Scenario: Study of the “Visa Premier" dataset

Summary

Modelisation (and calibration) of a classification task is illustrated in this practical session. We are interested in studying the Visa Premier dataset and aim to predict with logistic regression, Random Forest and support vector machine the behaviour of customers for a bank product.

During the practical session, please follows linearly the text. You are asked to produce a report on the Visa Premier project and send it back to me by email, before February 11 : sebastien.gadat@math.univ-toulouse.fr

Your file must be in pdf and must be called :

"firstname-lastname-m2-eee.pdf".

Some questions are indicated in red. Please do not try to answer these questions during the current practical session but at home !

1 Introduction

1.1 Pitch - Data - Aim of the study

This work is interested in a problem of prediction for a bank product : the visa premier card to optimize the advertisement of the product. The orginal data comes from a real dataset of Caisse d'Epargne. The main objective is to answer the following question : is it possible to infer from a large set of data the answer to question : will I am interested in the visa premier card ?

You will have to download the following programs (please note that - is indeed the underscore symbol)

- "transf-visprem.R"
- "lecture-visprem.R"
- "cod-visprem.R"

You will need the following files :

- "vispremR.txt"
- "visprem.txt"
- "vispremt.txt"
- "vispremv.txt"

You will need to use the following packages

- "library(rpart)"
- "library(randomForest)"
- "library(e1071)"

2 Exploratory approach

2.1 Loading the dataset

To load the dataset, use the following commands.

```
rm(list=ls(all=TRUE))

source("lecture_visprem.R")
source("transf_visprem.R")
source("cod_visprem.R")
```

Check that at the end, you have the "vispremv.txt" and "vispremt.txt" files.

Explain briefly in your report what is done in these programs (not during the practical session !)

```
varquant = names(vispremv[sapply(vispremv,is.numeric)])
varqual = names(vispremv[sapply(vispremv,is.factor)])
```

Identify several features of the dataset (number of samples, variables, nature of the variables, etc.)

2.2 Training set - Test set

Build the training set and the test set (below XXX must be a reference number you have to choose to initialize the random generating number of R).

```
set.seed(XXX)
npop<-nrow(vispremv)
testi<-sample(1:npop,200)
```

```
appri<-setdiff(1:npop,testi)
visappt<-vispremv[appri,]
vistest<-vispremv[testi,]
visapptq=visappt[,c(varqual)]
vistestq=vistest[,c(varqual)]
visapptr=visappt[,c("CARVP",varquant)]
vistestr=vistest[,c("CARVP",varquant)]
```

Explain briefly in your report why we should do such a sub-sampling procedure. Comment on the size of the several subsets.

2.3 Descriptive result

In less than 10 lines, give a descriptive summary result of the dataset with PCA.

3 Classification with CART

We begin with a standard classification with tree.

3.1 On qualitative variables

Run the following code.

```
library(rpart)
vis.treeq=rpart(CARVP~.,data=visapptq,
parms=list(split='information'),cp=0.001)
```

Remind some theoretical elements on the algorithm. What is the "information" option ? What is cp ?

It is possible to obtain some graphical representation.

```
summary(vis.treeq)
plot(vis.treeq)
text(vis.treeq)
```

What are the conclusions for predicting well the visa variable on this particular example ?

3.2 Pruning

Why is it important to prune the tree ? Run the following command where xxx is replaced by a suitable value of the parameter Cp

```
plotcp(vis.treeq)

vis.treeq.cut=prune(vis.treeq,cp=xxx)

plot(vis.treeq.cut)
text(vis.treeq.cut)
```

Note the optimal value for pruning to compare the CART algorithm with other classification methods.

Comment on these lines, on the interest of pruning, and on the conclusions for this algorithm.

3.3 On quantitative variables

Follow the same guideline on quantitative variables.

```
visapptr=visappt[,c("CARVP",varquant)]
vistestr=vistest[,c("CARVP",varquant)]

vis.treer=rpart(CARVP~.,data=visapptr,
parms=list(split='information'),cp=0.001)

plot(vis.treer)
text(vis.treer)

...
```

Note the optimal value for pruning to compare the CART algorithm with other classification methods.

Each time, explain the R commands you will use, the theoretical construction for selecting variables at each node of the tree, the pruning, etc.

3.4 Important variables

Using the previous lines, what are the variables that seem important for prediction ?

3.5 Pruning with cross validation

Explain the principle of cross validation. Help yourself with www.

```
xmat=xpred.rpart(vis.treeq,xval=10,
        cp=seq(0.05,0.001,length=20))
xerr=as.integer(visapptq$CARVP)!=xmat
apply(xerr,2,sum)/nrow(xerr)

vis.treeq.cut2=rpart(CARVP~.,data=visapptq,
parms=list(split='information'),cp=XXX)
plot(vis.treeq.cut2)
text(vis.treeq.cut2)
```

3.6 Prediction on the test set

Using the previous selected models of CART, we now compute the final error of classification on the test set.

```
pred.vistestq=predict(vis.treeq.cut,
        newdata=vistestq,type="class")
table(pred.vistestq,vistestq$CARVP)

pred.vistestr=predict(vis.treer.cut,
        newdata=vistestr,type="class")
table(pred.vistestr,vistestr$CARVP)
```

Discuss briefly on the average performance of the algorithm.

