Algoritmo EM
Aplicaciones y Extensiones

Burgo Usarralde Casas

Trabajo de fin de grado en Matemáticas
Universidad de Zaragoza

Director del trabajo: José Tomás Alcalá Nalváiz
9 de Febrero de 2017

Prólogo

El trabajo de Fin de Grado que van a leer a continuación lleva el título de "Algoritmo EM. Aplicaciones y Extensiones". Es un algoritmo iterativo en auge en los últimos años para la estimación máximo verosímil en problemas con datos incompletos.

El motivo que me llevo a la realización de este trabajo es el interés promovido por mi tutor Tomás Alcalá Nalváiz por esta rama de las matemáticas al cursar la asignatura de Técnicas de Regresión y al cual quiero dar las gracias por guiarme con este trabajo.

En la Memoria podemos encontrar el estudio de las propiedades más importantes de este Algoritmo y su aplicación a un ejemplo basado en la estimación de la frecuencia de los Alelos en grupos sanguíneos. Además trataremos también algunas de sus extensiones más usadas.

Por último dedicar este trabajo a mi familia y amigos que siempre ha estado ahí para apoyarme, en especial a mis padres y hermano que son los que me han ayudado a llegar hasta donde estoy ahora.

Disfruten de su lectura,
Burgo Usarralde Casas

Summary

Introduction

In real world sometimes we are face to face with truncated observations that we cannot avoid or with data sets with censored observations. One method to solve these situation was discovered by Oscar Rothaus who used it for voice recognition. But the first reference to "EM Algorithm" which we can be found is in the paper of Dempster, Leird and Rubin in 1977. The EM algorithm has become a popular tool in statistical estimation problems involving incomplete data, or in problems which can be posed in a similar form, such as mixture estimation.

EM Algorithm

The Expectation-Maximization Algorithm is an iterative method that aims to find the maximum likelihood estimator of a parameter θ of a parametric probability distribution.
This algorithm consists of repeating two steps:

- **E Step:** Given the estimate from the previous iteration $\theta^{(k)}$, compute the conditional expectation

$$Q(\theta|\theta^{(k)}) = E\{\ln f(X|\theta)|Y,\theta^{(k)}\}$$

- **M Step:** The likelihood function is maximized under the assumption that the missing data are known. Choose $\theta^{(k+1)}$ to be any value of $\theta \in \Theta$ that maximizes $Q(\theta|\theta^{(k)})$.

$$\theta^{(k+1)} = arg\ max\ Q(\theta,\theta^{(k)}) \quad \theta \in \Theta$$

Properties of the EM Algorithm

- Any EM sequence $\theta^{(k)}$ increases the likelihood and $L(\theta^{(k)})$, if bounded above, converges to some $L*$.
- The convergence of L($\theta^{(k)}$) to $L*$ does not automatically imply the convergence of $\theta^{(k)}$ to a point $\theta*$.

The rate of convergence of the EM Algorithm are obtained when we calculate the information matrices for the missing data and the observed data. That rate can be faster than the value obtain for other types of first-order iterative algorithms.

However, one downside of this approach is the provision of standard error or the full covariance matrix in multivariate situations, of the MLE obtained via the EM Algorithm.It does not automatically provide an estimate of the covariance matrix of MLE. Many different procedures have been proposed to obtain the asymptotic covariance matrix for the parameters. In the present memoir, we propose two methods for calculate the covariance matrix:

- Louis's Method: requires first and second derivatives of the complete data log likelihood.
- Baker's Method: computing the observed information matrix in the case of categorical data.

Example: Analysis of blood group in the Spanish Civil War (Allele frequency estimation)

We provide an introductory example on the EM Algorithm applied to a multinomial distribution with cell probabilities depending on two unknown parameters.
We take an example of 4280 individuals data collected in the Spanish Civil War for ABO blood group from donations ([9]).

To finish this memoir, we analalyzed some extensions of EM Algorithm. We were focus in the study of the ECM Algorithm and in two Monte Carlo Versions, the Monte Carlo EM and the EM Bayesian.

