DATA Mining Project

Prepared By:
Parag Shah

Course:
Topics in statistics – Data Mining

Table of contents

Introduction

1.1 Problem Description and objectives

The classification goal is to predict if the client will subscribe a term deposit given other covariates. Other objective is to model that can explain successful subscription.

1.2 Data Description

The data is related with direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to access if the product (bank term deposit) would be (or not) subscribed.

The data contains 45211instances and can be found at,
http://archive.ics.uci.edu/ml/datasets/Bank+Marketing

The data set includes 16 attributes and 1 target variable. Out of 17, 5 is numeric data and rest is all categorical data.

Input variables: client data
1. age (numeric)
2. job : type of job (categorical:
    'admin.','unknown','unemployed','management','housemaid','entrepreneur',
    'student', 'blue-collar','self-employed','retired','technician','services')
3. Marital : marital status (categorical:
    'married','divorced','single'; note: 'divorced' means divorced or widowed)
4. education (categorical: 'unknown','secondary','primary','tertiary')
5. default: has credit in default? (binary: 'yes','no')
6. balance: average yearly balance, in euros (numeric)
7. housing: has housing loan? (binary: 'yes','no')
8. loan: has personal loan? (binary: 'yes','no')

# related with the last contact of the current campaign:

9. contact: contact communication type (categorical:
    'unknown','telephone','cellular')
10. day: last contact day of the month (numeric)
11. month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')
12. duration: last contact duration, in seconds (numeric)

13. campaign: number of contacts performed during this campaign and for this
client (numeric, includes last contact)
14. pdays: number of days that passed by after the client was last contacted from a
previous campaign (numeric, -1 means client was not previously contacted)
15. previous: number of contacts performed before this campaign and for this
client (numeric)
16. poutcome: outcome of the previous marketing campaign (categorical:
'unknown','other','failure','success')

Output variable (desired target):
17. y - has the client subscribed a term deposit? (binary: 'yes','no')

After reading the paper from creator of this data, I came to know that there were other variables which was part of their initial analysis but after careful evaluation (manual) they decided to discard it e.g Sex of person wasn't affecting outcome of data mining process (14% male vs 15% didn't subscribed to term deposit) they decided to discard it.

1.3 **Data Loading**

Data was loaded using read.csv function and setting Header=TRUE, so that R can recognize first row as header and won't try to define its own header.

1.4 **Task Complexity**

There are two tasks, one is to predict classification based on 16 feature variables and second is to model which variables are more relevant to increase success.
For the first one, many algorithms were tried and results were analyzed using our data set. For second one, the final algorithm was picked and used it to get most relevant feature variable.

Note: Since there are many feature variables, categorical and numeric, the data dimension is high. One can use principal component analysis or variant of such technique such as "correspondence" analysis to reduce the number of dimensions but due to time constraint and unavailability of algorithm to handle mix mode data such technique was not used.

Pre-processing data

2.1 **Missing values**

After reading the paper published by author, they already cleaned the data by removing missing data from data set. Since number of missing data were small this didn’t affected the outcome of the learning algorithm.

2.2 **Class unbalanced problem**

Only 11% of total number of outcome has positive response, so random sampling on original data set could unevenly distribute that outcome and that may affect efficiency of learning algorithms. To rectify this “smote” algorithm was used which generates extra minority cases for each selected majority case based on k nearest neighbor algorithm. Using this new data set was created and all the result were analyzed using original data set and this new data set.

Prediction models

3.1 **Training and test data partitioning**

Partitioning using random sampling:
Sample size of 2/3 and 1/3 of total length were chosen to create training and test data respectively. In R this could be done using “sample” function. So after this step there were two frames for original data set and two for modified data set.

3.2 **Modeling Algorithms**

Almost all of the classification algorithms were carried out using all the input variables, without interaction. Only support vector machines were used with interaction among input variables.

