Abstract

Effectively identifying and actively engaging prospective customers with a high probability of conversion is of utmost importance in optimising marketing expenditures. This research project employs the CRISP-DM methodology to systematically address the challenge. Through comprehensive data and processing, the Support Vector Machine (SVM) model, exhibiting superior accuracy among five constructed models, was chosen to establish a lead prediction system. Our findings indicate that the SVM model demonstrates robust predictive capabilities, showcasing its potential for impactful practical applications in customer acquisition and conversion efforts.

1 Introduction

A critical success factor for banking and financial service companies is the ability to identify and capitalise on prospective customers; therefore, World Plus, a mid-size private bank, plans to implement a lead conversion system for their new term deposit product to minimise their costs with the challenge of accurately predicting leads. This report aims to provide insight and solutions that align with World Plus's objectives to enhance accurately predicting targeted customers through data mining techniques. The Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology will be employed along with this report which each process can be observed in Figure 1. In the first section, the data processing will be discussed, followed by modelling and methods used for prediction including Decision Tree, Support Vector Machine (SVM), Logistic Regression, Random Forest, and Naïve Bayes with brief literature reviews. After modelling, each model will be evaluated using tools such as Confusion Matrix, Receiver Operating Characteristic curve (ROC), or Gain Chart to pinpoint the most effective model for enhancing accurate customer targeting, culminating in a conclusion that summarises insights and proposes actionable plans for World Plus.

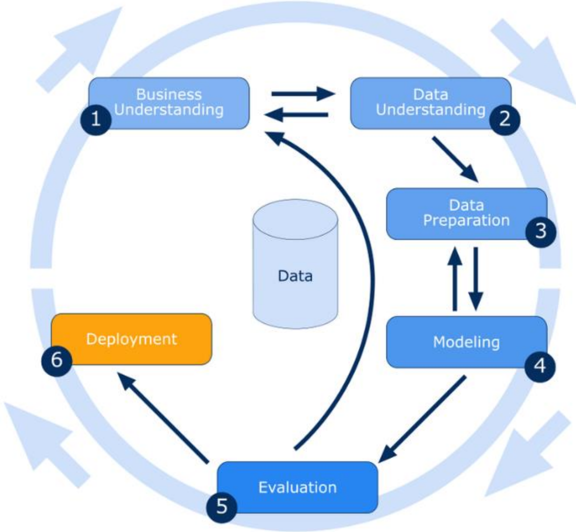


Figure 1 CRISP-DM Framework

2 Data Processing

The dataset, consisting of 220,000 records from past product offerings, focused on identifying customer purchases using 15 predictors (see Appendix A for data dictionary). To ensure the accuracy and efficiency of the predictive models, data preparation plays a fundamental role. The very first step is to clean the data by handling data type, missing values and outliers that show in the dataset. Overall, there are diverse ways to solve these problems, however, our approach is to apply A/B testing to test the different methods on the different models to find the best performance measured by confusion matrix that is suitable to our goals. Therefore, the missing values were omitted while the outliers remained because there were no strange or errors with this information. For data type, one hot encoding was used to deal with the nominal variables because SVM and Logistics Regression model, which will be used later, do not work well with this type of categorical variables (see Appendix B.1 for one hot encoding). In the next step, before doing the data partition, the information gain of the predictors on the target variable had been checked so, I could eliminate the attributes that had a low level of importance to the target variable (see Appendix B.2 for Information gain plot). This helps avoid overfitting and improve the performance of the predictive models. After that, stratified sampling with the ratio of 2:1 was used to create the training and test set for modelling because I want to ensure the homogeneity among the groups and avoid information loss from false negative results.

There is a huge gap between the non-target and target customers as can be observed in Figure 2.

