Homework on Supervised Learning
Dolores Romero Morales

The minicreditgerman dataset (creditgerman.txt) contains 20 explanatory variables and 750 observations. The class informs us whether the client was a good payer ('1') or not ('2'):

Using the best practices in Data Science, address the following questions making sure to describe the modeling approach used:

(i) Use Logistic Regression to model the creditgerman data.

(ii) Use Support Vector Machines to model the creditgerman data.

(iii) Use Classification Trees to model the creditgerman data.

(iv) Use Random Forests to model the creditgerman data.

(v) Use *k*-Nearest Neighbors to model the creditgerman data.

(vi) Use the models built in Questions 1(i)-1(v) to predict the class membership of the objects in testcreditgerman.txt.

(vii) ...

(viii) ...

Answers

Note1: With the workshop and homework answers provided in CANVAS, you should be able to code in R the models.
Note2: I kindly remind you that you have been asked to describe the modeling approach
Note3: I have kicked off the process adding some lines of code here below. You can fill in the remaining lines.
Note4: Recall that you need to upload the corresponding packages

```
#these lines are to read the two files provided
mymini <- read.table(file.choose(),header=TRUE, stringsAsFactors=TRUE)
mytesting <- read.table(file.choose(),header=TRUE, stringsAsFactors=TRUE)

#these lines are to inspect the nature of the variables
str(mymini)
str(mytesting)

#these lines are to convert the target variable into a factor and the categorical variables into dummies
library(dummies)
myXmini <- mymini[,-21]
myXminiD <- dummy.data.frame(myXmini, sep= ".")
classmembership <- as.factor(mymini$classmembership)
myminiD <-  cbind(myXminiD, classmembership)
myXtesting <- mytesting[,-21]
myXtestingD <- dummy.data.frame(myXtesting, sep= ".")
classmembership <- as.factor(mytesting$classmembership)
mytestingD <-  cbind(myXtestingD, classmembership)

#these lines are to understand in which position is our target variable now
#we can see that we know have 61 explanatory variables and our response variable is in column 62
dim(myminiD)

#these lines are to normalize the continuous and the integer data
myminiNORM<- myminiD
mytestingNORM<- mytestingD
for(i in 1:length(colnames(myminiD))-1) {
      if(class(myminiD[,i]) == "numeric" || class(myminiD[,i]) == "integer") {
         minimum<-min(myminiD[,i])
         maximum<-max(myminiD[,i])
         myminiNORM[,i] <- as.vector(scale(myminiD[,i],center=minimum,scale=maximum-minimum))
         mytestingNORM[,i] <- as.vector(scale(mytestingD[,i],center=minimum,scale=maximum-minimum))
}
      }
summary(myminiNORM)
summary(mytestingNORM)

#this line is to run a Logistic Regression model
#please note that the glm function can handle numeric as well as categorical variables, without you needing to transform them yourself, although internally, the transformation is occurring.
LRmodel <- glm(classmembership~ ., family='binomial', myminiNORM)
#this line is to eliminate irrelevant variables
LRmodelR <- step(LRmodel)

#this line is to predict on another sample, note that we need to do a bit of more work as glm() is a function that can be used for regression and for classification
#we find first the probabilities and then we find the predicted class
#remember that the classes are "1" and "2"
probabilitiesLR <- predict(LRmodelR, mytestingNORM[,-62],type= "response")
```

```
predictionLR <- ifelse(probabilitiesLR > 0.5, "2", "1")
classificationtable <- table(pred= predictionLR, mytestingNORM[,62])
acctestLR <- sum(diag(classificationtable))/sum(classificationtable)
acctestLR

#lines to run SVM models with several kernels
…

#lines to run Classification Trees models and to prune them
….

#lines to run Random Forests models
….

#lines to run k-NN models
….
```