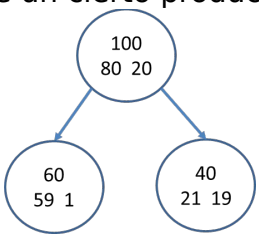
ÁRBOLES DE DECISIÓN

Xavier Gonzalez Arriola

1. Suponga el siguiente árbol simple T con sólo dos nodos (hojas) terminales. En el nodo raíz se tiene 100 individuos que se dividen en dos nodos hijos de 60 y 40 individuos cada uno. La variable de respuesta indica la compra (No o Si) de un cierto producto:



Calcule la reducción de impureza que se obtiene al pasar del nodo padre a los dos nodos hijos.

La impureza del padre:

$$i(t)=1-(80/100)^2-(20/100)^2=1-0.64-0.04=0.32$$

Impureza del *left* child:

$$i(t_L)=1-(59/60)^2-(1/60)^2=0.032778$$

Impureza del *right* child:

$$i(t_R)=1-(21/40)^2-(19/40)^2=0.49875$$

Reducción de impureza:

$$\Delta i(t)=i(t)-n_{tL}/n_t\, i(t_L)-n_{tR}/n_t\, i(t_R)=0.32-60/100\, i(t_L)-40/100\, i(t_R)=0.10083$$

2. Con el mismo árbol precedente, calcule su coste de mal clasificación $R(T)$.

$$r(t_L)=1-Max(p(j/t))=1-59/60=0.01667$$

$$r(t_R)=1-Max(p(j/t))=1-21/40=0.475$$

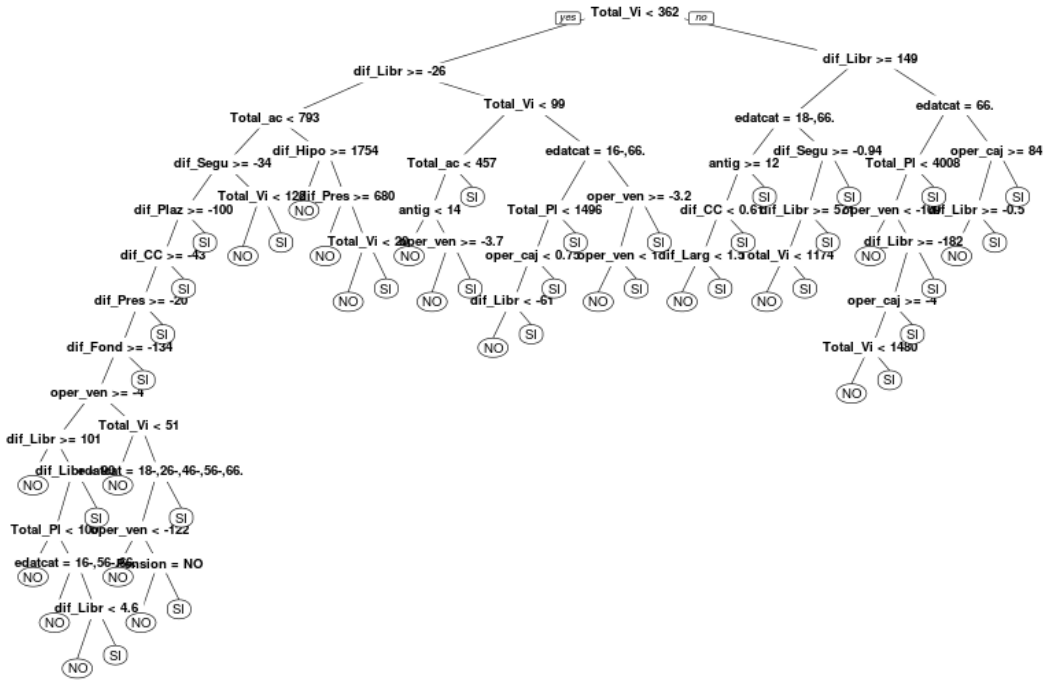
$$r(t_{ROOT})=1-Max(p(j/t))=1-80/100=0.2$$

$$R(T)=\frac{\sum(p(t)r(t))}{r(t_{ROOT})}=\frac{0.6\times r(t_L)+0.4\times(t_R)}{r(t_{ROOT})}=\frac{0.6\times 0.01667+0.4\times 0.475}{0.2}=1.00001$$

