Random Forests - Prospective Customers

V2 Maestros

Contents

Problem Statement

The input data contains surveyed information about potential customers for a bank. The goal is to build a model that would predict if the prospect would become a customer of a bank, if contacted by a marketing exercise.

Techniques Used

1. Random Forests
2. Training and Testing
3. Confusion Matrix
4. Indicator Variables
5. Binning
6. Variable Reduction

Data Engineering & Analysis

```
setwd("C:/Personal/V2Maestros/Modules/Machine Learning Algorithms/Random Forests")

bank_data <- read.table("bank.csv", header=TRUE,sep=";")

str(bank_data)
```

Loading and understanding the dataset

```
## 'data.frame':    4521 obs. of  17 variables:
##  $ age      : int  30 33 35 30 59 35 36 39 41 43 ...
##  $ job      : Factor w/ 12 levels "admin.","blue-collar",..: 11 8 5 5 2 5 7 10 3 8 ...
##  $ marital  : Factor w/ 3 levels "divorced","married",..: 2 2 3 2 2 3 2 2 2 2 ...
##  $ education: Factor w/ 4 levels "primary","secondary",..: 1 2 3 3 2 3 3 2 3 1 ...
##  $ default  : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ balance  : int  1787 4789 1350 1476 0 747 307 147 221 -88 ...
##  $ housing  : Factor w/ 2 levels "no","yes": 1 2 2 2 2 1 2 2 2 2 ...
##  $ loan     : Factor w/ 2 levels "no","yes": 1 2 1 2 1 1 1 1 1 2 ...
##  $ contact  : Factor w/ 3 levels "cellular","telephone",..: 1 1 1 3 3 1 1 1 3 1 ...
##  $ day      : int  19 11 16 3 5 23 14 6 14 17 ...
##  $ month    : Factor w/ 12 levels "apr","aug","dec",..: 11 9 1 7 9 4 9 9 9 1 ...
##  $ duration : int  79 220 185 199 226 141 341 151 57 313 ...
##  $ campaign : int  1 1 1 4 1 2 1 2 2 1 ...
##  $ pdays    : int  -1 339 330 -1 -1 176 330 -1 -1 147 ...
##  $ previous : int  0 4 1 0 0 3 2 0 0 2 ...
##  $ poutcome : Factor w/ 4 levels "failure","other",..: 4 1 1 4 4 1 2 4 4 1 ...
##  $ y        : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
```

```
summary(bank_data)
```

```
##       age                job          marital         education
##  Min.   :19.0   management :969   divorced: 528   primary  : 678
##  1st Qu.:33.0   blue-collar:946   married :2797   secondary:2306
##  Median :39.0   technician :768   single  :1196   tertiary :1350
##  Mean   :41.2   admin.     :478                   unknown  : 187
##  3rd Qu.:49.0   services   :417
##  Max.   :87.0   retired    :230
##                 (Other)    :713
##  default       balance      housing     loan          contact
##  no :4445   Min.   :-3313   no :1962   no :3830   cellular :2896
##  yes:  76   1st Qu.:   69   yes:2559   yes: 691   telephone: 301
##             Median :  444                         unknown  :1324
##             Mean   : 1423
##             3rd Qu.: 1480
##             Max.   :71188
##
##       day           month         duration       campaign
##  Min.   : 1.0   may    :1398   Min.   :   4   Min.   : 1.00
##  1st Qu.: 9.0   jul    : 706   1st Qu.: 104   1st Qu.: 1.00
##  Median :16.0   aug    : 633   Median : 185   Median : 2.00
##  Mean   :15.9   jun    : 531   Mean   : 264   Mean   : 2.79
##  3rd Qu.:21.0   nov    : 389   3rd Qu.: 329   3rd Qu.: 3.00
##  Max.   :31.0   apr    : 293   Max.   :3025   Max.   :50.00
##                 (Other): 571
##      pdays          previous         poutcome      y
##  Min.   : -1.0   Min.   : 0.000   failure: 490   no :4000
##  1st Qu.: -1.0   1st Qu.: 0.000   other  : 197   yes: 521
##  Median : -1.0   Median : 0.000   success: 129
##  Mean   : 39.8   Mean   : 0.543   unknown:3705
##  3rd Qu.: -1.0   3rd Qu.: 0.000
##  Max.   :871.0   Max.   :25.000
##
```

```
head(bank_data)
```

```
##   age         job marital education default balance housing loan  contact
## 1  30  unemployed married   primary      no    1787      no   no cellular
## 2  33    services married secondary      no    4789     yes  yes cellular
## 3  35  management  single  tertiary      no    1350     yes   no cellular
## 4  30  management married  tertiary      no    1476     yes  yes  unknown
## 5  59 blue-collar married secondary      no       0     yes   no  unknown
## 6  35  management  single  tertiary      no     747      no   no cellular
##   day month duration campaign pdays previous poutcome  y
## 1  19   oct       79        1    -1        0  unknown no
## 2  11   may      220        1   339        4  failure no
## 3  16   apr      185        1   330        1  failure no
## 4   3   jun      199        4    -1        0  unknown no
## 5   5   may      226        1    -1        0  unknown no
## 6  23   feb      141        2   176        3  failure no
```

