Introdução ao Algoritmo EM

Erick Amorim

Departamento de Estatística
Universidade Federal de Minas Gerais

30 de Setembro de 2016

SUMÁRIO

Expectation-Maximization

- O algoritmo EM (Expectation-Maximization) é uma técnica de otimização originalmente introduzida por Dempster, Laird e Rubin (1977).

- É um algoritmo usado para obter o estimador que maximiza a verossimilhança (EMV).

- Segundo Casella e Berge (2010), o EM é um algoritmo que seguramente converge para o EMV e tem como base substituir uma difícil maximização da verossimilhança por uma sequência de maximizações mais fáceis, cujo limite é o EMV.

- A demostração original da Convergência do EM realizada or Dempster, Lair e Rubin (1977) tinha uma falha, mas provas válidas da convergência foram apresentadas por Boyles (1983) e Wu (1983).

EM

- O algoritmo EM tem grande utilidade em problemas de dados incompletos, onde a estimação de máxima verossimilhança é difícil devido a ausência de parte dos dados.

- O algoritmo é muito útil em problemas de: Clustering, reconhecimento de padrões, modelos de Markov oculto, entre outros.

O algoritmo consiste de duas etapas:

Etapa E Calcula-se o valor esperado do logaritmo da verossimilhança;

Etapa M Encontra-se o máximo deste valor esperado, iterando a esperança até a convergência ou critério de parada.

Aplicação 1

Este problema foi proposto por Rao (1973, p. 368-369). McLachlan and Krishnan (1997) também discutem este problema e apresentaremos o modelo deles.

Os dados abaixo se referem a $n = 197$ animais que se distribuem em 4 caterogias (C_1, C_2, C_3, C_4) de modo que o vetor de frequência observado é $\mathbf{x} = (x_1, x_2, x_3, x_4)' = (125, 18, 20, 34)'$.

Neste caso temos uma distribuição Multinomial com 4 caterogrias e função de probabilidade dada por:

$$P(X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4) = g(\mathbf{x} \mid \theta)$$

com $g(\mathbf{x} \mid \theta) = \dfrac{n!}{x_1!x_2!x_3!x_4!} \left(\dfrac{1}{2} + \dfrac{\theta}{4}\right)^{x_1} \left(\dfrac{1}{4} - \dfrac{\theta}{4}\right)^{x_2} \left(\dfrac{1}{4} - \dfrac{\theta}{4}\right)^{x_3} \left(\dfrac{\theta}{4}\right)^{x_4}$

onde $n = x_1 + x_2 + x_3 + x_4$ e $0 \leq \theta \leq 1$.

Solução Analítica da Aplicação 1

A função de verossimilhança $L(\theta;\mathbf{x}) = g(\mathbf{x} \mid \theta)$

$$\mathrm{g}(\mathrm{x}|\ \theta) = \frac{n!}{x_1!x_2!x_3!x_4!}\left(\frac{1}{2}+\frac{\theta}{4}\right)^{x_1}\left(\frac{1}{4}(1-\theta)\right)^{x_2+x_3}\left(\frac{\theta}{4}\right)^{x_4}$$

$$\mathrm{g}(\mathrm{x}|\ \theta) \propto (2+\theta)^{x_1}\,(1-\theta)^{x_2+x_3}\,\theta^{x_4}$$

e o logaritmo da verossimilhança será:

$$l(\theta) \propto x_1 \ln(2+\theta) + (x_2+x_3)\ln(1-\theta) + x_4 \ln\theta$$

e além do mais:

$$\frac{d}{d\theta}l(\theta) = \frac{x_1}{2+\theta} - \frac{x_2+x_3}{1-\theta} + \frac{x_4}{\theta}$$

fazendo $\frac{d}{d\theta}l(\theta) = 0$ chegaremos na equação

Solução Analítica da Aplicação 1

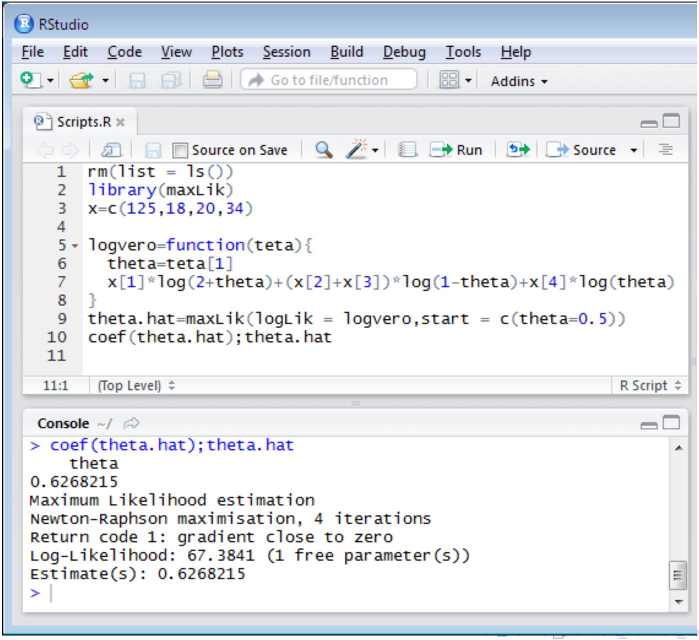
$n\theta^2 + (2x_2 + 2x_3 - x_1 + x_4)\theta - 2x_4 = 0$ cuja solução será:

$$\hat{\theta}_{MV} = \frac{-(2x_2 + 2x_3 - x_1 + x_4) + \sqrt{(2x_2 + 2x_3 - x_1 + x_4)^2 + 8nx_4}}{2n}$$

então para $\mathbf{x} = (125, 18, 20, 34)'$ teremos

$$\hat{\theta}_{MV} = \frac{15 + \sqrt{53809}}{394} = 0,626821498$$

Script da Aplicação 1



Alguns detalhes do Método

Suponha que $\mathbf{X} = (X_1, ..., X_n)$ sejam v.a. i.i.d. com distribuição conjunta $g(\mathbf{x} \mid \theta)$. Neste caso $g(\mathbf{x} \mid \theta) = L(\theta; \mathbf{x})$. E o que se deseja é $\hat{\theta} = \arg\max L(\theta; \mathbf{x})$.

Vamos considerar que os dados completos $\mathbf{w}$ são provenientes de uma amostra aleatória $\mathbf{W} = (\mathbf{X}, \mathbf{Z})$. Neste caso a verossimilhança completa é:

$$f(\mathbf{w} \mid \theta) = f(\mathbf{x}, \mathbf{z} \mid \theta) = k(\mathbf{z} \mid \mathbf{x}, \theta) g(\mathbf{x} \mid \theta)$$

O objetivo é maximizar $L(\theta \mid \mathbf{x})$ usando a verossimilhança completa $L^c(\theta \mid \mathbf{x}, \mathbf{z}) = f(\mathbf{x}, \mathbf{z} \mid \theta)$ neste processo.

Como calculamos $L^c(\theta \mid \mathbf{x}, \mathbf{z})$ se não conhecemos $\mathbf{z}$?

Alguns Detalhes do Método

Como não conhecemos $\mathbf{z}$ em $L^c(\theta \mid \mathbf{x}, \mathbf{z})$ vamos considerar que $L^c(\theta \mid \mathbf{x}, \mathbf{z})$ é uma v.a.

Calcularemos uma média de funções de verossimilhanças sobre variáveis não observadas (latentes).

