|  |
| --- |
| **Bank Marketing Campaign Classification** |

**Group 8**

Marcus Chan Hua Rui

Huang Cheng

Catur Indra Pratisto

**Abstract**

Using dataset from Portuguese banking institution from May 2008 to November 2010, we applied various machine learning classification methods to predict the effectiveness of the bank direct campaigns (by phone calls) to get potential customers to subscribe to term deposit. By comparing Type-II error (False Negative Rate), we found that Random Forest yielded best prediction.

# Introduction

Banks regularly conduct direct marketing campaigns to get the potential customers to subscribe to certain products. It is therefore important to analyse the dataset to predict the effectiveness of direct marketing campaign.

# Objective

The classification goal is to predict if the direct marketing campaign is effective to get the potential customers will subscribe to a bank product.

For this project, the direct marketing campaign is done by phone calls and the bank product is term deposit.

# Dataset

All the analysis is done in R, files saved as R Markdown (.Rmd).

This dataset is based on "Bank Marketing" UCI dataset ‘bank-additional-full.csv’ containing 41188 number of instances. There are 20 + output attribute, the binary classification goal is to predict if the client will subscribe a bank term deposit (variable y).

Detail descriptions of each variables can be found in Appendix.

# Methodologies

There are several machine learning techniques employed for comparison :

* K-nearest neighbors (KNN)
* Logistic Regression
* Decision tree
* Random Forest
* Gradient Boosting
* Adaptive Boosting (AdaBoost)
* Extreme Gradient Boosting (XGboost)
* Support Vector Machine (SVM) with linear kernel
* Support Vector Machine (SVM) with polynomial kernel
* Support Vector Machine (SVM) with Radial Basis Function (RBF) kernel

# Preprocessing and Exploratory Analysis

## 5.1 Preprocessing

This dataset comprises of a significant amount of missing data which was indicated by “unknown” values. For instance, from the 41188 data points in the “default” column, there were 8597 “unknown” values, 1731 in “education”, 990 in “housing” and “loan”, 330 in “job”, and 80 in “marital”. The details of how we deal with the missing values are shown in the next section

We did further data pre-processing to deal with the missing values as well as to reduce the amount of noise in our dataset. As our dataset was hugely imbalanced with 88% of respondents saying “no” and only 12% saying “yes”, we also did oversampling, which helped to reduce the type II error that we were interested in as it helped our model better generalise for the “yes” values we wanted to characterise.

## 5.2 Exploratory Analysis

In the “default” column, the fields filled with “unknown” values had 32588 “no”, 8597 “unknown”, and 3 “yes”. Inside “default”’s “no”s, there were 87.1% that eventually said no to the bank for the marketing campaign, which is lower than 94.8% in the respondents that had unknown values. As all the yes in default replied no to the marketing campaign, I will recode the unknown and yes values to be the same in the dataset.

For education’s unknown values, I also assessed the behaviour of the unknown values, finding that they are most similar to that of university students, as can be seen from our code, and therefore recoded them to be university students as well.

The “housing” and “loan” columns were found to be statistically insignificant, from a chi-squared analysis and were dropped from the dataset to reduce noise in the dataset, which seemed to improve the results.

Unknown values in “job” was found to have distinct composition of result to the marketing campaign, and hence were relabeled as unconventional.

From reading the description of some of the fields, we found that poutcome and previous seemed to be characterising similar information. Hence it seemed to be a good idea to merge this fields to output a single column. Failures were given a weight of 0.5, non-existent = 0.2, and success given 1. These weights were then used to multiply against “previous”+1. I added 1 as I did not want to multiple any values with 0. I then dropped the 2 original columns.

Age in this dataset was particularly interesting. We noticed that the proportion of people that said yes to the marketing campaign decreased with age, except that trend became significantly different after 60, where many began to reply yes and the trend with age seemed random. I recatagorised age into 7 values, for whichever age group the individual was in to reduce noise, and those over the age of 51 would be in one group.

