

Applying Machine Learning to the Development of Prediction Models for Bank Deposit Subscription

Sipu Hou, California State University, East Bay, USA

Zongzhen Cai, California State University, East Bay, USA

Jiming Wu, California State University, East Bay, USA*

Hongwei Du, California State University, East Bay, USA

Peng Xie, California State University, East Bay, USA

ABSTRACT

It is not easy for banks to sell their term-deposit products to new clients because many factors will affect customers' purchasing decisions and because banks may have difficulties in identifying their target customers. To address this issue, the authors use different supervised machine learning algorithms to predict if a customer will subscribe a bank term deposit and then compare the performance of these prediction models. Specifically, the current paper employs these five algorithms: naïve bayes, decision tree, random forest, support vector machine, and neural network. This paper thus contributes to the artificial intelligence and big data field with an important evidence of the best performed model for predicting bank term deposit subscription.

KEYWORDS

Machine Learning, Naïve Bayes, Decision Tree, Random Forest, Support Vector Machine, Neural Network

INTRODUCTION

In the past few decades, banks have experienced some problems of selling their term-deposit products to new clients (Elsalamony, 2014). This is mainly due to the fact that it is difficult for banks to figure out who are their target customers and that many factors will affect a customer's decision to purchase a term deposit (Moro et al., 2014). Meanwhile, more and more customers start to complain about the irrelevant phone calls received from banks (Elsalamony, 2014). To address these business issues, banks start to leverage their huge customer data to gain insight into customer behavior and buying preferences, and thus to improve their marketing effectiveness (Moro et al., 2014). This strategic business initiative relies on artificial intelligence and Big Data technologies.

Artificial intelligence refers to using computer systems to perform tasks that normally require human intelligence and can be considered as a concept broader than machine learning, even though

DOI: 10.4018/IJBAN.288514

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

they are often used interchangeably (Bali et al., 2017). Nowadays, more and more organizations are employing artificial intelligence and Big Data to gain an edge over the rest of the market. There are a variety of machine learning based predictive techniques to be used to predict if customers will subscribe a term deposit. Many researchers consider machine learning as one of the most useful research methodologies to transform data into intelligent action and to capture the meaningful information and hidden patterns from the historic datasets (Kohavi & Provost, 1998).

In this paper, we use machine learning with R to develop prediction models to forecast whether customers will purchase bank term-deposit products. Five different machine learning algorithms have been employed: Naïve Bayes (NB), Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), and Neural Network (NN). A real dataset from a European bank has been used for this research. The dataset includes 45221 observations and 21 variables. The current study uses R to analyze the data and to examine the results. Below, we present the details about the design and implementation of this research.

RESEARCH QUESTIONS

It is always challenging for banking institutions to market to new potential clients and to retain them successfully for a long time. To reach out to profitable customers, banks often deploy mediums such as social media, digital media, customer service, and strategic partnerships. But would it be possible for a bank to accurately market to a specific location, community, and group of people? Fortunately, with the inception of machine learning technologies, banking institutions can leverage their data and analytics solutions to reach out to specific target clients, and to accurately and intelligently predict which customers are more likely to purchase financial products and services.

The objective of this study is to (1) explain how a banking institution can utilize its client data and machine learning technology to predict which client would subscribe to a bank term deposit, (2) use five different machine learning algorithms to build up five predictive models, and (3) compare these five predictive models and find out which algorithm is best suited for term deposit subscription prediction. By doing so, our aim is to address two main research questions: (1) whether client purchasing behavior is predictable and how accurate the prediction can be? and (2) which machine learning algorithm would be more effective in such a prediction task? This study thus helps IS researchers expand their baseline knowledge of different machine learning algorithms, better understand how to build various predictive models, and more effectively design and conduct research on financial client behavior.

RESEARCH METHODOLOGY

A time deposit or term deposit is a deposit with a specified period of maturity and earns interest (Kagan, 2020). In other words, it is a money deposit at banking institution that cannot be withdrawn for a specific term or period unless a penalty is paid (Kagan, 2020). Although banks have various outreach plans to raise the capitals, one key plan is through engaging in direct marketing campaigns to sell their deposit products to clients. Therefore, identifying potential clients is of key importance for any banking institution. With the development of artificial intelligence, identifying potential clients becomes more feasible and manageable. Machine learning, as a powerful form of artificial intelligence, is one of the most exciting technological developments in history. Machine learning based predictive techniques can help banks determine which customers are more likely to subscribe a term deposit. Therefore, it is worthwhile for researchers to analyze historic datasets by making a prediction model to forecast whether clients will subscribe a term deposit (Elsalamony, 2014).

