Bank Direct Marketing Campaign Prediction using Machine Learning Methods

Author: Qianhui Li

AAE 724 Capstone Project

Professor: Guanming Shi

University of Wisconsin, Madison

Introduction

There are two common approaches to marketing promotions, which are mass marketing and direct marketing. Mass marketing does not directly engage with customers, it uses TV, radio, and newspapers to broadcast their promotional message. While direct marketing engages with customers directly like by phone calls, it helps in selecting targeted customers and focus more on customers' specific needs (Elsalamony, 2014). Direct marketing is in theory more effective than mass marketing. The development in technology makes direct marketing becomes more and more possible. The bank did mass marketing only previously because they have fewer ways to do direct marketing back then. Hence, the first motivation of my research is the trend among the banks that shifts from mass marketing to direct marketing.

The second motivation behind this is that the machine learning method has an increasing popularity in the banks. It saves marketing campaign costs and increases customers' responses rate. Also, the machine learning method can be applied to direct marketing, which utilizes customers' historical purchasing data and predictive models to measure whether a customer will respond to an offer or not more accurately (Sing'oei &Wang, 2013). Additionally, although direct marketing such as telemarketing is an interactive and powerful tool, it annoys customers sometimes (Vajiramedhin & Suebsing, 2014). The machine learning prediction can help to identify target customers efficiently without bothering customers by frequent phone calls.

My overall goal is to compare the four most commonly used machine learning methods and select the most accurate mechanism that predicts customers' responses to the bank's direct marketing offer, which in this case is a long-term deposit offer, and identify the relevant factors that affect customers' responses.

Data

1. Description

My proposed methods performance is assessed using the real data from the University of California, Irvine (UCI) Machine Learning Repository (Moro et al., 2014). The dataset is obtained from a Portuguese banking institution from May 2008 to November 2010, and the marketing campaigns were based on phone calls. It was often the case that more than one contact to the same client was required to obtain whether the bank term deposit would be ('yes') or not be ('no') subscribed. The original dataset involves 41,188 phone contacts in total with 20 input

variables and one output variable y, which will be listed in Table 1. The classification goal for output variable y is to predict whether the customer will or not (yes=1/no=0) subscribe to the long-term deposit. There are two types of input variables, which are numerical and categorical, and details are listed below in Table 1.

Table 1. Variable Descriptions

Number	Variable Name	Description	Type
1	Age	Age of the customer	numeric
2	Job	Type of job ('admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')	categorical
3	Marital	Marital status ('divorced','married','single','unknown'; note: 'divorced' means divorced or widowed)	categorical
4	Education	Education status('basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown')	categorical
5	Default	Has credit in default? ('no','yes','unknown')	categorical
6	Housing	Has housing loan? ('no','yes','unknown')	categorical
7	Loan	Has personal loan? ('no','yes','unknown')	categorical
8	Contact	Contact communication type ('cellular','telephone')	categorical
9	Month	Last contact month of year ('jan', 'feb', 'mar', ..., 'nov', 'dec')	categorical
10	Day_of_week	Last contact day of the week (categorical: 'mon','tue','wed','thu','fri')	categorical
11	Duration	Last contact duration, in seconds. Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.	numeric
12	Campaign	Number of contacts performed during this campaign and for this client (includes last contact)	numeric
13	pdays	Number of days that passed by after the client was last contacted from a previous campaign (999 means client was not previously contacted)	numeric
14	Previous	Number of contacts performed before this campaign and for this client	numeric
15	poutcome	Outcome of the previous marketing campaign ('failure','nonexistent','success')	categorical
16	emp.var.rate	Employment variation rate - quarterly indicator	numeric
17	cons.price.idx	Consumer price index - monthly indicator	numeric
18	cons.conf.idx	Consumer confidence index - monthly indicator	numeric
19	euribor3m	euribor 3 month rate - daily indicator	numeric
20	nr.employed	Number of employees - quarterly indicator	numeric
21	y	Has the client subscribed a term deposit?	(binary: 'yes','no')

2. Data Anomalies

2.1. Unknown values

First of all, I notice some unknown values labeled as “unknown”, “none-existent”, or “999”. The unknown values present most in observations of variable “pdays”, “default”, and “duration”.

- “pdays”: Number of days that passed by after the client was last contacted from a previous campaign . Since "999" in pdays means the client was not previously contacted, and there is a large percentage of “999”, so I convert “pdays” into a binary variable, never contacted(999)=0, contacted=1;
- “default”: Has credit in default. The second variable that has the largest proportion of unknown values is "default”. However, it may be possible that the customer is not willing to disclose this information to the banking representative. Hence, the unknown value in 'default' is actually a separate value. Thus I keep the unknown category and rename it as “refuse to disclose”, and I treat "loan" and "housing" variables in the same way;
- “duration”: Last contact duration, in seconds. As indicated by the data contributor, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. This input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model. Hence, I remove "duration".

