Machine Learning Assignment 2

Subramani.M

31 May 2018

```
setwd("C:/Users/Administrator/Desktop/Machine Learning/assignments")
library(dplyr)

## Warning: package 'dplyr' was built under R version 3.4.3

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(xlsx)

## Warning: package 'xlsx' was built under R version 3.4.3

## Loading required package: rJava

## Warning: package 'rJava' was built under R version 3.4.3

## Loading required package: xlsxjars

## Warning: package 'xlsxjars' was built under R version 3.4.3

library(tree)

## Warning: package 'tree' was built under R version 3.4.4

library(rpart)
library(rattle)

## Warning: package 'rattle' was built under R version 3.4.4

## Rattle: A free graphical interface for data science with R.
## Version 5.1.0 Copyright (c) 2006-2017 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.

library(caret)

## Warning: package 'caret' was built under R version 3.4.4

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

## Warning: package 'ggplot2' was built under R version 3.4.3
```

Sanitizing Dataset

```
# Reading the data file
credit <- read.xlsx("credit_default.xlsx",sheetName = "Sheet1")

# Exploring the Data
summary(credit)

##           X.      months_loan_duration                credit_history
##  < 0 DM    :274   Min.   : 4.0         critical                :293
##  > 200 DM  : 63   1st Qu.:12.0         delayed                 : 88
##  1 - 200 DM:269   Median :18.0         fully repaid            : 40
##  unknown   :394   Mean   :20.9         fully repaid this bank: 49
##                   3rd Qu.:24.0         repaid                  :530
##                   Max.   :72.0
##
##        purpose       amount          savings_balance  employment_length
##  radio/tv  :280   Min.   :  250   < 100 DM     :603   > 7 yrs   :253
##  car (new) :234   1st Qu.: 1366   > 1000 DM    : 48   0 - 1 yrs :172
##  furniture :181   Median : 2320   101 - 500 DM :103   1 - 4 yrs :339
##  car (used):103   Mean   : 3271   501 - 1000 DM: 63   4 - 7 yrs :174
##  business  : 97   3rd Qu.: 3972   unknown      :183   unemployed: 62
##  education : 50   Max.   :18424
##  (Other)   : 55
##  installment_rate      personal_status      other_debtors
##  Min.   :1.000    divorced male: 50    co-applicant: 41
##  1st Qu.:2.000    female       :310    guarantor   : 52
##  Median :3.000    married male : 92    none        :907
##  Mean   :2.973    single male  :548
##  3rd Qu.:4.000
##  Max.   :4.000
##
##  residence_history                     property        age
##  Min.   :1.000     building society savings:232   Min.   :19.00
##  1st Qu.:2.000     other                   :332   1st Qu.:27.00
##  Median :3.000     real estate             :282   Median :33.00
##  Mean   :2.845     unknown/none            :154   Mean   :35.55
##  3rd Qu.:4.000                                    3rd Qu.:42.00
##  Max.   :4.000                                    Max.   :75.00
##
##  installment_plan     housing    existing_credits    default
##  bank  :139       for free:108   Min.   :1.000     Min.   :1.0
##  none  :814       own     :713   1st Qu.:1.000     1st Qu.:1.0
##  stores: 47       rent    :179   Median :1.000     Median :1.0
##                                  Mean   :1.407     Mean   :1.3
##                                  3rd Qu.:2.000     3rd Qu.:2.0
##                                   Max.   :4.000    Max.   :2.0
##
##    dependents    telephone  foreign_worker                      job
##  Min.   :1.000   none:596   no : 37        mangement self-employed:148
##  1st Qu.:1.000   yes :404   yes:963        skilled employee       :630
##  Median :1.000                             unemployed non-resident: 22
##  Mean   :1.155                             unskilled resident     :200
##  3rd Qu.:1.000
##  Max.   :2.000
##
```

```
glimpse(credit)
```

```
## Observations: 1,000
## Variables: 21
## $ X.                   <fctr> < 0 DM, 1 - 200 DM, unknown, < 0 DM, < 0...
## $ months_loan_duration <dbl> 6, 48, 12, 42, 24, 36, 24, 36, 12, 30, 12...
## $ credit_history       <fctr> critical, repaid, critical, repaid, dela...
## $ purpose              <fctr> radio/tv, radio/tv, education, furniture...
## $ amount               <dbl> 1169, 5951, 2096, 7882, 4870, 9055, 2835,...
## $ savings_balance      <fctr> unknown, < 100 DM, < 100 DM, < 100 DM, <...
## $ employment_length    <fctr> > 7 yrs, 1 - 4 yrs, 4 - 7 yrs, 4 - 7 yrs...
## $ installment_rate     <dbl> 4, 2, 2, 2, 3, 2, 3, 2, 2, 4, 3, 3, 1, 4,...
## $ personal_status      <fctr> single male, female, single male, single...
## $ other_debtors        <fctr> none, none, none, guarantor, none, none,...
## $ residence_history    <dbl> 4, 2, 3, 4, 4, 4, 4, 2, 4, 2, 1, 4, 1, 4,...
## $ property             <fctr> real estate, real estate, real estate, b...
## $ age                  <dbl> 67, 22, 49, 45, 53, 35, 53, 35, 61, 28, 2...
## $ installment_plan     <fctr> none, none, none, none, none, none, none...
## $ housing              <fctr> own, own, own, for free, for free, for f...
## $ existing_credits     <dbl> 2, 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 1, 2,...
## $ default              <dbl> 1, 2, 1, 1, 2, 1, 1, 1, 1, 2, 2, 2, 1, 2,...
## $ dependents           <dbl> 1, 1, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ telephone            <fctr> yes, none, none, none, none, yes, none, ...
## $ foreign_worker       <fctr> yes, yes, yes, yes, yes, yes, yes, yes, ...
## $ job                  <fctr> skilled employee, skilled employee, unsk...
```

```
# Checking for NA Values
colSums(is.na(credit))            # No presence of NA Values
```

```
##                   X. months_loan_duration       credit_history
##                    0                    0                    0
##              purpose               amount      savings_balance
##                    0                    0                    0
##    employment_length     installment_rate      personal_status
##                    0                    0                    0
##        other_debtors    residence_history             property
##                    0                    0                    0
##                  age     installment_plan              housing
##                    0                    0                    0
##     existing_credits              default           dependents
##                     0                    0                    0
##             telephone       foreign_worker                  job
##                     0                    0                    0
```

