qbs124_hw2_gibran

Gibran Erlangga

4/9/2022

Use mortgageROC.csv data from Example 5.5 of the book.

```
data <- read.csv('mortgageROC.csv')
head(data)
```

```
##   Default FamilyIncome
## 1     yes           27
## 2     yes           36
## 3     yes           44
## 4     yes           25
## 5     yes           43
## 6     yes           40
```

Question 1

(20 points). Plot the two empirical cdfs with family income on the log scale with actual numbers displayed in thousand dollars. Use rug, legend, and different colors.

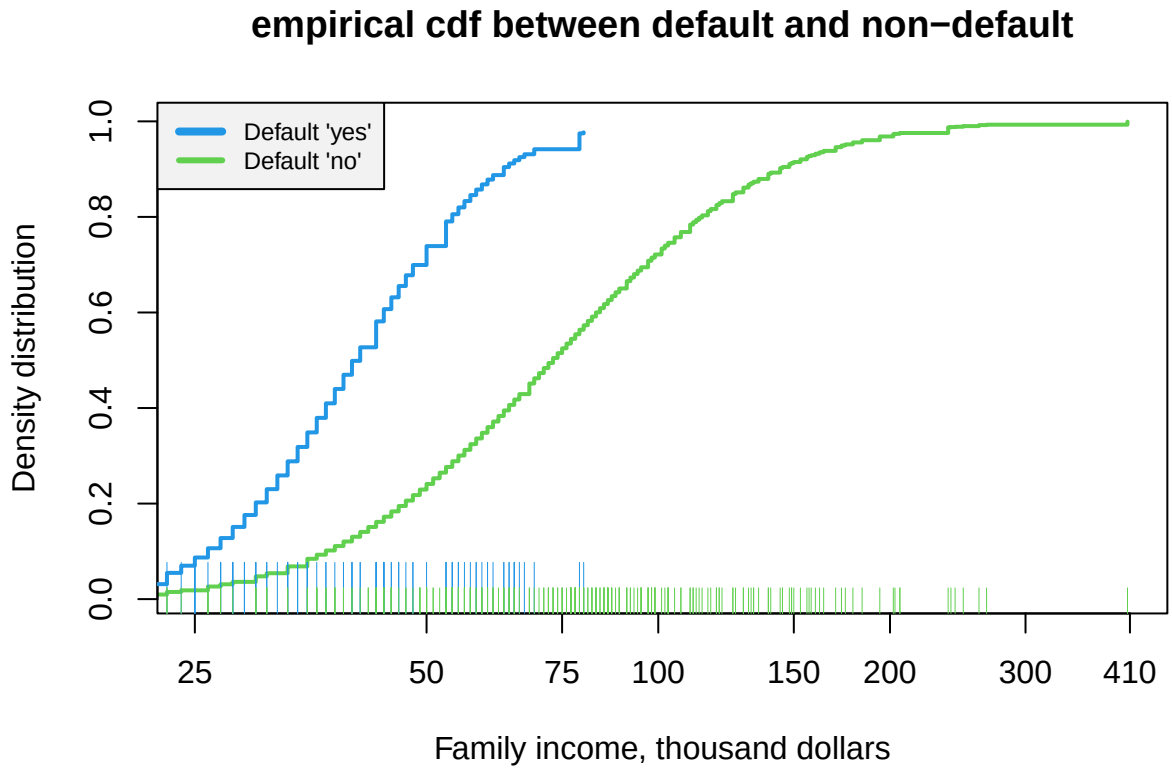
```
data$FamilyIncomeLog = log(data$FamilyIncome)

#a = default, b = no default
a = sort(data$FamilyIncomeLog[data$Default=="yes"])
b = sort(data$FamilyIncomeLog[data$Default=="no"])
label_range = range(c(2.5:7))

# compare cdfs
dollar_val = c(25,50,75,100,150,200,300,410)
xlabel_log = log(dollar_val)
label_range = range(xlabel_log)
x=seq(from=label_range[1],to=label_range[2],length=279)
plot(b, pnorm(b,mean=mean(b),sd=sd(b)), xlim=label_range,
     col=3, lwd=2, type='s',
     main="empirical cdf between default and non-default",
     xlab='Family income, thousand dollars',
     xaxt = "n",
     ylab='Density distribution')
axis(side=1,at=xlabel_log,labels=as.character (dollar_val))
lines(a, pnorm(a,mean=mean(a),sd=sd(a)), col=4, lwd=2, type='s')
rug(a,ticksize=0.1, col=4)
```

```
## Warning in rug(a, ticksize = 0.1, col = 4): some values will be clipped
```

```
rug(b,ticksize=0.05, col=3)
```

```
## Warning in rug(b, ticksize = 0.05, col = 3): some values will be clipped
```

```
legend("topleft",c("Default 'yes'","Default 'no'"),
       col=c(4,3),lwd=c(4,3),lty=1,cex=0.75,bg="gray95")
```