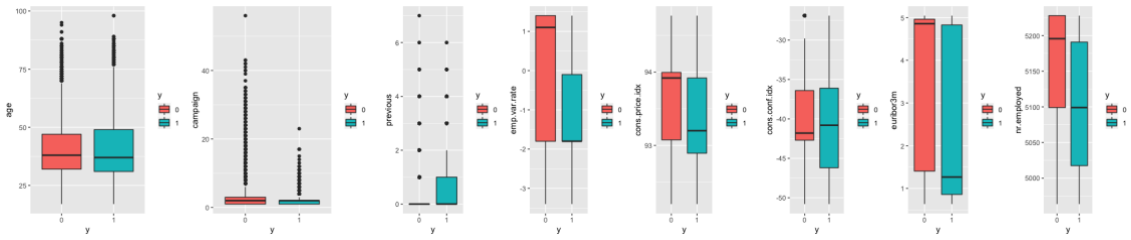
2.2. Outliers

During data preparation, I identify some outliers in both numeric and categorical variables observations. Figure 1 shows that “previous” has the largest amount of outliers, and then “campaign”, “age”, and “cons.conf.idx”. The threshold I used to remove outliers is :

$$\boldsymbol{Remove\ observations} > \boldsymbol{3rd\ Quartile} + \mathbf{1.5} * (\boldsymbol{3rd\ Quartile} - \mathbf{1}\boldsymbol{st\ Quartile})$$

- Age: Remove observations > 69.5
- Campaign: Remove observations > 6
- Previous: Remove observations> 2
- Cons.conf.idx: Remove observations >-26.95

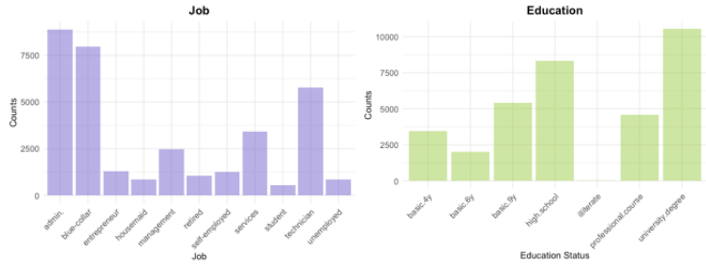
Figure 1. Numeric Variables Boxplots



I also check outliers for categorical variables with more than 3 categories. Figure 2 indicates that there is an extremely small number of observations for the "illiterate" category in Education, which was only 16. So I treat “illiterate” observations as an outlier category and drop it.

- Job: No removal
- Education: Drop observations within "illiterate” category

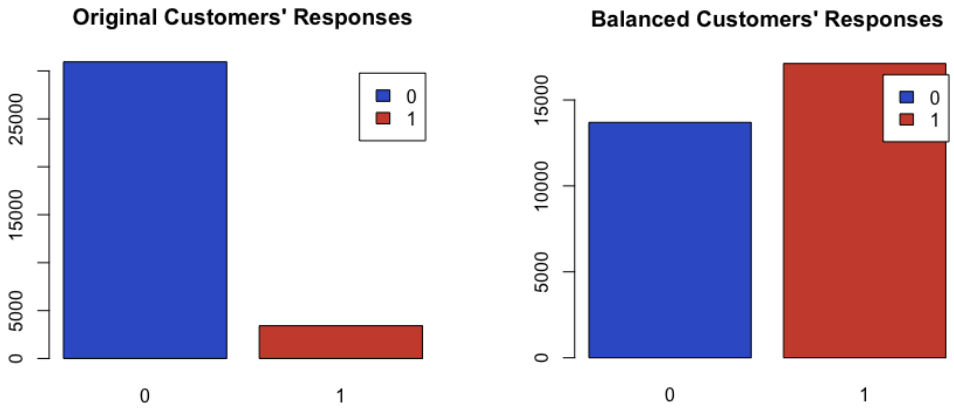
Figure 2. Categorical Variables Histogram



2.3. Data Imbalance

Thirdly, during data processing, I notice the data is highly imbalanced. Among 34,354 observations, only 11% of customers subscribed to the term deposit (y=1), remaining almost 90% who did not (y=0). So I use a popular resampling method called SMOTE to synthetic the minority group. After resampling, I get 30,816 observations, among which 44.4% of customers responded with no and 55.6% responded with yes.