Data Cleansing

1. The ranges of values in each of the variables (columns) look ok without any kind of outliers
2. There is equal distribution of the three classes - setosa, versicolor and virginia

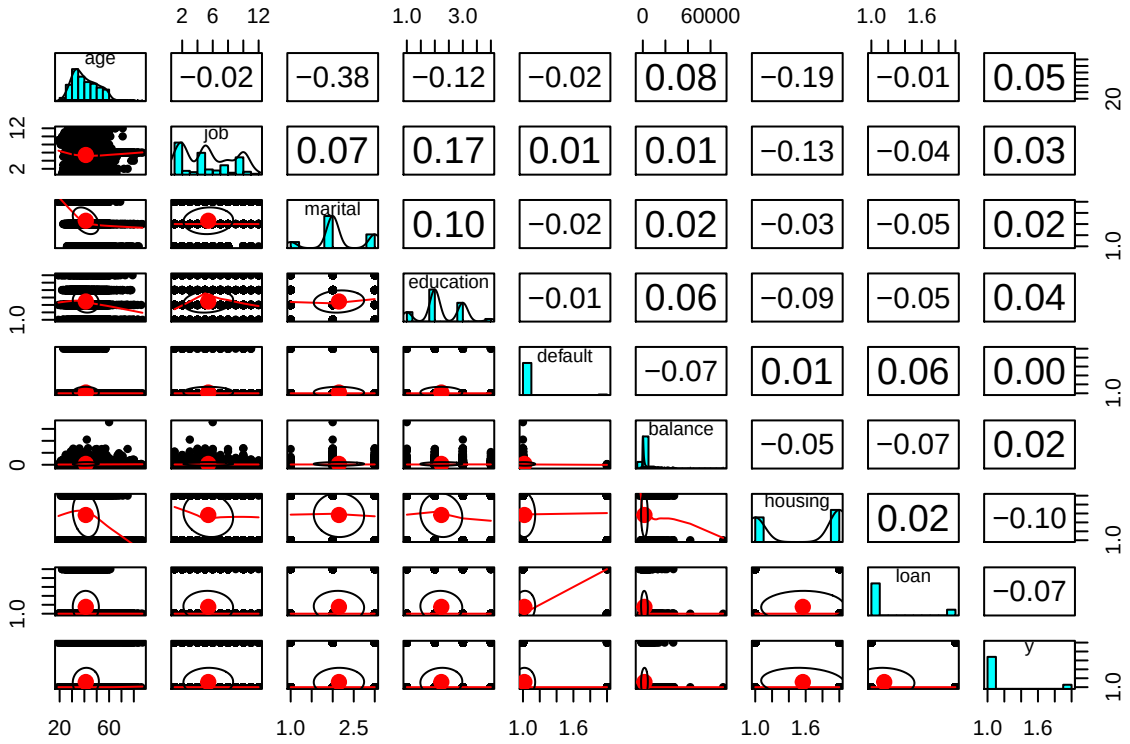
No cleansing required

Correlations Given the large number of predictors, we would like to start with a correlation analysis to see if some variables can be dropped

