Bayesian Methods - Assignment 1

Ryan Durfey

April 4, 2016

```
set.seed(11)
```

```
options(width = 80)
library (faraway)
```

```
## Warning: package 'faraway' was built under R version 3.2.3
```

```
data(femsmoke)
head(femsmoke)
```

```
##    y smoker dead   age
## 1  2    yes  yes 18-24
## 2  1     no  yes 18-24
## 3  3    yes  yes 25-34
## 4  5     no  yes 25-34
## 5 14    yes  yes 35-44
## 6  7     no  yes 35-44
```

```
# convert factors into predefinted numeric values
fem<-femsmoke # dataset copy
fem$smoker<-as.numeric(fem$smoker)
fem[fem$smoker==2,2]<-0
fem$dead<-as.numeric(fem$dead)
fem[fem$dead==2,3]<-0
fem$age<-as.numeric(fem$age)
head(fem)
```

```
##    y smoker dead age
## 1  2      1    1   1
## 2  1      0    1   1
## 3  3      1    1   2
## 4  5      0    1   2
## 5 14      1    1   3
## 6  7      0    1   3
```

```
#-----------------------------------------------#
# Create Join Distributions of 3 count variables #
#-----------------------------------------------#

# reshape data into table format that we want
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 3.2.3
```

```r
mat.u0<-spread(fem[fem$dead==0,-3],key=age,value=y)[,-1]
mat.u1<-spread(fem[fem$dead==1,-3],key=age,value=y)[,-1]

rownames(mat.u1)<-rownames(mat.u0)<-c("smoke.no","smoke.yes")
colnames(mat.u1)<-colnames(mat.u0)<-c("age.1","age.2","age.3","age.4","age.5","age.6","age.7")

mat.u0<-mat.u0[c(2,1),] # reorder rows to align with what assignment doc shows
mat.u1<-mat.u1[c(2,1),]
mat.u0
```

```
##           age.1 age.2 age.3 age.4 age.5 age.6 age.7
## smoke.yes    53   121    95   103    64     7     0
## smoke.no     61   152   114    66    81    28     0
```

```r
mat.u1
```

```
##           age.1 age.2 age.3 age.4 age.5 age.6 age.7
## smoke.yes     2     3    14    27    51    29    13
## smoke.no      1     5     7    12    40   101    64
```

```r
(femsmoke.joint.p<-list(dead=mat.u1/sum(fem$y),alive=mat.u0/sum(fem$y)))
```

```
## $dead
##                 age.1       age.2       age.3      age.4      age.5      age.6
## smoke.yes 0.001522070 0.002283105 0.010654490 0.02054795 0.03881279 0.02207002
## smoke.no  0.000761035 0.003805175 0.005327245 0.00913242 0.03044140 0.07686454
##                 age.7
## smoke.yes 0.009893455
## smoke.no  0.048706240
##
## $alive
##                age.1      age.2      age.3      age.4      age.5       age.6
## smoke.yes 0.04033486 0.09208524 0.07229833 0.07838661 0.04870624 0.005327245
## smoke.no  0.04642314 0.11567732 0.08675799 0.05022831 0.06164384 0.021308980
##           age.7
## smoke.yes     0
## smoke.no      0
```

```r
Reduce("+",lapply(femsmoke.joint.p,sum)) # confirms that all the join probs add to one
```

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

```r
#------------------------------#
# Create Marginal Distributions #
#------------------------------#

# u
uMarginal<-c(uAlive=sum(femsmoke.joint.p$alive),uDead=sum(femsmoke.joint.p$dead))
uMarginal
```

```
##    uAlive     uDead
## 0.7191781 0.2808219
```

```r
# v
marg.v1<-sum(femsmoke.joint.p$alive[1,],femsmoke.joint.p$dead[1,])
marg.v2<-sum(femsmoke.joint.p$alive[2,],femsmoke.joint.p$dead[2,])
vMarginal<-c(smoke.yes=marg.v1,smoke.no=marg.v2)
vMarginal
```

```
## smoke.yes  smoke.no
## 0.4429224 0.5570776
```

```r
# w
marg.w1<-sum(femsmoke.joint.p$alive[,1],femsmoke.joint.p$dead[,1])
marg.w2<-sum(femsmoke.joint.p$alive[,2],femsmoke.joint.p$dead[,2])
marg.w3<-sum(femsmoke.joint.p$alive[,3],femsmoke.joint.p$dead[,3])
marg.w4<-sum(femsmoke.joint.p$alive[,4],femsmoke.joint.p$dead[,4])
marg.w5<-sum(femsmoke.joint.p$alive[,5],femsmoke.joint.p$dead[,5])
marg.w6<-sum(femsmoke.joint.p$alive[,6],femsmoke.joint.p$dead[,6])
marg.w7<-sum(femsmoke.joint.p$alive[,7],femsmoke.joint.p$dead[,7])