Figure 3. Customers’ Responses before & after Resampling

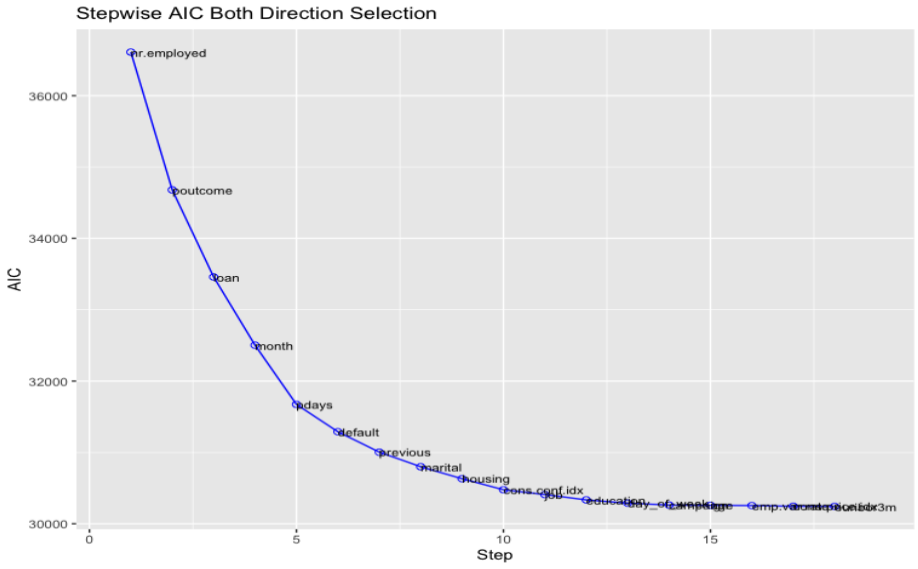


Methodologies

1. **Feature Selection**

I use feature selection for Logistic Model and Neural Network, since the Classification Tree and random forest can automatically select variables by themselves, while the Logistic Model and Neural Network cannot. I apply the mixed selection method for the Logistic Model and Neural Network, and the selected variables are listed in Table 2. It starts with no variables in the model, and as with forwarding selection, it adds the variable that provides the best fit, then continues to add variable one-by-one. If any point the p-value for one of the variables in the model rose above a certain threshold, it removes that variable, then continues until all variables in the model have a sufficiently low p-value. After screening variables with balanced data, no input variable is removed. Hence, I include all 19 input variables in all four methods.

Figure 4. Feature Selection by Steps



2. Model Performance Measurement

To predict customers' responses to bank direct marketing term deposit subscription, 4 algorithms are used, which are the Logistic model, Classification Tree, Random Forest, and Artificial Neural Network. The balanced data is then randomly divided into training data (50% of the balanced data), and the remaining data serves as the test dataset. Since the dataset is already balanced, the accuracy comparison is now a direct approach to validate each model and find the model that is the most accurate. The accuracy measures how accurate the certain test performs the classification, and it can be computed from the confusion matrix. Table 2 below shows the classical layout of the confusion matrix, and it is followed by the Accuracy formula. The reason

that I don't use RMSE as performance measurement is that RMSE is commonly used in regression, where a predictor variable is a real number. However, in classification, I have class labels or categories, so it does not correspond to numbers. Besides, it is hard for RMSE to find the difference between "a" and "b", so I use accuracy that can be computed from the confusion matrix instead.

Table 2. Sample Confusion Matrix

		Predicted class: *P*	Predicted class: *N*
Actual Class	*P*	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

$$\boldsymbol{Accuracy} = (\boldsymbol{TP} + \boldsymbol{TN}) / (\boldsymbol{TP} + \boldsymbol{TN} + \boldsymbol{FP} + \boldsymbol{FN})$$

3. Classification Models

3.1. Logistic Model

$$\boldsymbol{Y_i} = \ln\left(\frac{\boldsymbol{p(Y_i = y|X_i)}}{\boldsymbol{1 - p(Y_i = y|X_i)}}\right) = \boldsymbol{\beta_0} + \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{\beta_i X_i}$$

I am interested in predicting customers' responses to bank direct marketing campaigns of term deposit subscriptions, and y=1 if the customer subscribes the term deposit, while y=0 if the customer does not subscribe. Since my dependent variable is binary, I would like to start with fitting and testing the Logistic Model. The Logistic Model produces a linear decision boundary, and the classification in the Logistic Model is allowed to be uncertain, which is reflected by the intermediate values between 0 and 1.

Logistic model Accuracy Table 3 shows that the Logistic model does a bad job in prediction accuracy, and it only achieves 75.97% prediction accuracy in both training and test data.

Table 3. Logistic Model Accuracy

	Training Data	**Test Data**

Accuracy	75.97%	75.97%

3.2. Classification Tree

Unlike logistic and linear regression, Classification Tree does not have a specific equation. Data are partitioned along with the predictor variables into subsets with homogeneous values of the dependent variable and to reduce class mixing at each split. It is similar to growing a large tree and then prune it. Pruning can be done by randomly selecting a test sample and computing the error by running it down the large tree and subtrees. The tree with the smallest error will be the final tree. The result of the pruned tree achieves 89.19% training accuracy and 89.15% test accuracy. Figure 5 indicates that the final tree with the lowest error has 11 terminal nodes, with cp = 0.012.