Question 2 (20 points). Create an R animation where at left you show empirical cdfs from the previous task and the growing stepwise ROC curve at right (use type="s") as in cdf.dyn(job=3). Submit as a standalone gif file.

```
n=100
# a => income for non-defaulters
X = b
# a => income for defaulters
Y = a
XY=sort(c(X,Y))
X=sort(X);Y=sort(Y)
nX=length(X);nY=length(Y)
n=length(XY)
th=XY
niY=niX=rep(NA,n)
for(i in 1:n)
```

```
{
  ch=as.character(i)
  if(i<10) ch=paste("00",ch,sep="")
  if(i>=10 & i<100) ch=paste("0",ch,sep="")
  jpeg(paste("cdf",ch,".jpg",sep=""),width=1200,height=600)
  par(mfrow=c(1,2),mar=c(4.5,4.5,3,1),cex.lab=1.75,cex.main=1.75)
  dollar_val = c(25,50,75,100,150,200,300,410)
   xlabel_log = log(dollar_val)
   label_range = range(xlabel_log)
   x=seq(from=label_range[1],to=label_range[2],length=279)
  plot(XY,XY,type="n",ylim=c(0,1),
       xlab="Family income, thousand dollars",
       xaxt = "n",
       ylab="Proportion",
       main="Two CDFs for uniform data comparison: Y < X")
  axis(side=1,at=xlabel_log,labels=as.character (dollar_val))
  legend("topleft",c("Data","Threshold","cdf of default yes","cdf of default no"),col=c(1,1,3,2),lwd=c(
  rug(X,ticksize=0.075,col=2)
  rug(Y,ticksize=0.05,col=3)
  segments(th[i],-1,th[i],.05,lwd=3)
  segments(min(XY)-1,0,th[i],0)
  Xi=c(X[X<=th[i]],th[i])
  niX[i]=length(Xi)
  lines(Xi,(1:niX[i])/nX,type="s",col=2,lwd=2)
  Yi=c(Y[Y<=th[i]],th[i])
  niY[i]=length(Yi)
  lines(Yi,(1:niY[i])/nY,type="s",col=3,lwd=2)

  plot(niX[1:i]/nX,niY[1:i]/nY,xlim=c(0,1),ylim=c(0,1),lwd=3,type="s",xlab="1-Specificity (false positi
  segments(-1,-1,2,2,col=4)

  dev.off()
}

#I couldn't figure out magick in mac so I made the gif file from online source (ezgif.com)
```

Question 3

(20 points). (a) Compute AUC and provide its layman interpretation.

```
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 4.1.1
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
data$default_flag<-ifelse(data$Default=="yes",1,0)
model<-glm(default_flag~FamilyIncome, family="binomial", data=data)
predicted <- predict(model, data, type="response")

paste("AUC score:", round(auc(data$default_flag, predicted), 4))
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
## [1] "AUC score: 0.822"
```