Há uma relação entre as verossimilhanças dada por:

$$\mathrm{L}^c(\theta \mid \mathbf{x}, \mathbf{z}) = k(\mathbf{z} \mid \mathbf{x}, \theta) L(\theta \mid \mathbf{x})$$

$$\ln L^c(\theta \mid \mathbf{x}, \mathbf{z}) = \ln k(\mathbf{z} \mid \mathbf{x}, \theta) + \ln L(\theta \mid \mathbf{x})$$

Usando a igualdade $k(\mathbf{z} \mid \mathbf{x}, \theta) = \dfrac{f(\mathbf{x}, \mathbf{z} \mid \theta)}{g(\mathbf{x} \mid \theta)}$ e para um valor $\theta_0 \in \Theta$ fixado teremos:

Com Alguns Cálculos

$$\ln L(\theta \mid \mathbf{x}) = \int \ln\left[L(\theta \mid \mathbf{x})\right] k(z \mid \mathbf{x}, \theta_0) d\mathbf{z}$$

$$\ln L(\theta \mid \mathbf{x}) = \int \ln\left[g(\mathbf{x} \mid \theta)\right] k(z \mid \mathbf{x}, \theta_0) d\mathbf{z}$$

$$\ln L(\theta \mid \mathbf{x}) = \int \ln\left[\frac{f(\mathbf{x}, \mathbf{z} \mid \theta)}{k(\mathbf{z} \mid \mathbf{x}, \theta)}\right] k(z \mid \mathbf{x}, \theta_0) d\mathbf{z}$$

$$\ln L(\theta \mid \mathbf{x}) = \int \left[\ln f(\mathbf{x}, \mathbf{z} \mid \theta) - \ln k(\mathbf{z} \mid \mathbf{x}, \theta)\right] k(z \mid \mathbf{x}, \theta_0) d\mathbf{z}$$

$$\ln L(\theta \mid \mathbf{x}) = \int \ln f(\mathbf{x}, \mathbf{z} \mid \theta) k(z \mid \mathbf{x}, \theta_0) d\mathbf{z} - \int \ln k(\mathbf{z} \mid \mathbf{x}, \theta) k(z \mid \mathbf{x}, \theta_0) d\mathbf{z}$$

$$\underbrace{\ln L(\theta \mid \mathbf{x})}_{\text{Dados Observados}} = \underbrace{E_{\theta_0}\left[\ln L^c(\theta \mid \mathbf{x}, \mathbf{z}) \mid \theta_0, \mathbf{x}\right]}_{\text{Dados Completos}} - \underbrace{E_{\theta_0}\left[\ln k(\mathbf{z} \mid \mathbf{x}, \theta) \mid \theta_0, \mathbf{x}\right]}_{\text{Dados Ausentes}}$$

Alguns Detalhes do Método

Definimos a função Q do termo da equação anterior que será Maximizada no passo M como $Q(\theta \mid \theta_0, \mathbf{x}) = E_{\theta_0}\left[\ln L^c(\theta \mid \mathbf{x}, \mathbf{z}) \mid \theta_0, \mathbf{x}\right]$

No processamento do algoritmo, maximizar $Q(\theta \mid \theta_0, \mathbf{x})$ é equivalente a maximizar $\ln L(\theta \mid \mathbf{x})$

Denote por $\theta^{(0)}$ uma estimativa inicial de θ. Então $\theta^{(1)}$ é o argumento que maximiza $Q(\theta \mid \theta_0, \mathbf{x})$. Esse é o primeiro passo para estimar θ. Se nós procedermos desta forma teremos o algoritimo definido como:

Algoritmo EM

Seja $\theta^{(m)}$ a estimativa na m-ésima iteração. Para calcular a estimativa na iteração $(m+1)$ proceda da seguinte maneira:

1. Etapa E: Calcule
$Q(\theta \mid \hat{\theta}_m, \mathbf{x}) = E_{\hat{\theta}^{(m)}}\left[\ln L^c(\theta \mid \mathbf{x}, \mathbf{z}) \mid \hat{\theta}_m, \mathbf{x}\right]$
2. Etapa M: Calcule $\hat{\theta}^{(m+1)} = \arg\max Q(\theta \mid \hat{\theta}_m, \mathbf{x})$

Sob fortes condições pode ser mostrado que $\hat{\theta}^{(m)}$ converge em probabilidade para o EMV quando $m \to \infty$. Isso não será mostrado aqui.

Mas pode-se mostrar que $\hat{\theta}^{(m+1)}$ sempre aumenta a verossimilhança em relação a $\hat{\theta}^{(m)}$.

Alguns Detahes do Método

Teorema: A sequência de estimadores $\hat{\theta}^{(m)}$ satisfaz

$$L(\hat{\theta}^{(m+1)} \mid \mathbf{x}) \geq L(\hat{\theta}^{(m)} \mid \mathbf{x}).$$

Prova: Isso é devido a $\hat{\theta}^{(m+1)}$ maximizar $Q(\theta \mid \hat{\theta}^{(m)}, \mathbf{x})$, pois temos:

$$Q(\hat{\theta}^{(m+1)} \mid \hat{\theta}^{(m)}, \mathbf{x}) \geq Q(\hat{\theta}^{(m)} \mid \hat{\theta}^{(m)}, \mathbf{x})$$

que pode ser escrito como,

$$E_{\hat{\theta}^{(m)}}\left[\ln L^c(\hat{\theta}^{(m+1)} \mid \mathbf{x}, \mathbf{z})\right] \geq E_{\hat{\theta}^{(m)}}\left[\ln L^c(\hat{\theta}^{(m)} \mid \mathbf{x}, \mathbf{z})\right],$$

onde a esperança é em relação a $k(\mathbf{z} \mid \hat{\theta}^{(m)}, \mathbf{x})$.

A prova é completada mostrando que
$E_{\hat{\theta}^{(m)}}\left[\ln k(\mathbf{z} \mid \hat{\theta}^{(m+1)}, \mathbf{x})\right] \leq E_{\hat{\theta}^{(m)}}\left[\ln k(\mathbf{z} \mid \hat{\theta}^{(m)}, \mathbf{x})\right]$

Alguns Detalhes do Método

Aplicando a desigualdade de Jensen tem-se:

$$E_{\hat{\theta}^{(m)}}\left[\ln \frac{k(\mathbf{z} \mid \hat{\theta}^{(m+1)}, \mathbf{x})}{k(\mathbf{z} \mid \hat{\theta}^{(m)}, \mathbf{x})}\right] \quad \leq \ln E_{\hat{\theta}^{(m)}}\left[\frac{k(\mathbf{z} \mid \hat{\theta}^{(m+1)}, \mathbf{x})}{k(\mathbf{z} \mid \hat{\theta}^{(m)}, \mathbf{x})}\right]$$

$$= \ln \int \frac{k(\mathbf{z} \mid \hat{\theta}^{(m+1)}, \mathbf{x})}{k(\mathbf{z} \mid \hat{\theta}^{(m)}, \mathbf{x})} k(\mathbf{z} \mid \hat{\theta}^{(m)}, \mathbf{x}) d\mathbf{z} = \ln(1) = 0.$$

Aplicação 1

Recordando o exemplo que apresentamos referente aos dados de $n = 197$ animais que se distribuem em 4 categorias, de modo que o vetor de frequência é $\mathbf{x} = (x_1, x_2, x_3, x_4)' = (125, 18, 20, 34)'$.

- A função de distribuição conjunta para os dados observados é

$$g(\mathbf{x} \mid \theta) = \frac{n!}{x_1! x_2! x_3! x_4!} \left(\frac{1}{2} + \frac{\theta}{4}\right)^{x_1} \left(\frac{1}{4} - \frac{\theta}{4}\right)^{x_2} \left(\frac{1}{4} - \frac{\theta}{4}\right)^{x_3} \left(\frac{\theta}{4}\right)^{x_4}$$

- Cujo núcleo é $g(\mathbf{x} \mid \theta) \propto (2+\theta)^{x_1} (1-\theta)^{x_2+x_3} \theta^{x_4}$.

Agora suponha que os dados da maior categoria seja proveniente das categorias z_1 e z_2, com probabilidades $\frac{1}{2}$ e $\frac{\theta}{4}$, respectivamente.

Aplicaçao 1

Desta maneira se introduzem variaveis latentes ($x_1 = z_1 + z_2$) resultando em 5 categorias: $w = (z_1, z_2, x_2, x_3, x_4)'$ que representam os dados completos e $\mathbf{z} = (z_1, z_2)$ os dados perdidos.