Índice general

Prólogo III

Summary V

1. Introducción 1
1.1. Contexto . . . 1
1.2. Objetivo . . . 1
1.2.1. Propiedades del Algoritmo . . . 2

2. El Algoritmo EM 3
2.1. Estimación mediante máxima verosimilitud . . . 3
2.2. Algoritmo EM . . . 6
2.2.1. Introducción . . . 6
2.2.2. Formulación del Algoritmo . . . 7
2.3. Estimación de la frecuencia de los Alelos . . . 8
2.3.1. Algoritmo EM Generalizado (GEM) . . . 11
2.3.2. Propiedades del Algoritmo . . . 11
2.3.3. Elección de los valores iniciales . . . 15
2.4. Estimación de los errores estándar . . . 16
2.4.1. Extracción de la matriz de la información observada en términos de la función de log-verosimilitud de los datos completos . . . 16
2.4.2. Algoritmo EM suplementado (SEM) . . . 17
2.4.3. Métodos para el cálculo de error estándar . . . 18

3. Extensiones Algoritmo EM 21
3.1. Algoritmo ECM . . . 21
3.1.1. Motivación . . . 21
3.1.2. Definición formal . . . 21
3.2. Versiones Monte Carlo del Algoritmo EM . . . 22
3.2.1. Motivación . . . 22
3.2.2. Monte Carlo EM . . . 22
3.2.3. EM Bayesiano . . . 23

Bibliografía 25

A. Ley de Hardy-Weinberg 1

B. Programación algortimos en R 3

Capítulo 1

Introducción

1.1. Contexto

En la vida real no siempre podemos trabajar con unos datos con la calidad que a nosotros nos gustaría. En muchas ocasiones nos encontramos ante una situación con valores perdidos, esta situación es relativamente frecuente, por ejemplo

- en diagnósticos médicos, ya que los historiales contienen un número limitado de pruebas
- fallos durante transmisiones de datos
- pruebas médicas imposibles de realizar
- enmascaramiento de señales debido al ruido (reconocimiento de voz)

Ante esta situación podemos actuar de distintas formas, antiguamente la más común era desechar todas las muestras en las que detectemos la falta de algún dato. A pesar de que es el procedimiento más usado, también es el que más problemas puede causar, ya que en algunos casos nos podemos quedar con un número muy reducido de datos y este procedimiento sólo puede ser aceptable cuando el número de datos faltantes es pequeño y tienen origen aleatorio.
Para resolver este tipo de problemas se creo el Algoritmo de Esperanza-Maximización (EM),el nombre de este viene dado por Dempster, Laird y Rubin (DLR) ([8]) en 1977 quienes también dieron una variedad de ejemplos de su aplicabilidad y establecieron su convergencia y otras propiedades básicas en condiciones bastante generales. Las situaciones en las que se puede aplicar este algoritmo incluyen no solo los casos antes mencionados en los que tenemos datos perdidos, distribuciones truncadas o censuradas, sino también donde la falta de datos no es tan evidente (esto es porque el algoritmo EM explota la reducción de la complejidad de la estimación de máxima verosimilitud dada la de los datos completos). Estas situaciones incluyen modelos estadísticos tales como efectos aleatorios, mixturas, modelos lineales logarítmicos, y de clases latentes y estructuras variables latentes. Se puede aplicar en casi todos los contextos estadísticos o donde se hayan aplicado técnicas estadísticas: tratamiento de imágenes médicas, epidemiología del SIDA, redes neuronales...

1.2. Objetivo

El Algoritmo EM es un método iterativo para realizar una estimación de máxima verosimilitud (ML) de parámetros de problemas en los que existen datos perdidos. Lo que trata de conseguir este algoritmo es asociar a un problema de datos incompletos otro problema con datos completos estableciendo relaciones entre la verosimilitud de estos dos problemas.
En cada iteración del algoritmo tenemos que realizar dos pasos:

1. ***Paso E o Paso de Esperanza***

   En este paso lo que hacemos es "rellenar" los datos que faltan, es decir, creamos el nuevo problema con los datos completos. Además tenemos que calcular una función de verosimilitud para el conjunto de datos completos.

2. ***Paso M o Paso de Maximización***

   Encontrar los parámetros que maximizan la función de log-verosimilitud calculada en el paso E.

3. Estos dos pasos son repetidos iterativamente hasta llegar a la convergencia.

El Algoritmo EM posee algunas ventajas comparado con otros algoritmos iterativos pero tiene ciertas limitaciones ya que en algunas situaciones nos podemos encontrar con una convergencia muy lenta.

1.2.1. Propiedades del Algoritmo

Este algoritmo presenta algunas ventajas y desventajas a la hora de su aplicación.