From the World Wide Web, we found a dataset about direct marketing campaigns of a European bank. The dataset can be downloaded from the UCI Machine Learning Repository at <https://archive.ics.uci.edu/ml/datasets/bank+marketing>. This direct marketing campaign dataset was collected from 2008 to 2013 by making phone calls to contact clients to collect their information and by calling the same client again to assess if the client will subscribe a term deposit or not (Lantz, 2015).

DATA DESCRIPTION

This bank marketing dataset includes 45211 observations, each with one dependent variable and 20 independent variables. The dependent variable shows the result if clients will subscribe the term deposit or not and thus is binary, with two values of “yes” and “no”. The 20 independent variables are described in Table 1.

Table 1.

NO.	Name	Type and Values	Description
1	age	Numeric	
2	job	Categorical: ('admin.', 'blue-collar', 'entrepreneur', 'housemaid', 'management', 'retired', 'self-employed', 'services', 'student', 'technician', 'unemployed', 'unknown')	Type of job
3	marital	Categorical: ('divorced', 'married', 'single', 'unknown'); Note: 'divorced' means divorced or widowed.	Marital status
4	education	Categorical: ('basic.4y', 'basic.6y', 'basic.9y', 'high.school', 'illiterate', 'professional.course', 'university.degree', 'unknown')	
5	default	Categorical: ('no', 'yes', 'unknown')	Has credit in default?
6	housing	Categorical: ('no', 'yes', 'unknown')	Has housing loan?
7	loan	Categorical: ('no', 'yes', 'unknown')	Has personal loan?
8	contact	Categorical: ('cellular', 'telephone')	Contact communication type
9	month	Categorical: ('jan', 'feb', 'mar', ..., 'nov', 'dec')	Last contact month of year
10	day_of_week	Categorical: ('mon', 'tue', 'wed', 'thu', 'fri')	Last contact day of the week
11	duration	Numeric; in seconds	Last contact duration; note this attribute highly affects the output target (e.g., if duration=0 then y='no')
12	campaign	Numeric; includes last contact	Number of contacts performed during this campaign and for this client
13	pdays	Numeric; 999 means client was not previously contacted	Number of days that passed by after the client was last contacted from a previous campaign
14	previous	Numeric	Number of contacts performed before this campaign and for this client
15	poutcome	Categorical: ('failure', 'nonexistent', 'success')	Outcome of the previous marketing campaign
16	emp.var.rate	Numeric	Employment variation rate - quarterly indicator
17	cons.price.idx	Numeric	Consumer price index - monthly indicator
18	cons.conf.idx	Numeric	Consumer confidence index - monthly indicator
19	euribor3m	Numeric	Euribor 3 month rate - daily indicator
20	nr.employed	Numeric	Number of employees - quarterly indicator

Data Preparation

Once we have gathered the data for this research, our next step is to prepare data for model training and evaluation. A key focus of this stage is to recognize missing data and then appropriately handle them. This is important because they may substantially bias the results. Before jumping to the methods of addressing missing data, we first need to know the reasons why data will be missing. Normally, there are three typical mechanisms causing missing data: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR). MCAR is quite straightforward and means what it says: the propensity for a data point to be missing is completely random (Little and Rubin 2002). In other words, whether a data point is missing has nothing to do with any missing or observed values in the dataset; the missing data are just a random subset of the data. MAR means that the propensity for a data point to be missing is not related to the missing data, but related to some of the observed data (Little and Rubin 2002). This means the missing is not completely at random, but is conditional on another variable. MNAR means that propensity for a data point to be missing varies for reasons that are unknown to us (Bland, 2015).

Two possible reasons for MNAR are that the missing value depends on the hypothetical value (e.g. individuals with high salaries usually do not want to reveal their incomes in surveys) or the missing value depends on the value of another variable (e.g., females generally don't want to reveal their ages and thus the missing value for age is related to the value of gender variable).

After we obtain a better understanding of the missing data, we can then choose an appropriate approach to handle them. For the first two mechanisms, it is safe to remove the data with missing values depending on the number of their occurrences, whereas for the third mechanism, removing missing data can create a bias in the model (Little and Rubin 2002).