3. Retome los datos del problema *churn*. Se trata ahora de obtener un árbol de decisión que nos permita efectuar predicciones sobre la probabilidad de baja de los clientes. Cargue en R la Liberia *rpart* y obtenga un árbol máximo (cp=0.0001) con crossvalidación (xval=10).

```
# set working directory
setwd("/media/xabee/XABEE_USB/BIG_DATA/Analytics/Sessio_02_PROFILING/exer2")

# load libraries
print("loading libraries....")
library(rpart)
library(rpart.plot)
library(rattle)

#  read "churn.txt"
df <- read.table("churn.txt", header=TRUE)
```

```
############################################################## TREE

n <- nrow(dd)

# 2/3 OF THE DATA TO BUILD THE OPTIMAL DECISION TREE AND THE REMAINING 1/3 TO ASSESS THE QUALITY

# learn <--- training individuals

set.seed(7)
learn <- sample(1:n, round(0.67*n))

nlearn <- length(learn)
ntest <- n - nlearn

# selection of decision tree by crossvalidation
# 1st maximal tree cp=0.0001, val=10

set.seed(27)
t = rpart(Baja ~ ., data=dd[learn,], control=rpart.control(cp=0.0001, xval=10))
rpart.plot::prp(t, cex=0.6)
...
```



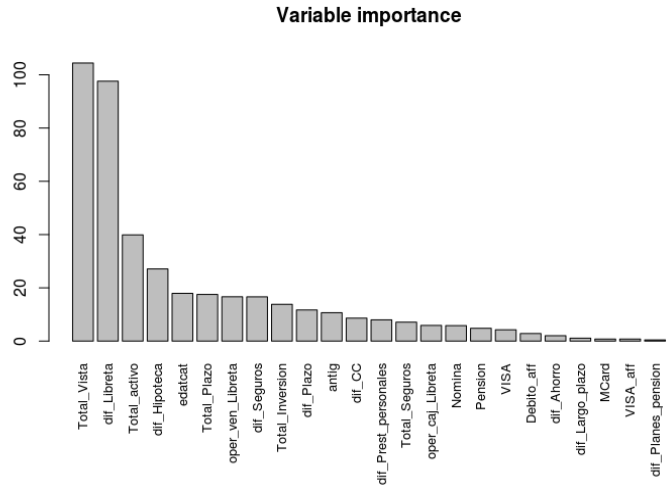
4. Determine ahora el árbol óptimo y su valor del *complexity parameter* (cp). Diga cuales son las variables más importantes en la definición del árbol óptimo.

Arbol óptimo:

```
...
t$cptable=as.data.frame(t$cptable)
ind = which.min(t$cptable$xerror)
xerr <- t$cptable$xerror[ind]
xstd<-t$cptable$xstd[ind]
i=1
while (t$cptable$xerror[i]>xerr+xstd) i=i+1
alfa=t$cptable$CP[i]
alfa (0.008982036, complexity parameter)
p=prune(t,cp=alfa)
...
```