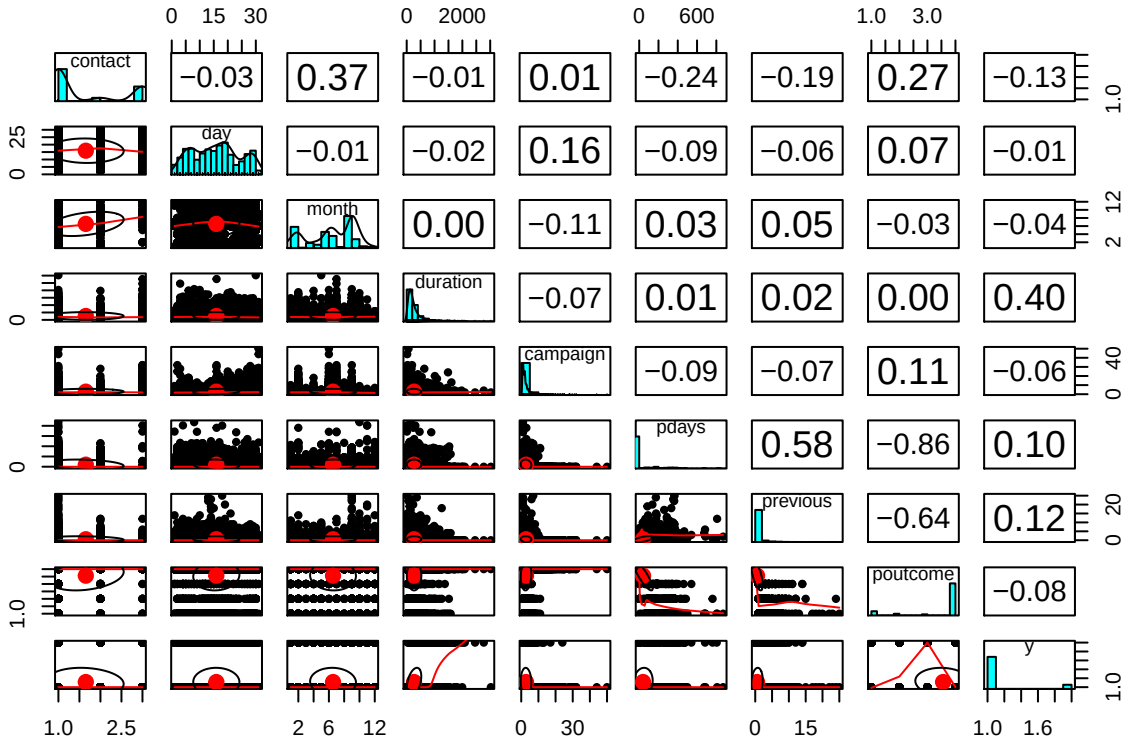
```
library(psych)
```

```
## Warning: package 'psych' was built under R version 3.1.1
```

```
#Given the large number of variables, we will do correlation analysis in 2 parts
pairs.panels(bank_data[, c(1:8,17)])
```




```
pairs.panels(bank_data[, c(9:17)])
```