3.2.1 **Linear Discriminant Analysis (LDA)**

R implementation of LDA has in built implementation of Leave-one-out (LOO) cross validation, so prediction error can be accurately calculated. So there was no need for using training and testing data separately to get “estimated prediction error”. One can get “estimated prediction error” by running on entire data set. Using R package “MASS”, which implements “lda” analysis was carried out.

BankData.lda = lda(Formula, data=cbind(X, Y), method="mle", CV=TRUE)

Using cbind, data set was aggregated and instead of default “moment” method to get mean and variance, “mle” was used since it gave better prediction. Last option was to set CV=true, for running Leave-one-out cross validation on predicted values.

3.2.2 **Quadratic Discriminant Analysis (QDA)**

Same as “LDA”, entire data set was analyzed since LOO was used by this method to calculate prediction errors.
Using R package “MASS”, which implements “lda” analysis was carried out.
BankData.lda = qda(Formula, data=cbind(X, Y), method="mle", CV=TRUE)

3.2.3 **Logistic regression**

Logistic regression was carried out using glm library and using binomial classifier to handle categorical data. First training data was used and using 10-fold cross validation training error was calculated. Training model was run on test data to get predicted output. Predicted output of Logistic regression contains prediction probabilities (probability of “yes”) and not the “class” variables, so to get class simple method was used:
If P(X>limit) then choose 1 (“yes”) otherwise target = 0 (“no”)
And based on this prediction error was calculated.

R method to do logistic regression is:
BankData.glm.lrm = glm(Formula, data=X, family=binomial)
10-fold Cross validation method was run number of times to cover entire data set and then taken average generated error rates.

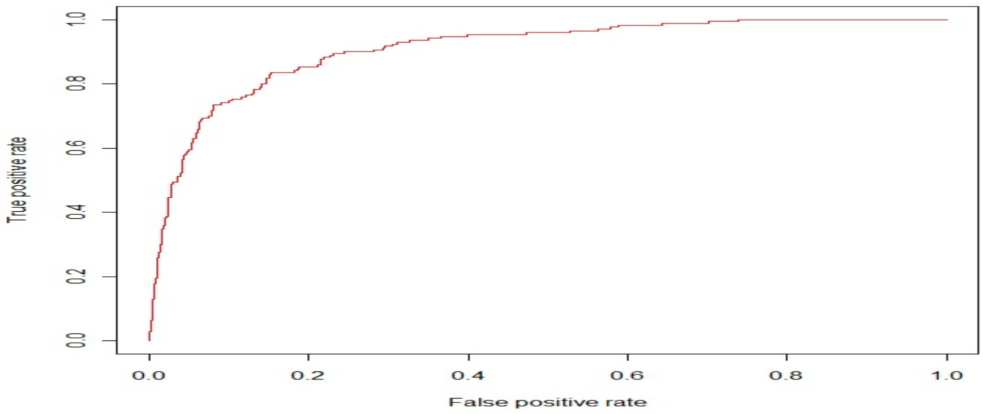
Since predict function of logistic regression returned probability, got another performance metric.

True and false - positive rate, which can be defined by,

Tpr = tp/total pos (Type I error)

Fpr = fp/N (type II error)

Using prediction and performance function type I and type II error was plotted.



3.2.4 **Artificial Neural nets**

Neural nets procedure was carried out using “nnet” package, which is implemented using feed forward network with one hidden layer. By default, nnet function sets initial of links between -0.5 and +0.5 randomly. So for ensuring I get the same result between two successive runs, set.seed(1234) was used. Limiting size of hidden layer to 10 and trying different weight decay, analysis of error was carried out. Weight decay of 0.01 gave best error rate.

As in logistic regression, 10-fold cross validation was carried out using “errorest” function and using training model prediction task was performed. Prediction function gives class values rather than probabilities so error prediction was much more straightforward.