wMarginal<-c(age.1=marg.w1,age.2=marg.w2,age.3=marg.w3,age.4=marg.w4,age.5=marg.w5,
             age.6=marg.w6,age.7=marg.w7)
wMarginal
```

```
##     age.1     age.2     age.3     age.4     age.5     age.6     age.7
## 0.0890411 0.2138508 0.1750381 0.1582953 0.1796043 0.1255708 0.0585997
```

```r
# alternative way to calc
# wMarginal.age<-colSums(rbind(colSums(femsmoke.joint.p$alive),colSums(femsmoke.joint.p$dead)))



#-------------------------------------------------#
# Create conditional distribution p(w,v|u="alive") #
#-------------------------------------------------#
(cond.v.w.given.uAlive<-femsmoke.joint.p[["alive"]])
```

```
##                age.1      age.2      age.3      age.4      age.5       age.6
## smoke.yes 0.04033486 0.09208524 0.07229833 0.07838661 0.04870624 0.005327245
## smoke.no  0.04642314 0.11567732 0.08675799 0.05022831 0.06164384 0.021308980
##           age.7
## smoke.yes     0
## smoke.no      0
```

```r
#-------------------------------------------------#
# Create conditional distribution p(v|u="alive") #
#-------------------------------------------------#
(cond.v.given.uAlive<-apply(femsmoke.joint.p[["alive"]],1,sum)/uMarginal["uAlive"])
```

```
## smoke.yes  smoke.no
## 0.4687831 0.5312169
```

```
#-----------------------------------------------------------#
# Create conditional distribution p(w|v="alive",u="smoker") #
#-----------------------------------------------------------#
(cond.w.given.uAlive.vSmoke<-(femsmoke.joint.p[["alive"]]["smoke.yes",])/
  (cond.v.given.uAlive[["smoke.yes"]]*uMarginal["uAlive"]))
```

```
##               age.1     age.2    age.3     age.4     age.5      age.6 age.7
## smoke.yes 0.1196388 0.2731377 0.214447 0.2325056 0.1444695 0.01580135     0
```

```
#-------------------------------------------------------------------------------------------#
# Compare the vectors p(w|v2,u1)p(v2|u1)p(u1)p(w|v2,u1)p(v2|u1)p(u1) & p(w,v,u)[,v2,u1] #
#-------------------------------------------------------------------------------------------#
rbind(uMarginal["uAlive"]*cond.v.given.uAlive["smoke.yes"]*cond.w.given.uAlive.vSmoke,
      femsmoke.joint.p[["alive"]]["smoke.yes",])
```

```
##                 age.1      age.2      age.3      age.4      age.5       age.6
## smoke.yes  0.04033486 0.09208524 0.07229833 0.07838661 0.04870624 0.005327245
## smoke.yes1 0.04033486 0.09208524 0.07229833 0.07838661 0.04870624 0.005327245
##            age.7
## smoke.yes      0
## smoke.yes1     0
```

```
#-------------------------------------------------------------------------------------------#
# Given simulated age group, simulate variable v using conditional distribution p(v|w) #
#-------------------------------------------------------------------------------------------#
set.seed(11)
simulatedData<-data.frame(ages=rep(NA,100),smokers=rep(NA,100),dead=rep(NA,100))

# simulate 100 people with age groups based on wMarginal
simulatedData$ages<-sample(x=1:7,size=100,prob = wMarginal,replace = TRUE)

# conditional distribution p(v|w)
cond.vSmokeYes.given.age<-(femsmoke.joint.p[["alive"]]+femsmoke.joint.p[["dead"]])[1,]/
  wMarginal
cond.vSmokeNo.given.age<-(femsmoke.joint.p[["alive"]]+femsmoke.joint.p[["dead"]])[2,]/
  wMarginal
(cond.Smoke.given.age<-rbind(cond.vSmokeYes.given.age,cond.vSmokeNo.given.age))
```

```
##               age.1     age.2    age.3 age.4     age.5     age.6     age.7
## smoke.yes 0.4700855 0.4412811 0.473913 0.625 0.4872881 0.2181818 0.1688312
## smoke.no  0.5299145 0.5587189 0.526087 0.375 0.5127119 0.7818182 0.8311688
```

```
# simulate v based on simulated w
for(i in 1:100){
  simulatedData[i,2]<-rbinom(n=1,size=1,prob=cond.Smoke.given.age[1,simulatedData[i,1]])
}
# check to make sure columns sum to 1
colSums(cond.Smoke.given.age)
