Sesión 4: Algoritmo EM

Aplicaciones en Computación Estadística

Natalie Julian - www.nataliejulian.com

Estadística UC y Data Scientist en Zippedi Inc.

Algoritmo EM para imputación de datos faltantes caso normal bivariado

Ejercicio 1

El archivo `Colesterol.RData` contiene los niveles de colesterol para $n = 28$ pacientes que han sido tratados por un ataque cardíaco. Los niveles de colesterol fueron medidos 2 y 14 días después del ataque.
Se asume que los datos $Y_i = (Y_{i1}, Y_{i2})$ sigue una distribución Normal bivariada, con Y_i e $Y_{i'}$ i.i.d si $i \neq i'$.

Ejercicio 1

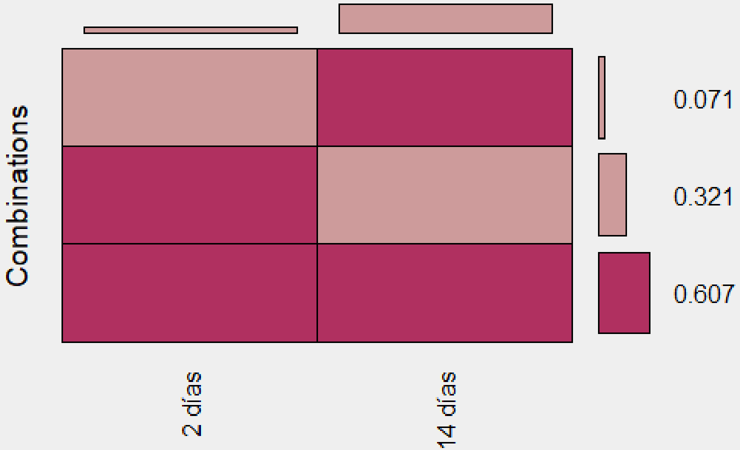
a) Hay algunos pacientes a los que no se le pudo realizar el seguimiento completo. Realice análisis de la incertidumbre en los datos. ¿Qué casos serán necesarios incorporar al imputar los datos faltantes?

a) Hay algunos pacientes a los que no se le pudo realizar el seguimiento completo. Realice análisis de la incertidumbre en los datos. ¿Qué casos serán necesarios incorporar al imputar los datos faltantes?

```
#install.packages("VIM")
library(VIM)

aggr(colesterol,col=c('maroon','rosybrown3'), combined=TRUE, numbers=TRUE)
```

Ejercicio 1

a) Hay algunos pacientes a los que no se le pudo realizar el seguimiento completo. Realice análisis de la incertidumbre en los datos. ¿Qué casos serán necesarios incorporar al imputar los datos faltantes?



Ejercicio 1

b) Plantee el algoritmo EM correspondiente para imputar las observaciones faltantes.

Pasos EM para imputación Caso normal

Paso 1:
Determinar la distribución del vector

Paso 2:
Determinar los casos de imputación. Si sólo falta una observación se utilizará la distribución condicional de dicha variable dada la otra. Si faltan ambas variables se debe imputar por el valor esperado

Paso 3: Paso E
Obtener los valores esperados de los datos faltantes utilizando la distribución condicional

Paso 4: Paso M
Los valores esperados quedarán sujetos a los parámetros, pero estos deben estimarse utilizando el EMV, en el caso normal quedan en función de estadísticos suficientes

Paso 5:
Se induce iteración en cada parámetro, se comienza el algoritmo utilizando los EMV sin considerar los datos faltantes

Pasos EM para imputación Caso normal

P1 La distribución del vector bivariado es una normal bivariada i.i.d

P2 Los casos de imputación son dos, cuando falta el primer registro ó (exclusivo) el segundo registro. Por lo tanto, en ambos casos se imputará utilizando la distribución condicional.

P3 Dada la distribución bivariada, se sabe que:

$$\theta = (\mu_1, \mu_2, \sigma_1, \sigma_2, \rho = \frac{\sigma_{12}}{\sqrt{\sigma_1^2 \sigma_2^2}})$$

$$Y_{i1}|Y_{i2}, \theta \sim N\left(\mu_1 + \rho\frac{\sigma_1}{\sigma_2}(Y_{i2} - \mu_2), (1 - \rho^2)\sigma_1^2\right)$$

$$Y_{i2}|Y_{i1}, \theta \sim N\left(\mu_2 + \rho\frac{\sigma_2}{\sigma_1}(Y_{i1} - \mu_1), (1 - \rho^2)\sigma_2^2\right)$$