* A distribuição conjunta de $\mathbf{w}$ é $f(\mathbf{w}) = f(\mathbf{x}, \mathbf{z} \mid \theta)$, onde

$$f(\mathbf{x}, \mathbf{z} \mid \theta) = \frac{n!}{z_1! z_2! x_2! x_3! x_4!} \left(\frac{1}{2}\right)^{z_1} \left(\frac{\theta}{4}\right)^{z_2} \left(\frac{1}{4} - \frac{\theta}{4}\right)^{x_2+x_3} \left(\frac{\theta}{4}\right)^{x_4}$$

* E o núcleo da distribuição é

$$f(\mathbf{x}, \mathbf{z} \mid \theta) \propto \theta^{z_2+x_4} (1-\theta)^{x_2+x_3}$$

Aplicação 1

Usando as distribuições dos dados completos e dos dados observados podemos calcular a distribuição de $[\mathbf{z} \mid \mathbf{x}, \theta]$ que é

$$k(\mathbf{z} \mid \mathbf{x}, \theta) = \frac{f(\mathbf{x}, \mathbf{z} \mid \theta)}{g(\mathbf{x} \mid \theta)} = \binom{x_1}{z_2} \left(\frac{\theta}{2+\theta} \right)^{z_2} \left(1 - \frac{\theta}{2+\theta} \right)^{x_1 - z_2}$$

Veja que $Z_2 \sim \text{Binomial}\left(x_1, \frac{\theta}{2+\theta}\right)$ e sua média é $E\left[Z_2\right] = x_1 \left(\frac{\theta}{2+\theta} \right)$.

Para o **passo E** precisamos calcular a esperança de $\ln L^c\left(\theta \mid \mathbf{x}, \mathbf{z}\right)$ em relação a $k(\mathbf{z} \mid \theta^{(0)}, \mathbf{x})$.

Aplicação 1

$$\ln L^c(\theta \mid \mathbf{x}, \mathbf{z}) = \ln f(\mathbf{x}, \mathbf{z} \mid \theta) \propto (Z_2 + x_4) \ln \theta + (x_2 + x_3) \ln(1 - \theta)$$

$$\mathrm{E}_{\hat{\theta}^{(m)}} \left[\ln L^c(\theta \mid \mathbf{x}, \mathbf{z})\right] \propto E_{\hat{\theta}^{(m)}} \left[(Z_2 + x_4) \ln \theta + (x_2 + x_3) \ln(1 - \theta)\right]$$

$$\mathrm{E}_{\hat{\theta}^{(m)}} \left[\ln L^c(\theta \mid \mathbf{x}, \mathbf{z})\right] \propto E_{\hat{\theta}^{(m)}} \left[Z_2\right] \ln \theta + x_4 \ln \theta + (x_2 + x_3) \ln(1 - \theta)$$

$$\mathrm{Q}\left(\theta \mid \hat{\theta}^{(m)}\right) \propto \frac{x_1 \theta^{(m)}}{2 + \theta^{(m)}} \ln \theta + x_4 \ln \theta + (x_2 + x_3) \ln(1 - \theta)$$

Para o **passo M** precisamos encontrar o maximo de $Q\left(\theta \mid \hat{\theta}^{(m)}\right)$;

$$\frac{d}{d\theta} Q\left(\theta \mid \hat{\theta}^{(m)}\right) = 0$$

chega-se a $\hat{\theta} = \dfrac{x_4 + x_1 \dfrac{\hat{\theta}^{(m)}}{2 + \hat{\theta}^{(m)}}}{x_2 + x_3 + x_4 + x_1 \dfrac{\hat{\theta}^{(m)}}{2 + \hat{\theta}^{(m)}}}$

Aplicação 1

Subistituindo os dados $x_1 = 125$, $x_2 = 18$, $x_3 = 20$ e $x_4 = 34$ tem-se

$$\hat{\theta} = \frac{34 + 125\dfrac{\hat{\theta}^{(m)}}{2 + \hat{\theta}^{(m)}}}{72 + 125\dfrac{\hat{\theta}^{(m)}}{2 + \hat{\theta}^{(m)}}}$$

Em cada iteração tem-se

$$\hat{\theta}^{(m+1)} = \frac{68 + 159\hat{\theta}^{(m)}}{144 + 197\hat{\theta}^{(m)}}$$

Veja o Script no R.

Aplicação 2

Vamos considerar um problema envolvendo misturas de normais. Suponha que $Y_1 \sim N(\mu_1, \sigma_1^2)$ e $Y_2 \sim N(\mu_2, \sigma_2^2)$. Considere também uma v.a. W sendo uma Bernoulli independente de Y_1 e Y_2 com probabilidade de sucesso $\epsilon = P(W = 1)$.

Suponha que a variavél aleatória que observamos seja

$$X = (1 - W)Y_1 + WY_2.$$

Nesse caso o vetor de parâmetros é $\theta' = (\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \epsilon)$ e a função densidade de X é:

$$f(x) = (1 - \epsilon)f_1(x) + \epsilon f_2(x)$$

Aplicação 2

Em problemas de misturas os dados não observáveis são v.a. que identificam as componentes da mistura. Então para $i = 1, ..., n$ definimos variáveis

$$W_i = \begin{cases} 0, & \text{se } X_i \text{ tem densidade } f_1(x) \\ 1, & \text{se } X_i \text{ tem densidade } f_2(x). \end{cases}$$

Desta forma os $W_{i's}$ são i.i.d. com probabilidade de sucesso ϵ.

A verossimilhança completa será:

$$L^c(\theta \mid \mathbf{x}, \mathbf{w}) = \prod_{W_i=0} f_1(x_i) \prod_{W_i=1} f_2(x_i)$$

Aplicação 2

O logaritmo da verossimilhança completa é:

$$\ln L^c(\theta \mid \mathbf{x}, \mathbf{w}) = \sum_{W_i=0} \ln f_1(x_i) + \sum_{W_i=1} \ln f_2(x_i)$$

$$\ln L^c(\theta \mid \mathbf{x}, \mathbf{w}) = \sum_{i=1}^{n} \left[(1 - W_i) \ln f_1(x_i) + W_i \ln f_2(x_i)\right].$$

Então calculando o valor esperado tem-se:

$$E_{\hat{\theta}^{(m)}}\left[\ln L^c(\theta \mid \mathbf{x}, \mathbf{w})\right] = \sum_{i=1}^{n} \left[\left(1 - E_{\hat{\theta}^{(m)}}[W_i]\right) \ln f_1(x_i) + E_{\hat{\theta}^{(m)}}[W_i] \ln f_2(x_i)\right]$$

Para o **passo E** precisamos da esperança condicional de W_i dado $\mathbf{x}$ e $\theta^{(m)}$ que é $E\left[W_i \mid \theta^{(m)}, \mathbf{x}\right] = P\left[W_i = 1 \mid \theta^{(m)}, \mathbf{x}\right]$

Aplicação 2

Uma estimativa desse valor esperado é dado por:

$$\gamma_i = \hat{P}\left[W_i = 1 \mid \theta^{(m)}, \mathbf{x}\right] = \frac{\hat{\epsilon} f_{2,(m)}(x_i)}{(1-\hat{\epsilon})\, f_{1,(m)}(x_i) + \hat{\epsilon} f_{2,(m)}(x_i)},$$

detalhes sobre a expressão acima podem ser vistas em McLachlan and Krishnan (1997).

