pdays’, ‘housing’, ‘CatAge’, ‘contact’ and ‘marital are used to split samples which results under-fitting situation. Hence, as a robust model, random forest is selected as the final winner with the highest accuracy, precision and AUC. Since it contains numerous decision trees and cannot be visualised, figure 14 displays relative feature importance. To further justify the model performance, recursive feature engineering used combine with feature importance. ‘CatBalance’ is deleted for testing leading no improvement on module performance. Hence 15 features are retained for final model build up.

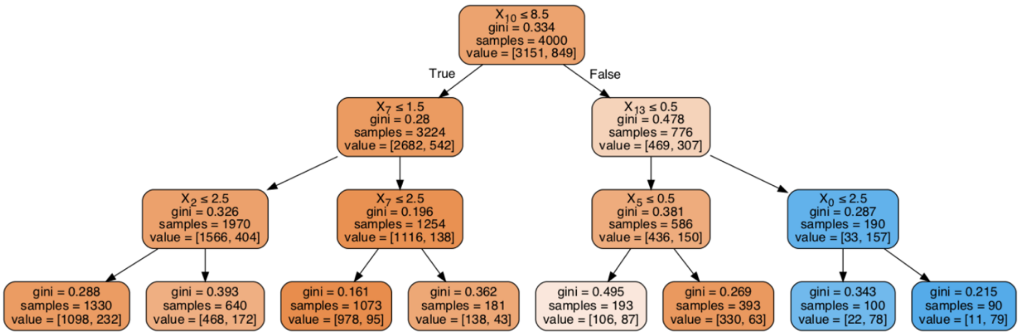


Figure 13 Visualisation of predictive decision tree model

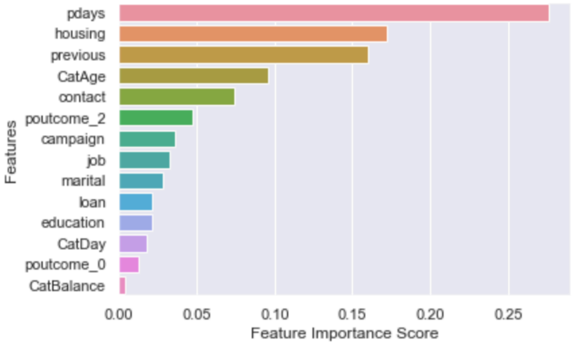


Figure 14 Feature importance score for predictive decision tree

Section E: Model Implementation

RF model is trained to whole training dataset as section E in the ‘Evaluation code’ documents. Previous evaluation code is provided in the document ‘Evaluation.ipynb’. Before running the code, to encode ‘poutcome’ into 3 dimensions and visualise decision tree, python module ‘category_encoders’ and ‘pydotplus’ are required to install. ‘pip install pydotplus’ and ‘pip install category_encoders’ are the command in terminal for installing. To implement the model for testing a new data set, ‘Final Model Code.ipynb’ is provided. Part A is same as the process in the ‘Evaluation Code.ipynb’ for training RF model. Before predicting the new data, part A should be run first. Part B is for predicting new data. Starting with loading new dataset firstly. ‘duration’, ’previous’ are not required for new data collection but if they exist, there is code to delete them first. Step 3 and 4 is for encoding categorical variables followed by grouping ‘age’, ‘balance’ and ‘day’ and dropping ‘poutcome_1’ in this process. Previous steps are meant for preparing the data for prediction, after those steps there should be 14 input variables. The final step is to get the prediction result and derives it as a CSV file. Part C is for evaluating the model performance after obtaining the final result. As getting the actual result, it can be added to previous data for better training of the model in part A. However, the quality of new data should be carefully examined before this process.