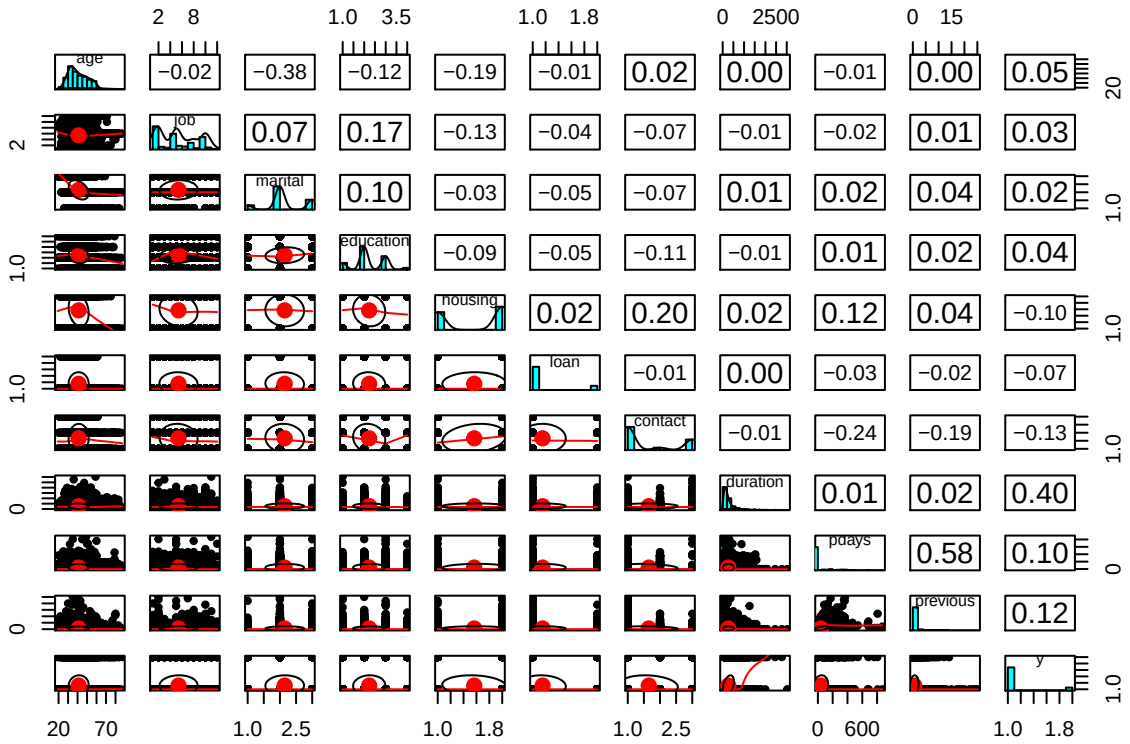


Based on the correlation co-efficients, let us eliminate default, balance, day, month, campaign, poutcome because of very low correlation. There are others too with very low correlation, but let us keep it for example sake.

```
new_data <-bank_data[, c(1:4,7:9,12,14,15,17)]
str(new_data)
```

```
## 'data.frame':    4521 obs. of  11 variables:
##  $ age      : int  30 33 35 30 59 35 36 39 41 43 ...
##  $ job      : Factor w/ 12 levels "admin.","blue-collar",..: 11 8 5 5 2 5 7 10 3 8 ...
##  $ marital  : Factor w/ 3 levels "divorced","married",..: 2 2 3 2 2 3 2 2 2 2 ...
##  $ education: Factor w/ 4 levels "primary","secondary",..: 1 2 3 3 2 3 3 2 3 1 ...
##  $ housing  : Factor w/ 2 levels "no","yes": 1 2 2 2 2 1 2 2 2 2 ...
##  $ loan     : Factor w/ 2 levels "no","yes": 1 2 1 2 1 1 1 1 1 2 ...
##  $ contact  : Factor w/ 3 levels "cellular","telephone",..: 1 1 1 3 3 1 1 1 3 1 ...
##  $ duration : int  79 220 185 199 226 141 341 151 57 313 ...
##  $ pdays    : int  -1 339 330 -1 -1 176 330 -1 -1 147 ...
##  $ previous : int  0 4 1 0 0 3 2 0 0 2 ...
##  $ y        : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
```

```
pairs.panels(new_data)
```



Data Transformations Let us do the following transformations

1. Convert age into a binned range.
2. Convert marital status into indicator variables. We could do the same for all other factors too, but we choose not to. Indicator variables may or may not improve predictions. It is based on the specific data set and need to be figured out by trials.

```
new_data$age <- cut(new_data$age, c(1,20,40,60,100))

new_data$is_divorced <- ifelse( new_data$marital == "divorced", 1, 0)
new_data$is_single <- ifelse( new_data$marital == "single", 1, 0)
new_data$is_married <- ifelse( new_data$marital == "married", 1, 0)
new_data$marital <- NULL

str(new_data)
```