Substituindo γ_i na Esperança, temos a expressão do **passo E**:

$$Q(\theta \mid \hat{\theta}^{(m)}, \mathbf{x}) = \sum_{i=1}^{n} \left[(1-\gamma_i) \ln f_1(x_i) + \gamma_i \ln f_2(x_i)\right]$$

Para o **passo M** basta derivar em relação a cada parâmetro por exemplo:

$$\frac{\partial}{\partial \mu_1} Q(\theta \mid \hat{\theta}^{(m)}, \mathbf{x}) = \sum_{i=1}^{n} (1-\gamma_i) \left(-\frac{1}{2\sigma_1^2}\right) (-2)(x_i - \mu_1)$$

Aplicação 2

Fazendo o cálculo para cada μ e σ^2 obtemos as seguintes estimativas:

$$\hat{\mu}_1 = \frac{\sum_{i=1}^{n}(1-\gamma_i)x_i}{\sum_{i=1}^{n}(1-\gamma_i)}$$

$$\hat{\sigma^2}_1 = \frac{\sum_{i=1}^{n}(1-\gamma_i)(x_i-\hat{\mu}_1)^2}{\sum_{i=1}^{n}(1-\gamma_i)}$$

$$\hat{\mu}_2 = \frac{\sum_{i=1}^{n}\gamma_i x_i}{\sum_{i=1}^{n}\gamma_i}$$

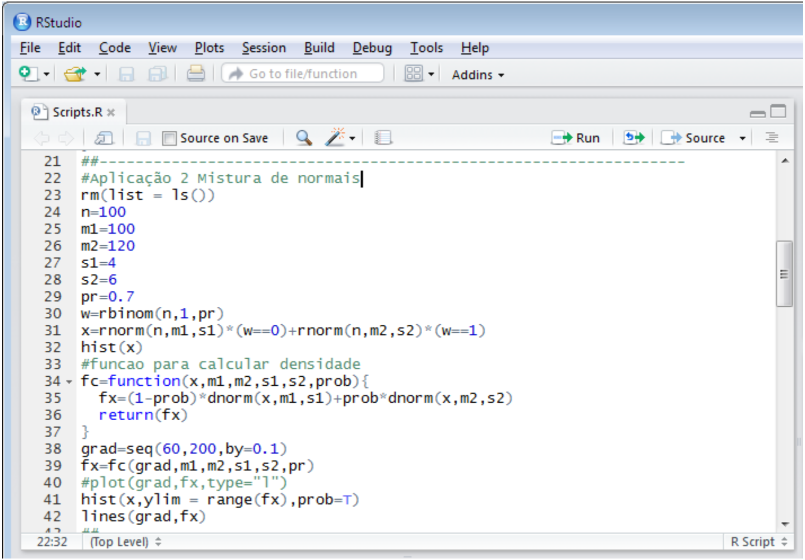
$$\hat{\sigma^2}_2 = \frac{\sum_{i=1}^{n}\gamma_i(x_i-\hat{\mu}_2)^2}{\sum_{i=1}^{n}\gamma_i}$$

Já que a estimativa de $P\left[W_i = 1 \mid \theta^{(m)}, \mathbf{x}\right]$ é γ_i, então para estimar $\epsilon = P(W_i = 1)$ usamos a média $\hat{\epsilon} = \dfrac{\sum_{i=1}^{n}\gamma_i}{n}$

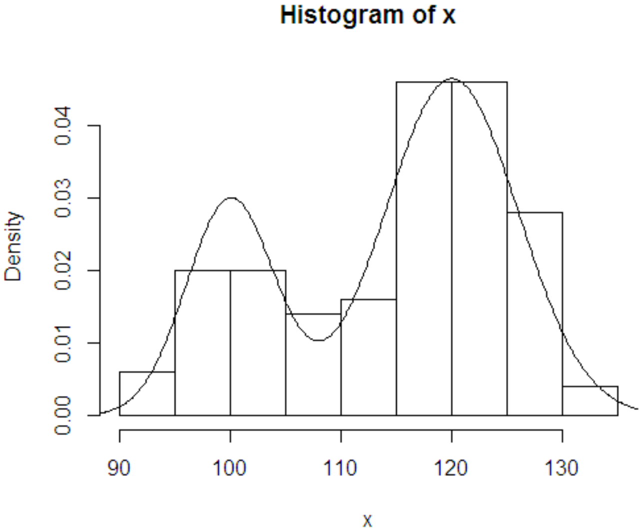
Aplicação 2

Os seguintes dados foram gerados da v.a. $X = (1-W)Y_1 + WY_2$ com $Y_1 \sim N(100, 4^2)$, $Y_2 \sim N(120, 6^2)$ e $W \sim$ Bernoull(0.7). Veja o Script no R.

Aplicação 2


RStudio
File Edit Code View Plots Session Build Debug Tools Help
Go to file/function
Addins
Scripts.R
Source on Save
Run
Source
##--
#Aplicação 2 Mistura de normais
rm(list = ls())
n=100
m1=100
m2=120
s1=4
s2=6
pr=0.7
w=rbinom(n,1,pr)
x=rnorm(n,m1,s1)*(w==0)+rnorm(n,m2,s2)*(w==1)
hist(x)
#funcao para calcular densidade
fc=function(x,m1,m2,s1,s2,prob){
fx=(1-prob)*dnorm(x,m1,s1)+prob*dnorm(x,m2,s2)
return(fx)
}
grad=seq(60,200,by=0.1)
fx=fc(grad,m1,m2,s1,s2,pr)
#plot(grad,fx,type="l")
hist(x,ylim = range(fx),prob=T)
lines(grad,fx)
22:32
(Top Level)
R Script

Aplicação 2



Aplicação 2

```r
#param[1] Media 1
#param[2] Media 2
#param[3] variancia 1
#param[4] variancia 2
#param[5] epsilon (peso da componente 2)

#y=c(195,166,188,195,179,198,161,179,200,191)
param=c(180,100,15,20,0.5) #chute inicial
e=0.0001
erro=0.5; it=0
while(erro>e){
  param0=param
  #Passo E
  part1 = (1-param[5])*dnorm(x,param[1],param[3])
  part2 = param[5]*dnorm(x,param[2],param[4])
  gam = part2/(part1+part2)
  #passo M
  aux= c(sum( (1-gam)*x)/sum(1-gam),sum(gam*x)/sum(gam))
  param[1] = min(aux)
  param[2] = max(aux)
  param[3] = sqrt(sum( ((1-gam)*(x-param[1])^2) )/sum(1-gam))
  param[4] = sqrt(sum( (gam*(x-param[2])^2) )/sum(gam))
  param[5] = mean(gam)
  #erro = sum((param0-param)^2)/(sum(param))^2
  erro = max(abs(param0-param)/(abs(param0)+0.001))
  it=it+1
}
round(param,3); it
```

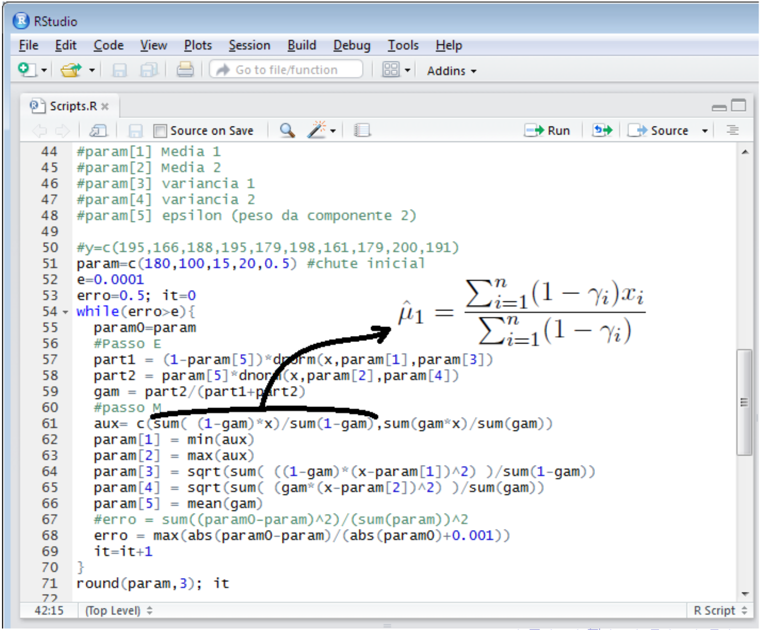
Aplicação 2

```r
#param[1] Media 1
#param[2] Media 2
#param[3] variancia 1
#param[4] variancia 2
#param[5] epsilon (peso da componente 2)

#y=c(195,166,188,195,179,198,161,179,200,191)
param=c(180,100,15,20,0.5) #chute inicial
e=0.0001
erro=0.5; it=0
while(erro>e){
  param0=param
  #Passo E
  part1 = (1-param[5])*dnorm(x,param[1],param[3])
  part2 = param[5]*dnorm(x,param[2],param[4])
  gam = part2/(part1+part2)
  #passo M
  aux= c(sum( (1-gam)*x)/sum(1-gam),sum(gam*x)/sum(gam))
  param[1] = min(aux)
  param[2] = max(aux)
  param[3] = sqrt(sum( ((1-gam)*(x-param[1])^2) )/sum(1-gam))
  param[4] = sqrt(sum( (gam*(x-param[2])^2) )/sum(gam))
  param[5] = mean(gam)
  #erro = sum((param0-param)^2)/(sum(param))^2
  erro = max(abs(param0-param)/(abs(param0)+0.001))
  it=it+1
}
round(param,3); it
```

$$\gamma_i = \frac{\hat{\epsilon} f_{2,(m)}(x_i)}{(1-\hat{\epsilon})\, f_{1,(m)}(x_i) + \hat{\epsilon} f_{2,(m)}(x_i)}$$