R function for nnet:

```
BankData.nnet = nnet(Formula, data=X, size=10, decay=0.01, maxit=1000,
                                        linout=FALSE, trace=FALSE)
```

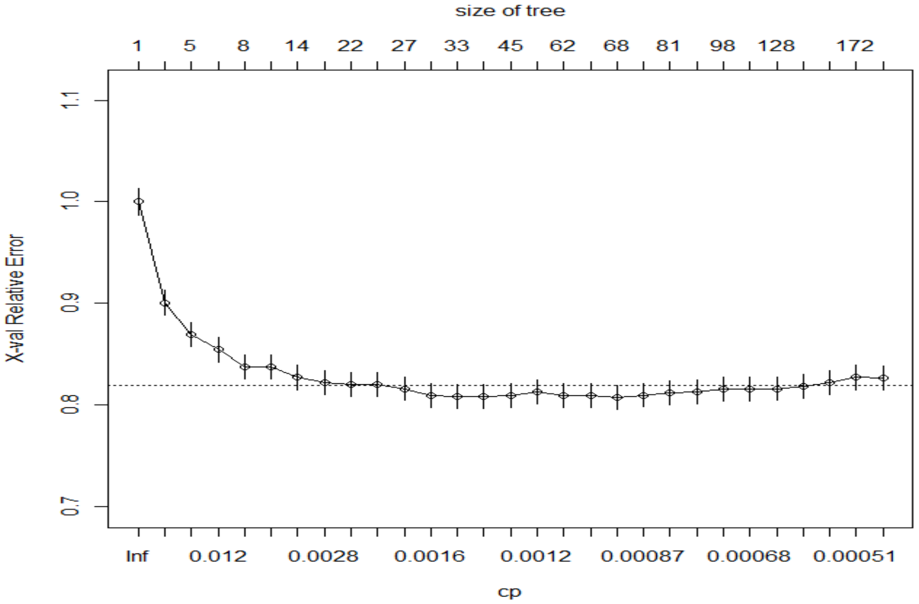
setting linout to FALSE to indicate classification problem and not regression problem.

3.2.5 **Naïve Bayes**

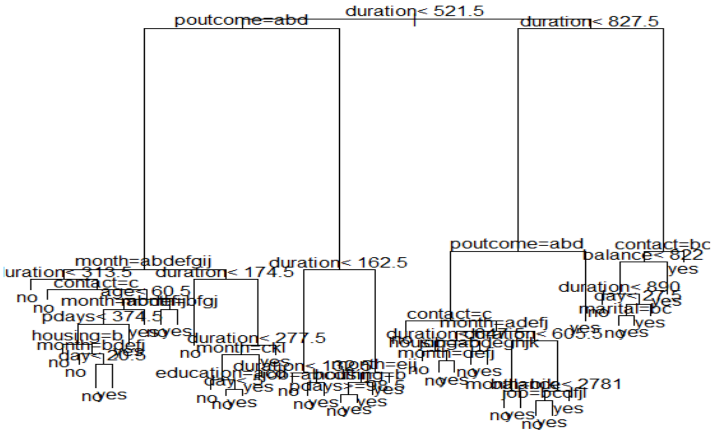
Analysis was done using naiveBayes function in package e1071 with type=”class” to generate model based on categorical data.

3.2.6 **Classification Trees**

Classification trees were constructed using rpart function. To control tree size different value of control parameter (“cp”) was tried and in the end settled with cp = 0.0005. Rpart returns cp values which can be used to prune the trees and resulting model was used to get prediction.