The histograms and other exploratory analysis visualisations may be found in the appendix.

# Findings

Because of the highly imbalance data (only 12% said Yes to term deposit), any classification models will classify label as ‘No’ so much more than ‘Yes’, therefore resulting in very good accuracy or low classification error rate.

However, looking more closely to the problem, Bank would not want to miscategorise potential customers that will say ‘Yes’ as ‘No’. In other words, Bank are more likely to be concerned on reducing the number of **False Negative** – the number of customers identified as saying ‘No’ but in fact would have said ‘Yes’.

Therefore, in order to choose the best prediction models, the False Negative Rate or type-II errors are compared instead of misclassification accuracy rate.

In addition to looking at type-II errors, we also want to know if we can reduce the **False Positive** – the number of customers identified as saying ‘Yes’ but in fact would have said ‘No’.

**Receiver Operating Characteristic (ROC)** curves are created to illustrate both error types for all possible probability of prediction occurrence (threshold). The x-axis on ROC curve shows the **False Positive Rate** / type-I error and the y-axis shows the **True Positive Rate** / 1 – type II error (also called recall or sensitivity).

For binary classification like this project, the prediction is ‘Yes’ if the probability is more than 0.5 (threshold = 0.5). Since Bank might be more concerned of not missing customers that will say ‘Yes’, we can reduce the threshold from 0.5 to smaller number like 0.3 or 0.2.

As the threshold is reduced, the False Negative Rate/type-II error will be reduced (True Positive Rate in y-axis increases). However, at the same time, False Positive Rate / type-I error in x-axis will increase.

So which threshold is the best and should be chosen?

Unfortunately, there is no easy answer to this.

Domain knowledge is required to choose the threshold. For example: Bank needs to calculate the cost of performing more marketing campaign for more potential customer and compare it with the actual profit for each actual term deposit signup.

In addition to showing the trade-off between type-I and type-II errors, ROC curves can also be used to indicate the overall performance of classification over all possible thresholds. This is given by Area Under ROC Curve or AUC

## 5.1 Classification Results

The following table summarizes classification results:

Table 5-1: Classification Result

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Misclass error** | **Type-II error** | **Precision** | **Recall** |
| KNN | 0.0920 | 0.5240 | 0.6255 | 0.4760 |
| Logistic regression | 0.0893 | 0.5827 | 0.6730 | 0.4173 |
| Decision tree | 0.0932 | 0.6531 | 0.6757 | 0.3469 |
| **Random forest** | 0.0827 | **0.4482** | 0.6645 | 0.5518 |
| Gradient boosting | 0.0819 | 0.4578 | 0.6737 | 0.5422 |
| Adaboost | 0.0895 | 0.4632 | 0.6241 | 0.5368 |
| XGBoost | 0.0816 | 0.4568 | 0.676 | 0.5432 |
| SVM (linear kernel) | 0.0972 | 0.6894 | 0.6525 | 0.3106 |
| SVM (polynomial kernel) | 0.0944 | 0.7375 | 0.7387 | 0.2625 |
| SVM (RBF kernel) | 0.0915 | 0.6553 | 0.6976 | 0.3447 |