Aplicação 2

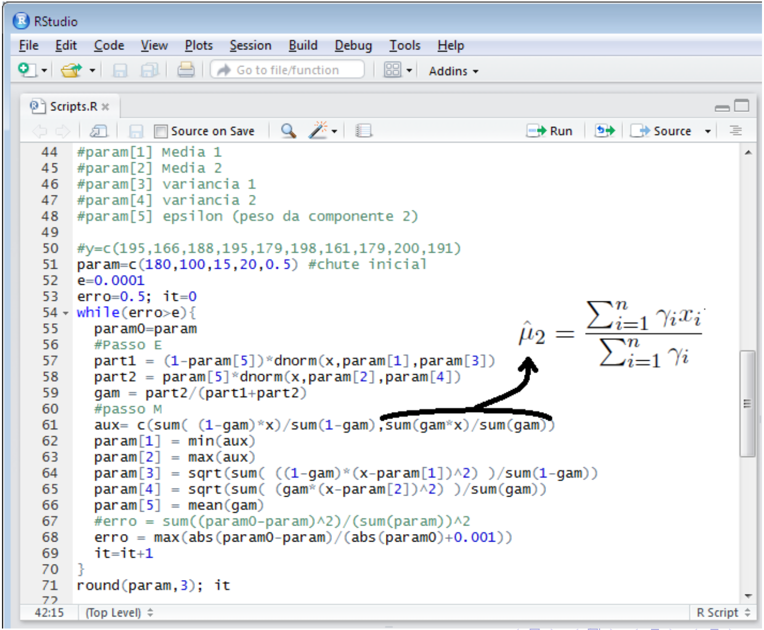



```r
#param[1] Media 1
#param[2] Media 2
#param[3] variancia 1
#param[4] variancia 2
#param[5] epsilon (peso da componente 2)

#y=c(195,166,188,195,179,198,161,179,200,191)
param=c(180,100,15,20,0.5) #chute inicial
e=0.0001
erro=0.5; it=0
while(erro>e){
  param0=param
  #Passo E
  part1 = (1-param[5])*dnorm(x,param[1],param[3])
  part2 = param[5]*dnorm(x,param[2],param[4])
  gam = part2/(part1+part2)
  #passo M
  aux= c(sum( (1-gam)*x)/sum(1-gam),sum(gam*x)/sum(gam))
  param[1] = min(aux)
  param[2] = max(aux)
  param[3] = sqrt(sum( ((1-gam)*(x-param[1])^2) )/sum(1-gam))
  param[4] = sqrt(sum( (gam*(x-param[2])^2) )/sum(gam))
  param[5] = mean(gam)
  #erro = sum((param0-param)^2)/(sum(param))^2
  erro = max(abs(param0-param)/(abs(param0)+0.001))
  it=it+1
}
round(param,3); it
```

$$\hat{\mu}_1 = \frac{\sum_{i=1}^{n}(1-\gamma_i)x_i}{\sum_{i=1}^{n}(1-\gamma_i)}$$

Aplicação 2




```r
#param[1] Media 1
#param[2] Media 2
#param[3] variancia 1
#param[4] variancia 2
#param[5] epsilon (peso da componente 2)

#y=c(195,166,188,195,179,198,161,179,200,191)
param=c(180,100,15,20,0.5) #chute inicial
e=0.0001
erro=0.5; it=0
while(erro>e){
  param0=param
  #Passo E
  part1 = (1-param[5])*dnorm(x,param[1],param[3])
  part2 = param[5]*dnorm(x,param[2],param[4])
  gam = part2/(part1+part2)
  #passo M
  aux= c(sum( (1-gam)*x)/sum(1-gam),sum(gam*x)/sum(gam))
  param[1] = min(aux)
  param[2] = max(aux)
  param[3] = sqrt(sum( ((1-gam)*(x-param[1])^2) )/sum(1-gam))
  param[4] = sqrt(sum( (gam*(x-param[2])^2) )/sum(gam))
  param[5] = mean(gam)
  #erro = sum((param0-param)^2)/(sum(param))^2
  erro = max(abs(param0-param)/(abs(param0)+0.001))
  it=it+1
}
round(param,3); it
```

$$\hat{\mu}_2 = \frac{\sum_{i=1}^{n} \gamma_i x_i}{\sum_{i=1}^{n} \gamma_i}$$

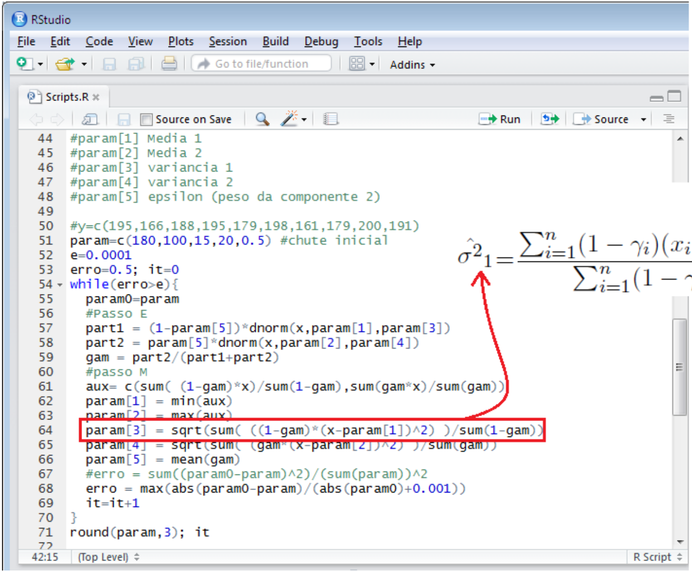
Aplicação 2

```r
#param[1] Media 1
#param[2] Media 2
#param[3] variancia 1
#param[4] variancia 2
#param[5] epsilon (peso da componente 2)

#y=c(195,166,188,195,179,198,161,179,200,191)
param=c(180,100,15,20,0.5) #chute inicial
e=0.0001
erro=0.5; it=0
while(erro>e){
  param0=param
  #Passo E
  part1 = (1-param[5])*dnorm(x,param[1],param[3])
  part2 = param[5]*dnorm(x,param[2],param[4])
  gam = part2/(part1+part2)
  #passo M
  aux= c(sum( (1-gam)*x)/sum(1-gam),sum(gam*x)/sum(gam))
  param[1] = min(aux)
  param[2] = max(aux)
  param[3] = sqrt(sum( ((1-gam)*(x-param[1])^2) )/sum(1-gam))
  param[4] = sqrt(sum( (gam*(x-param[2])^2) )/sum(gam))
  param[5] = mean(gam)
  #erro = sum((param0-param)^2)/(sum(param))^2
  erro = max(abs(param0-param)/(abs(param0)+0.001))
  it=it+1
}
round(param,3); it
```