The tree automatically selected 9 out of 19 variables before building the final model, which are “cons.conf.idx”, “nr.employed”, “cons.price.idx”, “month”, “euribor3m”, “emp.var.rate”, “previous”, “contact”, “loan”. Figure 6 shows that among those 9 variables, 4 variables are used in the final tree model, which are “nr.employed”, “cons.conf.idx”, “cons.price.idx”, and “month”.

Table 4. Classification Tree Accuracy

	Training Data	**Test Data**
Accuracy	89.19%	89.15%

Figure 5. Size of tree

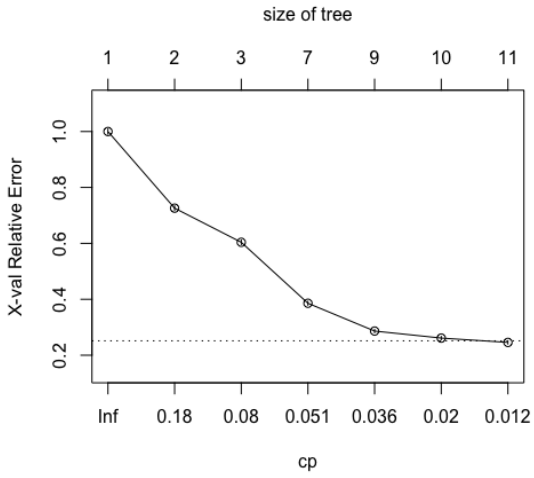
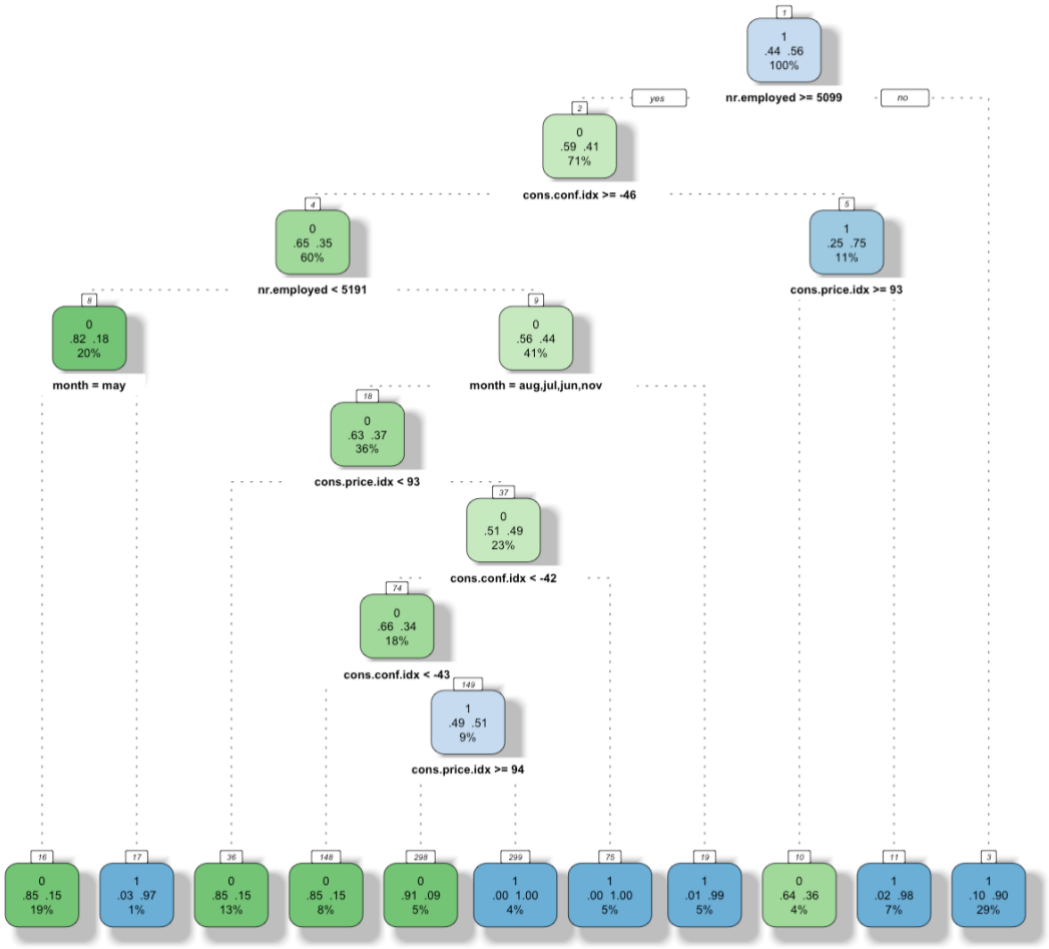