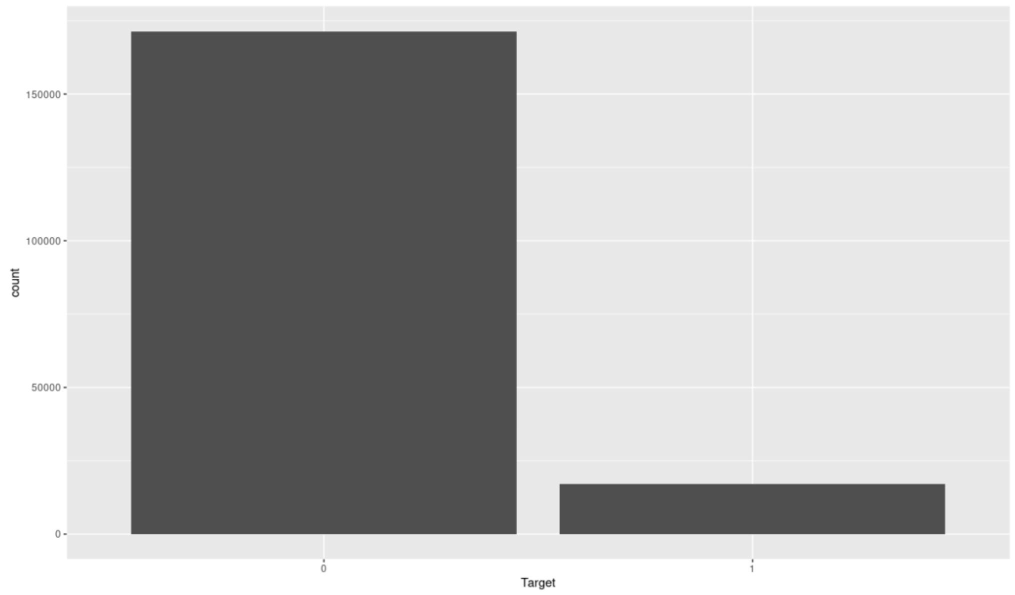


Figure 2 The number of non-targets (0) and targets (1)

I applied Random Oversampling (ROS), Random Under sampling (RUS) and both, ROS and RUS, at the same time to balance the dataset and compared their performance with the original unbalanced dataset. Logistics Regression and Random Forest models were randomly picked to test on to avoid biased results.

Logistics Regression Models	Accuracy	Precision	Recall	AUC	F1
Unbalanced Dataset	93.03%	72.45%	39.38%	87.30%	51.02%
ROS	89.07%	43.74%	64.66%	87.32%	52.18%
RUS	88.80%	42.90%	64.93%	87.33%	51.67%
ROS + RUS	88.87%	43.15%	65.09%	87.31%	51.89%

Figure 3 The results of different sampling methods on LR model

Random Forest	Accuracy	Precision	Recall	AUC	F1
Unbalanced Dataset	93.44%	75.13%	54.75%	87.34%	54.75%
ROS	92.01%	56.85%	55.35%	87.83%	56.09%
RUS	88.17%	41.63%	70.47%	87.91%	52.34%
ROS + RUS	91.08%	51.36%	61.04%	87.82%	55.78%

Figure 4 The results of different sampling methods on RF model

According to the results from Figure 3 and 4, ROS + RUS performed quite well in both models and did not overfit compared to the other strategies in models. Therefore, I decided to apply both ROS + RUS at the same time to balance the original dataset.

3 Evaluation

I used the data to create five models to predict the lead conversion for World Plus. The results of these models shown in Figure 5 were used to predict the best model for the bank. There is always a trade-off between precision and recall due to their inverse relation illustrated in formulations below. I settle for the results where our models are performing well in terms of identifying and correctly classifying the positive instances.

$$Precision = \frac{TruePositive}{TruePositive+FalsePositive} \qquad (4\text{-}1)$$

$$Recall = \frac{TruePositive}{TruePositive+FalseNegative} \qquad (4\text{-}2)$$

According to Figure 5, logistic regression exhibits the highest recall in comparison, although it does not achieve the highest precision. The Random Forest (RF) and Support Vector Machine (SVM) models yield comparable results, with slightly elevated accuracy and precision observed in the SVM model, along with a superior Gain Chart outcome considered in Figure 6. The SVM model emerges as the most accurate classification model, achieving 91.31% accuracy, 52.54% precision, 55.54% F-measure, and 58.90% recall (see Appendix C for more detail of results).

Despite the SVM model's superior performance, certain limitations should be acknowledged. These include potential computational intensity when handling large datasets, sensitivity to noise and outliers, and the critical impact of kernel selection on model performance and outcomes. These considerations are crucial in the meticulous construction of the model.

Model	Accuracy	Precision	Recall	AUC	F1
Random Forest	91.04%	51.16%	60.95%	87.84%	55.63%
Decision Tree	83.76%	46.41%	64.26%	82.13%	53.89%
Logistic Regression	88.87%	43.15%	65.09%	87.31%	51.89%
SVM-radial	91.31%	52.54%	58.90%	86.16%	55.54%
Naive Bayes	76.94%	34.61%	63.11%	85.07%	44.70%