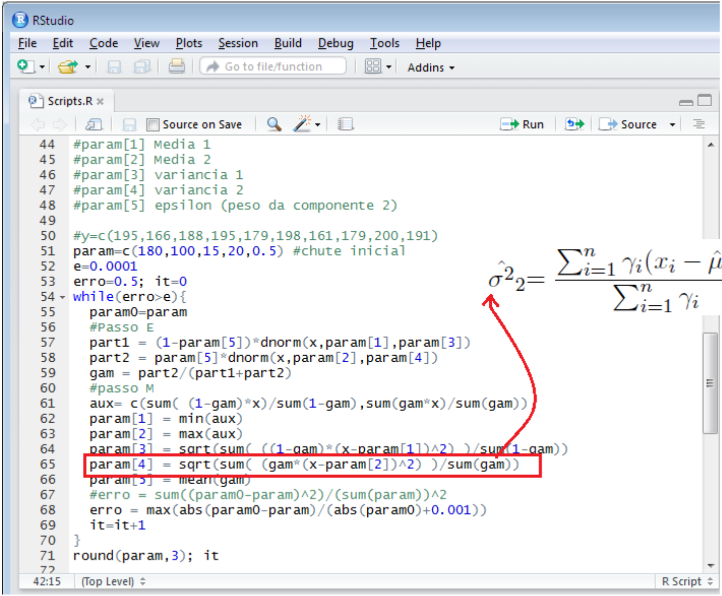
$\mu_1 < \mu_2$

Aplicação 2



$$\hat{\sigma}^2{}_1 = \frac{\sum_{i=1}^{n}(1-\gamma_i)(x_i-\hat{\mu}_1)^2}{\sum_{i=1}^{n}(1-\gamma_i)}$$

Aplicação 2



$$\hat{\sigma}^2_2 = \frac{\sum_{i=1}^{n} \gamma_i (x_i - \hat{\mu}_2)^2}{\sum_{i=1}^{n} \gamma_i}$$

Aplicação 2

```r
#param[1] Media 1
#param[2] Media 2
#param[3] variancia 1
#param[4] variancia 2
#param[5] epsilon (peso da componente 2)

#y=c(195,166,188,195,179,198,161,179,200,191)
param=c(180,100,15,20,0.5) #chute inicial
e=0.0001
erro=0.5; it=0
while(erro>e){
  param0=param
  #Passo E
  part1 = (1-param[5])*dnorm(x,param[1],param[3])
  part2 = param[5]*dnorm(x,param[2],param[4])
  gam = part2/(part1+part2)
  #passo M
  aux= c(sum( (1-gam)*x)/sum(1-gam),sum(gam*x)/sum(gam))
  param[1] = min(aux)
  param[2] = max(aux)
  param[3] = sqrt(sum( ((1-gam)*(x-param[1])^2) )/sum(1-gam))
  param[4] = sqrt(sum( (gam*(x-param[2])^2) )/sum(gam))
  param[5] = mean(gam)
  #erro = sum((param0-param)^2)/(sum(param))^2
  erro = max(abs(param0-param)/(abs(param0)+0.001))
  it=it+1
}
round(param,3); it
```

$$\hat{\epsilon} = \frac{\sum_{i=1}^{n} \gamma_i}{n}$$

Aplicação 2

```
Console ~/
+ }
> round(param,3); it
[1] 100.999 120.742   4.422   5.225   0.714
[1] 58
> 
```

Aplicação 3

Vamos considerar nesta aplicação um modelo dicotômico sobe o enfoque de variaveis latentes.

Nesse exemplo existe uma variavel latente contínua (Z) que não observamos diretamente e portanto não podemos mensurar.

O que observamos de fato é um indicador da variável não observada ou latente.

Sob este enfoque considere o modelo probit:
$\mathrm{Y}_i = \mathbf{1}_{[Z_i>0]}$,
$\mathrm{Z}_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \epsilon_i$,
$\epsilon_i \sim N(0,1)$

Induzindo a variável latente conseguimos modelar a probabilidade.

$$P\left(Y_i = 1\right) = P\left(Z_i > 0\right) = P\left(Z_i - X_i'\beta > -X_i'\beta\right) = \Phi\left(X_i'\beta\right),$$

podemos escrever a distribuição de Y_i sob forma de funções indicadoras:

$$f(y_i \mid z_i) = \mathbf{1}_{[Y_i=1]}\mathbf{1}_{[Z_i>0]} + \mathbf{1}_{[Y_i=0]}\mathbf{1}_{[Z_i\leq 0]}.$$

Veja que a densidade de $[Z_i \mid Y_i, \beta]$ é:

$$f(z_i \mid y_i, \beta) \propto f(y_i \mid z_i) f(z_i \mid \beta),$$

$$\mathrm{f}(z_i \mid y_i, \beta) \propto \left\{\mathbf{1}_{[Y_i=1]}\mathbf{1}_{[Z_i>0]} + \mathbf{1}_{[Y_i=0]}\mathbf{1}_{[Z_i\leq 0]}\right\} N\left(X_i'\beta, 1\right)$$

$$\mathrm{f}(z_i \mid y_i, \beta) = \begin{cases} N\left(X_i'\beta, 1\right)\mathbf{1}_{[Z_i>0]}, & \text{se } Y_i = 1 \\ N\left(X_i'\beta, 1\right)\mathbf{1}_{[Z_i\leq 0]}, & \text{se } Y_i = 0. \end{cases}$$

Aplicação 3

Escrevemos a verossimilhança completa da seguinte forma:

$\mathrm{L}^c(\beta \mid \mathbf{y}, \mathbf{z}) = \prod_{i=1}^{n} f(y_i \mid z_i) f(z_i \mid \beta)$

$$\ln L^c(\beta \mid \mathbf{y}, \mathbf{z}) = \sum_{i=1}^{n} [\ln f(y_i \mid z_i) + \ln f(z_i \mid \beta)]$$

$$\ln L^c(\beta \mid \mathbf{y}, \mathbf{z}) \propto \sum_{i=1}^{n} \ln f(z_i \mid \beta)$$

$$\ln L^c(\beta \mid \mathbf{y}, \mathbf{z}) \propto -\tfrac{1}{2} \sum_{i=1}^{n} (Z_i - X_i'\beta) = -\tfrac{1}{2} (\mathbf{Z} - \mathbf{X}\beta)' (\mathbf{Z} - \mathbf{X}\beta)$$

$$\ln L^c(\beta \mid \mathbf{y}, \mathbf{z}) \propto -\tfrac{1}{2} (\mathbf{Z}'\mathbf{Z} - 2\beta'\mathbf{X}'\mathbf{Z} + \beta'\mathbf{X}'\mathbf{X}\beta)$$

$$\underbrace{E_{\hat{\theta}^{(m)}} [\ln L^c(\beta \mid \mathbf{y}, \mathbf{z})]}_{Q\left(\beta \mid \beta^{(m)}, \mathbf{X}\right)} \propto -\tfrac{1}{2} \left(E_{\hat{\theta}^{(m)}} [\mathbf{Z}'\mathbf{Z}] - 2\beta'\mathbf{X}' E_{\hat{\theta}^{(m)}} [\mathbf{Z}] + \beta'\mathbf{X}'\mathbf{X}\beta \right)$$

$$\frac{\partial}{\partial \beta} Q\left(\beta \mid \beta^{(m)}, \mathbf{X}\right) = -\frac{1}{2} \left(0 - 2\mathbf{X}' E_{\hat{\theta}^{(m)}} [\mathbf{Z}] + 2\mathbf{X}'\mathbf{X}\beta\right)$$

Aplicação 3

$$2\mathbf{X}'\mathbf{X}\beta = 2\mathbf{X}'E_{\hat{\theta}^{(m)}}\left[\mathbf{Z}\right]$$

$$\beta_{\text{est}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'E_{\hat{\theta}^{(m)}}\left[\mathbf{Z}\right]$$

$$E\left[Z_i \mid \theta^{(m)}, \mathbf{y}\right] = \begin{cases} X_i'\beta - \dfrac{\phi(-X_i\beta)}{\Phi(-X_i\beta)}, & \text{se } Z_i \leq 0 \\[2ex] X_i'\beta + \dfrac{\phi(-X_i\beta)}{1-\Phi(-X_i\beta)}, & \text{se } Z_i > 0. \end{cases}$$