Figure 6. Final Tree Structure



3.3. Random Forest

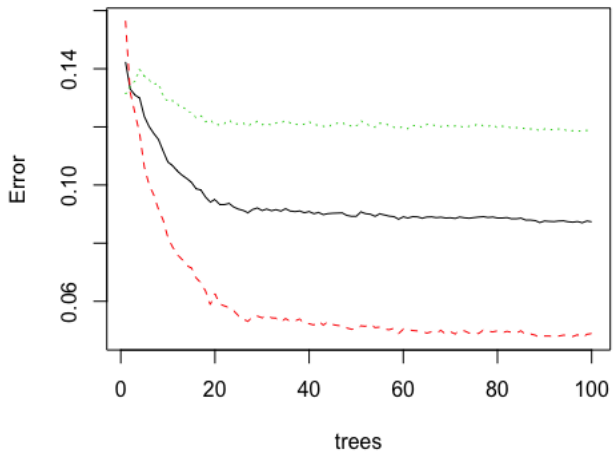
Unlike the decision tree performs as an individual trivial tree, Random Forest forms a forest with many decision trees. In my random forest model, there are 100 trees in the forest. It uses all 19 variables to build the model, and the number of variables tried at each split is 4. On average, each bagged tree makes use of around 2/3 of the observations, and the remaining 1/3 that is not used to fit a given bagged tree are the OOB (out-of-bag) observations. The OOB error is the classification error for the test data. The OOB error for an individual tree is accumulated and averaged as a measure of prediction accuracy of test data, and my OOB estimate of error rate is 8.74%. Table 5. Random Forest Accuracy backs up my initial assumption that Random Forest always works well in terms of classification accuracy. The accuracy for training data is 96.09%, with an accuracy of 91.61% for test data that is higher than other models.

Figure 7 shows the error rate of the Random Forest. The black line represents the overall averaged OOB error (1- test accuracy), and the red line is the prediction error rate for the y=0 (1-specificity), and the green line shows the prediction error rate for y=1 (1-sensitivity). The black OOB error line is always in the middle, because it measures the averaged error, while the up or low positions of green error rate line for y=1 and the red error rate line for y=0 are not always the same, and it depends on the data. This figure also indicates that the errors do not continue to decrease sharply after the number of trees in the forest reaches 20.

Table 5. Random Forest Accuracy

	Training Data	**Test Data**
Accuracy	96.09%	91.61%

Figure 7. The Error Rate of Random Forest



3.4. Artificial Neural Network

Artificial Neural Network is composed of many nodes, and these nodes are connected and function together, by passing information. They consist of several layers, and each layer performs a different function on the received data. Figure 9 shows the final structure of Neural Network with 1 input layer, 1 output layer, and 1 hidden layer which has 5 neurons. Figure 9 just provides a general structure of my neural network model, and no more detailed information should be specified. My final neural network model uses 58 predictors, 25 reps bootstrapped resampling, and accuracy is used to select the optimal model using the largest value. Table 7 Resampling Results across Tuning Parameters suggests that the best accuracy of my neural

network model is obtained at 5 neurons with decay of 0. As a result, the accuracy with training data is 90.31%, and with test data is 89.17%. The results shows the Neural Network does a good job in classification accuracy.