Pasos EM para imputación Caso normal

P3 Por lo tanto, los valores esperados para las observaciones faltantes (Paso E) son:

$$E(Y_{i1}|Y_{i2},\theta) = \mu_1 + \rho\frac{\sigma_1}{\sigma_2}(Y_{i2} - \mu_2)$$

$$E(Y_{i1}^2|Y_{i2},\theta) = E(Y_{i1}|Y_{i2},\theta)^2 + Var(Y_{i1}|Y_{i2},\theta)$$

$$E(Y_{i1}^2|Y_{i2},\theta) = \left(\mu_1 + \rho\frac{\sigma_1}{\sigma_2}(Y_{i2} - \mu_2)\right)^2 + (1-\rho^2)\sigma_1^2$$

Para $Y_{i2}|Y_{i1},\theta$ es análogo.

Pasos EM para imputación Caso normal

P4 Cada parámetro es estimado mediante máxima verosimilitud en cada iteración. A priori sabemos que:

$$\hat{\mu}_1 = \frac{\sum_{i=1}^{n} Y_{i1}}{n} \qquad \hat{\mu}_2 = \frac{\sum_{i=1}^{n} Y_{i2}}{n}$$

$$\hat{\sigma}_1 = \sqrt{\frac{\sum_{i=1}^{n} Y_{i1}^2}{n} - \left(\frac{\sum_{i=1}^{n} Y_{i1}}{n}\right)^2}$$

$$\hat{\sigma}_2 = \sqrt{\frac{\sum_{i=1}^{n} Y_{i2}^2}{n} - \left(\frac{\sum_{i=1}^{n} Y_{i2}}{n}\right)^2}$$

$$\hat{\rho} = \left(\frac{\sum_{i=1}^{n} Y_{i1} Y_{i2}}{n} - \frac{\sum_{i=1}^{n} Y_{i1}}{n} \cdot \frac{\sum_{i=1}^{n} Y_{i2}}{n}\right) / \hat{\sigma}_1 \hat{\sigma}_2$$

Pasos EM para imputación Caso normal

P5 Se induce iteración en cada parámetro:

1. Comenzamos con un valor inicial para el vector de parámetros: $\theta^{(0)}$ puede ser el EMV con datos completos o estimador de momentos, etcétera.
2. Utilizando la distribución condicional, se obtendrán los siguientes valores esperados (para las observaciones faltantes):

$$E(Y_{i1}|Y_{i2},\theta) = \mu_1^{(0)} + \rho^{(0)}\frac{\sigma_1^{(0)}}{\sigma_2^{(0)}}(Y_{i2}-\mu_2^{(0)})$$

$$E(Y_{i1}^2|Y_{i2},\theta) = \left(\mu_1^{(0)} + \rho^{(0)}\frac{\sigma_1^{(0)}}{\sigma_2^{(0)}}(Y_{i2}-\mu_2^{(0)})\right)^2 + (1-\rho^{(0)2})\sigma_1^{(0)2}$$

Análogo para Y_{i2} faltantes.

Pasos EM para imputación Caso normal

3. Una vez que se tienen las observaciones completas (pues se realizó imputación) se obtiene $\theta^{(1)}$ utilizando los estimadores encontrados. Se vuelve a iterar, pero utilizando los valores de los parámetros actualizados.

c) Implemente su algoritmo en R. Obtenga los datos completos y estimaciones para los parámetros.

Opción 1 de implementación

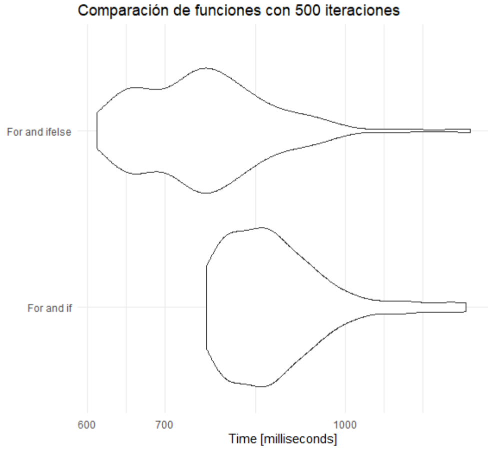
```
EMnormal<-function(datos,mu1,mu2,sigma1,sigma2,rho,N){
  n<-dim(datos)[1]

  M=cbind(datos,rep(0,n),rep(0,n))

  for(k in 1:N){

    for(i in 1:n){
      if(is.na(datos[i,1])==TRUE){ #Si la primera medicion es faltante
        M[i,1]<-mu1+rho*sigma1/(sigma2)*(datos[i,2]-mu2) #E(Y_i1|Y_i2,theta)
        M[i,3]<-(1-rho^{2})*sigma1^{2} #Var(Y_i1|Y_i2,theta)
      }

      if(is.na(datos[i,2])==TRUE){ #Si la segunda medicion es faltante
        M[i,2]<-mu2+rho*sigma2/(sigma1)*(datos[i,1]-mu1) #E(Y_i2|Y_i1, theta)
        M[i,4]<-(1-rho^{2})*sigma2^{2} #Var(Y_i2^{2}|Y_i1, theta)
      }
    }

    mu1<-sum(M[,1])/n
    mu2<-sum(M[,2])/n
    sigma1<-sqrt((sum(M[,1]^{2}+M[,3]))/n-mu1^{2})
    sigma2<-sqrt((sum(M[,2]^{2}+M[,4]))/n-mu2^{2})
    rho<-((sum(M[,1]*M[,2]))/n-mu1*mu2)/(sigma1*sigma2)
  }

  M<-M[,1:2]

  return(list=c("media1"=mu1,"media2"=mu2,"sd1"=sigma1,"sd2"=sigma2,
              "rho"=rho,"datos"=M))
}
```