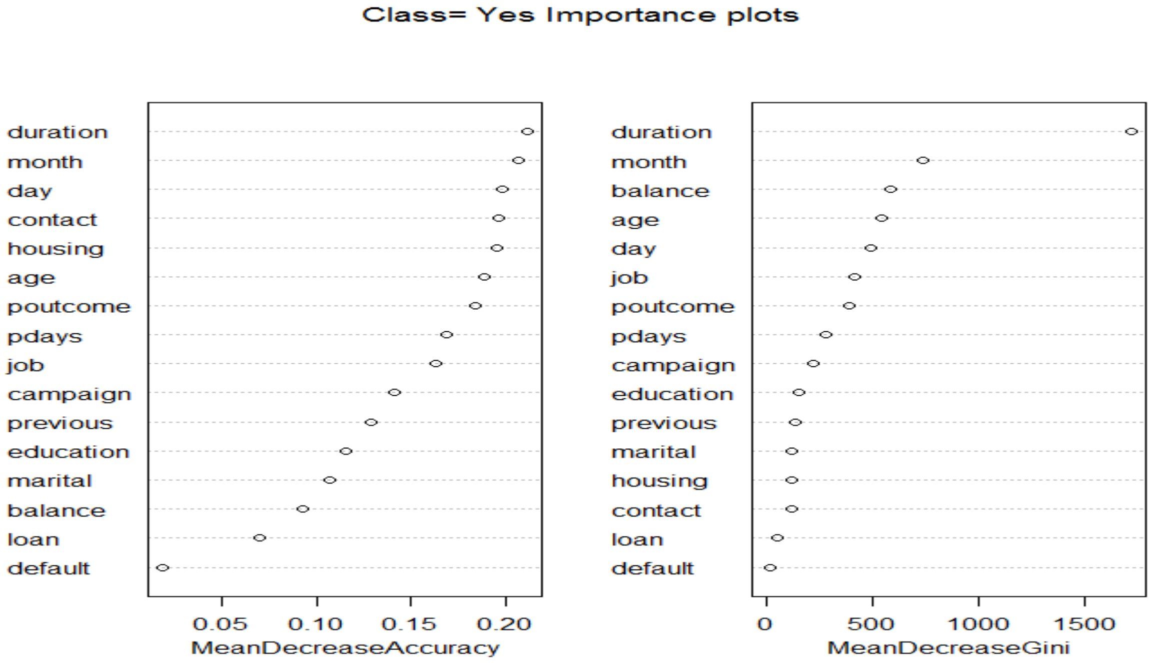


As per above graph, relative x-error is minimized when CP value is around 0.0015. This value was used to get pruned tree. Below is graph of pruned tree.



3.2.7 **Random Forest**

To determine which predictor variables are more important than others, random forest was used which returns importance rate with each variables.



From above graph, seems like most important variables, which are most contributing towards positive answer are housing, contact, day, month and duration.

Using the above model and test data, prediction error was calculated.

3.2.9 **Support Vector Machines**

Since SVMs are computationally demanding, author of data created second data set which contains 10% of the original data. This data set was created using stratified sampling.

SVM analysis was carried out using "ksvm" function. Several variation was tried to get best outcome. Interaction between variables was analyzed using polynomial of degree 2 and straightforward analysis without interaction was carried out using polynomial of degree 1.

Since I am doing classification problem, type of classifier used was "C-svc" and "nu-svc" with "polydot" kernel method.

Also 10-fold cross validation was used to get models using training data and using these model predictions were found.

Model Evaluation and selection

4.1 **Model Evaluation**

Since target variable takes binary values (“yes” or “no”), Misclassification rate is used to evaluate model.

Misclassification rate is defined by:

1-sum(diagonal (matrix C))/sum(C)

Where C = Confusion matrix

Confusion Matrix is is a specific table layout that allows visualization of the performance of an algorithm typically supervised learning one.

E.g. Confusion Matrix of SVM on test data.

```
                          Y.test
BankData.ksvm.degree1.prd  no    yes
                     no   1315   152
                     yes   10     30
```

4.2 **Result Analysis**

Error rate of different algorithm

	Balanced Data	**Original Data**
LDA		0.0995
QDA		0.13008
Logistic Regression		0.0883
Naïve Bayes	0.53	0.18
Artificial Neural Nets		0.0956
Classification Trees	0.16	0.0954
Bagging	0.55	0.0944
Random Forest	0.56	0.0919
SVM Degree 2, C-SVC	0.466	0.1207
SVM Degree 1, C-SVC	0.529	0.09953
SVM Degree 2, Nu-SVC		0.0948
SVM Degree 1, Su-SVC		0.1003

Model is selected based on minimum error rate.

Based on above result, minimum error rate is for, *Random forest* was chosen model for original data but for balanced data, *Classification trees* performed much better than any other model.