Devoir 1 - Math pour l'IA - Partie pratique

Yunfan CAI

2022/10/22

EXERCICE 2

```
data(marks)

notes_reussite = (marks >=45)*1
#notes_reussite[notes_reussite==1] = "R"
#notes_reussite[notes_reussite==0] = "E"


f1 <- function(x,p1){
  a=0
  if((x==0) || (x==1) )
  {
    a = p1^x * (1-p1)^(1-x)
  }
  return(a)
}

f2 <- function(x,y,p2,p3){
  a=0
  if( ((x==0) || (x==1)) && ((y==0) || (y==1)) )
  {
    a = (p2^x * (1-p2)^(1-x))^y * (p3^x * (1-p3)^(1-x))^(1-y)
  }
  return(a)
}

f3 <- function(x,y,z,p4,p5,p6,p7){
  a=0
  if(((x==0) || (x==1)) && ((y==0) || (y==1)) && ((z==0) || (z==1)) )
  {
    a = ((p4^x * (1-p4)^(1-x))^y * (p5^x * (1-p5)^(1-x))^(1-y))^(z) *
      ((p6^x * (1-p6)^(1-x))^y * (p7^x * (1-p7)^(1-x))^(1-y))^(1-z)
  }
  return(a)
}

f4 <- function(x,y,p8,p9){
  a=0
  if( ((x==0) || (x==1)) && ((y==0) || (y==1)) )
```

```
  {
    a = (p8^x * (1-p8)^(1-x))^y * (p9^x * (1-p9)^(1-x))^(1-y)
  }
  return(a)
}

f5 <- function(x,y,z,p10,p11,p12,p13){
  a=0
  if(((x==0) || (x==1)) && ((y==0) || (y==1)) && ((z==0) || (z==1)) )
  {
    a = ((p10^x * (1-p10)^(1-x))^y * (p11^x * (1-p11)^(1-x))^(1-y))^(z) *
      ((p12^x * (1-p12)^(1-x))^y * (p13^x * (1-p13)^(1-x))^(1-y))^(1-z)
  }
  return(a)
}


L <- function(x1,x2,x3,x4,x5,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13){
  return(f1(x5,p1)*f2(x4,x5,p2,p3)*f3(x3,x4,x5,p4,p5,p6,p7)*
           f4(x2,x3,p8,p9)*f5(x1,x2,x3,p10,p11,p12,p13))
}
# = L(x1,x2,x3,x4,x5,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13)




Nstat_1=0
NAnl_1_Stat_1=0
NAnl_1_Stat_0=0
NAlg_1_Anl_1_Stat_1=0
NAlg_1_Anl_0_Stat_1=0
NAlg_1_Anl_1_Stat_0=0
NAlg_1_Anl_0_Stat_0=0
NAlg_1=0
NVect_1_NAlg_1=0
NVect_1_NAlg_0=0
NMech_1_Vect_1_Alg_1=0
NMech_1_Vect_0_Alg_1=0
NMech_1_Vect_1_Alg_0=0
NMech_1_Vect_0_Alg_0=0
for(i in 1:88){
  if(notes_reussite[i,5]==1){
    Nstat_1=Nstat_1+1
  }
  if((notes_reussite[i,4]==1)&&(notes_reussite[i,5]==1)){
    NAnl_1_Stat_1=NAnl_1_Stat_1+1
  }
  if((notes_reussite[i,4]==1)&&(notes_reussite[i,5]==0)){
    NAnl_1_Stat_0=NAnl_1_Stat_0+1
  }
  if((notes_reussite[i,3]==1)&&(notes_reussite[i,4]==1)&&(notes_reussite[i,5]==1)){
    NAlg_1_Anl_1_Stat_1=NAlg_1_Anl_1_Stat_1+1
  }
```

```
  if((notes_reussite[i,3]==1)&&(notes_reussite[i,4]==0)&&(notes_reussite[i,5]==1)){
    NAlg_1_Anl_0_Stat_1=NAlg_1_Anl_0_Stat_1+1
  }
  if((notes_reussite[i,3]==1)&&(notes_reussite[i,4]==1)&&(notes_reussite[i,5]==0)){
    NAlg_1_Anl_1_Stat_0=NAlg_1_Anl_1_Stat_0+1
  }
  if((notes_reussite[i,3]==1)&&(notes_reussite[i,4]==0)&&(notes_reussite[i,5]==0)){
    NAlg_1_Anl_0_Stat_0=NAlg_1_Anl_0_Stat_0+1
  }
  if(notes_reussite[i,3]==1){
    NAlg_1=NAlg_1+1
  }
  if((notes_reussite[i,2]==1)&&(notes_reussite[i,3]==1)){
    NVect_1_NAlg_1=NVect_1_NAlg_1+1
  }
  if((notes_reussite[i,2]==1)&&(notes_reussite[i,3]==0)){