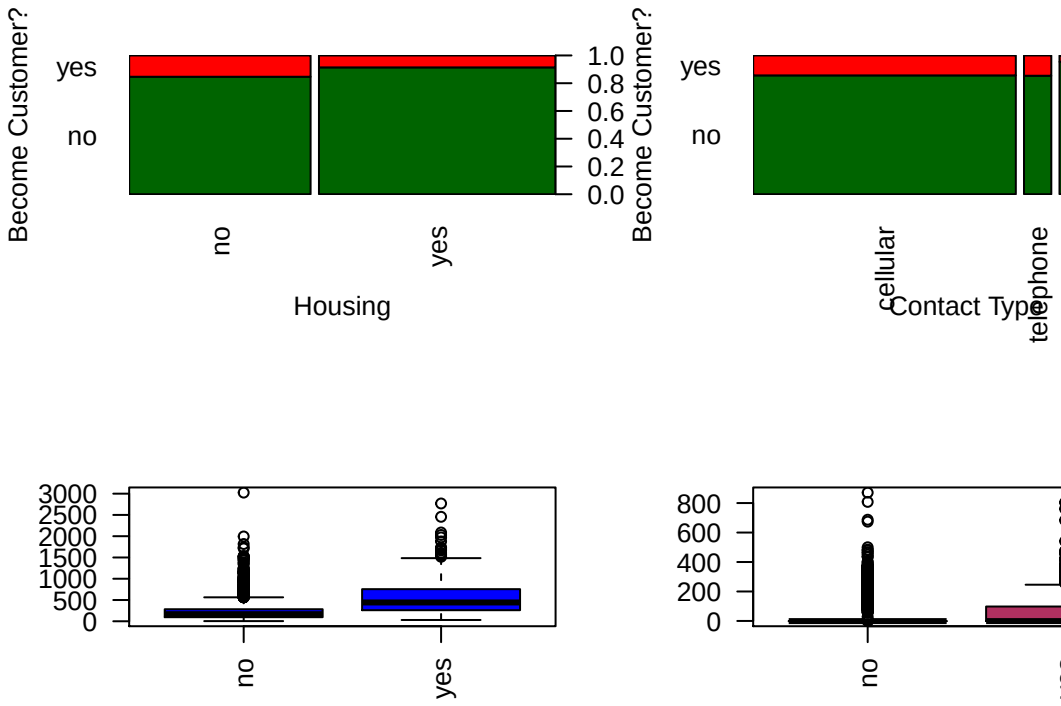
```
## 'data.frame':    4521 obs. of  13 variables:
##  $ age         : Factor w/ 4 levels "(1,20]","(20,40]",..: 2 2 2 2 3 2 2 2 3 3 ...
##  $ job         : Factor w/ 12 levels "admin.","blue-collar",..: 11 8 5 5 2 5 7 10 3 8 ...
##  $ education   : Factor w/ 4 levels "primary","secondary",..: 1 2 3 3 2 3 3 2 3 1 ...
##  $ housing     : Factor w/ 2 levels "no","yes": 1 2 2 2 2 1 2 2 2 2 ...
```

```
##  $ loan       : Factor w/ 2 levels "no","yes": 1 2 1 2 1 1 1 1 1 2 ...
##  $ contact    : Factor w/ 3 levels "cellular","telephone",..: 1 1 1 3 3 1 1 1 3 1 ...
##  $ duration   : int  79 220 185 199 226 141 341 151 57 313 ...
##  $ pdays      : int  -1 339 330 -1 -1 176 330 -1 -1 147 ...
##  $ previous   : int  0 4 1 0 0 3 2 0 0 2 ...
##  $ y          : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ is_divorced: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ is_single  : num  0 0 1 0 0 1 0 0 0 0 ...
##  $ is_married : num  1 1 0 1 1 0 1 1 1 1 ...
```

```
par(mfrow=c(2,2),las=2)

plot( new_data$housing, new_data$y,
      xlab="Housing", ylab="Become Customer?", col=c("darkgreen","red"))
plot( new_data$contact, new_data$y,
      xlab="Contact Type", ylab="Become Customer?", col=c("darkgreen","red"))

boxplot( duration ~ y, data=new_data,col="blue")
boxplot( pdays ~ y, data=new_data,col="maroon")
```



Exploratory Data Analysis

Model Building Build model based on the training data

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.1.1
```

```
## Loading required package: lattice
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.1.1
```

```
##
## Attaching package: 'ggplot2'
##
## The following object is masked from 'package:psych':
##
##     %+%
```

```
inTrain <- createDataPartition(y=new_data$y ,p=0.7,list=FALSE)
training <- new_data[inTrain,]
testing <- new_data[-inTrain,]
dim(training);dim(testing)
```

```
## [1] 3165   13
```

```
## [1] 1356   13
```

```
table(training$y); table(testing$y)
```

```
##
##   no  yes
## 2800  365
```

```
##
##   no  yes
## 1200  156
```

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 3.1.1
```

```
## randomForest 4.6-10
## Type rfNews() to see new features/changes/bug fixes.
##
## Attaching package: 'randomForest'
##
## The following object is masked from 'package:psych':
##
##     outlier
```

```
model <- randomForest(y ~ ., data=training)
model
```

```
##
## Call:
##  randomForest(formula = y ~ ., data = training)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 10.74%
## Confusion matrix:
##       no yes class.error
## no  2705  95     0.03393
## yes  245 120     0.67123
```

```
#importance of each predictor
importance(model)
```

```
##             MeanDecreaseGini
## age                   22.707
## job                   63.382
## education             26.274
## housing               16.620
## loan                   8.544
## contact               18.653
## duration             219.012
## pdays                 56.497
## previous              32.506
## is_divorced            7.625
## is_single              8.819
## is_married             9.886
```

Testing

Now let us predict the class for each sample in the test data. Then compare the prediction with the actual value of the class.

```
library(caret)
predicted <- predict(model, testing)
table(predicted)
```

```
## predicted
##   no  yes
## 1287   69
```

```
confusionMatrix(predicted, testing$y)
```