Figure 5 The evaluation of each model

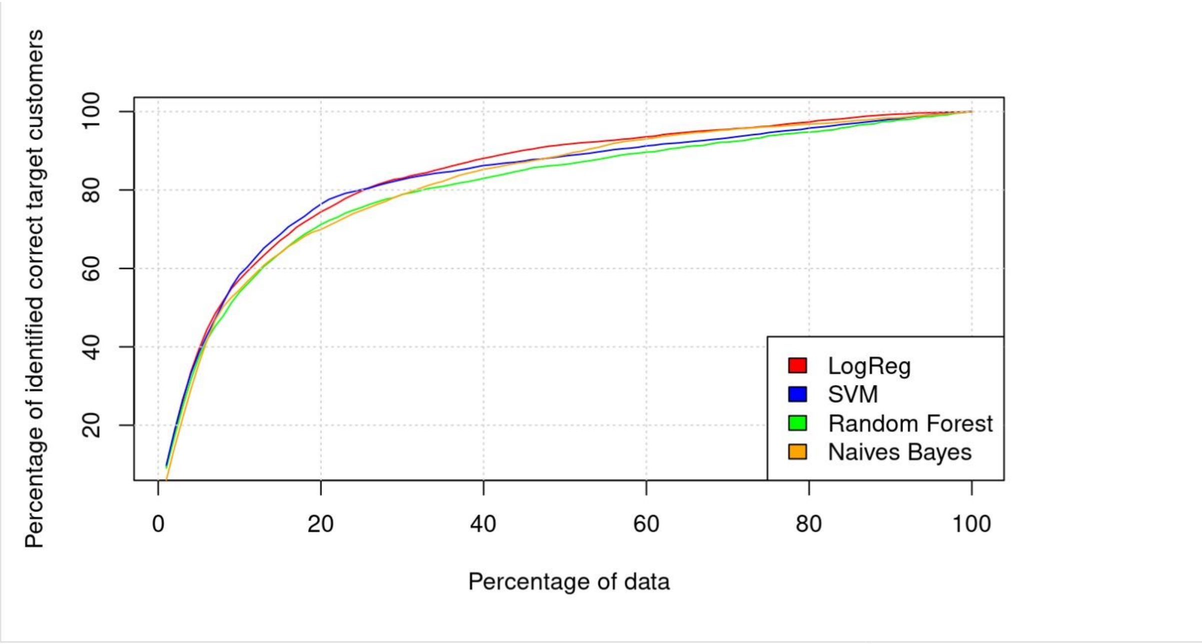


Figure 6 Gain Chart of all models

The ROC in Figure 7 shows the performance of the classification model at all classification thresholds to understand the trade-off between true positive rate and false positive rate which RF, Logistic regression and SVM models achieve high and almost identical Area Under the Curve (AUC) values.

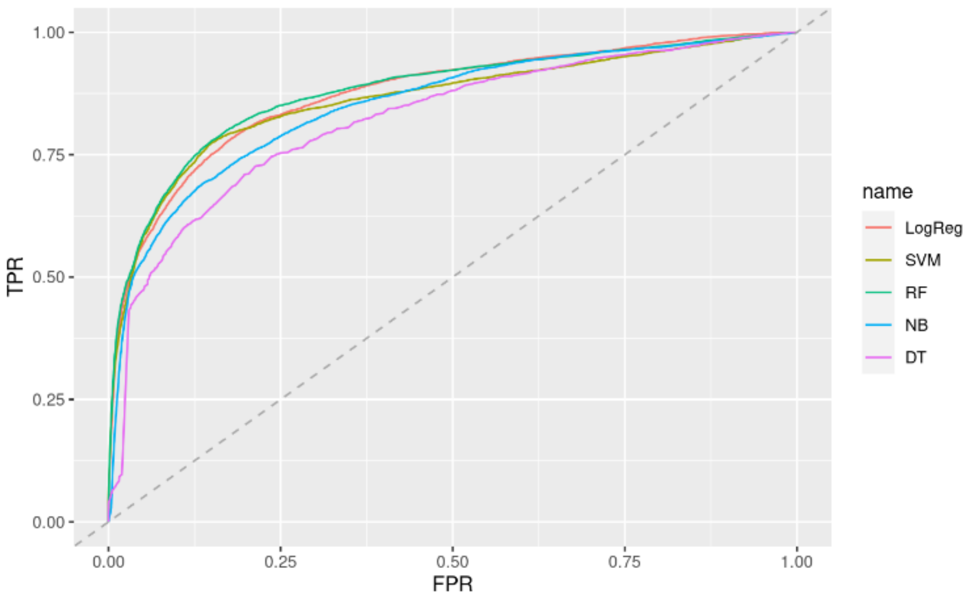


Figure 7 ROC compared the performance of each model and baseline

4 Conclusion

In conclusion, selecting the model with the highest accuracy is crucial for enhancing World Plus's ability to accurately identify leads. Considering the high cost of time and money, I prioritise the model with the highest precision to minimise the cost of false positives in targeting uninterested customers. Additionally, the F-Measure is considered to avoid missing potential customers. To maintain market competitiveness, the bank should make full use of machine learning, specifically the SVM model, for lead prediction. Our evaluation tools include Precision, Recall, accuracy, and area under the ROC curve to identify the best model to develop a lead conversion prediction aiming to target prospective customers through strategic communication channels which SVM proves to be the most efficient model with high precision and accuracy.