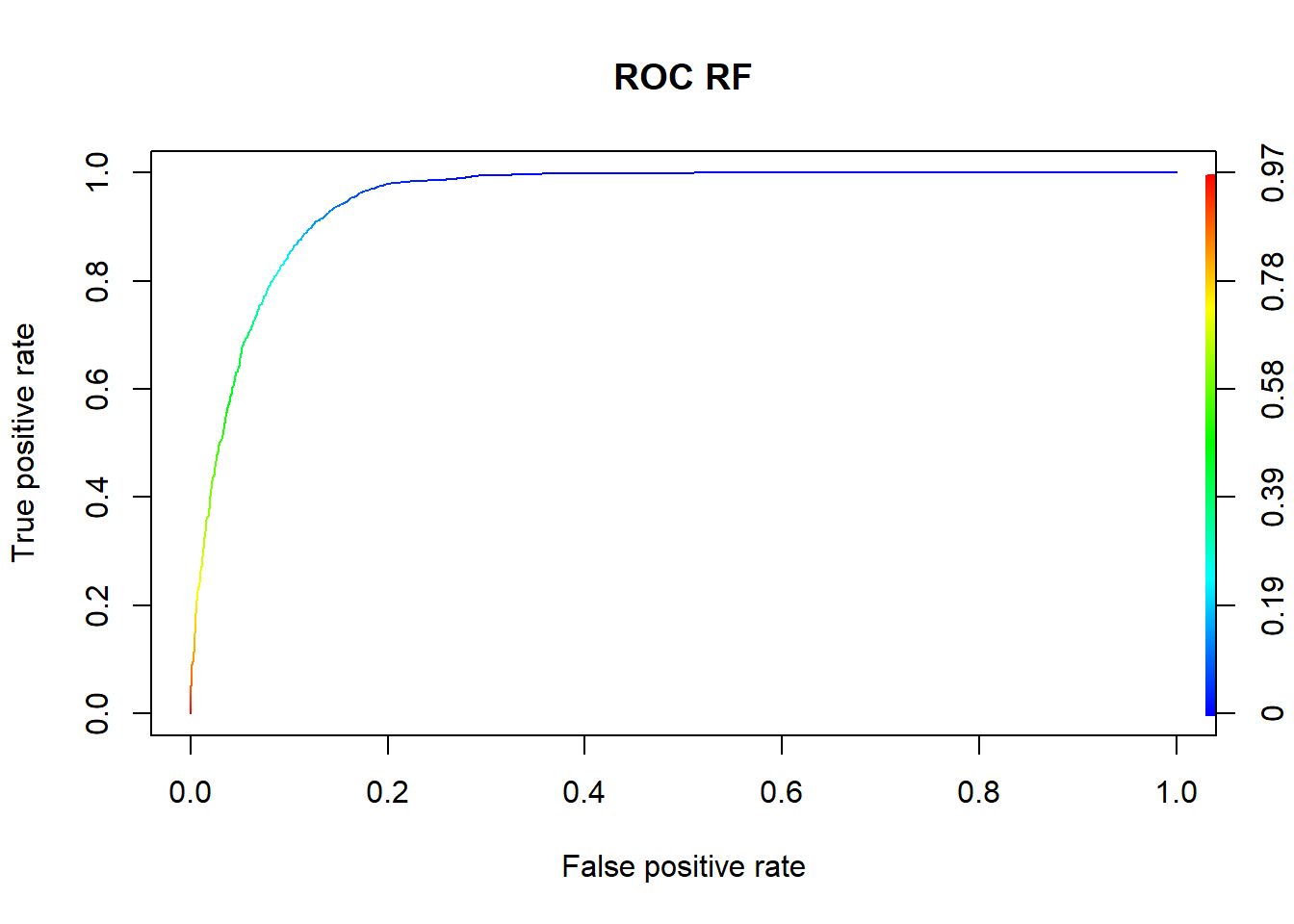


Figure 5-1: Random Forest ROC Curve (**AUC = 0.951**)

From the table, Random Forest has the best classification result (lowest type-II error).