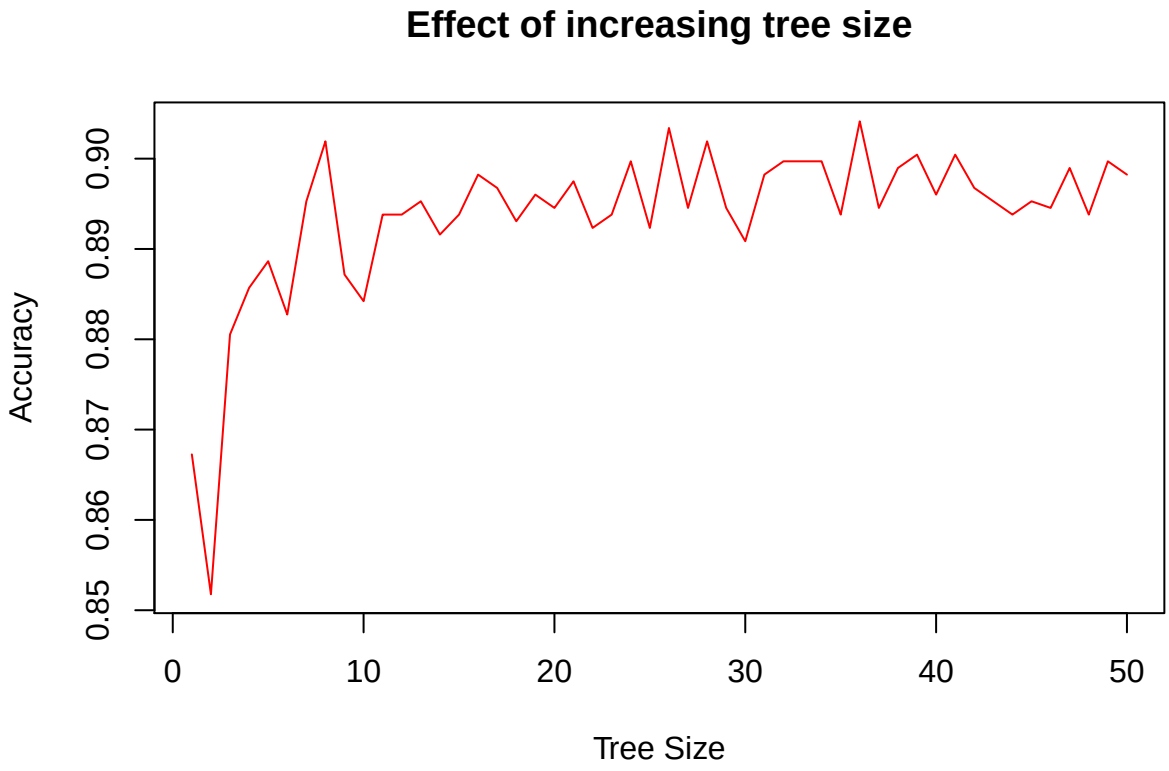
```
## Confusion Matrix and Statistics
##
```

```
##           Reference
## Prediction   no  yes
##        no  1177  110
##        yes   23   46
##
##                Accuracy : 0.902
##                  95% CI : (0.885, 0.917)
##     No Information Rate : 0.885
##     P-Value [Acc > NIR] : 0.0258
##
##                   Kappa : 0.364
##  Mcnemar's Test P-Value : 8.84e-14
##
##             Sensitivity : 0.981
##             Specificity : 0.295
##          Pos Pred Value : 0.915
##          Neg Pred Value : 0.667
##              Prevalence : 0.885
##          Detection Rate : 0.868
##    Detection Prevalence : 0.949
##       Balanced Accuracy : 0.638
##
##        'Positive' Class : no
##
```

Inspite of the correlations being not so high, the accuracy is high because of the combined effect of predictors as well as the power of building multiple trees.

Effect of increasing tree count Let us try to build different number of trees and see the effect of that on the accuracy of the prediction

```
accuracy=c()
for (i in seq(1,50, by=1)) {
  modFit <- randomForest(y ~ ., data=training, ntree=i)
  accuracy <- c(accuracy, confusionMatrix(predict(modFit, testing, type="class"), testing$y)$overall[1])
}
par(mfrow=c(1,1))
plot(x=seq(1,50, by=1), y=accuracy, type="l", col="red",
     main="Effect of increasing tree size", xlab="Tree Size", ylab="Accuracy")
```



Conclusions

Random forests provide better accuracy than plain decision trees because of the power of the number of trees built. The example shows that as we increase the number of trees, accuracy also increases. But that comes at a cost of increased time and resource usage.
