```
#Exercise 8 in Classification and Discriminant analysis
#Isabella Sulz
rm(list = ls())
require(randomForest)
library(mvpart)

#--------------------------------------------------------------------------------------------
------------
#--------------------------------------------------------------------------------------------
------------
#1a
filepath<-"http://www.statistik.tuwien.ac.at/public/filz/students/klassdis/bank.zip"
temp<-tempfile()
download.file(filepath,temp)
d<-read.csv2(unz(temp,"bank.csv"))
str(d)

set.seed(1337)
j<-sample(nrow(d),3000)

forest<-randomForest(y~.,data=d[j,])
plot(forest)
pred_forest<-predict(forest,d[-j,],type="class")
tab<-table(d[-j,"y"],pred_forest)
tab
#as we can see, a lot of yes is wrong predicted
mklrate <- 1-sum(diag(tab))/sum(tab)
mklrate
#mklrate for signed customers
c(tab[2,1]/sum(tab[2,]),forest$confusion[2,3])
#very high

#--------------------------------------------------------------------------------------------
------------
#--------------------------------------------------------------------------------------------
------------
#1b
#a lot of possibilites to get more yes right

#first with the cutoff parameter
forest2<-randomForest(y~.,data=d[j,],cutoff=c(0.8,0.2))
pred_forest2<-predict(forest2,d[-j,],type="class")
tab2<-table(d[-j,"y"],pred_forest2)
mklrate2 <- 1-sum(diag(tab2))/sum(tab2)
c(mklrate2,tab2[2,1]/sum(tab2[2,]))
#not much higher mklrate, but much lower wrong yes

#with quantile for predict
table(d$y)
prob_yes<-round(100*521/4000)/100
pred_forest_new <-  predict(forest,d[-j,],type="prob")
pred_forest_new2 <-  as.numeric(pred_forest_new[,2]>quantile(forest$votes[,2],prob=1-
prob_yes))
true_forest <-  as.numeric(d$y[-j])-1
tab_new <- table(true_forest,pred_forest_new2)
tab_new
c(1-sum(diag(tab_new))/sum(tab_new),tab_new[2,1]/sum(tab_new[2,]))
#equal mklrate, lower wrong signed rate
```

```
#another idea:
#the same from cutoff in forest for rpart with the prior parameter
baum<-rpart(y~.,data=d[j,],method="class",parms=list(prior=c(0.25,0.75)))
printcp(baum)
plotcp(baum)
baum_pr<-prune(baum,cp=0.018)
plot(baum_pr)
text(baum_pr,cex=0.65)
pred_baum2<-predict(baum_pr,d[-j,],type="class")
tab3<-table(d[-j,"y"],pred_baum2)
tab3
mklrate3 <- 1-sum(diag(tab3))/sum(tab3)
mklrate3
tab3[2,1]/sum(tab3[2,])
#good for signed, bad for whole mklrate
#but now we would have a strategy to follow

#-------------------------------------------------------------------------------------------
------------
#now for the full data

d1<-read.csv2(unz(temp,"bank-full.csv"))
str(d1)
set.seed(1337)
j<-sample(nrow(d1),25000)
#wanted 30000 but my laptop didn't like it

forest3<-randomForest(y~.,data=d1[j,])
forest3.1<-randomForest(y~.,data=d1[j,],cutoff=c(0.8,0.2))
#takes long!!

pred_forest3<-predict(forest3,d1[-j,],type="class")
tab4<-table(d1[-j,"y"],pred_forest3)
c(1-sum(diag(tab4))/sum(tab4),tab4[2,1]/sum(tab4[2,]))
pred_forest3.1<-predict(forest3.1,d1[-j,],type="class")
tab4.1<-table(d1[-j,"y"],pred_forest3.1)
c(1-sum(diag(tab4.1))/sum(tab4.1),tab4.1[2,1]/sum(tab4.1[2,]))
#much better, both mklrate and signed rate

#with quantiles
table(d1$y)
prob_yes2<-round(100*5289/39922)/100
pred_forest_new3 <-  predict(forest3,d1[-j,],type="prob")
pred_forest_new3.1 <-  as.numeric(pred_forest_new3[,2]>quantile(forest3$votes[,2],prob=1-
prob_yes2))
true_forest2 <-  as.numeric(d1$y[-j])-1
tab_new2 <- table(true_forest2,pred_forest_new3.1)
tab_new2
c(1-sum(diag(tab_new2))/sum(tab_new2),tab_new2[2,1]/sum(tab_new2[2,]))
#good mklrate, not so bad wrong signed rate

#rpart
baum2<-rpart(y~.,data=d1[j,],cp=0.018,method="class",parms=list(prior=c(0.25,0.75)))
pred_baum3<-predict(baum2,d1[-j,],type="class")
tab5<-table(d1[-j,"y"],pred_baum3)
c(1-sum(diag(tab5))/sum(tab5),tab5[2,1]/sum(tab5[2,]))
#high mklrate, but low wrong signed rate
```