Aplicação 3

```r
#Exemplo probit
rm(list=ls())
resultado = matrix(0,nrow=1000,ncol=3)
for(jj in 1:1000){
  n=500
  beta0=0.2
  beta1=1.2
  beta2=-0.6
  x1=rbinom(n,1,0.5)
  x2=rnorm(n)
  pred=beta0+beta1*x1+beta2*x2
  prob=pnorm(pred)
  y=rbinom(n,1,prob)

  x=cbind(1,x1,x2)
  beta=array(0,c(3,1))
  e = 0.0001 ;it=0 ; erro = 0.5
  while(erro > e){
    beta.ini = beta
    xb = x%*%beta
    EZ_aux1= xb - ( dnorm(-xb)/pnorm(-xb) )        # se Zi <= 0
    EZ_aux2= xb + ( dnorm(-xb)/(1-pnorm(-xb)) )  # se Zi > 0
    EZ = EZ_aux1*(y==0) + EZ_aux2*(y==1)          # vetor de esperancas
    beta = solve( t(x)%*%x ) %*% t(x)%*%EZ        # estimativas na iteracao m
    erro = max(abs(beta.ini - beta)/(abs(beta.ini)+0.001))
    it = it+1
  }
#it;round(t(beta),3); cbind(beta0,beta1,beta2)
resultado[jj,]=t(beta)
}
```

Aplicação 3

RStudio

File Edit Code View Plots Session Build Debug Tools Help

Go to file/function Addins

Scripts.R

Source on Save Run Source

```r
#Exemplo probit
rm(list=ls())
resultado = matrix(0,nrow=1000,ncol=3)
for(jj in 1:1000){
  n=500
  beta0=0.2
  beta1=1.2
  beta2=-0.6
  x1=rbinom(n,1,0.5)
  x2=rnorm(n)
  pred=beta0+beta1*x1+beta2*x2
  prob=pnorm(pred)
  y=rbinom(n,1,prob)

  x=cbind(1,x1,x2)
  beta=array(0,c(3,1))
  e = 0.0001 ;it=0 ; erro = 0.5
  while(erro > e){
    beta.ini = beta
    xb = x%*%beta
    EZ_aux1= xb - ( dnorm(-xb)/pnorm(-xb) )       # se zi <= 0
    EZ_aux2= xb + ( dnorm(-xb)/(1-pnorm(-xb)) )  # se zi > 0
    EZ = EZ_aux1*(y==0) + EZ_aux2*(y==1)         # vetor de esperancas
    beta = solve( t(x)%*%x ) %*% t(x)%*%EZ       # estimativas na iteracao m
    erro = max(abs(beta.ini - beta)/(abs(beta.ini)+0.001))
    it = it+1
  }
#it;round(t(beta),3); cbind(beta0,beta1,beta2)
resultado[jj,]=t(beta)
}
```

Gerando Dados

73:1 (Top Level) R Script

Aplicação 3

```r
#Exemplo probit
rm(list=ls())
resultado = matrix(0,nrow=1000,ncol=3)
for(jj in 1:1000){
  n=500
  beta0=0.2
  beta1=1.2
  beta2=-0.6
  x1=rbinom(n,1,0.5)
  x2=rnorm(n)
  pred=beta0+beta1*x1+beta2*x2
  prob=pnorm(pred)
  y=rbinom(n,1,prob)

  x=cbind(1,x1,x2)
  beta=array(0,c(3,1))
  e = 0.0001 ;it=0 ; erro = 0.5
  while(erro > e){
    beta.ini = beta
    xb = x%*%beta
    EZ_aux1= xb - ( dnorm(-xb)/pnorm(-xb) )      # se Zi <= 0
    EZ_aux2= xb + ( dnorm(-xb)/(1-pnorm(-xb)) )  # se Zi > 0
    EZ = EZ_aux1*(y==0) + EZ_aux2*(y==1)        # vetor de esperancas
    beta = solve( t(x)%*%x ) %*% t(x)%*%EZ      # estimativas na iteracao m
    erro = max(abs(beta.ini - beta)/(abs(beta.ini)+0.001))
    it = it+1
  }
#it;round(t(beta),3); cbind(beta0,beta1,beta2)
resultado[jj,]=t(beta)
}
```

Aplicação 3

```r
#Exemplo probit
rm(list=ls())
resultado = matrix(0,nrow=1000,ncol=3)
for(jj in 1:1000){
  n=500
  beta0=0.2
  beta1=1.2
  beta2=-0.6
  x1=rbinom(n,1,0.5)
  x2=rnorm(n)
  pred=beta0+beta1*x1+beta2*x2
  prob=pnorm(pred)
  y=rbinom(n,1,prob)

  x=cbind(1,x1,x2)
  beta=array(0,c(3,1))
  e = 0.0001 ;it=0 ; erro = 0.5
  while(erro > e){
    beta.ini = beta
    xb = x%*%beta
    EZ_aux1= xb - ( dnorm(-xb)/pnorm(-xb) )       # se zi <= 0
    EZ_aux2= xb + ( dnorm(-xb)/(1-pnorm(-xb)) )   # se zi > 0
    EZ = EZ_aux1*(y==0) + EZ_aux2*(y==1)          # vetor de esperancas
    beta = solve( t(x)%*%x ) %*% t(x)%*%EZ        # estimativas na iteracao m
    erro = max(abs(beta.ini - beta)/(abs(beta.ini)+0.001))
    it = it+1
  }
#it;round(t(beta),3); cbind(beta0,beta1,beta2)
resultado[jj,]=t(beta)
}
```

$$\boxed{E\left[Z_i \mid \theta^{(m)}, \mathbf{y}\right] = X_i'\beta - \frac{\phi(-X_i\beta)}{\Phi(-X_i\beta)}}$$

Aplicação 3

```r
#Exemplo probit
rm(list=ls())
resultado = matrix(0,nrow=1000,ncol=3)
for(jj in 1:1000){
  n=500
  beta0=0.2
  beta1=1.2
  beta2=-0.6
  x1=rbinom(n,1,0.5)
  x2=rnorm(n)
  pred=beta0+beta1*x1+beta2*x2
  prob=pnorm(pred)
  y=rbinom(n,1,prob)

  x=cbind(1,x1,x2)
  beta=array(0,c(3,1))
  e = 0.0001 ;it=0 ; erro = 0.5
  while(erro > e){
    beta.ini = beta
    xb = x%*%beta
    EZ_aux1= xb - ( dnorm(-xb)/pnorm(-xb) )      # se Zi <= 0
    EZ_aux2= xb + ( dnorm(-xb)/(1-pnorm(-xb)) )  # se Zi > 0
    EZ = EZ_aux1*(y==0) + EZ_aux2*(y==1)         # vetor de esperancas
    beta = solve( t(x)%*%x ) %*% t(x)%*%EZ       # estimativas na iteracao m
    erro = max(abs(beta.ini - beta)/(abs(beta.ini)+0.001))
    it = it+1
  }
#it;round(t(beta),3); cbind(beta0,beta1,beta2)
resultado[jj,]=t(beta)
}
```

$$\boxed{E\left[Z_i \mid \theta^{(m)}, \mathrm{y}\right] = X_i'\beta + \frac{\phi(-X_i\beta)}{1-\Phi(-X_i\beta)}}$$

Aplicação 3

```r
#Exemplo probit
rm(list=ls())
resultado = matrix(0,nrow=1000,ncol=3)
for(jj in 1:1000){
  n=500
  beta0=0.2
  beta1=1.2
  beta2=-0.6
  x1=rbinom(n,1,0.5)
  x2=rnorm(n)
  pred=beta0+beta1*x1+beta2*x2
  prob=pnorm(pred)
  y=rbinom(n,1,prob)

  x=cbind(1,x1,x2)
  beta=array(0,c(3,1))
  e = 0.0001 ;it=0 ; erro = 0.5
  while(erro > e){
    beta.ini = beta
    xb = x%*%beta
    EZ_aux1= xb - ( dnorm(-xb)/pnorm(-xb) )       # se Zi <= 0
    EZ_aux2= xb + ( dnorm(-xb)/(1-pnorm(-xb)) )  # se Zi > 0
    EZ = EZ_aux1*(y==0) + EZ_aux2*(y==1)         # vetor de esperancas
    beta = solve( t(x)%*%x ) %*% t(x)%*%EZ       # estimativas na iteracao m
    erro = max(abs(beta.ini - beta)/(abs(beta.ini)+0.001))
    it = it+1
  }
#it;round(t(beta),3); cbind(beta0,beta1,beta2)
resultado[jj,]=t(beta)
}
```