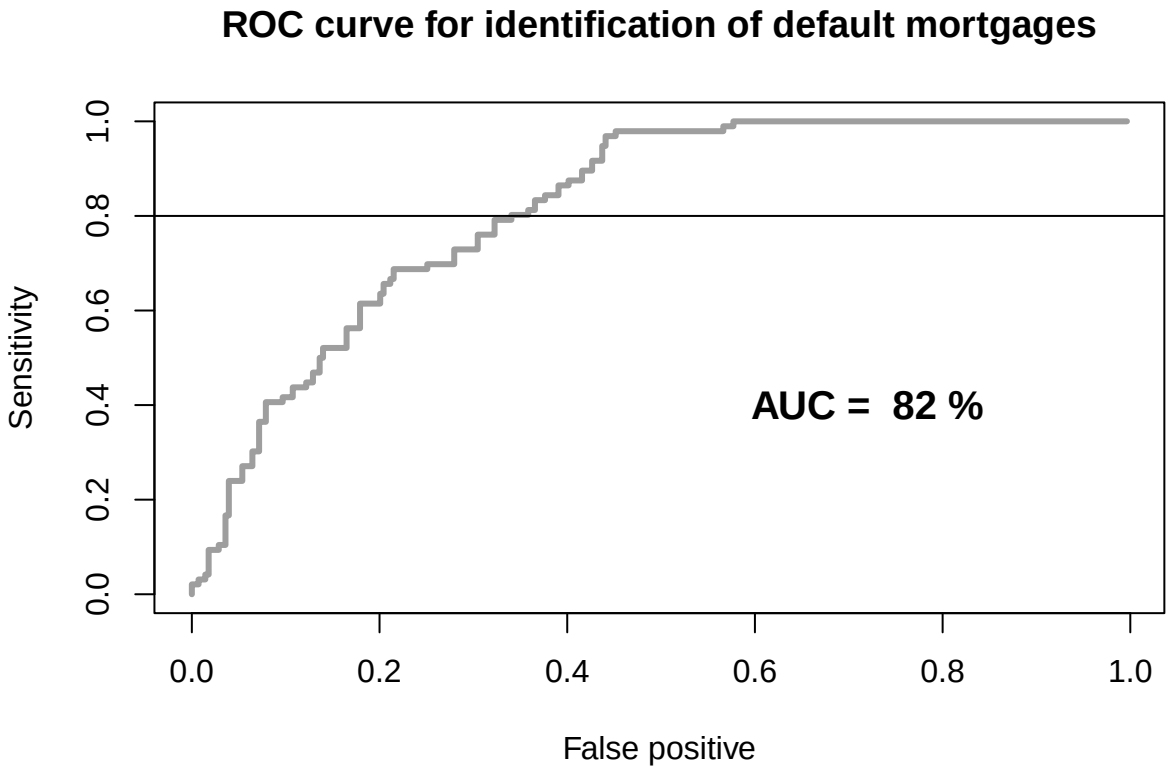
AUC is the probability of Y < X, where Y represents the positive instance and X represents the negative instance. The AUC score in this particular case is 0.822 / 82.2%, which is a relatively high AUC value. An AUC score of 82% indicates that the probability of Y ranking higher than X is 82%.