Opción 2 de implementación

```
library(dplyr)

EMnormalfast<-function(datos,mu1,mu2,sigma1,sigma2,rho,N){
    for(i in 1:N){
        datosimputados<- datos%>%
             mutate(Ey1=ifelse(is.na(‘2 días‘)==TRUE, mu1+rho*sigma1/(sigma2)*(‘14 días‘-mu2), 0),
                Ey2=ifelse(is.na(‘14 días‘)==TRUE, mu2+rho*sigma2/(sigma1)*(‘2 días‘-mu1), 0),
                VarY1=ifelse(is.na(‘2 días‘)==TRUE, (1-rho^{2})*sigma1^{2}, 0),
                VarY2=ifelse(is.na(‘14 días‘)==TRUE, (1-rho^{2})*sigma2^{2}, 0))

        params<-datosimputados  %>%
             summarise(mu1=sum(‘2 días‘, Ey1, na.rm = TRUE)/length(Ey1),
                mu2=sum(‘14 días‘, Ey2, na.rm = TRUE)/length(Ey1),
                sigma1=sqrt(sum(‘2 días‘^{2}, Ey1^{2}, VarY1, na.rm = TRUE)/length(Ey1)-mu1^{2}),
                sigma2=sqrt(sum(‘14 días‘^{2}, Ey2^{2}, VarY2, na.rm = TRUE)/length(Ey1)-mu2^{2}),
                rho=(sum(‘2 días‘*‘14 días‘,Ey1*‘14 días‘,Ey2*‘2 días‘,na.rm=TRUE)/length(Ey1)-mu1*mu2)
                /(sigma1*sigma2))

mu1<-params$mu1
mu2<-params$mu2
sigma1<-params$sigma1
sigma2<-params$sigma2
rho<-params$rho
}

datosimputados[which(is.na(datos$‘2 días‘)==TRUE), 1]<-datosimputados[which(is.na(datos$‘2 días‘)==TRUE),"Ey1"]
datosimputados[which(is.na(datos$‘14 días‘)==TRUE), 2]<-datosimputados[which(is.na(datos$‘14 días‘)==TRUE),"Ey2"]

return(list=c("media1"=mu1,"media2"=mu2,"sd1"=sigma1,"sd2"=sigma2,
              "rho"=rho,"datos"=datosimputados[, 1:2]))
}
```

Comparación de 500 iteraciones con *microbenchmark*

```
 mbm
Unit: milliseconds
           expr      min       lq     mean   median       uq      max
     For and if 759.3425 797.5849 874.4770 853.2133 912.8107 1269.341
 For and ifelse 610.9763 672.5100 763.1615 752.2869 809.3485 1280.160
 neval cld
   100   b
   100  a
```



Algoritmo EM para mezcla de distribuciones beta

Ejercicio 2

Al querer modelar datos multimodales, la mezcla de distribuciones gaussianas es una alternativa popular pero no siempre la mejor, sobretodo cuando el soporte es limitado, por ejemplo, entre [0,1], en dicho caso resulta más adecuado utilizar una mezcla beta.

a) Plantee el algoritmo EM para un caso de mezclas beta.