    NVect_1_NAlg_0=NVect_1_NAlg_0+1
  }
  if((notes_reussite[i,1]==1)&&(notes_reussite[i,2]==1)&&(notes_reussite[i,3]==1)){
    NMech_1_Vect_1_Alg_1=NMech_1_Vect_1_Alg_1+1
  }
  if((notes_reussite[i,1]==1)&&(notes_reussite[i,2]==0)&&(notes_reussite[i,3]==1)){
    NMech_1_Vect_0_Alg_1=NMech_1_Vect_0_Alg_1+1
  }
  if((notes_reussite[i,1]==1)&&(notes_reussite[i,2]==1)&&(notes_reussite[i,3]==0)){
    NMech_1_Vect_1_Alg_0=NMech_1_Vect_1_Alg_0+1
  }
  if((notes_reussite[i,1]==1)&&(notes_reussite[i,2]==0)&&(notes_reussite[i,3]==0)){
    NMech_1_Vect_0_Alg_0=NMech_1_Vect_0_Alg_0+1
  }



}
p1=Nstat_1/88
p2=NAnl_1_Stat_1/Nstat_1
p3=NAnl_1_Stat_0/(88-Nstat_1)
p4=NAlg_1_Anl_1_Stat_1/(p2*p1*88)
p5=NAlg_1_Anl_0_Stat_1/((1-p2)*p1*88)
p6=NAlg_1_Anl_1_Stat_0/(p3*(1-p1)*88)
p7=NAlg_1_Anl_0_Stat_0/(p3*(1-p1)*88)
p8=NVect_1_NAlg_1/NAlg_1
p9=NVect_1_NAlg_0/(88-NAlg_1)
p10=NMech_1_Vect_1_Alg_1/(p8*(NAlg_1/88)*88)
p11=NMech_1_Vect_0_Alg_1/((1-p8)*(NAlg_1/88)*88)
p12=NMech_1_Vect_1_Alg_0/(p8*(1-(NAlg_1/88))*88)
p13=NMech_1_Vect_0_Alg_0/(p8*(1-(NAlg_1/88))*88)

#p1=P(Stat=1)
print(p1)
```

```
## [1] 0.3863636
```

```
#p2=P(Anl=1|Stat=1)
print(p2)
```

```
## [1] 0.8529412
```

```
#p3=P(Anl=1|Stat=0)
print(p3)
```

```
## [1] 0.5555556
```

```
#p4=P(Alg=1|Anl=1,Stat=1)
print(p4)
```

```
## [1] 0.9655172
```

```
#p5=P(Alg=1|Anl=0,Stat=1)
print(p5)
```

```
## [1] 0.8
```

```
#p6=P(Alg=1|Anl=1,Stat=0)
print(p6)
```

```
## [1] 0.8333333
```

```
#p7=P(Alg=1|Anl=0,Stat=0)
print(p7)
```

```
## [1] 0.3333333
```

```
#p2=P(Vect=1|Alg=1)
print(p8)
```

```
## [1] 0.7761194
```

```
#p3=P(Vect=1|Alg=0)
print(p9)
```

```
## [1] 0.2857143
```

```
#p4=P(Mech=1|Vect=1,Alg=1)
print(p10)
```

```
## [1] 0.5576923
```

```
#p5=P(Mech=1|Vect=0,Alg=1)
print(p11)
```

```
## [1] 0.3333333
```

```
#p6=P(Mech=1|Vect=1,Alg=0)
print(p12)
```

```
## [1] 0.1227106
```

```
#p7=P(Mech=1|Vect=0,Alg=0)
print(p13)
```

```
## [1] 0.1227106
```

```
petoile <- function(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13) {
  P=1
  for(i in 1:88 ){
    Xi = notes_reussite[i,]
    P=P*L(Xi[1],Xi[2],Xi[3],Xi[4],Xi[5],p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13)
  }
  return (P)
}


petoile2 <- function(Plist) {
  P=1
  for(i in 1:88 ){
    Xi = notes_reussite[i,]
    #P=P*L(Xi[1],Xi[2],Xi[3],Xi[4],Xi[5],Plist[1],Plist[2],Plist[3],Plist[4],Plist[5],
          #Plist[6],Plist[7],Plist[8],Plist[9],Plist[10],Plist[11],Plist[12],Plist[13])
    P=P+log(L(Xi[1],Xi[2],Xi[3],Xi[4],Xi[5],Plist[1],Plist[2],Plist[3],Plist[4],Plist[5],
          Plist[6],Plist[7],Plist[8],Plist[9],Plist[10],Plist[11],Plist[12],Plist[13]))
  }
  return (-P) # on fait -P pour avoir argmax au lieux de argmin
}



notes_reussite[1,]
```

```
## MECH VECT  ALG  ANL STAT
##    1    1    1    1    1
```

```