1. Ventajas

   - Es numéricamente estable con cada iteración en la que aumenta la verosimilitud.
   - En condiciones generales, tiene una convergencia global fiable. Es decir, si empezamos de un punto arbitrario $\theta^{(0)}$ en el espacio de los parámetros, convergeremos casi siempre a un máximo local. Esto no ocurre cuando hacemos una mala elección del punto $\theta^{(0)}$ o por alguna patología local de la función de verosimilitud.
   - Es fácil de implementar ya que se basa en el cálculo de datos completos y también es fácil de programar.
   - Requiere poco espacio de almacenamiento y se puede realizar en un ordenador pequeño.
   - Ya que el problema con los datos completos es más o menos estándar, el paso M lo podemos realizar con frecuencia usando paquetes estadísticos estándar en situaciones donde la estimación de máxima verosimilitud de datos completos no existe de forma cerrada.
   - El trabajo analítico necesario es mucho más simple que con otros métodos.
   - El coste por cada iteración es bajo, lo que compensa el mayor número de iteraciones necesitadas respecto a otros algoritmos.
   - Observando el crecimiento monótono de la verosimilitud a lo largo de las iteraciones, es fácil controlar errores de convergencia y de programación.
   - El Algoritmo EM puede usarse para proporcionar estimaciones de los valores de los datos perdidos.

2. Desventajas

   - No tiene un procedimiento incorporado para proporcionar una estimación de la matriz de covarianza de las estimaciones de los parámetros. (Esta desventaja puede ser eliminada empleando una metodología adecuada asociada con el Algoritmo EM)
   - Puede converger de una manera muy lenta en algunos problemas con demasiados datos incompletos o en algunos problemas aparentemente inofensivos.
   - No garantiza la convergencia al máximo global cuando hay múltiples máximos. En estos casos la estimación obtenida depende del valor inicial.
   - En algunos casos el paso E puede ser analíticamente intratable, aunque en estas situaciones tenemos la posibilidad de aplicarlo mediante el método Monte Carlo.

Capítulo 2

El Algoritmo EM

En este capítulo presentaremos algunos conceptos previos al algoritmo, su desarrollo y sus propiedades más importantes.

2.1. Estimación mediante máxima verosimilitud

Nos podemos encontrar ante muchos métodos de estimación puntuales como el método de los momentos, estimación bayesiana o el método de máxima verosimilitud. Algunos autores se han dedicado a comparar los métodos para elegir el más efectivo de ellos y han demostrado que el último nombrado es el mejor.

Definición. Sea $\mathbf{X}$ una muestra aleatoria de una población X con función de probabilidad (pdf) o función de masa de probabilidad (pmf) $f(x|\theta)$ donde $\theta = (\theta_1, ..., \theta_k)$ es el vector de parámetros desconocidos que queremos estimar.
La ***función de verosimilitud*** de la muestra observada $\mathbf{x}$ se define como

$$L(\theta, \mathbf{x}) = f(\mathbf{x}|\theta) = \prod_{i=1}^{n} f(\mathbf{x}_i|\theta)$$

La función de verosimilitud de $\mathbf{x}$ coincide con la de probabilidad o la de densidad muestral, pero tiene un *punto de vista diferente*. En este caso es la muestra observada la que consideramos dada y $L(\theta, \mathbf{x})$ se ve como una función de θ, en el sentido de que, dada $\mathbf{x}$ hay valores de θ (el parámetro que define la distribución F de X) que hacen que $L(\theta, \mathbf{x})$ sea grande o pequeña: la muestra es más verosímil para unos valores de θ que para otros.

Definición. Sea X un vector aleatorio y T una función medible cuyo dominio incluye el espacio muestral del vector, entonces

$$T = T(X)$$

es una v.a función del vector aleatorio X que llamaremos *estadístico*.

Definición. Para estimar θ, escalar, o la componente θ_i de θ, buscaremos un *estadístico* $T = T(\mathbf{X})$, una v.a función de $\mathbf{X}$, a la que llamaremos *estimador*.

Definición. Un estimador de máxima verosimilitud $\hat{\theta}$ de θ es un valor que maximiza $L(\theta, \mathbf{x})$, es decir,

$$\hat{\theta} = arg_{\theta \in \Theta} max L(\theta)$$

donde Θ representa el espacio paramétrico.