For the current bank deposit dataset, we found that the missing value of “unknown” exists in several variables that are *job*, *marital*, *education*, *default*, *housing*, *loan*. They all belong to categorical data type. For the *default* variable, there are only three “yes” observations, but the number of “unknown” observations is much larger, with a total of 8,598. Because of this extremely small number of “yes” observations, we can't impute *default* values to replace “unknown”. Therefore, we decided to keep “unknown” for *default* as a new category and use it in our following data analysis. For the other variables with the missing value of “unknown”, it is safe to exclude them.

After the removal, now the dataset has 38245 observations, which is split into training data and testing data. We use stratified sampling to split the data, so that the distribution of the outcome within training and testing datasets is preserved. We split the data with the ratio of 66% (25243) for training the model and 34% (or 13002) for testing the model. Now, the data is clean and ready for the modeling phase.

BUILDING THE MODELS

(1) Naïve Bayes

The Naive Bayes classifier is an easy and simple probabilistic classifier based on Bayes' theorem with the independence assumptions between predictors (Elsalamony, 2014). It is considered as “naive” because it is created to simplify the computation involved (Han & Kamber, 2006). The Naive Bayes classifier can predict class membership probabilities, that is, the probability that a given sample belongs to a particular class. In the statistics and data/computer science literature, Naive Bayes models are also known under a variety of names, including Simple Bayes and Independence Bayes (Hand & Yu, 2001). All these names reference the use of Bayes' theorem in the decision rules for classification.

A Naive Bayes model is easy to build and requires no complicated iterative parameter estimation, which makes it particularly useful for very large datasets (Marin & Robert, 2014). Using the class labels from the training dataset, a Naive Bayes algorithm learns the conditional attributes and then

calculates the probability of a class value, given the particular instance. Despite its simplicity, the Naive Bayes classifier often predicts the class value with the highest probability and is thus widely used in multi-label classification research problems like this study tackles. Moreover, many researchers and statisticians believe that the Naive Bayes classifier can often outperform other more sophisticated classification methods (Marin & Robert, 2014).

We start with loading the necessary libraries and build an initial model with all features and then evaluate the model on the testing data. The results are shown in Table 2.

The initial model yields quite decent results, especially for the prediction for 'no'. We see that the misclassified prediction rate for 'no' is 5.21% ($570/(570+10375)$), which is quite small. However, the misclassified prediction rate for 'yes' is 57.37% ($1180/(1180+877)$). The overall accuracy rate is 86.54% and the sensitivity rate is 89.79% ($10375/(10375+1180)$). These two rates are both excellent.

Sensitivity analysis can be defined as the study of how the variation in the output of a predictive model can be apportioned, qualitatively or quantitatively, to different sources of input variation (Saltelli et al., 2004). In other words, it can inform researchers how a given predictive model depends upon the information fed into it. It is widely accepted that sensitivity analysis plays a crucial role in assessing the robustness of the results or findings of a predictive model (Saltelli et al., 2004). For this reason, the current study also employs sensitivity analysis to analyze and identify the effectiveness of our models.

(2) Decision Tree

Another important machine learning technique is the Decision Tree approach. Decision Tree is a type of supervised learning algorithm in the sense that users can explain what the input is and what the corresponding output is in the training dataset. Decision Tree provides powerful techniques for both classification and prediction, and works for both categorical and continuous input and output variables (Pang & Gong, 2009). During the Decision Tree analysis process, the data is continuously split based on certain parameters. The goal of using a Decision Tree is to build up a training model

Table 2.

```
> confusionMatrix(naiveprediction, mytest$y)
Confusion Matrix and Statistics

              Reference
Prediction    no    yes
no      10375  570
yes     1180   877

              Accuracy : 0.8654
              95% CI   : (0.8594, 0.8712)
No Information Rate : 0.8887
P-Value [Acc > NIR] : 1

              Kappa : 0.4255
McNemar's Test P-Value : <2e-16

              Sensitivity : 0.8979
              Specificity : 0.6061
              Pos Pred Value : 0.9479
              Neg Pred Value : 0.4263
              Prevalence : 0.8887
              Detection Rate : 0.7980
              Detection Prevalence : 0.8418
              Balanced Accuracy : 0.7520

              'Positive' Class : no
```

that can predict the class or value of the target variable by learning simple decision rules inferred from the training data. Figure 1 shows an example of Decision Tree (Analytics Vidhya, 2016).

A Decision Tree can be seen as a flowchart with two entities, namely decision nodes and leaves. Decision nodes include non-leaf nodes and leaf nodes. Each non-leaf node represents a conditional test on one of the parameters and each leaf node represents a class label where predictions or classifications are made for the final outcome. Each leaf represents an outcome of the conditional test. Paths from the root to all the leaf nodes give us all the decision rules. Decision Trees are not difficult to represent, construct, and understand, but they have one major drawback, that is, they are very prone to overfitting (Bali et al., 2017). Because of this reason, Decision Tree models usually do not generalize well. In the current research, we will follow the similar analytics modeling approaches in the literature to build a model based on Decision Trees (Pang & Gong, 2009).