(b) Compute and provide a layman interpretation for True Positive, True negative, False Positive, and False negative when Sensitivity=0.8.

```
n_income <- length(data$FamilyIncome)

# prep data
default<- data[,1]
income <- data[,2]

income
```

```
##   [1]  27  36  44  25  43  40  41  63  25  66  30  25  30  44  43  35  27  31
##  [19]  65  43  32  27  37  21  33  30  64  60  39  25  57  56  54  54  35  53
##  [37]  67  41  27  55  55  35  79  29  46  16  37  63  33  28  53  29  45  20
##  [55]  23  33  26  69  58  29  50  58  44  46  64  33  34  59  24  43  40  48
##  [73]  60  45  44  54  65  47  27  38  80  35  53  65  55  40  44  28  48  28
##  [91]  28  27  31  61  34  39  56  92 112  63  95  57 119  95 116  95  78  27
## [109]  50  89  56  61  57 157  49  56  71  78  77  68  33  84  91  55  98  48
## [127]  85  50  30 125  99  75 148  31  75  21 144  45  36  45  39  26  43  65
## [145] 110  86  38  51 184  86  37 261  89 107 121  78  72  43  36 240  50 110
## [163] 113 156 175  99 129  40  98 129  79  57  79  53  54 125 206  30  31  42
## [181]  82  45  85  79  56  53  36  98 206  57  36  50  97 103 267  35  72  63
## [199]  37  50  36  80 120  31  56 145  30 153  33  39  59  44 107  55 101  51
## [217]  83 173 145  86  77 162  43  97  53 238  60  66 114  94  38  60  86  30
## [235]  49  62  42  44  87  59  61 133  81 132 117  98  60  88  89 249  54 139
## [253]  75  47 179  92  28  53  69  24 170 102  55  55  83  43  54  72  74 126
## [271]  85  65  58 133  49  93  81  78 111  45  70  63  23  89 158  38 160  78
## [289] 140  46  99  52  50  85 135  82  54 131  60 105  37 103  47  58  35 407
## [307]  38  77  73  45  80  97 103  23  45  64  68  54  91  71  44 111  59  73
## [325] 203  77  43  61  76  40  27  55  91  44  99 150  77 164  80  53  65 243
## [343] 120  83  78  72  82 149  95  73  21  72  83  87  78  74  26 149  84  87
## [361] 103  86  26  41 119 202  54  86 194  64 139  54  49  64  57
```

```
income0 <- sort(income[default=="no"])
n0 <- length(income0)
m0 <- mean(income0)
s0 <- sd(income0)
income1 <- sort(income[default=="yes"])
n1 <- length(income1)
m1 <- mean(income1)
s1 <- sd(income1)
income<- sort(income)

sens=fp=toter10=rep(NA,n)
AUC=toter=0
for (i in 1:n) {
  sens[i] <- sum(income1 < income[i])/n1
  fp[i] <- sum(income0 < income[i])/n0
  toter10[i] <- (4.5/10)*(1-sens[i])+(5.5/10)*fp[i]
  if(i>1) AUC <- AUC+(fp[i]-fp[i-1])*sens[i]
}
opt.thresh <- unique(income[which(toter10==min(toter10))])
fp10 <- sum(income0<opt.thresh)/n0
plot(fp,sens,type="s",lwd=3, col=8,
     main="ROC curve for identification of default mortgages",
     xlab="False positive",
     ylab="Sensitivity")
abline(h = 0.8)
AUC.th <- pnorm((m1-m0)/sqrt(s0^2+s1^2))
segments(opt.thresh,-1,opt.thresh,2,col=2)
text(.72,.4, paste("AUC = ",round(100*AUC),"%"),cex=1.25,font=2)
```




```
# index position where sensitivity = .8, approximately
idx_80_sens <- 173

tp=sens[idx_80_sens]*n1
paste("The true positive is",tp)
```

```
## [1] "The true positive is 77"
```

```
fp_number=fp[idx_80_sens]*n0
paste("The false positive is",fp_number)
```

```
## [1] "The false positive is 95"
```

```
fn=n1-tp
paste("The false negative is",fn)
```

```
## [1] "The false negative is 19"
```