decision trees

```
# converting to catagorical columns
credit$months_loan_duration = as.factor(credit$months_loan_duration)
credit$installment_rate = as.factor(credit$installment_rate)
credit$residence_history = as.factor(credit$residence_history)
credit$dependents = as.factor(credit$dependents)
credit$default = as.factor(credit$default)
credit$existing_credits = as.factor(credit$existing_credits)

# Training and testing dataset
credit_train <- credit[sample(seq(1,nrow(credit)),700),]
credit_test <- credit[sample(seq(1,nrow(credit)),300),]

glimpse(credit)
```

```
## Observations: 1,000
## Variables: 21
## $ X.                   <fctr> < 0 DM, 1 - 200 DM, unknown, < 0 DM, < 0...
## $ months_loan_duration <fctr> 6, 48, 12, 42, 24, 36, 24, 36, 12, 30, 1...
## $ credit_history       <fctr> critical, repaid, critical, repaid, dela...
## $ purpose              <fctr> radio/tv, radio/tv, education, furniture...
## $ amount               <dbl> 1169, 5951, 2096, 7882, 4870, 9055, 2835,...
## $ savings_balance      <fctr> unknown, < 100 DM, < 100 DM, < 100 DM, <...
## $ employment_length    <fctr> > 7 yrs, 1 - 4 yrs, 4 - 7 yrs, 4 - 7 yrs...
## $ installment_rate     <fctr> 4, 2, 2, 2, 3, 2, 3, 2, 2, 4, 3, 3, 1, 4...
## $ personal_status      <fctr> single male, female, single male, single...
## $ other_debtors        <fctr> none, none, none, guarantor, none, none,...
## $ residence_history    <fctr> 4, 2, 3, 4, 4, 4, 4, 2, 4, 2, 1, 4, 1, 4...
## $ property             <fctr> real estate, real estate, real estate, b...
## $ age                  <dbl> 67, 22, 49, 45, 53, 35, 53, 35, 61, 28, 2...
## $ installment_plan     <fctr> none, none, none, none, none, none, none...
## $ housing              <fctr> own, own, own, for free, for free, for f...
## $ existing_credits     <fctr> 2, 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 1, 2...
## $ default              <fctr> 1, 2, 1, 1, 2, 1, 1, 1, 1, 2, 2, 2, 1, 2...
## $ dependents           <fctr> 1, 1, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1...
## $ telephone            <fctr> yes, none, none, none, none, yes, none, ...
## $ foreign_worker       <fctr> yes, yes, yes, yes, yes, yes, yes, yes, ...
## $ job                  <fctr> skilled employee, skilled employee, unsk...
```

```
# model building
credit_model = rpart(default ~ .,data = credit_train)
credit_predict = predict(credit_model,credit_test,type = "class")

summary(credit_predict)
```

```
##   1   2
## 231  69

credit_predict = as.factor(credit_predict)
final_result = table(credit_test$default,credit_predict)
confusion_matrix = confusionMatrix(final_result,positive = '2')
confusion_matrix$byClass[5]

## Precision
## 0.4761905
```