We start with loading all the necessary libraries and test data into machine learning with R system. We then build a decision tree model with all the variables using training data and finally, we predict the classification value and evaluate the model with the test data. The results are also excellent. As shown in the following decision tree output, the overall accuracy rate is 91.82% and sensitivity rate is 96.55%. The misclassified prediction rate for 'no' is 5.94% ($664/(11156+664)$), which is quite small. However, the misclassified prediction rate for 'yes' is much larger, with a value of 33.76% ($399/(399+783)$).

(3) Random Forest

To greatly improve the predictive ability of a Decision Tree, we can create many trees and combine their results. The random forest technique indeed does this for us. The logic of the algorithm can be briefly described as follows. At any point in time, each tree in the ensemble of decision trees is created from a bootstrap sample, which is basically sampling with replacement (Bali & Sarkar, 2016). This sampling is mainly conducted on the training dataset. During the creation of a decision tree, the split that was earlier being recognized as the best split for all the variables is not valid anymore. Now the best split is always chosen from a randomly selected subset of the variables each time. The adoption of randomness into the model will slightly increase the bias of the model but will also significantly decrease the variance of the model (Bali & Sarkar, 2016). Such a decrease may prevent the overfitting

Figure 1.

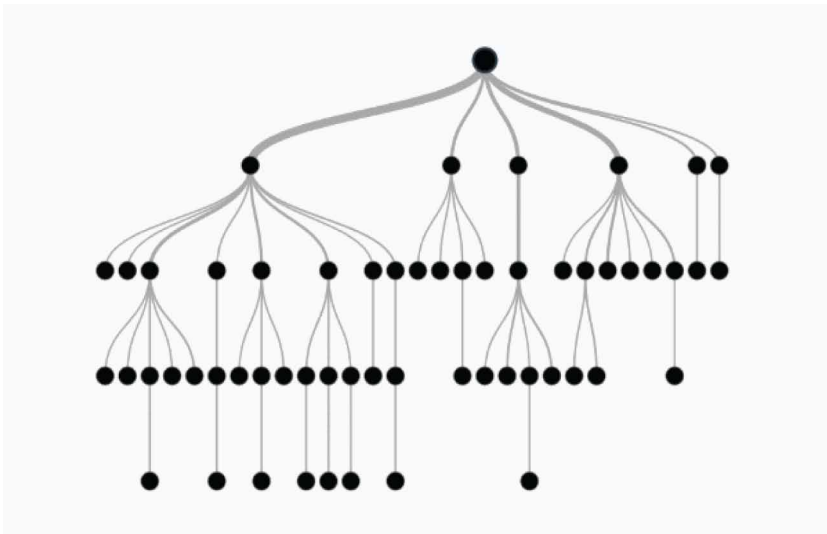


Table 3

```
> confusionMatrix(fPredictions, mytest$y)
Confusion Matrix and Statistics

          Reference
Prediction no  yes
no      11156  664
yes      399   783

          Accuracy : 0.9182
          95% CI   : (0.9134, 0.9229)
    No Information Rate : 0.8887
    P-Value [Acc > NIR] : < 2.2e-16

          Kappa   : 0.5507
    Mcnemar's Test P-Value : 5.622e-16

          Sensitivity : 0.9655
          Specificity : 0.5411
    Pos Pred Value   : 0.9438
    Neg Pred Value   : 0.6624
          Prevalence : 0.8887
    Detection Rate   : 0.8580
    Detection Prevalence : 0.9091
    Balanced Accuracy : 0.7533

    'Positive' Class : no
```

of a model, which is a serious concern for Decision Trees. Therefore, this process will overall yield much better general Decision Tree models (Bali & Sarkar, 2016).

We start our analytical process by loading the necessary libraries into the R system and then build the random forest training model with all the 20 variables. Using the importance() function, we can find the importance value for each variable in the Random Forest algorithm as shown in Table 4. This gives us the variable importance plot in Figure 2.

Next, we perform the predictions on the test data and obtain the results as follows. The results are also very good, showing that the overall accuracy rate is 91.89% and the sensitivity rate is 96.58%.