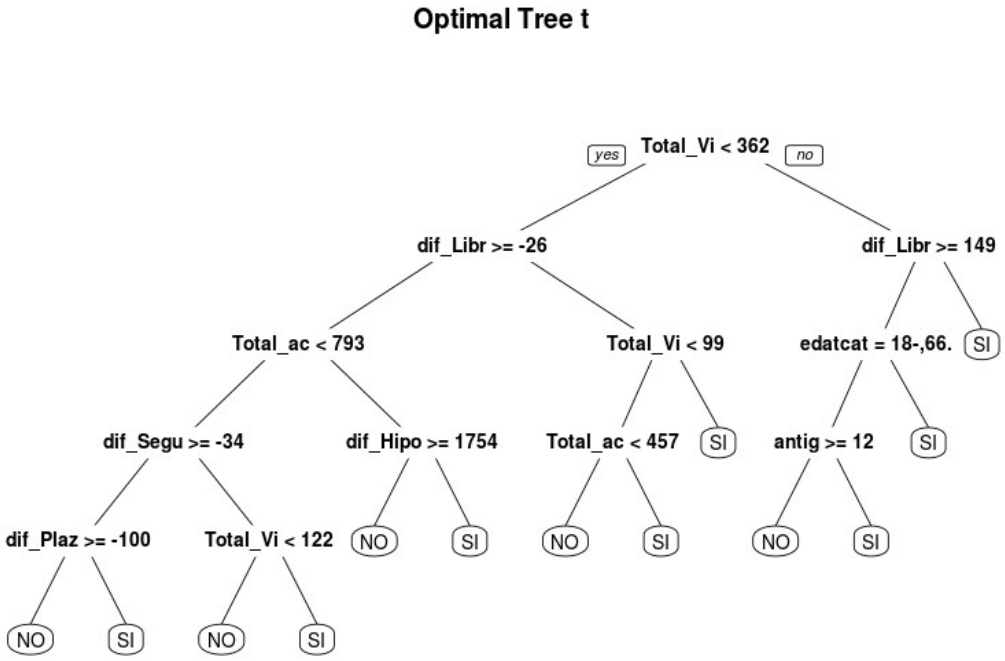
Variables más importantes:

```
...
par(mai=c(2,1,1,1))
barplot(t$variable.importance, las=3, cex.main=0.1, cex.lab=0.1, cex.axis=0.75, cex.names=0.8)
...
```



5. Represente gráficamente el árbol óptimo y liste sus reglas de decisión.

```
...
> rpart.plot::prp(p, main="Optimal Tree t")
...
```



Reglas de decisión:

```
> asRules(p)
```

```
Rule number: 21 [Baja=SI cover=16 (1%) prob=1.00]
  Total_Vista< 361.5
  dif_Libreta< -25.54
  Total_Vista< 99
  Total_activo>=457
```

```
Rule number: 35 [Baja=SI cover=21 (2%) prob=0.95]
 Total_Vista< 361.5
 dif_Libreta>=-25.54
 Total_activo< 793
 dif_Seguros< -34
 Total_Vista>=121.5

Rule number: 33 [Baja=SI cover=10 (1%) prob=0.90]
 Total_Vista< 361.5
 dif_Libreta>=-25.54
 Total_activo< 793
 dif_Seguros>=-34
 dif_Plazo< -100

Rule number: 7 [Baja=SI cover=299 (22%) prob=0.87]
 Total_Vista>=361.5
 dif_Libreta< 148.9

Rule number: 25 [Baja=SI cover=16 (1%) prob=0.81]
 Total_Vista>=361.5
 dif_Libreta>=148.9
 edatcat=18-25,66..
 antig< 11.5

Rule number: 19 [Baja=SI cover=64 (5%) prob=0.78]
 Total_Vista< 361.5
 dif_Libreta>=-25.54
 Total_activo>=793
 dif_Hipoteca< 1754

Rule number: 11 [Baja=SI cover=124 (9%) prob=0.74]
 Total_Vista< 361.5
 dif_Libreta< -25.54
 Total_Vista>=99

Rule number: 13 [Baja=SI cover=103 (8%) prob=0.71]
 Total_Vista>=361.5
 dif_Libreta>=148.9
 edatcat=16-17,26-35,36-45,46-55,56-65

Rule number: 20 [Baja=NO cover=59 (4%) prob=0.29]
 Total_Vista< 361.5
 dif_Libreta< -25.54
 Total_Vista< 99
 Total_activo< 457

Rule number: 24 [Baja=NO cover=82 (6%) prob=0.24]
 Total_Vista>=361.5
 dif_Libreta>=148.9
 edatcat=18-25,66..
 antig>=11.5

Rule number: 32 [Baja=NO cover=524 (39%) prob=0.19]
 Total_Vista< 361.5
 dif_Libreta>=-25.54
 Total_activo< 793
 dif_Seguros>=-34
 dif_Plazo>=-100

Rule number: 34 [Baja=NO cover=11 (1%) prob=0.18]
 Total_Vista< 361.5
 dif_Libreta>=-25.54
 Total_activo< 793
 dif_Seguros< -34
 Total_Vista< 121.5

Rule number: 18 [Baja=NO cover=11 (1%) prob=0.00]
 Total_Vista< 361.5
 dif_Libreta>=-25.54
 Total_activo>=793
 dif_Hipoteca>=1754
```