Appendices

Appendix A

Data Dictionary

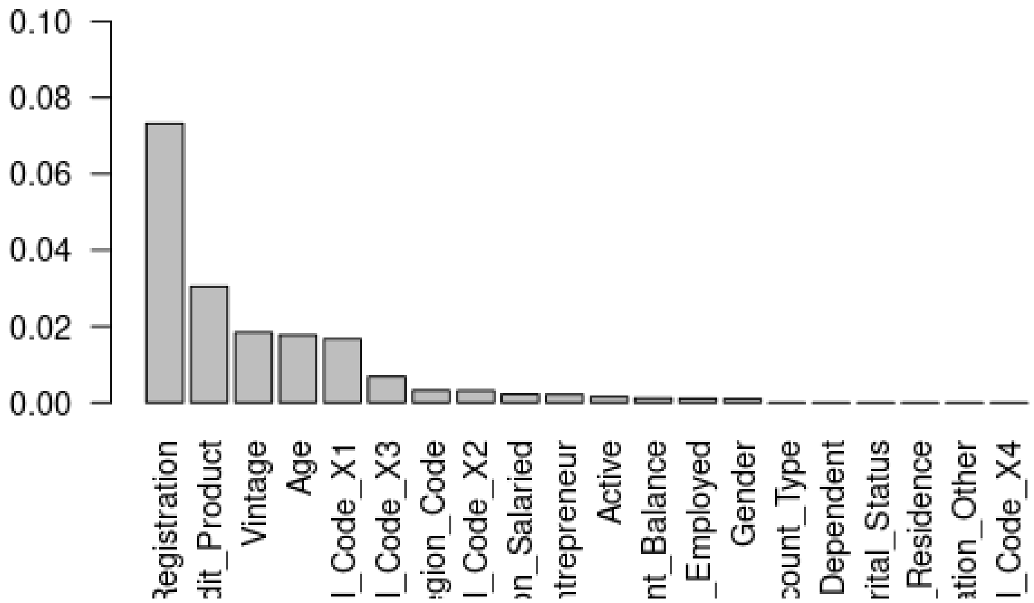
Variable	Description
ID	customer identification number
Gender	gender of the customer
Age	age of the customer in years
Dependent	whether the customer has a dependent or not
Marital_Status	marital state (1=married, 2=single, 0 = others)
Region_Code	code of the region for the customer
Years_at_Residence	the duration in the current residence (in years)
Occupation	occupation type of the customer
Channel_Code	acquisition channel code used to reach the customer when they opened their bank account
Vintage	the number of months that the customer has been associated with the company
Credit_Product	if the customer has any active credit product (home loan, personal loan, credit card etc.)
Avg_Account_Balance	average account balance for the customer in last 12 months
Account_Type	account type of the customer with categories Silver, Gold and Platinum
Active	if the customer is active in last 3 months
Registration	whether the customer has visited the bank for the offered product registration (1 = yes; 0 = no)
Target	whether the customer has purchased the product, 0: Customer did not purchase the product; 1: Customer purchased the product