## 5.2. Classification with Scaled Numerical Variables Results

The following table summarizes classification results:

Table 5-2: Classification Results (Scaled Numerical Variables)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Misclass error** | **Type-II error** | **Precision** | **Recall** |
| KNN | 0.0906 | 0.5560 | 0.6490 | 0.4440 |
| Logistic regression | 0.0893 | 0.5827 | 0.6730 | 0.4173 |
| Decision tree | 0.0932 | 0.6531 | 0.6757 | 0.3469 |
| Random forest | 0.0839 | **0.4536** | 0.6581 | 0.5464 |
| Gradient boosting | 0.0819 | 0.4578 | 0.6737 | 0.5422 |
| Adaboost | 0.0896 | 0.4642 | 0.6236 | 0.5358 |
| XGBoost | 0.0816 | 0.4568 | 0.6760 | 0.5432 |
| SVM (linear kernel) | 0.0972 | 0.6894 | 0.6525 | 0.3106 |
| SVM (polynomial kernel) | 0.0944 | 0.7375 | 0.7387 | 0.2625 |
| SVM (RBF kernel) | 0.0915 | 0.6553 | 0.6976 | 0.3447 |

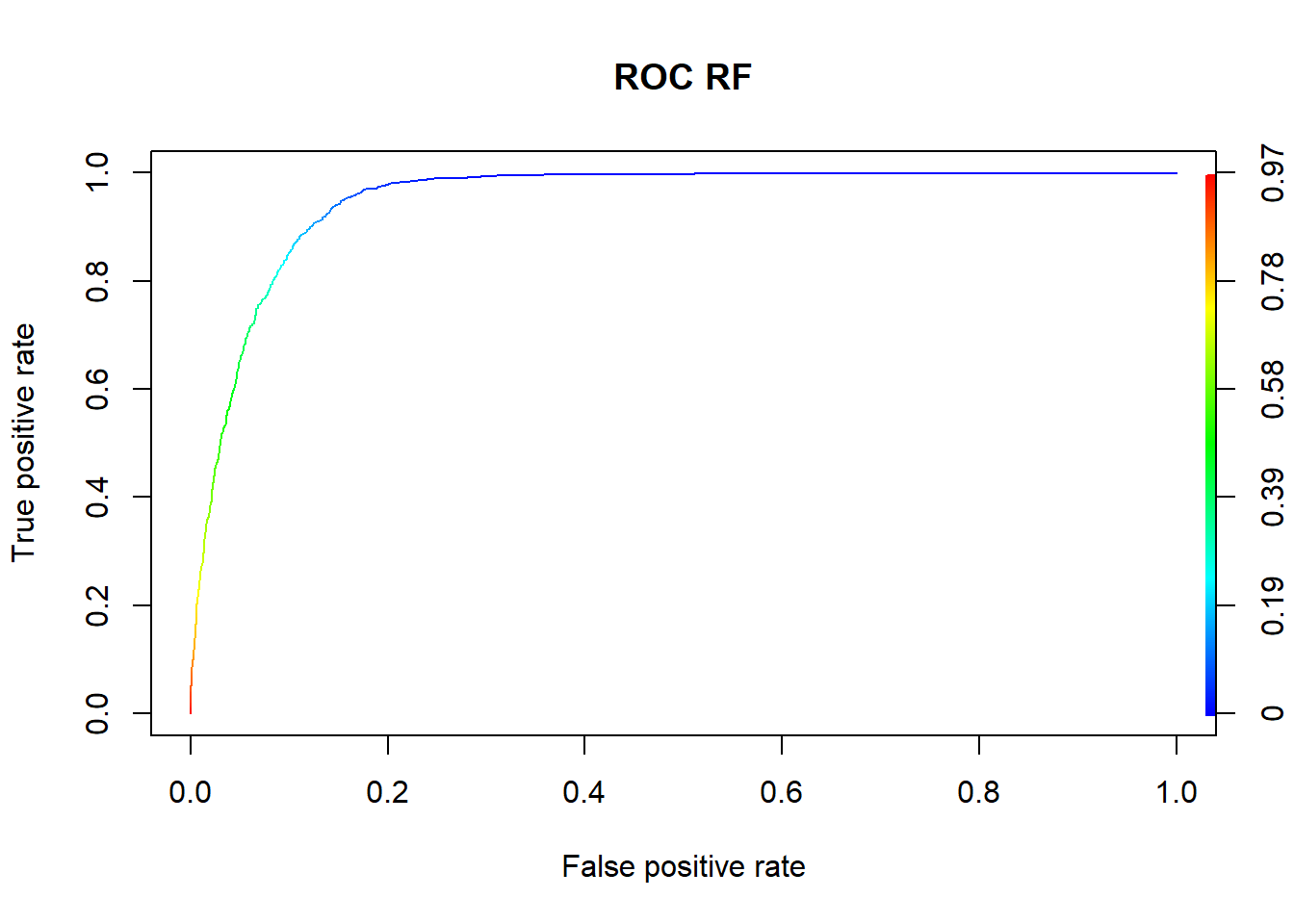


Figure 5-2: Random Forest ROC Curve (**AUC = 0.951**)

There is slight improvement / reduction on type-II errors, but generally scaling the numerical variables do not have any significant effect to the classification results.

## 5.3 Classification with Oversampling Variables Results

The following table summarizes classification results:

Table 5-3: Classification Results (with Oversampling)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Misclass error** | **Type-II error** | **Precision** | **Recall** |
| KNN | 0.0900 | 0.0106 | 0.8544 | 0.9894 |
| Logistic regression | 0.1249 | 0.1103 | 0.8654 | 0.8897 |
| Decision tree | 0.2064 | 0.0598 | 0.7281 | 0.9402 |
| Random forest | 0.0422 | **0.0015** | 0.9236 | 0.9985 |
| Gradient boosting | 0.1064 | 0.0655 | 0.8646 | 0.9345 |
| Adaboost | 0.0879 | 0.0489 | 0.8829 | 0.9511 |
| XGBoost | 0.0981 | 0.0504 | 0.8676 | 0.9496 |
| SVM (linear kernel) | 0.1187 | 0.0832 | 0.8569 | 0.9168 |
| SVM (polynomial kernel) | 0.1211 | 0.0891 | 0.8569 | 0.9109 |
| SVM (RBF kernel) | 0.1113 | 0.0598 | 0.8532 | 0.9402 |