#test = petoile(0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.1,0.2,0.3,0.4,0.5,0.6)
#test

#x0 <- c(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13)
x0 <- c(0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5)

solution <- optim(x0, petoile2)
print(solution$counts)
```

```
## function gradient
##      502       NA
```

```
#la solution est déjà minimum
print(solution$par)
```

```
##  [1] 0.38589839 0.81888775 0.55986154 0.96479956 0.52862065 0.82099768
##  [7] 0.43109717 0.77892814 0.32375379 0.54137566 0.26393970 0.38264238
## [13] 0.08296025
```

```
print(solution$value)
```

```
## [1] 250.305
```

```
#minf <- optimize(v,c(c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,
```

```
#minf = optimize(petoile2,c(c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1
```

```
#notes_reussite[1,]
```

```
#test = petoile(0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.1,0.2,0.3,0.4,0.5,0.6)
#test
```

```
#x0 <- c(0,0,0,0,0,0,0,0,0,0,0,0,0.3)
```

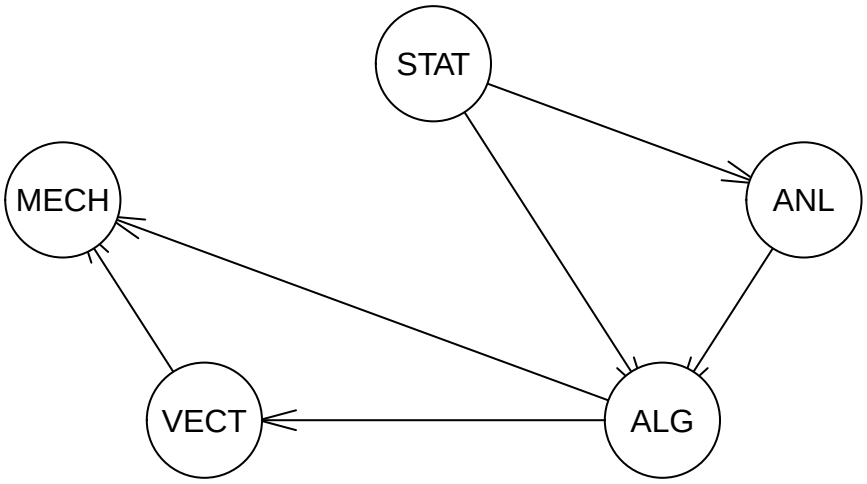
```
#solution <- optim(x0, petoile2)
```

```
#print(solution$counts)
#print(solution$par)
#print(solution$value)
```

```
#minf <- optimize(petoile2, c(c(0,1), c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(0,1),c(
```

```
#print(minf$minimum)
#print(minf$objective)
```

```
dag_notes = empty.graph(names(marks))
arcs(dag_notes) = matrix(
c("VECT", "MECH",
"ALG", "MECH",
"ALG", "VECT",
"ANL", "ALG",
"STAT", "ALG",
"STAT", "ANL"),
ncol = 2, byrow = TRUE, dimnames = list(c(), c("from", "to")))
plot(dag_notes)
```




```
bn.fit(dag_notes, data = as.data.frame(notes_reussite))
```

```
##
##   Bayesian network parameters
##
##   Parameters of node MECH (Gaussian distribution)
##
## Conditional density: MECH | VECT + ALG
## Coefficients:
## (Intercept)         VECT          ALG
##   0.1282463    0.2178046    0.2101740
## Standard deviation of the residuals: 0.4756269
##
##   Parameters of node VECT (Gaussian distribution)
##
## Conditional density: VECT | ALG
## Coefficients:
## (Intercept)          ALG
##   0.2857143    0.4904051
## Standard deviation of the residuals: 0.4303528
##
##   Parameters of node ALG (Gaussian distribution)
##
## Conditional density: ALG | ANL + STAT
## Coefficients:
```

```
## (Intercept)          ANL         STAT
##   0.4504797    0.3558032    0.1872176
## Standard deviation of the residuals: 0.3752541
##
##   Parameters of node ANL (Gaussian distribution)
##
## Conditional density: ANL | STAT
## Coefficients:
## (Intercept)         STAT
##   0.5555556    0.2973856
## Standard deviation of the residuals: 0.4523587
##
##   Parameters of node STAT (Gaussian distribution)
##
## Conditional density: STAT
## Coefficients:
## (Intercept)
##   0.3863636
## Standard deviation of the residuals: 0.4897059
```