Table 6. Neural Network Accuracy

	Training Data	**Test Data**
Accuracy	90.31%	89.17%

Figure 9. Neural Network Structure

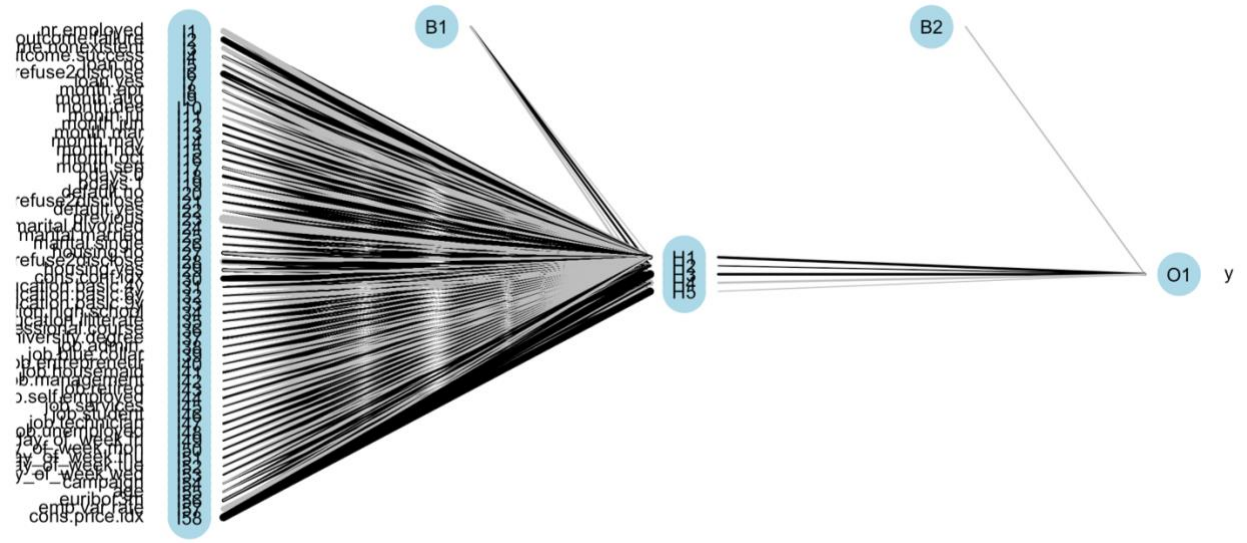


Table 7. Resampling Results across Tuning Parameters

size	decay	Accuracy	Kappa
1	0e+00	0.7543957	0.5162276
1	1e-04	0.7581908	0.5158607
1	1e-01	0.7617808	0.5293895
3	0e+00	0.8315494	0.6570095
3	1e-04	0.8727522	0.7453642
3	1e-01	0.8768841	0.7537362
5	0e+00	0.8937919	0.7862600
5	1e-04	0.8863729	0.7722653
5	1e-01	0.8820252	0.7638854

Result Analysis

1. Model Performance Comparison

Figure 10. Model Accuracy

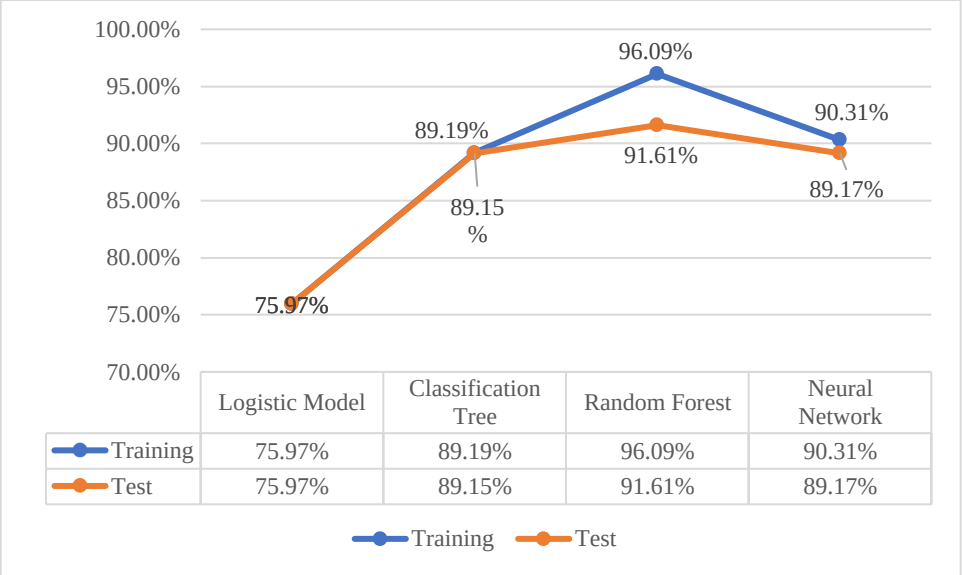


Figure 10 illustrates the model accuracy comparisons, and it indicates that Random Forest stands out and performs the best no matter in training data or test data, it achieves 96.09% accuracy with training data and 91.61% accuracy with the test dataset. Therefore, Random Forest has the best classification accuracy compared with the Logistic Model, Classification Tree, and Neural Networks.

Random Forest won the game in my case might because it is very stable. Even if a new data point is introduced in the dataset, the overall forest is not affected much since the new data may impact one tree, but it is very hard for it to impact all the trees in the forest. Besides, it has the ability to reduce overfitting and variance so as to improve the accuracy, and overfitting is a very common problem in other algorithms.

The Neural Network ranks the second, and the third is Classification Tree, which has similar accuracies as Neural Network. These two methods work well because they are good classifying a mix of categorical and numerical variables. Also, both methods can model data that have nonlinear relationships between variables.

The logistic regression works the worst in prediction accuracy. The Logistic Regression lost the game in my case might due to the parameter estimates are unstable. Besides, from the histogram in the data part, we know there are some categorical variables that have unbalanced distributions, thus it may perform way worse than others.

2. Variable Importance

Although Random Forest achieves the highest accuracy in classification prediction, the Neural Network and Classification Tree test accuracy results are close to Random Forest. So it's better to take a look at and compare the important variables in all those 3 models. The Random Forest selects its own variables, and it keeps all 19 variables. Figure 11 shows that if a variable is assigned values by random permutation, and how much the accuracy will decrease in the random forest model. It indicates that "cons.price.idx" and "cons.conf.idx" are the most important two variables in the Random Forest model, followed by "nr.employed". Figure 12 indicates that "cons.conf.idx", "nr.employed", and "cons.price.idx" are the top important variables in Classification Tree. Figure 13 illustrates that "previous", "poutcome.failure", "cons.price.idx" are the most important three variables, followed by "cons.conf.idx". Comparing the important variables in all those three models, I identify three variables that are most important, which are "cons.price.idx", "cons.conf.idx", and "nr.employed".