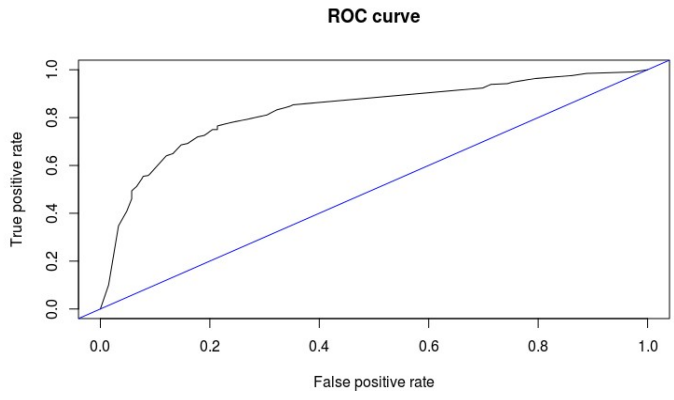
6. Las probabilidades de baja no están por fortuna equidistribuidas, sino que la probabilidad de baja es muy inferior (un 5%). Exporte a Excel la tabla de resultados por hoja y pondere estos resultados de acuerdo con las probabilidades a priori mencionadas. Obsérvese que en este caso no utilizamos una muestra test de validación del árbol obtenido (en general deberíamos obtener la predicción del árbol en una muestra independiente (test) y validar la calidad del árbol con los resultados obtenidos en esta muestra test).

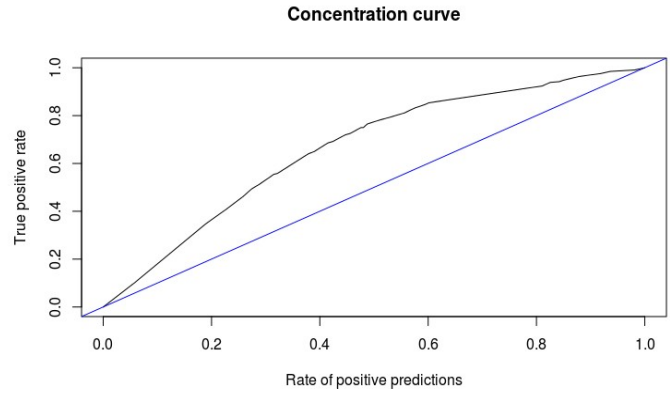
	n	n1	n2	p2
35	24	1	23	0.9583333333
21	23	2	21	0.9130434783
33	15	2	13	0.8666666667
7	452	64	388	0.8584070796
25	23	5	18	0.7826086957
11	179	48	131	0.7318435754
19	106	31	75	0.7075471698
13	166	50	116	0.6987951807
20	85	54	31	0.3647058824
24	120	88	32	0.2666666667
34	15	11	4	0.2666666667
32	777	630	147	0.1891891892
18	15	14	1	0.0666666667

....

7. Obtenga gráficamente las curvas de concentración y ROC correspondientes.

```
> library(ROCR)
> pred_test = as.data.frame(predict(p1, newdata=dd[-learn,],type="prob"))
> pred <- prediction(pred_test$positiu, dd$Dictamen[-learn])
> roc <- performance(pred,measure="tpr",x.measure="fpr")
> plot(roc, main="ROC curve")
> abline(0,1,col="blue")
```





8. Decida un umbral de decisión para la predicción de “baja” y obtenga el “error_rate”, la precisión en la predicción positiva, la precisión en la predicción negativa, el promedio de ambas precisiones y el Recall asociado al umbral escogido.

Tomamos un umbral de 0.5

	dec.test	
	pred_neg.test	pred_pos.test
NO	248	84
SI	77	251

$$Error\ rate = \frac{n_{FN} + n_{FP}}{n} = \frac{84+77}{660} = 0.2439$$

$$Precision_P = \frac{n_{TP}}{n_{TP} + n_{FP}} = \frac{251}{251+84} = 0.7493$$

$$Precision_N = \frac{n_{TN}}{n_{TN} + n_{FN}} = \frac{248}{248+77} = 0.7631$$

$$\overline{Precision} = \frac{Precision_P + Precision_N}{2} = 0.7562$$

$$Recall = \frac{n_{TP}}{n_{TP} + n_{FN}} = \frac{251}{251+77} = 0.7652$$