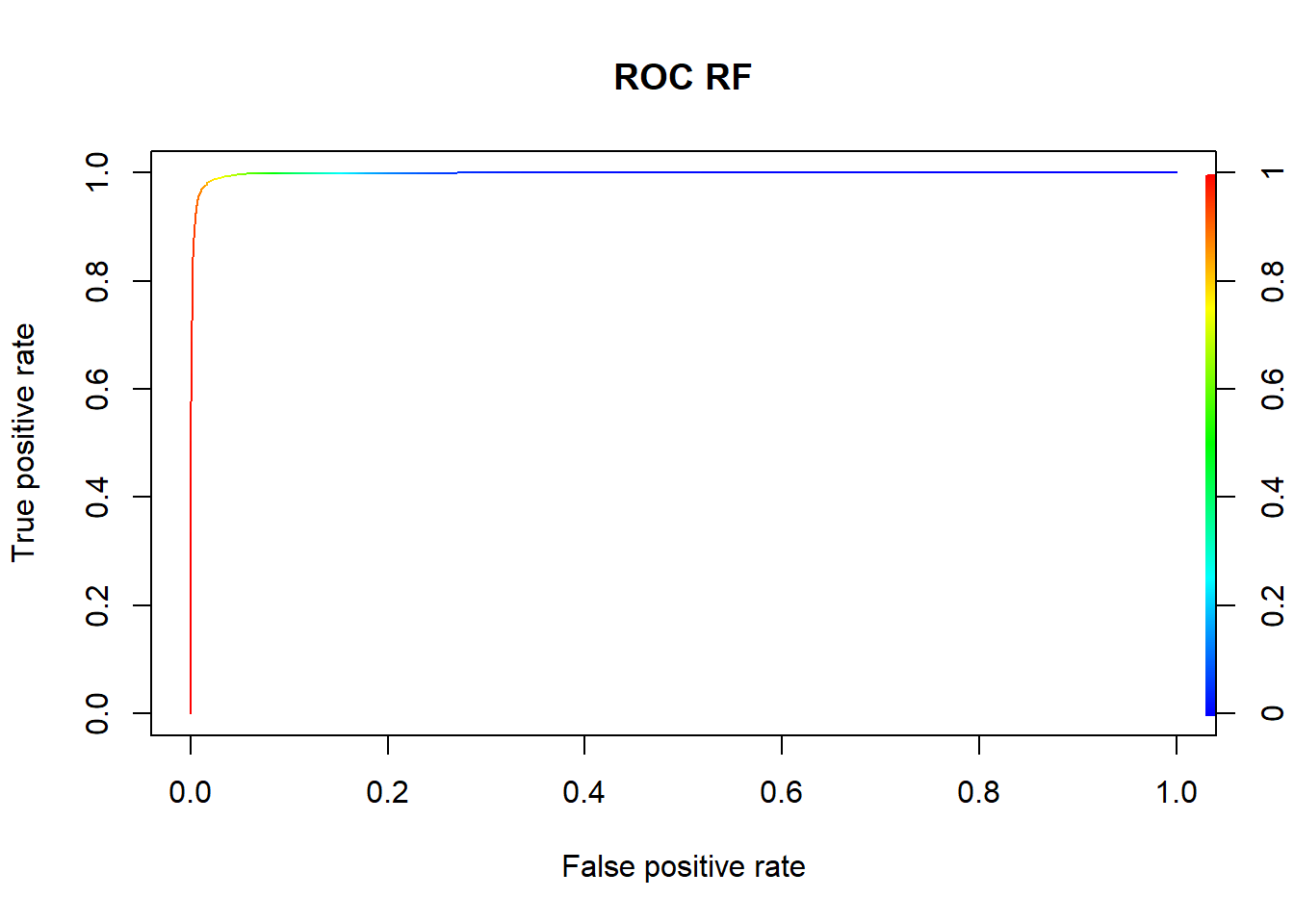


Figure 5-3: Random Forest ROC Curve (**AUC = 0.998**)

Oversampling significantly reduces type-II error in Random Forest.

## 5.4 Variables Importance

|  |  |  |
| --- | --- | --- |
|  |  |  |

Figure 5-4: Random Forest Variables Importance

Duration call is self-explanatory: the longer the call being made, the more likely that potential customers will subscribe to term deposit.

Euribor is short for Euro Interbank Offered Rate, which is the average interest rates at which a large panel of European banks borrow funds from one another (equivalent to SIBOR – Singapore Interbank Offered Rate in Singapore). The Euribor rates are the reference rates in the European money market, to be used as basis for the price and interest rates of all kinds of financial products like interest rate swaps, interest rate futures, saving accounts and mortgages.

Therefore, it makes sense that more potential customer will subscribe to term deposit when the interest rate (based on Euribor rate) is higher.

# 6 Lesson Learnt

[Ronnie]

## 6.1 Challenges

XXX

## 6.2 Future Improvement

XXX

**References**

References follow the acknowledgments. Use unnumbered third level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the font size to ‘small’ (9-point) when listing the references. **Remember that this year you can use a ninth page as long as it contains *only* cited references.**

[1] Alexander, J.A. & Mozer, M.C. (1995) Template-based algorithms for connectionist rule extraction. In G. Tesauro, D. S. Touretzky and T.K. Leen (eds.), *Advances in Neural Information Processing Systems 7*, pp. 609-616. Cambridge, MA: MIT Press.

[2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System*. New York: TELOS/Springer-Verlag.

[3] Hasselmo, M.E., Schnell, E. & Barkai, E. (1995) Dynamics of learning and recall at excitatory recurrent synapses and cholinergic modulation in rat hiippocampal region CA3. *Journal of Neuroscience* **15**(7):5249-5262.