Figure 11. Random Forest Variable Importance

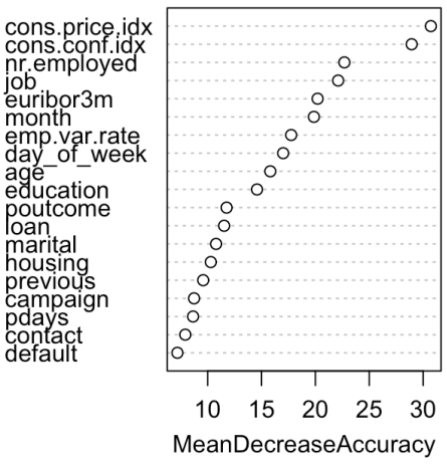


Figure 12. Tree Variable Importance

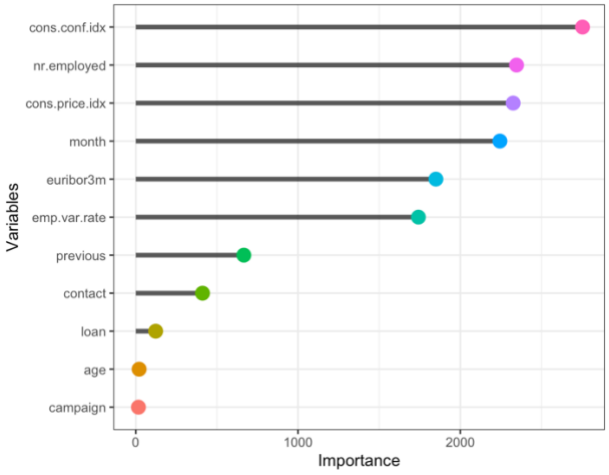
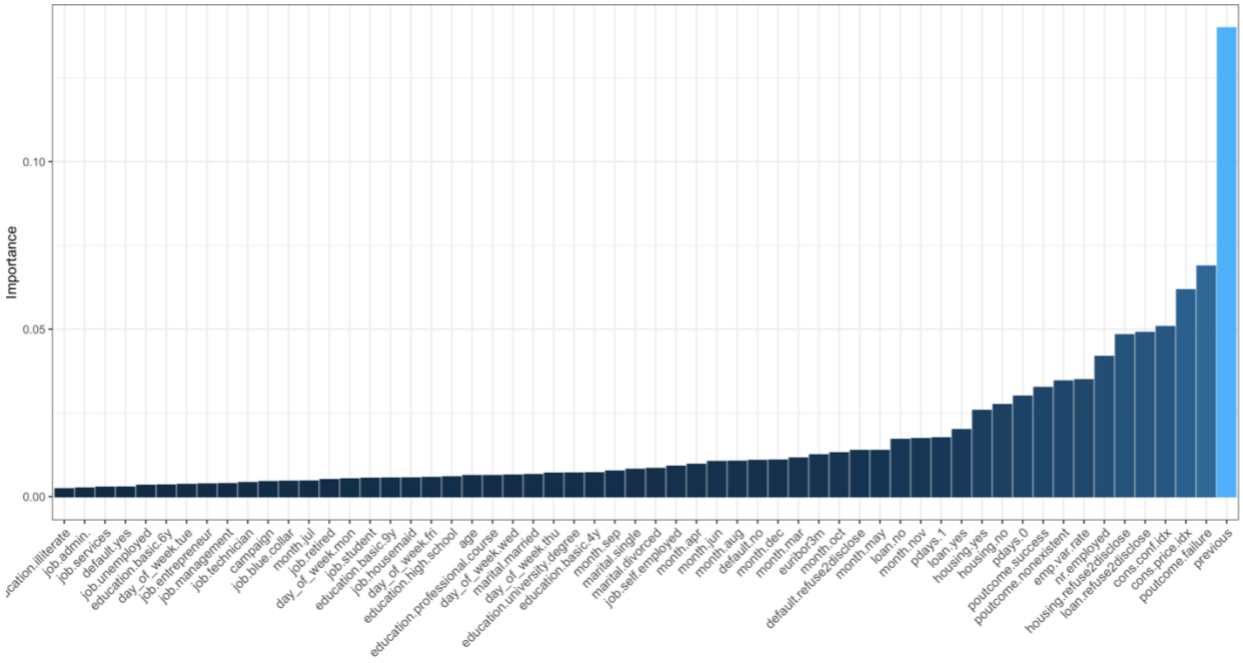


Figure 13. Neural Network Variable Importance



To identify the relationship between those 3 important variables and customers' response (y variable), I use the Partial Dependence Plot (PDP) from Random Forest. The PDP shows the marginal effect one or two features have on the predicted outcome of a machine learning model. For classification where the machine learning model outputs probabilities, PDP displays the probability for a certain class given different values for that feature.