Section F: Business Case Recommendations

Random Forest model is built up for targeting potential clients of “N/LAB Platinum Deposit” telemarketers to contact. As figure 15 indicates ‘housing’ as an important feature, it also displays a negative correlation with ‘y’. Figure 14 represents 28.75% clients without housing loan are tending buy product in campaign compared with 14.37% purchasing for clients with housing loan. Hence the telemarketer group should focus more on clients without housing loan. Secondly, ‘duration’ shows a moderate correlation and plays high gini importance in section A and B. Although it is not a predictive feature, figure 16 suggests telemarketers to extend the duration for the phone call to increase the possibility for successful sales. ‘Campaign’ means calls made in this campaign, displaying a negative relationship with ‘y’. Figure 17 suggests that on average successful sale has less number of calls for the reason that more calls may annoy customers and leading to failure. Generally, calls made should be thoughtfully examined. Furthermore, as figure 8 shows, repeating clients and clients without housing loan should be put more focus. The last suggestion is that beyond assessing precision with confusion matrix, the business value can be better addressed with profit confusion matrix to select model with highest expected value. The information of cost caused by false alarms and missing potential clients and benefits from true alarm and excluding non target class is required.

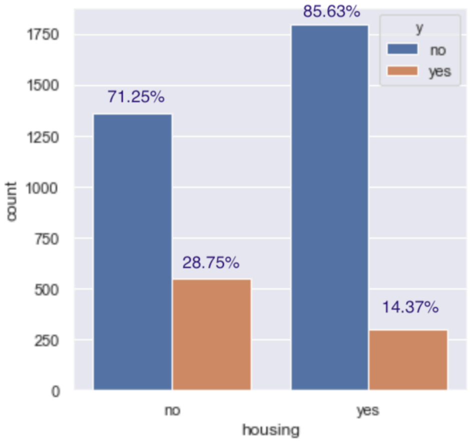


Figure 15 count plot ‘housing’ with ‘y’

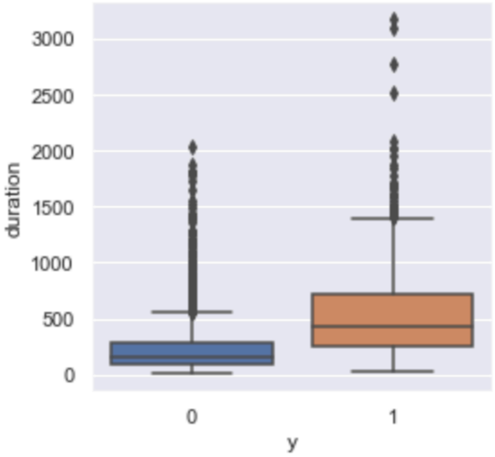


Figure 16 box plot ‘duration’ with ‘y’