Aplicação 3

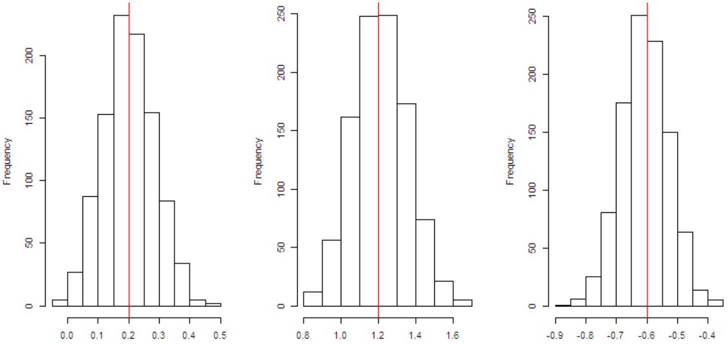
```r
#Exemplo probit
rm(list=ls())
resultado = matrix(0,nrow=1000,ncol=3)
for(jj in 1:1000){
  n=500
  beta0=0.2
  beta1=1.2
  beta2=-0.6
  x1=rbinom(n,1,0.5)
  x2=rnorm(n)
  pred=beta0+beta1*x1+beta2*x2
  prob=pnorm(pred)
  y=rbinom(n,1,prob)

  x=cbind(1,x1,x2)
  beta=array(0,c(3,1))
  e = 0.0001 ;it=0 ; erro = 0.5
  while(erro > e){
    beta.ini = beta
    xb = x%*%beta
    EZ_aux1= xb - ( dnorm(-xb)/pnorm(-xb) )        # se Zi <= 0
    EZ_aux2= xb + ( dnorm(-xb)/(1-pnorm(-xb)) )    # se Zi > 0
    EZ = EZ_aux1*(y==0) + EZ_aux2*(y==1)          # vetor de esperancas
    beta = solve( t(x)%*%x ) %*% t(x)%*%EZ        # estimativas na iteracao m
    erro = max(abs(beta.ini - beta)/(abs(beta.ini)+0.001))
    it = it+1
  }
#it;round(t(beta),3); cbind(beta0,beta1,beta2)
resultado[jj,]=t(beta)
}
```

$$\beta_{\text{est}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'E_{\hat{\theta}^{(m)}}\left[\mathbf{Z}\right]$$

Aplicação 3

```
Console ~/
> mean(resultado[,1]);mean(resultado[,2]);mean(resultado[,2])
[1] 0.200374
[1] 1.21238
[1] 1.21238
>
```



MCEM

Veja que no **passo E**, na iteração $m+1$ temos que calcular

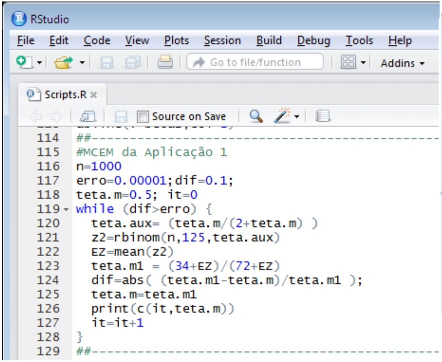
$$Q(\theta \mid \hat{\theta}^{(m)}, \mathbf{x}) = E_{\hat{\theta}^{(m)}}\left[\ln L^c(\theta \mid \mathbf{x}, \mathbf{z})\right]$$

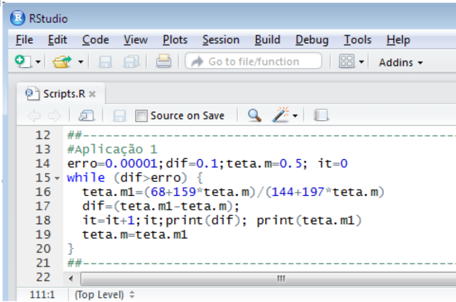
E algumas vezes é difícil obter analiticamente o valor $E\left[Z_i \mid \theta^{(m)}, \mathbf{y}\right]$, mas sabemos gerar, via Monte Carlo, os valores da distribuição condicional de $\left[Z_i \mid \theta^{(m)}, \mathbf{y}\right]$.

Suponha que temos uma amostra $Z_i^{m,1}, Z_i^{m,2}, ..., Z_i^{m,K}$ de $\left[Z_i \mid \theta^{(m)}, \mathbf{y}\right]$. Então aproxime $Q(\theta \mid \hat{\theta}^{(m)}, \mathbf{x})$ por:

$$Q(\theta \mid \hat{\theta}^{(m)}, \mathbf{x}) \approx \frac{1}{K}\sum_{i=1}^{K}\{\ln L^c(\theta \mid \mathbf{x}, \mathbf{z})\}$$

MCEM


RStudio
File Edit Code View Plots Session Build Debug Tools Help
Go to file/function
Addins
Scripts.R
Source on Save
##---
#MCEM da Aplicação 1
n=1000
erro=0.00001;dif=0.1;
teta.m=0.5; it=0
while (dif>erro) {
teta.aux= (teta.m/(2+teta.m))
z2=rbinom(n,125,teta.aux)
EZ=mean(z2)
teta.m1 = (34+EZ)/(72+EZ)
dif=abs((teta.m1-teta.m)/teta.m1);
teta.m=teta.m1
print(c(it,teta.m))
it=it+1
}
##---


RStudio
File Edit Code View Plots Session Build Debug Tools Help
Go to file/function
Addins
Scripts.R
Source on Save
##---
#Aplicação 1
erro=0.00001;dif=0.1;teta.m=0.5; it=0
while (dif>erro) {
teta.m1=(68+159*teta.m)/(144+197*teta.m)
dif=(teta.m1-teta.m);
it=it+1;it;print(dif); print(teta.m1)
teta.m=teta.m1
}
##---
111:1
(Top Level)

MCEM

```r
#Exemplo probit
rm(list=ls())
library(msm)
  n=500
  m=c(rep(10,30),rep(50,20),rep(100,20),rep(500,15),rep(1000,15))
  r=100
  beta0=0.2
  beta1=1.2
  beta2=-0.6
  x1=rbinom(n,1,0.5)
  x2=rnorm(n)
  pred=beta0+beta1*x1+beta2*x2
  prob=pnorm(pred)
  y=rbinom(n,1,prob)
  x=cbind(1,x1,x2)
  beta=array(0,c(3,1))
  e = 0.0001 ;it=0 ; erro = 0.5 ; result=array(0,c(r,3))
#  while(erro > e){
for(k in 1:r){
    beta.ini = beta
    xb = x%*%beta
    Z_aux1= t(sapply(1:n, function(i){rtnorm(m[k], mean=xb[i], sd=1, lower=-Inf, upper=0)}) ) # se Zi <= 0
    Z_aux2= t(sapply(1:n, function(i){rtnorm(m[k], mean=xb[i], sd=1, lower=0, upper= Inf)}) ) # se Zi > 0
    EZ_aux1=rowMeans(Z_aux1)
    EZ_aux2=rowMeans(Z_aux2)
    EZ = EZ_aux1*(y==0) + EZ_aux2*(y==1)        # vetor de esperancas
    beta = solve( t(x)%*%x ) %*% t(x)%*%EZ      # estimativas na iteracao m
    #erro = max(abs(beta.ini - beta)/(abs(beta.ini)+0.001))
    result[k,]=t(beta) ; #print(k)
  }
#  it;round(t(beta),3); cbind(beta0,beta1,beta2)
```

Obrigado!

Referências I

Boyles, R. A., On the Convergence of the EM algorithm. J. Roy. Statist. Soc. Ser. B 45, 47-50, 1983.

Casella, G., Berger, R. L. Inferência estatística - tradução da 2a edição norte americana. Centage Learning, 2010. Página 147-151, 291-294, 329.

Dempster, A. P., Laird, N.M. e Rubin, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, série B, 39, 1-22.

McLachlan, G. J. and Krishnan, T. (1997), The EM Algorithm and Extensions, New York: John Wiley Sons.

Rao, C. R. (1973), Linear Statistical Inference and Its Applications, 2nd Ed., New York: John Wiley Sons.

Wu, C. F. J. (1983) On the Convergence of the EM algorithm. Ann. Statist. 11, 95-113.