Observación 1: Este criterio para estimar θ es intuitivo.

Observación 2: Al ser la función logaritmo monótona creciente, suele ser más cómodo trabajar con la función de log-verosimilitud.

Definición. La ***función de log-verosimilitud*** de la muestra aleatoria simple $\mathbf{x}$ se define como

$$l(\theta,\mathbf{x}) = logL(\theta,\mathbf{x}) = log\prod_{i=1}^{n} f(\mathbf{x}_i|\theta) = \sum_{i=1}^{n} logf(\mathbf{x}_i|\theta)$$

si la muestra es iid.

Observación 3: El máximo en θ de $L(\theta,\mathbf{x})$ y $l(\theta,\mathbf{x})$ coinciden.

El procedimiento para hallar el EMV de θ, si $l(\theta,\mathbf{x})$ es derivable en θ_j, es encontrar los valores de $(\theta_1,..,\theta_k)$ que son solución del sistema de ecuaciones de verosimilitud.

Definición. Las ecuaciones de verosimilitud vienen dadas por

$$S(\mathbf{x}|\theta) = \frac{\partial l(\theta|\mathbf{x})}{\partial \theta_j} = 0 \quad \text{con} \quad j = 1,..,k$$

en el supuesto de que $\theta = (\theta_1,.....,\theta_k)$ sea un parámetro k dimensional.

Esta es una condición necesaria para un máximo, pero no es suficiente. Además, esa condición localiza los puntos extremos en el interior del espacio paramétrico Θ; la frontera de Θ, debe ser analizada separadamente. El objetivo es encontrar el máximo global.
Si $l(\theta,\mathbf{x})$ no es derivable o si el soporte de la distribución depende del parámetro, hay que recurrir al análisis directo de $L(\theta,\mathbf{x})$.

Para las siguientes definiciones vamos a denotar a x como el vector que representa los datos completos y a y como el vector que representa los datos reales observados.

Matrices para los datos incompletos

Definición. La matriz de información observada se denota por $I(\hat{\theta};y)$ donde

$$I(\theta;y) = -\frac{\partial^2 logL(\theta)}{\partial\theta\partial\theta^T}$$

Definición. La matriz de información esperada se denota por $\mathscr{I}(\theta)$ donde

$$\mathscr{I}(\theta) = E_\theta\{I(\theta;Y)\}$$

Matrices para los datos completos

Definición. La matriz de información observada es

$$I_c(\theta;x) = -\frac{\partial^2 logL_c(\theta)}{\partial\theta\partial\theta^T}$$

Definición. Mientras que su esperanza condicionada a y se denota

$$\mathscr{I}_c(\theta;y) = E_\theta\{I_c(\theta;X)|y\}$$

Definición. La matriz de información esperada se denota por $\mathscr{I}_c(\theta)$ donde

$$\mathscr{I}_c(\theta) = E_\theta\{I_c(\theta;X)\}$$

La matriz de información de datos perdidos se denota por $\mathscr{I}_m(\theta;y)$ y se define como

$$\mathscr{I}_m(\theta;y) = -E_\theta\{\partial^2 k(x|y;\theta)/\partial\theta\partial\theta^T|y\}$$

donde k es la función de probabilidad de X dada y, es decir

$$k(x|y;\theta) = \frac{g_c(x;\theta)}{g(y;\theta)}$$

Principio de información perdida

Tenemos que

$$I(\theta;y) = -\frac{\partial^2 logL(\theta)}{\partial\theta\partial\theta^T}$$

es una matriz negativa de las derivadas parciales de segundo orden de la función de log-verosimilitud (de los datos incompletos) con respecto a los elementos de θ. En condiciones normales, la matriz de información esperada $\mathscr{I}(\theta)$ viene dada por

$$\mathscr{I}(\theta) = E_\theta\{S(Y;\theta)S^T(Y;\theta)\} = E_\theta\{I(\theta;Y)\}$$

Con respecto a la función de log-verosimilitud de los datos completos, tenemos

$$I_c(\theta;x) = -\frac{\partial^2 logL_c(\theta)}{\partial\theta\partial\theta^T}$$