random forest

```
library(randomForest)

## Warning: package 'randomForest' was built under R version 3.4.4

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin

## The following object is masked from 'package:rattle':
##
##     importance

## The following object is masked from 'package:dplyr':
##
##     combine

#training and testing dataset
credit_train <- credit[sample(seq(1,nrow(credit)),700),]
credit_test <- credit[sample(seq(1,nrow(credit)),300),]

#model building
model = randomForest(default ~ . , data = credit_train,ntree=30)
result = as.factor(predict(model,credit_test))

#confusion matrix
cm = confusionMatrix(result,credit_test$default,positive = "2")
cm

## Confusion Matrix and Statistics
##
##             Reference
## Prediction   1   2
##          1 200  16
##          2   7  77
##
##                Accuracy : 0.9233
##                  95% CI : (0.8872, 0.9508)
##     No Information Rate : 0.69
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.8159
##  Mcnemar's Test P-Value : 0.09529
##
##             Sensitivity : 0.8280
##             Specificity : 0.9662
##          Pos Pred Value : 0.9167
##          Neg Pred Value : 0.9259
##              Prevalence : 0.3100
##          Detection Rate : 0.2567
##    Detection Prevalence : 0.2800
##       Balanced Accuracy : 0.8971
##
##        'Positive' Class : 2
##
```

Ada boost

```
library(adabag)

## Warning: package 'adabag' was built under R version 3.4.4

## Loading required package: foreach

## Warning: package 'foreach' was built under R version 3.4.4

## Loading required package: doParallel

## Warning: package 'doParallel' was built under R version 3.4.4

## Loading required package: iterators

## Warning: package 'iterators' was built under R version 3.4.4

## Loading required package: parallel

#training and testing dataset
credit_train <- credit[sample(seq(1,nrow(credit)),700),]
credit_test <- credit[sample(seq(1,nrow(credit)),300),]

# Building the model
boosting_model <- boosting(default ~ .,data = credit_train)
boosting_pred <- predict(boosting_model,credit_test)
boosting_pred$class <- as.factor(boosting_pred$class)
```

```r
#Confusion matrix
boosting_cm <- confusionMatrix(boosting_pred$class,credit_test$default)
boosting_cm
```

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction   1   2
##          1 205   9
##          2   8  78
## 
##                Accuracy : 0.9433
##                  95% CI : (0.9108, 0.9666)
##     No Information Rate : 0.71
##     P-Value [Acc > NIR] : <2e-16
## 
##                   Kappa : 0.8619
##  Mcnemar's Test P-Value : 1
## 
##             Sensitivity : 0.9624
##             Specificity : 0.8966
##          Pos Pred Value : 0.9579
##          Neg Pred Value : 0.9070
##              Prevalence : 0.7100
##          Detection Rate : 0.6833
##    Detection Prevalence : 0.7133
##       Balanced Accuracy : 0.9295
## 
##        'Positive' Class : 1
## 
```

Knn Algorithm