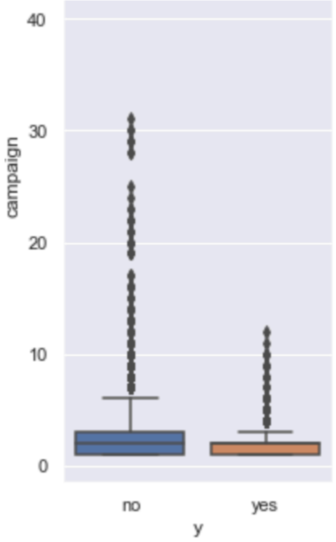
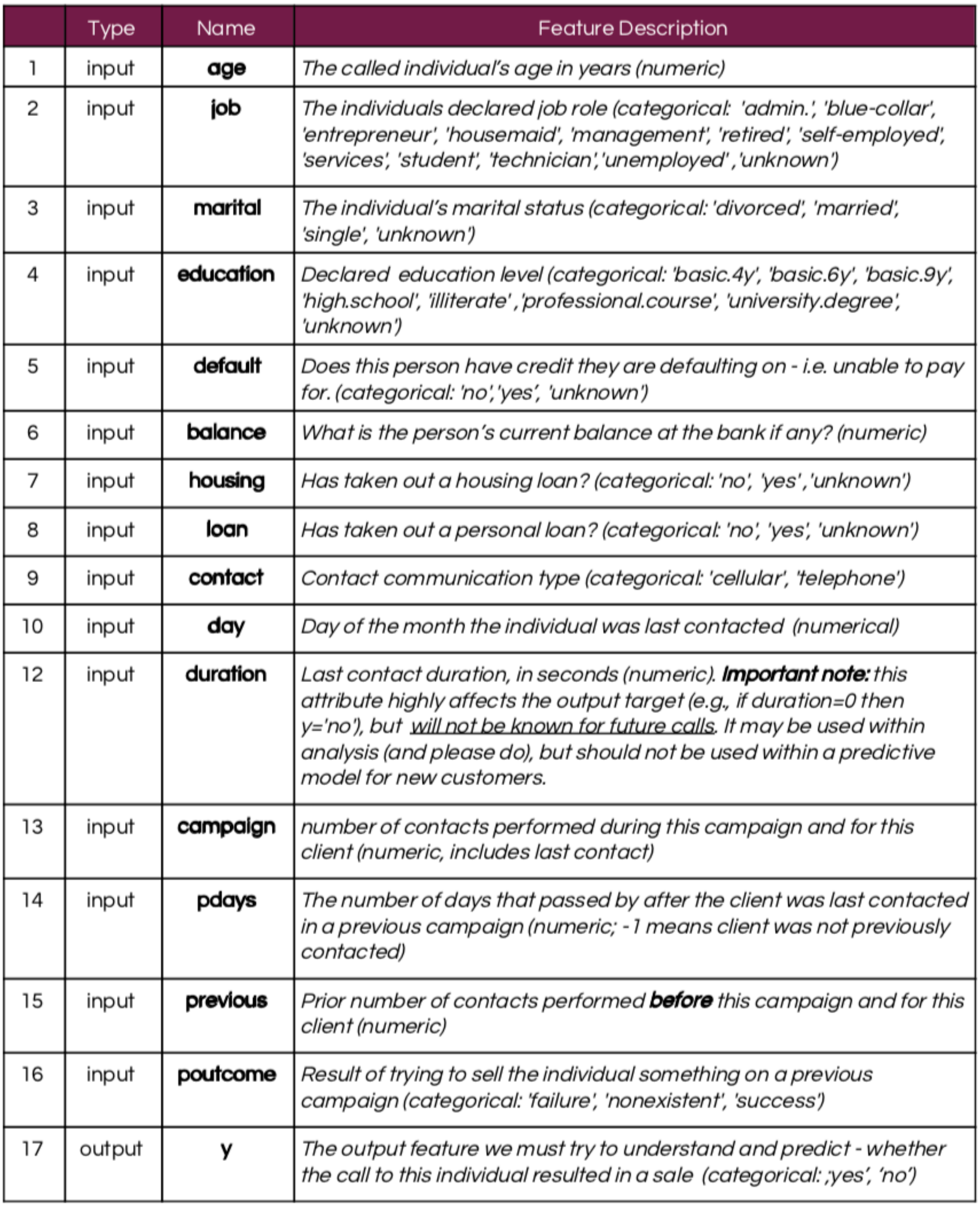


Figure 18 box plot ‘campaign’ with ‘y’

Appendix A



	Type	Name	Feature Description
1	input	**age**	*The called individual's age in years (numeric)*
2	input	**job**	*The individuals declared job role (categorical: 'admin.', 'blue-collar', 'entrepreneur', 'housemaid', 'management', 'retired', 'self-employed', 'services', 'student', 'technician','unemployed' ,'unknown')*
3	input	**marital**	*The individual's marital status (categorical: 'divorced', 'married', 'single', 'unknown')*
4	input	**education**	*Declared education level (categorical: 'basic.4y', 'basic.6y', 'basic.9y', 'high.school', 'illiterate' ,'professional.course', 'university.degree', 'unknown')*
5	input	**default**	*Does this person have credit they are defaulting on - i.e. unable to pay for. (categorical: 'no','yes', 'unknown')*
6	input	**balance**	*What is the person's current balance at the bank if any? (numeric)*
7	input	**housing**	*Has taken out a housing loan? (categorical: 'no', 'yes' ,'unknown')*
8	input	**loan**	*Has taken out a personal loan? (categorical: 'no', 'yes', 'unknown')*
9	input	**contact**	*Contact communication type (categorical: 'cellular', 'telephone')*
10	input	**day**	*Day of the month the individual was last contacted (numerical)*
12	input	**duration**	*Last contact duration, in seconds (numeric). **Important note:** this attribute highly affects the output target (e.g., if duration=0 then y='no'), but will not be known for future calls. It may be used within analysis (and please do), but should not be used within a predictive model for new customers.*
13	input	**campaign**	*number of contacts performed during this campaign and for this client (numeric, includes last contact)*
14	input	**pdays**	*The number of days that passed by after the client was last contacted in a previous campaign (numeric; -1 means client was not previously contacted)*
15	input	**previous**	*Prior number of contacts performed **before** this campaign and for this client (numeric)*
16	input	**poutcome**	*Result of trying to sell the individual something on a previous campaign (categorical: 'failure', 'nonexistent', 'success')*
17	output	**y**	*The output feature we must try to understand and predict - whether the call to this individual resulted in a sale (categorical: ;yes', 'no')*