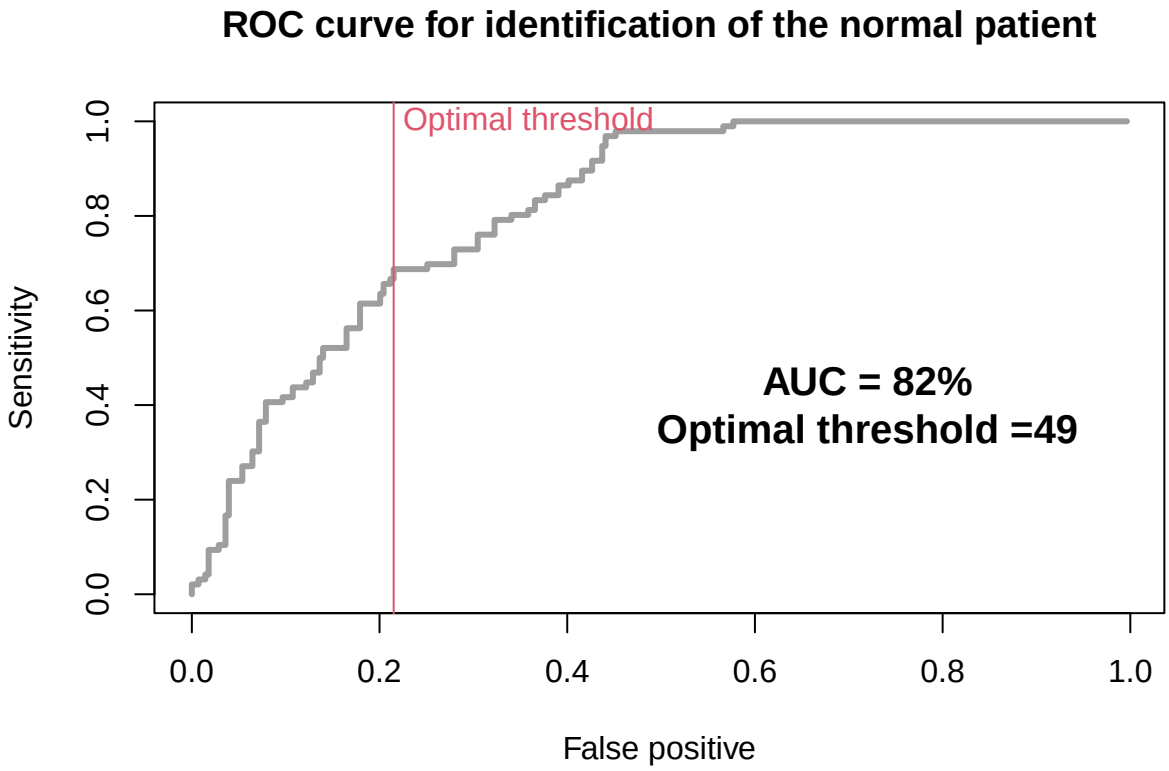
```
tn=n0-fp_number
paste("The true negative is",tn)
```

```
## [1] "The true negative is 184"
```

Sensitivity is showing the proportion of true positives over all actual positive cases. In this particular case, when sensitivity = 0.8, the value for true positive (TP), false positive (FP), false negative (FN), and true negative (TN) are 77, 95, 19, and 184, respectively.

(c) Compute and display the optimal threshold if the cost of overlooking a future defaulter is $200K and the cost of denying the mortgage application who will not default in the future is $100K.

```
sens=fp=toter10=rep(NA,n)
AUC=toter=0
for (i in 1:n) {
  sens[i] <- sum(income1 < income[i])/n1
  fp[i] <- sum(income0 < income[i])/n0
  # n1 = defaulters; n0 = non-defaulters
  toter10[i] <- (n1*200000)*(1-sens[i])+(n0*100000)*fp[i]
  if(i>1) AUC <- AUC+(fp[i]-fp[i-1])*sens[i]
}
opt.thresh <- unique(income[which(toter10==min(toter10))])
fp10 <- sum(income0<opt.thresh)/n0
plot(fp,sens,type="s",lwd=3, col=8,
     main="ROC curve for identification of the normal patient",
     xlab="False positive",
     ylab="Sensitivity")
segments(fp10,-1,fp10,2,col=2)
text(fp10+.01,1,"Optimal threshold",col=2,adj=0)
segments(opt.thresh,-1,opt.thresh,2,col=2)
AUC.th <- pnorm((m1-m0)/sqrt(s0^2+s1^2))
text(.72,.4,paste("AUC = ",round(100*AUC),
                  "%\nOptimal threshold =",
                  round(opt.thresh),sep=""),
                  cex=1.25,font=2)
```


ROC curve for identification of the normal patient
Optimal threshold
AUC = 82%
Optimal threshold =49
Sensitivity
False positive
0.0
0.2
0.4
0.6
0.8
1.0