```r
library(class)
library(BBmisc)
```

```
## Warning: package 'BBmisc' was built under R version 3.4.4
```

```
## 
## Attaching package: 'BBmisc'
```

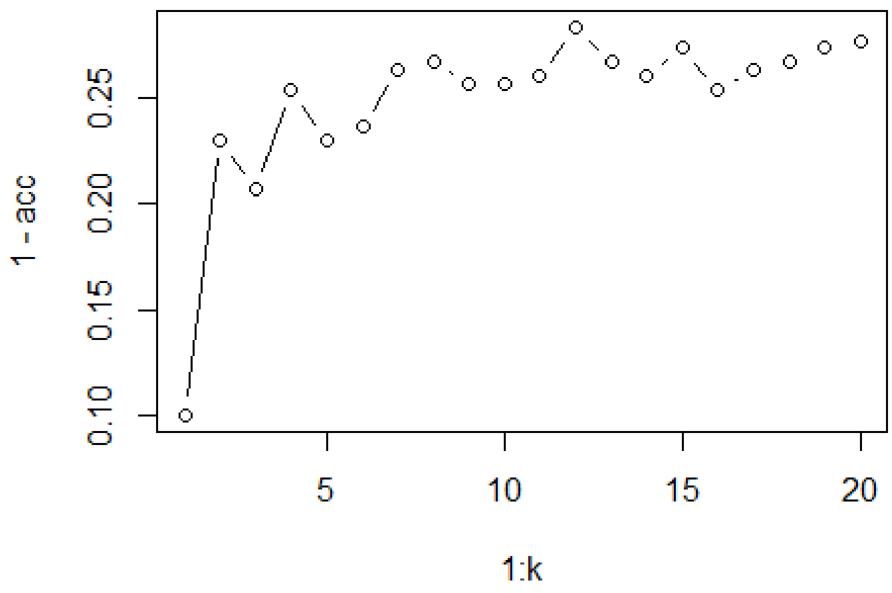
```
## The following objects are masked from 'package:dplyr':
## 
##     coalesce, collapse
```

```r
# Data reading
credit <- read.xlsx("credit_default.xlsx",sheetName = "Sheet1")

# Converting Categorical to Numerical Columns
```

```
knn_credit <- dummyVars(~.,data = credit)
knn_credit <- data.frame(predict(knn_credit,credit))

# Normalizing the Data
knn_credit_norm <- normalize(knn_credit,method = "range",range = c(0,1))

# Training and testing dataset
knn_train <- knn_credit_norm[sample(seq(1,nrow(knn_credit_norm)),700),]
knn_test <- knn_credit_norm[sample(seq(1,nrow(knn_credit_norm)),300),]

# Finding 'K' value
k <- round(sqrt(nrow(knn_train)))

#KNN Implementation
knn_pred <- knn(knn_train %>% select(-default),
    knn_test %>% select(-default),
    cl = as.factor(knn_train$default),k = k-1)

knn_pred <- as.factor(knn_pred)
knn_test$default <- as.factor(knn_test$default)
knn_cm <- confusionMatrix(knn_pred,knn_test$default,positive = "1")
knn_cm

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 183  67
##          1  21  29
##
##                Accuracy : 0.7067
##                  95% CI : (0.6516, 0.7576)
##     No Information Rate : 0.68
##     P-Value [Acc > NIR] : 0.1769
##
##                   Kappa : 0.2281
##  Mcnemar's Test P-Value : 1.61e-06
##
##             Sensitivity : 0.30208
##             Specificity : 0.89706
##          Pos Pred Value : 0.58000
##          Neg Pred Value : 0.73200
##              Prevalence : 0.32000
##          Detection Rate : 0.09667
##    Detection Prevalence : 0.16667
##       Balanced Accuracy : 0.59957
##
##        'Positive' Class : 1
##
```

Finding the suitable 'k' value

```
k <- 20
sens <- c()
acc <- c()
for (i in 1:k)
{
  knn_pred <- knn(knn_train %>% select(-default),
    knn_test %>% select(-default),
    cl = as.factor(knn_train$default),k = i)
  knn_pred <- as.factor(knn_pred)
  knn_test$default <- as.factor(knn_test$default)
  knn_cm <- confusionMatrix(knn_pred,knn_test$default,positive = "1")
  acc <- c(acc,knn_cm$overall["Accuracy"])
  sens <- c(sens,knn_cm$byClass["Sensitivity"])
}
plot(1:k,1-acc,type = "b")
```