```

```
## age.1 age.2 age.3 age.4 age.5 age.6 age.7
##     1     1     1     1     1     1     1
```

```r
head(simulatedData[,-3],25)
```

```
##    ages smokers
## 1     5       1
## 2     2       1
## 3     3       0
## 4     2       0
## 5     2       0
## 6     7       0
## 7     2       1
## 8     5       1
## 9     1       1
## 10    2       0
## 11    2       1
## 12    3       0
## 13    1       0
## 14    6       1
## 15    6       0
## 16    4       1
## 17    3       1
## 18    5       0
## 19    2       0
## 20    3       0
## 21    2       0
## 22    4       1
## 23    5       1
## 24    5       0
## 25    2       0
```

```r
#-----------------------------------------------------------------------#
# Given simulated variables for age and for smoke, simulate mortality #
# variable using distribution p(dead|v,u), p(alive|v,u)                #
#-----------------------------------------------------------------------#

# p(u,v,w)=p(u|v,w)p(v|w)p(w)
# p(u|v,w)=p(u,v,w)/(p(v|w)p(w))

p.uvw<-femsmoke.joint.p
p.v.w<-cond.Smoke.given.age
p.w<-wMarginal

pp.dead<-matrix(nrow=2,ncol=7)
for(i in 1:2){
  for(j in 1:7){
    pp.dead[i,j]<-p.uvw$dead[i,j]/(p.v.w[i,j]*p.w[j])
  }
}
pp.alive<-matrix(nrow=2,ncol=7)
for(i in 1:2){
  for(j in 1:7){
    pp.alive[i,j]<-p.uvw$alive[i,j]/(p.v.w[i,j]*p.w[j])
  }
}
```

```
# check to make sure each dead/alive pair sums to 1
for(i in 1:2){
  for(j in 1:7){
    print(sum(pp.dead[i,j],pp.alive[i,j]))
  }
}
```

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

```
# yay.

# now simulate

# first, make alternate simulatedData where smoker.no=2
# this will now also correspond to indices of appropriate probabilities for below
temp.sim<-simulatedData
temp.sim[temp.sim$smokers==0,2]<-2

# ok, NOW simulate
for(i in 1:100){
  simulatedData[i,3]<-rbinom(n=1,size=1,prob=pp.dead[temp.sim[i,2],temp.sim[i,1]])
}
simulatedData
```

```
##     ages smokers dead
## 1      5       1    0
## 2      2       1    0
## 3      3       0    0
## 4      2       0    0
## 5      2       0    0
## 6      7       0    1
## 7      2       1    0
## 8      5       1    0
## 9      1       1    0
## 10     2       0    0
## 11     2       1    0
## 12     3       0    0
## 13     1       0    0
```

```
## 14     6       1    1
## 15     6       0    1
## 16     4       1    0
## 17     3       1    1
## 18     5       0    1
## 19     2       0    0
## 20     3       0    0
## 21     2       0    0
## 22     4       1    1
## 23     5       1    1
## 24     5       0    1
## 25     2       0    0
## 26     3       0    0
## 27     3       0    0
## 28     2       1    0
## 29     2       1    0
## 30     3       1    0
## 31     3       1    0
## 32     5       1    0
## 33     3       0    0
## 34     2       0    0
## 35     6       0    1
## 36     4       1    0
## 37     5       0    0
## 38     2       1    0
## 39     5       1    1
## 40     2       1    0
## 41     5       1    0
## 42     5       1    1
## 43     3       0    0
## 44     4       0    0
## 45     5       0    1
## 46     1       0    0
## 47     4       1    0
## 48     2       0    0
## 49     3       1    0
## 50     6       0    1
## 51     3       1    0
## 52     4       0    1
## 53     3       0    0
## 54     5       1    1
## 55     5       1    0
## 56     3       1    0
## 57     3       0    0
## 58     6       0    0
## 59     2       1    0
## 60     4       1    0
## 61     2       0    0
## 62     6       0    1
## 63     2       1    0
## 64     4       1    0
## 65     6       0    1
## 66     2       1    0
## 67     2       0    0
```

```