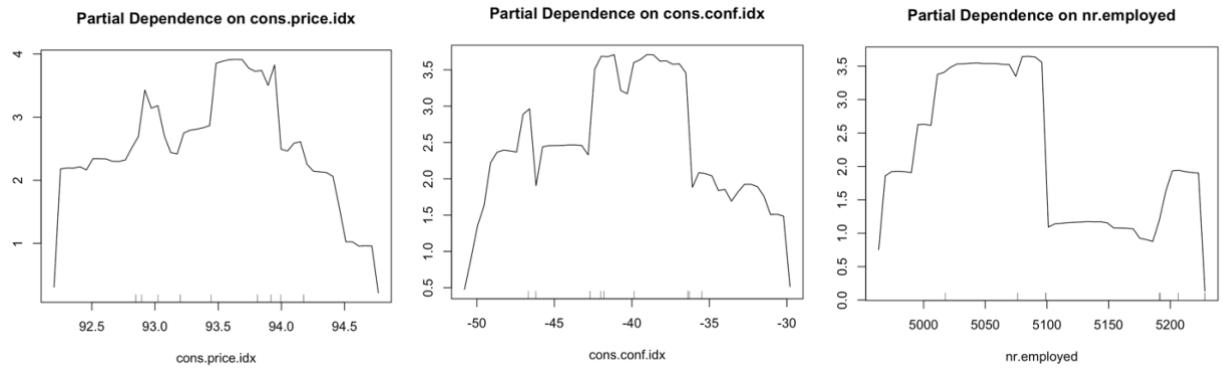
Figure 10 Partial Dependence Plot illustrates that for "cons.price.idx", the highest probability of customers' subscription (y=1) occurs when the consumers' price index is in the range of (93.5, 94). The probability of customers' subscription generally has an increasing trend in the consumers' price index range (92.0, 93.5), while it has a declining trend in the range of (94.0, 95.0). Within the reference peaking range of consumer price index, customers are more likely to subscribe to the long term deposit, because the price of goods might be high, and thus customers might prefer to put money in the bank, instead of spending it.

For "cons.conf.idx", the highest probability of customers' subscription peaks when the consumers' confidence index is in the range of (-47, -37). When the consumers' confidence index changes from (-50, -47), the probability of customers' subscription generally increases, while the probability of customers' subscription has a decreasing trend in the consumers' confidence index range (-37, -30). The result indicates that within the peaking range, consumers may feel less confident about the country's economy, so they will prefer to deposit their money

into the bank instead of spending their money, which is said customers are more willing to subscribe to the long term deposit offer.

For "nr.employed", the probability of customers' subscription peaks when the quarterly number of employees reaches (5010, 5090). Before the quarterly number of employees reaches 5,010, the probability of customers' subscription has an increasing trend; while after the quarterly number of employees reaches 5,090, the probability of customers' subscription will decrease. The result suggests that the bank should involve no more than 5,090 employees quarterly. It implies that if more than 5,090 employees are involved in direct marketing, customers may feel bothered and will be unwilling to subscribe.

Figure 10. Partial Dependence on "cons.price.idx", "cons.conf.idx", & "nr.employed" (for y=1)



Conclusions

Random Forest outperforms others in predicting customers' responses to direct marketing, and Neural Network and Classification Tree also do a good job in classification prediction accuracy, while the Logistic Model works the worst. It is pretty common to see that Random Forest always works well in terms of classification accuracy.

In addition, the Partial Dependence Plot generated from Random Forest suggests that the bank should pay attention to macro-economic situation indexes, and customers may react differently with different offer character and within different indexes ranges. From a micro-perspective, the bank should also notice that the appropriate number of employees in direct marketing is important. It is not necessary that the more employees, the more customers' subscriptions. Customers may feel bothered and will be unwilling to subscribe if the number of employees is beyond the appropriate level.

References

Elsalamony, H. (2013). Bank Direct Marketing Analysis of Data Mining Techniques. *International Journal of Computer Applications (0975 – 8887)*, Volume 85, No 7

Moro, Sérgio & Cortez, Paulo & Rita, Paulo. (2014). A Data-Driven Approach to Predict the Success of Bank Telemarketing. *Decision Support Systems.* 62. 10.1016/j.dss.2014.03.001.

Sing'oei L. & Wang J. (2013). Data Mining Framework for Direct Marketing: A Case Study of Bank Marketing. *IJCSI International Journal of Computer Science Issues*, Vol. 10, Issue 2, No 2, March 2013

Vajiramedhin C. & Suebsing A. (2014). Feature Selection with Data Balancing for Prediction of Bank Telemarketing. *Applied Mathematical Sciences*, Vol. 8, 2014, no. 114, 5667 – 5672. Retrived from http://dx.doi.org/10.12988/ams.2014.47222