(4) Support Vector Machine (SVM)

The Support Vector Machine is also a supervised machine learning algorithm mainly used for classification, but can also be used for prediction. The main idea of this algorithm is that based on the training data, the SVM model tries to find the optimal hyperplane that can be used to classify new data points (Scholkopf & Smola, 2018). When there are only two dimensions, the hyperplane is a simple line. Therefore, SVM has several different kernels that can be used to find linear or non-linear optimal hyperplane. These include, but not limited to, regular Linear kernel, Polynomial kernel, Sigmoid kernel, and Gaussian Radial Basis Function (RBF) kernel. They are based on the equations shown in Table 6.

The main principle behind these non-linear kernel functions is that even if linear separation is not possible in the original feature space, they enable the separation to happen in a higher dimensional transformed feature space, in which we can use a hyperplane to separate the classes (Bali & Sarkar, 2016). An important thing to note is that unfortunately, the curse of dimensionality applies here. Because we may end up working with higher dimensional feature spaces, the model generalization error will increase and the predictive power of the model will decrease (Bali & Sarkar, 2016). But if the dataset is large enough, it will still perform well. Here we are using the RBF kernel, also known as the radial basis function, to build up the SVM model.

Table 4

```
> importance
rf variable importance

only 20 most important variables shown (out of 53)

duration                Overall
euribor3m              1614.83
nr.employed            556.34
age                   527.66
campaign              435.66
pdays               184.02
cons.conf.idx         141.51
cons.price.idx        105.98
housingyes            78.78
poutcomesuccess      73.33
educationuniversity.degree 72.52
day_of_weekwed        61.62
day_of_weekmon        55.49
loanyes              54.12
day_of_weekthu        53.54
jobtechnician         52.89
maritalmarried        52.78
day_of_weektue        51.77
day_of_weekthu        51.37
educationhigh.school  50.26
previous              50.13
> |
```

Figure 2.

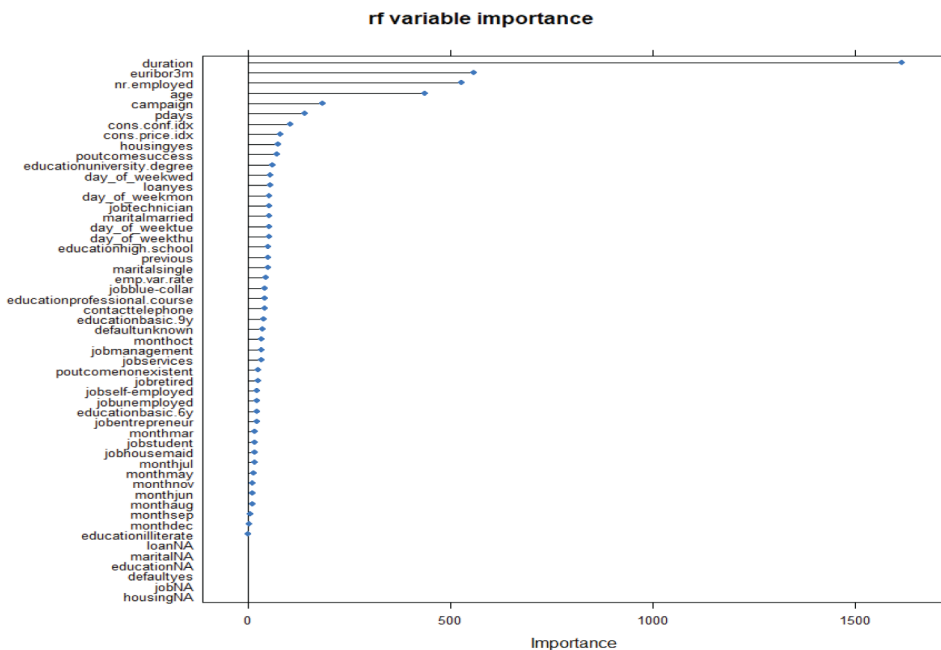


Table 5

```
> confusionMatrix(ranpredictions, mytest$y)
Confusion Matrix and Statistics

          Reference
Prediction no  yes
no      11160  660
yes      395   787

      Accuracy : 0.9189
      95% CI   : (0.914, 0.9235)
No Information Rate : 0.8887
P-Value [Acc > NIR] : < 2.2e-16

      Kappa   : 0.5541
McNemar's Test P-Value : 4.368e-16

      Sensitivity : 0.9658
      Specificity : 0.5439
      Pos Pred Value : 0.9442
      Neg Pred Value : 0.6658
      Prevalence : 0.8887
      Detection Rate : 0.8583
      Detection Prevalence : 0.9091
      Balanced Accuracy : 0.7548

      'Positive' Class : no
```