## 68     3       0    0
## 69     2       1    0
## 70     4       1    0
## 71     4       1    0
## 72     3       1    0
## 73     3       0    0
## 74     2       1    0
## 75     1       1    0
## 76     3       1    0
## 77     5       0    1
## 78     5       1    0
## 79     5       0    0
## 80     5       0    0
## 81     2       1    0
## 82     3       0    0
## 83     3       0    0
## 84     5       0    1
## 85     6       1    1
## 86     3       0    0
## 87     2       0    0
## 88     4       1    0
## 89     6       0    1
## 90     5       1    0
## 91     5       0    0
## 92     6       0    1
## 93     2       1    0
## 94     3       1    0
## 95     6       0    1
## 96     5       1    1
## 97     6       1    1
## 98     2       1    0
## 99     4       1    0
## 100    5       1    1
```

```r
#-------------------------------------------------------------------------------#
# SANITY CHECK: Compare joint distribution of simulated with original dataset #
#-------------------------------------------------------------------------------#
sim.mat.u0<-matrix(0,nrow=2,ncol=7)
sim.mat.u1<-matrix(0,nrow=2,ncol=7)

for(i in 0:1){
  for(j in 1:7){
    sim.mat.u0[i+1,j]<-nrow(simulatedData[simulatedData$ages==j & simulatedData$smokers==i &
                                          simulatedData$dead==0,])
  }
}
for(i in 0:1){
  for(j in 1:7){
    sim.mat.u1[i+1,j]<-nrow(simulatedData[simulatedData$ages==j & simulatedData$smokers==i &
                                          simulatedData$dead==1,])
  }
}

rownames(sim.mat.u1)<-rownames(sim.mat.u0)<-c("smoke.no","smoke.yes")
```

```r
colnames(sim.mat.u1)<-colnames(sim.mat.u0)<-c("age.1","age.2","age.3","age.4","age.5",
                                              "age.6","age.7")

sim.mat.u0<-sim.mat.u0[c(2,1),] # reorder rows to align with what assignment doc shows
sim.mat.u1<-sim.mat.u1[c(2,1),]
sim.mat.u0
```

```
##           age.1 age.2 age.3 age.4 age.5 age.6 age.7
## smoke.yes     2    15     8     9     7     0     0
## smoke.no      2    11    14     1     4     1     0
```

```r
sim.mat.u1
```

```
##           age.1 age.2 age.3 age.4 age.5 age.6 age.7
## smoke.yes     0     0     1     1     6     3     0
## smoke.no      0     0     0     1     5     8     1
```

```r
simulated.joint.p<-list(dead=sim.mat.u1,alive=sim.mat.u0)
N<-sum(unlist(simulated.joint.p))
(simulated.joint.p<-lapply(simulated.joint.p,"/",N))
```

```
## $dead
##           age.1 age.2 age.3 age.4 age.5 age.6 age.7
## smoke.yes     0     0  0.01  0.01  0.06  0.03  0.00
## smoke.no      0     0  0.00  0.01  0.05  0.08  0.01
##
## $alive
##           age.1 age.2 age.3 age.4 age.5 age.6 age.7
## smoke.yes  0.02  0.15  0.08  0.09  0.07  0.00     0
## smoke.no   0.02  0.11  0.14  0.01  0.04  0.01     0
```

```r
femsmoke.joint.p
```

```
## $dead
##                  age.1       age.2       age.3      age.4      age.5      age.6
## smoke.yes 0.001522070 0.002283105 0.010654490 0.02054795 0.03881279 0.02207002
## smoke.no  0.000761035 0.003805175 0.005327245 0.00913242 0.03044140 0.07686454
##                  age.7
## smoke.yes 0.009893455
## smoke.no  0.048706240
##
## $alive
##                 age.1      age.2      age.3      age.4      age.5       age.6
## smoke.yes 0.04033486 0.09208524 0.07229833 0.07838661 0.04870624 0.005327245
## smoke.no  0.04642314 0.11567732 0.08675799 0.05022831 0.06164384 0.021308980
##           age.7
## smoke.yes     0
## smoke.no      0
```

```r
# The joint distributions are not equal, but our simulated sample size is small,
# so variation is not unexpected.
#
# The values are still somewhat close and appear to follow the same general pattern,
# so I think it's ok.

# plot comparison of joint distribution values
plot(unlist(simulated.joint.p),type='b',col='blue')
lines(unlist(femsmoke.joint.p),type='b',col="red")
```