Naive Bayes and BAyesian Networks.

By Saba Akram

3/9/2021

churn <- read.csv("D:/M.Sc in Banking and Financial Analytics/Sem 3/Data Analytic and Machine learning/Data Mining and Predictive Analysis/Data sets/churn.txt", stringsAsFactors=TRUE)  
  
n<-dim(churn)[1]  
n

## [1] 3333

p.int.plan<-sum(churn$Int.l.Plan=="yes")/n  
p.int.plan

## [1] 0.09690969

p.vmail.plan<-sum(churn$VMail.Plan=="yes")/n  
p.vmail.plan

## [1] 0.2766277

p.churn<-sum(churn$Churn.=="True.")/n  
p.churn

## [1] 0.1449145

# Now we have calculated the probabilities of the international plan, vmail plan and the churn.  
  
# Now we will try to calculate the conditional probabilities.

# Calculate the conditional probabilities.  
  
n.churnT<-length(churn$Churn.[which(churn$Churn.=="True.")])  
n.churnT

## [1] 483

n.churnF<-length(churn$Churn.[which(churn$Churn.=="False.")])  
n.churnF

## [1] 2850

p.int.plan.g.ChurnT<-sum(churn$Int.l.Plan[which(churn$Churn.=="True.")]=="yes")/n.churnT  
p.int.plan.g.ChurnT

## [1] 0.2836439

p.int.plan.g.ChurnF<-sum(churn$Int.l.Plan[which(churn$Churn.=="False.")]=="yes")/n.churnF  
p.int.plan.g.ChurnF

## [1] 0.06526316

p.vmail.plan.g.ChurnT<-sum(churn$VMail.Plan[which(churn$Churn.=="True.")]=="yes")/n.churnT  
p.vmail.plan.g.ChurnT

## [1] 0.1656315

p.vmail.plan.g.ChurnF<-sum(churn$VMail.Plan[which(churn$Churn.=="False.")]=="yes")/n.churnF  
p.vmail.plan.g.ChurnF

## [1] 0.2954386

# Calculating Posterior Probablities  
  
p.churn.g.int.plan<-(p.int.plan.g.ChurnT\*p.churn)/p.int.plan  
p.churn.g.int.plan

## [1] 0.4241486

p.churn.g.vmail.plan<-(p.vmail.plan.g.ChurnT\*p.churn)/p.vmail.plan  
p.churn.g.vmail.plan

## [1] 0.0867679

## We can also calculate the posterior probablities directly.  
  
n.int<-sum(churn$Int.l.Plan=="yes")  
n.vmail<-sum(churn$VMail.Plan=="yes")  
p\_c\_t\_g\_int\_y<-sum(churn$Churn.[which(churn$Int.l.Plan=="yes")]=="True.")/n.int  
p\_c\_t\_g\_int\_y

## [1] 0.4241486

p\_c\_t\_g\_vmail\_y<-sum(churn$Churn.[which(churn$VMail.Plan=="yes")]=="True.")/n.vmail  
p\_c\_t\_g\_vmail\_y

## [1] 0.0867679

## Joint Conditional Probabilities.  
i.v<-i.vbar<-ibar.v<-ibar.vbar<-rep("no",n)  
for (i in 1:n) {  
 if(churn$Int.l.Plan[i]=="yes"&&churn$VMail.Plan[i]=="yes")i.v[i]<-"yes"  
 if(churn$Int.l.Plan[i]=="yes"&&churn$VMail.Plan[i]=="no")i.vbar[i]<-"yes"  
 if(churn$Int.l.Plan[i]=="no"&&churn$VMail.Plan[i]=="yes")ibar.v[i]<-"yes"  
 if(churn$Int.l.Plan[i]=="no"&&churn$VMail.Plan[i]=="no")ibar.vbar[i]<-"yes"  
}  
  
table.iv<-table(i.v,churn$Churn.,dnn = c("Int\_vmail","Churn"))  
table.iv

## Churn  
## Int\_vmail False. True.  
## no 2794 447  
## yes 56 36

table.i.vbar<-table(i.vbar,churn$Churn.,dnn = c("int\_vmail.bar","Churn"))  
table.i.vbar

## Churn  
## int\_vmail.bar False. True.  
## no 2720 382  
## yes 130 101

table.ibar.v<-table(ibar.v,churn$Churn.,dnn = c("int.bar\_vmail","Churn"))  
table.ibar.v

## Churn  
## int.bar\_vmail False. True.  
## no 2064 439  
## yes 786 44

table.ibar.vbar<-table(ibar.vbar,churn$Churn.,dnn = c("int.bar\_vmail.bar","churn"))  
table.ibar.vbar