4 Random Forest

4.1 Remainders

Explain briefly the principle of the Random Forest method. What are the important steps ? What is the major improvement comparing to CART ? What are the important parameters to choose ?

4.2 Random Forest with R

```
fit=randomForest(CARVP~.,data=visapptr,
xtest=vistestr[,varquant],ytest=vistestr[,"CARVP"],
do.trace=20,importance=TRUE,norm.vote=FALSE)

print(fit)
```

Read carefully the output of R. Many informations are shown. Please comment on the nature of the results obtained with your computer.

We can use many many trees in the forest :

```
fit=randomForest(CARVP~.,data=visapptr,
xtest=vistestr[,varquant],ytest=vistestr[,"CARVP"],
do.trace=20,importance=TRUE,norm.vote=FALSE,ntree=5000)
```

What is overfitting ? What are the typical situations where a classifier overfits ? Is there any overfitting in this example ?

4.3 Optimization with cross validation

The following lines make it possible to obtain an automatic calibration of the random forest algorithm with cross validation.

```
library(e1071)

res=tune(randomForest, CARVP~., data=visapptr,
tunecontrol=tune.control(sampling="cross",cross=10),
ranges=list(mtry=c(10,17,24),ntree=c(200,400,600)))
```

Note the optimal parameters, the misclassification error rate, the optimal parameters to be used, and the final average performance.

4.4 Variable selection with RF

It is possible to produce a variable selection with the R package. Help yourself to find how !

5 Logistic regression

5.1 On the Visa premier dataset

We detail in this practical session an old-fashioned method : the logistic regression.

Briefly remind the principle of such algorithm. Is it a regression method ? What is minimized ? What is the tecnical algorithm behind the optimization procedure ?

```
library(MASS)
var=names(vispremv)
varquant=var[1:30]
varqual=var[31:54]

visapptq=visappt[,c("CARVP",varqual)]
vistestq=vistest[,c("CARVP",varqual)]

vraisemblance.
visa.logit=glm(CARVP ~.,data=visapptq,
 family=binomial, na.action=na.omit)
```

What is produced by R ?

5.2 Variable selection

```
anova(visa.logit,test="Chisq")

visa.logit=glm(CARVP ~.,data=visapptq,
        family=binomial, na.action=na.omit)
visa.step <-step(visa.logit)

anova(visa.step,test="Chisq")
```

What is the principle of the variable selection with the logistic regression algorithm ? What is the AIC ?

5.3 Calibration with cross validation

Again, it is possible to compute the average performence of the logistic regression with a cross validation algorithm.

```
library(boot)

visa1.logit=glm(CARVP~SEXEQ+PCSPQ+kvunbq+uemnbq+nptagq+
endetq+gagetq+facanq+havefq+relatq+qsmoyq+opgnbq+
moyrvq+dmvtpq+boppnq+jnbjdq+itavcq, data=visapptq,
family=binomial, na.action=na.omit)

cv.glm(visapptq,visa1.logit,K=10)$delta[1]

anova(visa1.logit,test="Chisq")

visa2.logit=glm(CARVP~SEXEQ+PCSPQ+kvunbq+uemnbq+nptagq+
gagetq+facanq+havefq+relatq+qsmoyq+opgnbq+moyrvq+dmvtpq+
boppnq+endetq+itavcq, data=visapptq,
family=binomial, na.action=na.omit)

cv.glm(visapptq,visa2.logit,K=10)$delta[1]
anova(visa2.logit,test="Chisq")

...
```

Iterate the process until the CV classification rate is deteriorated by the selection procedure... And compute the final error rate on the test set !

6 Support vector machine

We end the practical session with an illustration of the support vector machine algorithm. You will have to do it almost alone !

6.1 First run

```
library(e1071)
```

```
vis.svm=svm(CARVP~., data=visapptr)
summary(vis.svm)

vis.pred=predict(vis.svm,data=visapptr)

table(vis.pred,visapptr$CARVP)
vis.pred=predict(vis.svm,newdata=vistestr)

table(vis.pred,vistestr$CARVP)
```

Explain the effect of each line above and the several computations of the error rates. What is the kernel used ? What is the value of C ? Etc.

6.2 Optimization

Try to optimize the behaviour of the SVM algorithm with the tune.svm command.

A not so bad calibration :

```
vis.svm=svm(CARVP~., data=visapptr,
gamma=0.015,cost=6)
```

Try other kernels with the option "kernel"... (polynomial, sigmoid, etc.)

6.3 Conclusion

Is the SVM method easy to use in this example ? Compare with the other (miss)-classification rates obtained by other methods.