Table 6. SVM Kernels

$$\begin{aligned} \text{Linear kernel: } K(\vec{x}_i, \vec{x}_j) &= \vec{x}_i \cdot \vec{x}_j \\ \text{Polynomial kernel: } K(\vec{x}_i, \vec{x}_j) &= (\vec{x}_i \cdot \vec{x}_j + 1)^d \\ \text{Sigmoid kernel: } K(\vec{x}_i, \vec{x}_j) &= \tan h(\kappa \vec{x}_i \cdot \vec{x}_j - \delta) \\ \text{Gaussian RBF kernel: } K(\vec{x}_i, \vec{x}_j) &= e^{-\frac{\|\vec{x}_i - \vec{x}_j\|^2}{2\sigma^2}} \end{aligned}$$

We build the model with all the variable and Figure 3 shows their importance in the SVM model. As demonstrated in Table 7, the overall accuracy rate is as good as that of the previous two models, with a value of 91.72% and the sensitivity rate is 97.21%.

(5) Neural Network

As one of the most important machine learning algorithms, neural networks have their roots in the artificial intelligence and thus often applied to classification problems like we are solving here in this paper (Elsalamony, 2014). A neural network mimics the human brain’s mechanism to process data and solve problems. Just as humans apply knowledge gained from past experience to new problems, a neural network takes previously solved examples to build a network of “neurons” that makes new forecasts, decisions, and classifications (Al-Osaimy, 1998). A neural network recognizes patterns in

Figure 3.

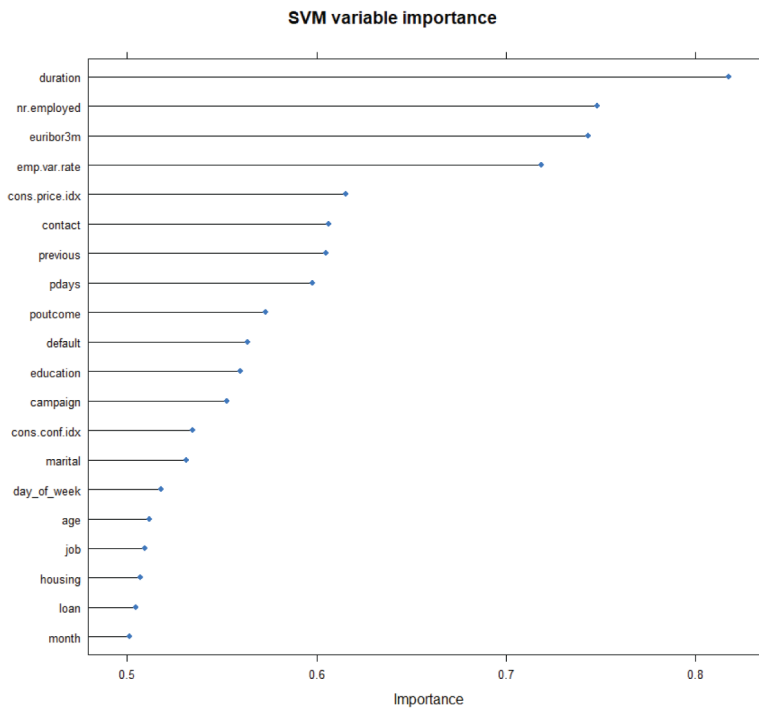


Table 7.

```
> confusionMatrix(svmpredictions, mytest$y)
Confusion Matrix and Statistics

          Reference
Prediction  no  yes
no      11233  755
yes      322   692

      Accuracy : 0.9172
      95% CI   : (0.9123, 0.9218)
No Information Rate : 0.8887
P-Value [Acc > NIR] : < 2.2e-16

      Kappa   : 0.5182
McNemar's Test P-Value : < 2.2e-16

      Sensitivity : 0.9721
      Specificity : 0.4782
      Pos Pred Value : 0.9370
      Neg Pred Value : 0.6824
      Prevalence : 0.8887
      Detection Rate : 0.8639
      Detection Prevalence : 0.9220
      Balanced Accuracy : 0.7252

      'Positive' Class : no
```

training datasets, learns from these patterns, and then develops the ability to accurately classify new patterns or to correctly make forecasts and predictions.

The basic building block of a neural network is a single simulated neuron that processes input data to produce an output. Independent neurons are of little use until they are all interconnected with each other in a network of neurons (Al-Osaimy, 1998). The neuron network usually processes a large number of inputs from the outside world to produce one or many outputs: the network's predictions or classifications. All the neurons are connected to each other by weights that are applied to values passed from one neuron to the next.