## churn  
## int.bar\_vmail.bar False. True.  
## no 972 181  
## yes 1878 302

prob\_iv\_churnT<-table.iv[4]/(table.iv[3]+table.iv[4])  
prob\_iv\_churnT

## [1] 0.07453416

prob\_iv\_churnF<-table.iv[2]/(table.iv[1]+table.iv[2])  
prob\_iv\_churnF

## [1] 0.01964912

prob\_iv\_churnT\*p.churn

## [1] 0.01080108

prob\_iv\_churnF\*(1-p.churn)

## [1] 0.01680168

## Posterior Odds Ratio  
(prob\_iv\_churnT\*p.churn)/(prob\_iv\_churnF\*(1-p.churn))

## [1] 0.6428571

## Balancing the data  
  
b.churn<-churn[which(churn$Churn.=="True."),]  
notchurn<-churn[which(churn$Churn.=="False."),]  
choose<-runif(dim(notchurn)[1],0,1)  
halfnotchurn<-notchurn[which(choose<0.5),]  
b.churn<-rbind(b.churn,halfnotchurn)  
  
## Updated probabilities  
  
nn<-dim(b.churn)[1]  
p.churnn<-sum(b.churn$Churn.=="True.")/nn  
p.churnn

## [1] 0.2559618

## Joint Probability distribution  
iv<-rep("no",nn)  
for (i in 1:nn) {  
 if(b.churn$Int.l.Plan[i]=="yes"&&b.churn$VMail.Plan[i]=="yes")iv[i]<-"yes"  
}  
  
table\_iv<-table(iv,b.churn$Churn.,dnn = c("int\_vmail","Churn"))  
table\_iv

## Churn  
## int\_vmail False. True.  
## no 1371 447  
## yes 33 36

p\_iv\_churnT<-table\_iv[2,2]/(table\_iv[1,2]+table\_iv[2,2])  
p\_iv\_churnF<-table\_iv[2,1]/(table\_iv[2,1]+table\_iv[1,1])  
p\_iv\_churnT\*p.churnn

## [1] 0.0190779

p\_iv\_churnF\*(1-p.churnn)

## [1] 0.01748808

## Naive Bayes classification using original Churn data  
  
n<-dim(churn)[1]  
p.churn<-sum(churn$Churn.=="True.")/n  
p.churn

## [1] 0.1449145

n.churnT<-length(churn$Churn.[which(churn$Churn.=="True.")])  
n.churnT

## [1] 483

n.churnF<-length(churn$Churn.[which(churn$Churn.=="False.")])  
n.churnF

## [1] 2850

p.int.plan.g.ChurnT<-sum(churn$Int.l.Plan[which(churn$Churn.=="True.")]=="yes")/n.churnT  
p.int.plan.g.ChurnT

## [1] 0.2836439

p.int.plan.g.ChurnF<-sum(churn$Int.l.Plan[which(churn$Churn.=="False.")]=="yes")/n.churnF  
p.int.plan.g.ChurnF

## [1] 0.06526316

p.vmail.plan.g.ChurnT<-sum(churn$VMail.Plan[which(churn$Churn.=="True.")]=="yes")/n.churnT  
p.vmail.plan.g.ChurnT

## [1] 0.1656315

p.vmail.plan.g.ChurnF<-sum(churn$VMail.Plan[which(churn$Churn.=="False.")]=="yes")/n.churnF  
p.vmail.plan.g.ChurnF

## [1] 0.2954386

p.int.plan.g.ChurnT\*p.vmail.plan.g.ChurnT\*p.churn

## [1] 0.006808134

p.int.plan.g.ChurnF\*p.vmail.plan.g.ChurnF\*(1-p.churn)

## [1] 0.01648712

## Log Posterior Odds Ratio  
  
log(p.int.plan.g.ChurnT/p.int.plan.g.ChurnF)

## [1] 1.469292

log(p.vmail.plan.g.ChurnT/p.vmail.plan.g.ChurnF)

## [1] -0.5786958

## Numeric Predictors for Naive Bayes Classification  
  
p.churnT.t800<-dnorm(800,mean = 635,sd=111)  
p.churnF.t800<-dnorm(800,mean = 585,sd=84)  
  
pivt800givenChurnT<-p.int.plan.g.ChurnT\*p.vmail.plan.g.ChurnF\*p.churnT.t800\*p.churn  
pivt800givenChurnF<-p.int.plan.g.ChurnF\*p.vmail.plan.g.ChurnF\*p.churnF.t800\*(1-p.churn)  
  
  
pivt800givenChurnT/pivt800givenChurnF

## [1] 4.885545