```
k = which(max(acc[-1]) == acc)
print(k)
```

```
## Accuracy
##        3
```

```
knn_pred <- knn(knn_train %>% select(-default),
    knn_test %>% select(-default),
    cl = as.factor(knn_train$default),k = k)
knn_pred <- as.factor(knn_pred)
```

```
knn_test$default <- as.factor(knn_test$default)
knn_cm <- confusionMatrix(knn_pred,knn_test$default,positive = "1")
knn_cm
```

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction   0   1
##          0 186  44
##          1  18  52
##                                          
##                Accuracy : 0.7933         
##                  95% CI : (0.743, 0.8377)
##     No Information Rate : 0.68           
##     P-Value [Acc > NIR] : 8.507e-06      
##                                          
##                   Kappa : 0.4884         
##  Mcnemar's Test P-Value : 0.001498       
##                                          
##             Sensitivity : 0.5417         
##             Specificity : 0.9118         
##          Pos Pred Value : 0.7429         
##          Neg Pred Value : 0.8087         
##              Prevalence : 0.3200         
##          Detection Rate : 0.1733         
##    Detection Prevalence : 0.2333         
##       Balanced Accuracy : 0.7267         
##                                          
##        'Positive' Class : 1              
## 
```

naive bayes

```
library(e1071)
```

```
## Warning: package 'e1071' was built under R version 3.4.4
```

```
library(dplyr)

#Building model
nb_model <- naiveBayes(default ~ .,data = credit_train)
nb_pred <- predict(nb_model,credit_test)

# confusion matrix
nb_cm <- confusionMatrix(nb_pred,credit_test$default)
nb_cm
```

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction   1   2
##          1 181  43
##          2  32  44
##
##                Accuracy : 0.75
##                  95% CI : (0.697, 0.798)
##     No Information Rate : 0.71
##     P-Value [Acc > NIR] : 0.07019
##
##                   Kappa : 0.3693
##  Mcnemar's Test P-Value : 0.24821
##
##             Sensitivity : 0.8498
##             Specificity : 0.5057
##          Pos Pred Value : 0.8080
##          Neg Pred Value : 0.5789
##              Prevalence : 0.7100
##          Detection Rate : 0.6033
##    Detection Prevalence : 0.7467
##       Balanced Accuracy : 0.6778
##
##        'Positive' Class : 1
##
```

logistic Regression

```
credit <- read.xlsx("credit_default.xlsx",sheetName = "Sheet1")

# Converting all Categorical columns to Numerical Columns
lm_credit <- dummyVars(~.,data = credit)
lm_credit <- data.frame(predict(lm_credit,credit))

#Training and testing data
credit_train <- lm_credit[sample(seq(1,nrow(lm_credit)),700),]
credit_test <- lm_credit[sample(seq(1,nrow(lm_credit)),300),]

#Building model
log_model <- lm(default ~ .,data = credit_train)
log_pred <- round(predict(log_model,credit_test))

## Warning in predict.lm(log_model, credit_test): prediction from a rank-
## deficient fit may be misleading

log_pred <- as.factor(log_pred)
credit_test$default <- as.factor(credit_test$default)

#Confusion matrix
log_cm <- confusionMatrix(log_pred,credit_test$default)
log_cm
```

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction   1   2
##          1 199  44
##          2  16  41
##                                           
##                Accuracy : 0.8             
##                  95% CI : (0.7502, 0.8438)
##     No Information Rate : 0.7167          
##     P-Value [Acc > NIR] : 0.0006027       
##                                           
##                   Kappa : 0.4531          
##  Mcnemar's Test P-Value : 0.0004909       
##                                           
##             Sensitivity : 0.9256          
##             Specificity : 0.4824          
##          Pos Pred Value : 0.8189          
##          Neg Pred Value : 0.7193          
##              Prevalence : 0.7167          
##          Detection Rate : 0.6633          
##    Detection Prevalence : 0.8100          
##       Balanced Accuracy : 0.7040          
##                                           
##        'Positive' Class : 1               
## 
```