Appendix B

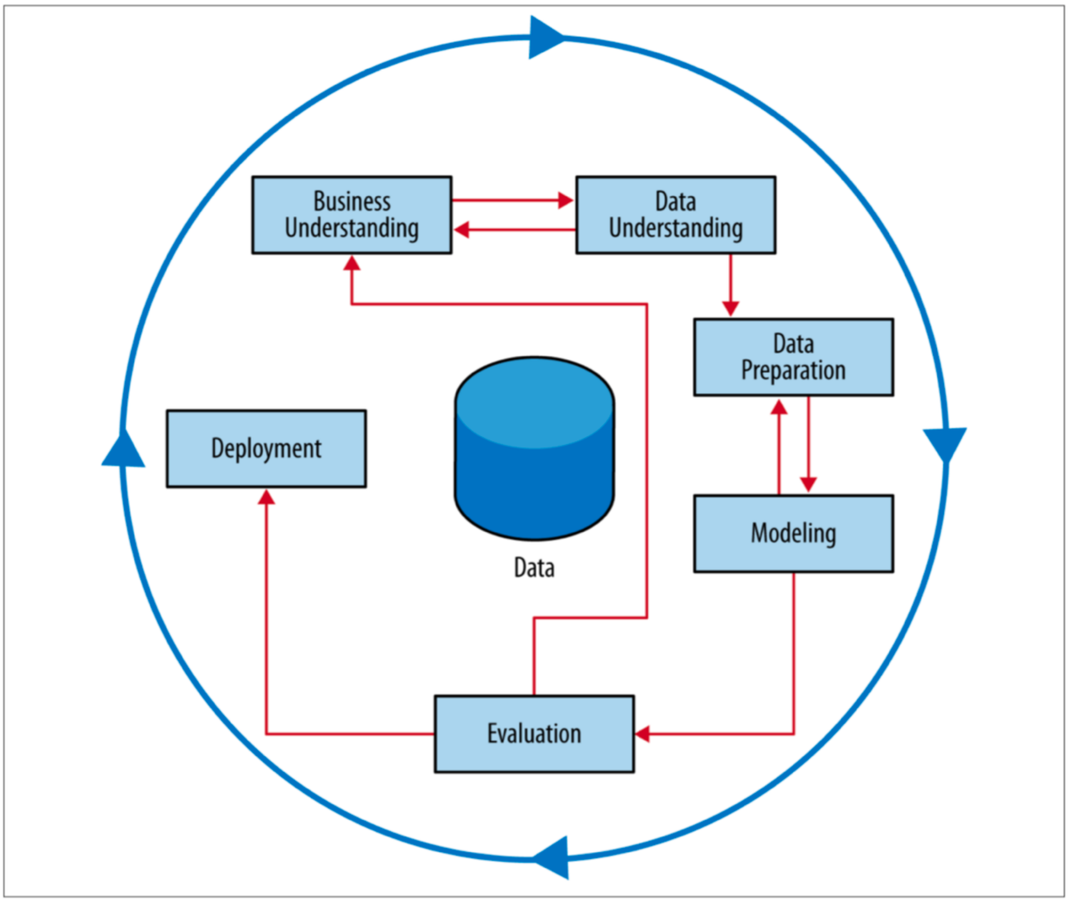


Figure 6-15. The CRISP data mining process.