Appendix B

Figure B.1 One hot encoding and removing irrelevant data

```
> mydata <- one_hot(as.data.table(mydata), cols = "Occupation")
> mydata <- one_hot(as.data.table(mydata), cols = "Channel_Code")
> mydata <- mydata %>% filter(Dependent != -1)
> mydata$ID <- NULL
> mydata$Target<- as.factor(mydata$Target)
> str(mydata)
Classes ‘data.table’ and 'data.frame':	219882 obs. of  21 variables:
 $ Gender                   : Factor w/ 2 levels "Female","Male": 1 1 1 2 1 2 1 1 1 2 ...
 $ Age                      : int  73 30 56 34 30 56 48 40 55 53 ...
 $ Dependent                : int  0 1 0 0 1 1 0 1 0 1 ...
 $ Marital_Status           : int  1 1 0 1 0 0 0 0 2 1 ...
 $ Region_Code              : Factor w/ 35 levels "RG250","RG251",..: 19 28 19 21 33 12 16 34 19 5
...
 $ Years_at_Residence       : int  1 1 3 5 3 2 2 4 5 5 ...
 $ Occupation_Entrepreneur : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Occupation_Other         : int  1 0 0 0 0 0 0 0 0 0 ...
 $ Occupation_Salaried      : int  0 1 0 1 1 0 0 0 0 0 ...
 $ Occupation_Self_Employed: int  0 0 1 0 0 1 1 1 1 1 ...
 $ Channel_Code_X1          : int  0 1 0 1 1 1 0 0 0 0 ...
 $ Channel_Code_X2          : int  0 0 0 0 0 0 0 1 1 0 ...
 $ Channel_Code_X3          : int  1 0 1 0 0 0 1 0 0 1 ...
 $ Channel_Code_X4          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Vintage                  : int  43 32 26 19 33 32 13 38 49 88 ...
 $ Credit_Product           : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 2 1 ...
 $ Avg_Account_Balance      : int  1045696 581988 1484315 470454 886787 544163 444724 1274284 2014239
980664 ...
 $ Account_Type             : Factor w/ 3 levels "Gold","Platinum",..: 1 1 3 3 1 1 3 1 1 2 ...
 $ Active                   : Factor w/ 2 levels "No","Yes": 1 1 2 1 1 2 2 1 1 2 ...
 $ Registration             : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Target                   : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

Figure B.2 Information Gain



Appendix C

The results of each model from Confusion Matrix

Figure C.1 Decision Tree

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 48767  2216
         1  6150  3361

               Accuracy : 0.8617
                 95% CI : (0.8589, 0.8644)
    No Information Rate : 0.9078
    P-Value [Acc > NIR] : 1

                  Kappa : 0.3726

 Mcnemar's Test P-Value : <2e-16

              Precision : 0.35338
                 Recall : 0.60265
                     F1 : 0.44552
             Prevalence : 0.09219
         Detection Rate : 0.05556
   Detection Prevalence : 0.15722
      Balanced Accuracy : 0.74533

       'Positive' Class : 1
```

Figure C.2 SVM

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 51950  2292
         1  2967  3285

               Accuracy : 0.9131
                 95% CI : (0.9108, 0.9153)
    No Information Rate : 0.9078
    P-Value [Acc > NIR] : 3.41e-06

                  Kappa : 0.5074

 Mcnemar's Test P-Value : < 2.2e-16

              Precision : 0.52543
                 Recall : 0.58903
                     F1 : 0.55541
             Prevalence : 0.09219
         Detection Rate : 0.05430
   Detection Prevalence : 0.10335
      Balanced Accuracy : 0.76750

       'Positive' Class : 1
```

Figure C.3 Random Forest

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 51672  2178
         1  3245  3399
                                          
               Accuracy : 0.9104          
                 95% CI : (0.9081, 0.9126)
    No Information Rate : 0.9078          
    P-Value [Acc > NIR] : 0.01522         
                                          
                  Kappa : 0.5068          
                                          
 Mcnemar's Test P-Value : < 2e-16         
                                          
              Precision : 0.51159         
                 Recall : 0.60947         
                     F1 : 0.55626         
             Prevalence : 0.09219         
         Detection Rate : 0.05619         
   Detection Prevalence : 0.10983         
      Balanced Accuracy : 0.77519         
                                          
       'Positive' Class : 1               
```

Figure C.4 Logistic Regression

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 50134  1947
         1  4783  3630
                                          
               Accuracy : 0.8887          
                 95% CI : (0.8862, 0.8912)
    No Information Rate : 0.9078          
    P-Value [Acc > NIR] : 1               
                                          
                  Kappa : 0.459           
                                          
 Mcnemar's Test P-Value : <2e-16          
                                          
              Precision : 0.43148         
                 Recall : 0.65089         
                     F1 : 0.51894         
             Prevalence : 0.09219         
         Detection Rate : 0.06001         
   Detection Prevalence : 0.13907         
      Balanced Accuracy : 0.78190         
                                          
       'Positive' Class : 1               
```

Figure C.5 Naïve Bayes from Confusion Matrix

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 47601  1748
         1  7316  3829

               Accuracy : 0.8502
                 95% CI : (0.8473, 0.853)
    No Information Rate : 0.9078
    P-Value [Acc > NIR] : 1

                  Kappa : 0.382

 Mcnemar's Test P-Value : <2e-16

              Precision : 0.34356
                 Recall : 0.68657
                     F1 : 0.45796
             Prevalence : 0.09219
         Detection Rate : 0.06330
   Detection Prevalence : 0.18423
      Balanced Accuracy : 0.77668

       'Positive' Class : 1
```