As we have done for the other four models, we build a neural network model based on all the variables. As shown in the confusion matrix output below, the overall accuracy rate has decreased to 88.86%, whereas the sensitivity has gone up to 99.99%. This means that the current model is very aggressive and just predicts every client will not subscribe a term deposit. This indicates the model may have some issues and can be further improved.

MODEL COMPARISON AND DISCUSSIONS

In this research, we have explored five machine learning techniques and built five models to predict if the client will subscribe a term deposit. All of the results are summarized in Table 9. We find that in general, all these five models performed very well and the prediction accuracy rates for DT, RF, and SVM are very close to each other. In terms of sensitivity, NN is the best, whereas NB has the lowest score.

Sensitivity analysis plays a critical role not only in assessing the robustness of a predictive model, but also in providing valuable information regarding the structure of the model and its reliance upon the input variables (Saltelli et al., 2000). The sensitivity of an input variable indicates the effect that a variation of that input will have on the output. In general, an input variable with higher sensitivity will cause a greater variation of the output and vice versa (Saltelli et al., 2000). The excellent sensitivity analysis results in Table 9 provide strong evidence for the stability and reliability of the five machine learning models developed in this study.

Table 8.

```
> confusionMatrix(nnpredictions, mytest$y)
Confusion Matrix and Statistics

          Reference
Prediction no  yes
no      11554 1447
yes         1    0

      Accuracy : 0.8886
      95% CI   : (0.8831, 0.894)
No Information Rate : 0.8887
P-Value [Acc > NIR] : 0.5181

      Kappa : -2e-04
McNemar's Test P-Value : <2e-16

      Sensitivity : 0.9999
      Specificity : 0.0000
      Pos Pred Value : 0.8887
      Neg Pred Value : 0.0000
      Prevalence : 0.8887
      Detection Rate : 0.8886
      Detection Prevalence : 0.9999
      Balanced Accuracy : 0.5000

      'Positive' Class : no
```

Table 9.

Methods	Accuracy	Sensitivity
Naive Byes	86.54%	89.79%
Decision Tree	91.79%	96.55%
Random Forest	91.89%	96.58%
Support Vector Machine	91.72%	97.21%
Neural Network	88.86%	99.99%

After we identified the best model, we take a step further to find out which independent variables are the most salient in predicting the dependent variable, that is, whether or not a client will subscribe the term deposit. According to Figure 2 and Figure 3, the most salient independent variables are *duration*, *nr.employed*, *euribor3m*, *age*, and *emp.var.rate*. *Duration* has a significant impact on clients' decision because the longer the last phone conversation is, the higher interest the client will have in the term deposit. The variable, *nr.employed*, refers to the number of employees in a bank and is also an influential variable on clients' intention to subscribe the term deposit. This may well be that if a bank has more employees, the bank will be more likely considered as influential and prestigious and thus attract more customers to purchase its products (Chen et al., 2014). As an indicator of the average interbank interest rates in Eurozone, *euribor3m* is a salient predictor because when the interest rate is high, people will be more willing to spend their money on financial products. *Age* also has positive impact on clients' saying "yes." This can be due to the fact that middle-aged clients are more likely to subscribe the term deposit than young bank clients. The variable, *emp.var.rate*, refers to employment variation rate and is an indicator of the macroeconomy (Chen et al., 2014). It has a negative impact on clients' purchasing decision, which means large employment rate change makes clients less likely to subscribe a term deposit. This makes sense and suggests that a stable employment rate indicates a stable economic environment, in which bank clients are more confident to make financial investments (Chen et al., 2014). This study thus contributes not only to the machine learning literature, but also to a better understanding of bank clients' behavior of purchasing financial products and services.

CONCLUSION

After understanding our data and exploring the several important models to predict if the client will subscribe a term deposit, we have arrived at the conclusion that all these models have done a good job and there is no one model that is all the best among them. One limitation of this research is that we do not further improve the models by using some techniques such as attribute selection. This may suggest that more advanced studies on these five machine learning algorithms are necessary in the future.