Pasos EM para mezcla de distribuciones

Paso 1:

Incorporar los pesos o ponderadores de cada componente, si son dos distribuciones entonces los pesos serán p y (1-p), notar que siempre suman 1. Se interpretan como la proporción de los datos que provienen de dicha componente

Paso 2:

Inducir las variables latentes z, que corresponden a los indicadores binarios. Indican la distribución de origen para determinado x

Paso 3:

Cada componente es ponderada por los pesos y elevada a la variable latente, esta indicatriz puede entenderse como una v.a Bernoulli

Paso 4: Paso E

Los parámetros corresponden a los pesos y a los parámetros de las distribuciones en cuestión. La variable latente corresponde a la información incompleta, por lo tanto, el paso E se traduce en calcular el valor esperado de Z

Paso 5: Paso M

Dado que se posee E(Z) se actualizan los valores de los pesos. Lo mismo se realiza con los parámetros, pero utilizando maximización

Ejercicio 2

a) Plantee de forma general el algoritmo EM para un caso de mezclas beta.

Considere la PDF de una distribución beta:

$$\text{Beta}(x;\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

Luego, la función de densidad para una mezcla de distribuciones beta tendrá la siguiente expresión:

$$f(x) = \sum_{i=1}^{I} P_i \text{Beta}(x;\alpha_i,\beta_i)$$

Donde I corresponde a la cantidad de distribuciones a considerar, P_i corresponde a la probabilidad de ocurrencia de la i ésima componente, notar que $\sum_{i=1}^{I} P_i = 1$.

Ejercicio 2

Defina el vector latente $z_j = (z_{j1}, ..., z_{jI})^T$, que toma sólo un valor 1 y el resto 0, corresponde al vector indicatriz que multiplica cada componente de la mezcla.

La verosimilitud (asumiendo independencia) puede escribirse como sigue:

$$L(\theta; x, z) = \prod_{j=1}^{N} \prod_{i=1}^{I} \left(P_i \text{Beta}(x_j; \alpha_i, \beta_i)\right)^{z_{ji}}$$

Y la log-verosimilitud:

$$l(\theta; x, z) = \sum_{j=1}^{N} \sum_{i=1}^{I} z_{ji}[log(P_i) + log(\text{Beta}(x_j; \alpha_i, \beta_i))]$$

Paso E Se calcula el valor esperado de las variables latentes:

$$Q(\theta|\theta^{(t)}, x) = \sum_{j=1}^{N} \sum_{i=1}^{I} E(z_{ji}|x_j, \theta^{(t)})[log(P_i) + log(\text{Beta}(x_j; \alpha_i, \beta_i))]$$

Dada la naturaleza Bernoulli de z_{ji} que corresponde a la indicatriz de si la observación j proviene de la componente i, es fácil notar dada la naturaleza de la variable latente, que:

$$E(z_{ji}|x_i, \theta^{(t)}) = 1 \times P(z_{ji} = 1|x_j, \theta^{(t)}) + 0 \times P(z_{ji} = 0|x_i, \theta^{(t)})$$

$$E(z_{ji}|x_j, \theta^{(t)}) = 1 \times P(z_{ji} = 1|x_j, \theta^{(t)})$$

$$E(z_{ji}|x_j, \theta^{(t)}) = \frac{P_i^{(t)}\text{Beta}(x_j; \alpha_i^{(t)}, \beta_i^{(t)})}{\sum^I P_k^{(t)}\text{Beta}(x_j; \alpha_i^{(t)}, \beta_i^{(t)})}$$

Ejercicio 2

Luego, sea

$$\widetilde{z}_{ji} = \frac{P_i^{(t)}\text{Beta}(x_j;\alpha_i^{(t)},\beta_i^{(t)})}{\sum^I P_k^{(t)}\text{Beta}(x_j;\alpha_i^{(t)},\beta_i^{(t)})}$$

El paso E queda expresado como sigue:

$$Q(\theta|\theta^{(t)},x) = \sum_{j=1}^{N}\sum_{i=1}^{I}\widetilde{z}_{ji}[log(P_i) + log(\text{Beta}(x_j;\alpha_i,\beta_i))]$$

Paso M Maximización de los parámetros

Obtenemos expresión para los pesos P_i utilizando $\widetilde{z}_{ji}$. Notar que está presente el contraste o restricción $\sum P_i = 1$ por lo tanto esta maximización debe realizarse utilizando multiplicadores de Lagrange (para el caso de dos mezclas es bastante directo). De todas formas un símil fue visto en clases, y se concluye que:

$$P_i^{(t)} = \frac{\sum_{j=1}^{N} \widetilde{z}_{ji}^{(t)}}{N}$$

Luego se deben obtener los EMV para α_i y β_i, derivando e igualando a cero es posible notar que no se llega a expresiones cerradas, por lo tanto es necesario utilizar algoritmos de aproximación como Newton Raphson.

Ejercicio 2

Un alcance muy interesante es que es necesario partir el algoritmo EM con valores iniciales. ¿Cómo obtener $\theta^{(0)}$? Puede utilizarse clustering con K-Means, para agrupar los datos y así estimar los parámetros para cada componente/mezcla.