REFERENCES

- Al-Osaimy, M. H. (1998). A neural networks system for predicting Islamic banks performance. *Journal of King Abdulaziz University: Economics and Administration*, 11, 33–46.
- Analytics Vidhya. (2016). *Tree based algorithms: A complete tutorial from scratch (in R & Python)*. <https://www.analyticsvidhya.com/blog/2016/04/tree-based-algorithms-complete-tutorial-scratch-in-python/>
- Bali, R., & Sarkar, D. (2016). *R machine learning by example*. Packt Publishing.
- Bali, R., Sarkar, D., & Lantz, B. (2017). *R: Unleash machine learning techniques: Smarter data analytics*. Packt Publishing.
- Bland, M. (2015). *An Introduction to Medical Statistics* (3rd ed.). Oxford University Press.
- Chen, J., Han, Y., Hu, Z., Lu, Y., & Sun, M. (2014). *Who will subscribe a term deposit?* Columbia University, <http://www.columbia.edu/~jc4133/ADA-Project.pdf>
- Elsalamony, H. A. (2014). Bank direct marketing analysis of data mining techniques. *International Journal of Computers and Applications*, 85, 12–22.
- Hand, J. W., & Kamber, M. (2006). *Data mining concepts and techniques* (2nd ed.). Morgan Kaufmann Publishers.
- Hand, D. J., & Yu, K. (2001). Idiot's Bayes - not so stupid after all? *International Statistical Review*, 69, 385–399.
- Kagan, J. (2020). *Time Deposit*. Investopedia.
- Kohavi, R., & Provost, F. (1998). Glossary of terms. *Machine Learning*, 30, 271–274.
- Lantz, B. (2015). *Machine Learning with R*.
- Little, R. J. A., & Rubin, D. B. (2002). *Statistical analysis with missing data* (2nd ed.). Wiley.
- Marin, J., & Robert, C. (2014). *Bayesian Essentials with R*. Springer.
- Moro, S., Cortez, P., & Rita, P. (2014). A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62, 22–31.
- Saltelli, A., Chan, K., & Scott, E. M. (Eds.). (2000). *Sensitivity Analysis*. John Wiley & Sons.
- Saltelli, A., Tarantola, S., Campolongo, F., & Ratto, M. (2004). *Sensitivity analysis in practice: A guide to assessing scientific models*. John Wiley and Sons.
- Scholkopf, B., & Smola, A. J. (2018). *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. The MIT Press.

Sipu Hou was a Master's student in Business Analytics at California State University East Bay. She is enthusiastic about utilizing data to make data-driven decisions and has hands-on experience in data modeling, data-based decision-making, and statistical analysis. She has solid technical knowledge in data mining techniques including dimension reduction, multiple linear regression, KNN, classification and regression trees, random forest, and clustering.

Zongzhen Cai has a background in business analysis and finance analysis. He is a data analyst at center for disease analysis foundation. He is proficient in data analyst with knowledge of Python, SQL, R and VBA. He is skilled in data mining, machine learning algorithm, statistical analysis of large dataset. He works with the modeling team to analyze data to develop disease modeling strategy and validate model analysis results to support epidemiology team. He has focus on improving the model while developing customized tools for data extraction, transforming and loading, and visualization. He graduated from California State University, East Bay with Master of Science in business analytics.

Jiming Wu is a Full Professor and Interim Associate Dean in the College of Business and Economics at California State University, East Bay. He received his B.S. from Shanghai Jiao Tong University, M.S. from Texas Tech University, and Ph.D. from the University of Kentucky. His research interests include Big Data analytics, IT adoption and acceptance, and computer and network security. His work has appeared in MIS Quarterly, Journal of the Association for Information Systems, European Journal of Information Systems, Information & Management, Decision Support Systems, and elsewhere.

Hongwei Du is the Chair of the Department of Management, College of Business and Economics at California State University, East Bay. He holds a Ph.D. in Operations Research from Florida Institute of Technology, a M.S. in Computer Science from Bowling Green State University and a M.S. in System Engineering from Beijing Institute of Automation. His works have been published in the European Journal of Information Systems, the Journal of Information Technology Theory and Application, the California Journal of Operations Management, the International Journal of Innovation and Learning, the International Journal of Intercultural Information Management, the International Journal of Information and Decision Science, the International Journal of Electronic Healthcare, the Journal of Economic Studies, the Journal of Supply Chain and Operations Management, and Financial Innovation.

Peng Xie is an assistant professor at the College of Business, California State University, East Bay. He received his Ph.D. degree in Business Administration from the Scheller School of Business, Georgia Institute of Technology. His research interest includes social media analytics, cryptocurrency, text mining, and financial technology. His research has appeared in top-quality journals such as Journal of Management Information Systems and Information System Frontiers. He has also been serving as Associate Editor in top IS conferences such as ICIS2021.