La matriz de información esperada viene dada por

$$\mathscr{I}_c(\theta) = -E_\theta\{I_c(\theta;X)\}$$

Teniendo en cuenta que por el cálculo de la función de log-verosimilitud de la función k se cumple que

$$logL(\theta) = logL_c(\theta) - logk(x|y;\theta) \tag{2.1}$$

Si derivamos ambos lados dos veces respecto a θ, tenemos que

$$I(\theta;y) = I_c(\theta;x) + \frac{\partial^2 logk(x|y;\theta)}{\partial\theta\partial\theta^T} \tag{2.2}$$

Y tomando esperanzas llegamos a que

$$I(\theta;y) = \mathscr{I}_c(\theta;y) - \mathscr{I}_m(\theta;y) \tag{2.3}$$

donde

$$\mathscr{I}_c(\theta;y) = E_\theta\{I_c(\theta;X)|y\}$$

es la esperanza condicionada de la matriz de la información de los datos completos $I_c(\theta;X)$ dada y, y donde

$$\mathscr{I}_m(\theta;y) = -E_\theta\{\partial^2 logk(X|y;\theta)/\partial\theta\partial\theta^T|y\}$$

es la matriz de la información esperada para θ en base a x. condicionada sobre y.

Ejemplo Vamos a calcular el EMV de una población multinomial.

Sabemos que si realizamos n experimentos que dan lugar a $X = \{X_1,...,X_k\}$ resultados con probabilidades $\pi_1,...\pi_k$ respectivamente y además tenemos que $\pi_1 + ... + \pi_k = 1$ nos encontramos ante un

modelo de distribución multinomial.
La función de probabilidad de una variable multinomial es la siguiente:

$$P(X_1 = x_1\ y\ ...\ y\ X_k = x_k) = \frac{n!}{x_1!x_2!...x_k!}\pi_1^{x_1}....\pi_k^{x_k}$$

Las propiedades más importantes de esta distribución son:

- $E(X_j) = \pi_j n$
- $V(X_j) = n\pi_j(1-\pi_j)$
- $Cov(X_iX_j) = -\pi_i\pi_j n \quad (i \neq j)$
- Cualquier subgrupo de los X_j condicionado a su suma, tiene una distribución multinomial, es decir,

$$X_1, X_2, X_3|(X_1+X_2+X_3 = m) \sim multinomial(m, p_1, p_2, p_3) \quad donde\ p_i = \frac{\pi_i}{\pi_1+\pi_2+\pi_3} \qquad (**)$$

La función de verosimilitud en este caso es:

$$L(\pi_1, ..., \pi_k) = \frac{n!}{x_1!x_2!...x_k!}\pi_1^{x_1}....\pi_k^{x_k}$$

y su función de log-verosimilitud será

$$log(L(\pi_1, ..., \pi_k)) = log\left(\frac{n!}{x_1!x_2!...x_k!}\right) + \sum_{j=1}^{k} x_j log\pi_j$$

Los estimadores de máxima verosimilitud de $\pi_1, ..., \pi_k$ los denotamos $\hat{\pi}_1, ..., \hat{\pi}_k$ y cumplen que maximizan la función de log-verosimilitud.
Para calcular estos estimadores hemos tenido que tener en cuenta que $\pi_1 + ... + \pi_k = 1$. Derivando obtenemos que los EMV son de la forma

$$\hat{\pi}_j = \frac{x_j}{n}$$

2.2. Algoritmo EM

2.2.1. Introducción

Definición. Denotamos al vector y=$(y_1, ..., y_n)$ de los datos observados de tamaño n, que será nuestro vector de datos incompleto, correspondiente a una realización de Y, con función de densidad $f(y|\theta)$ donde θ es el vector de parámetros que queremos estimar en un espacio paramétrico Θ. Sea Z= $(Z_1, ..., Z_n)$ el vector que representa los datos no observados (o valores perdidos). Si representamos el vector aleatorio X como (Y, Z), a este vector lo denotaremos como el vector de los datos completos.

Definición. Denotamos por $f(X|\theta)$ la función de densidad de probabilidad que genera los datos. A partir de esto podemos escribir:

$$f(X|\theta) = f(Y, Z|\theta) = f(Z|Y, \theta)f(Y, \theta) \qquad (2.4)$$

Aplicando logaritmos para calcular la verosimilitud a (2.4) obtenemos

$$lnf(X|\theta) = \ln f(Z|Y, \theta) + \ln f(Y, \theta)$$

Lo que nos interesa optimizar son los parámetros respecto a los datos observados,asi pues, tenemos que despejar de la siguiente manera

$$lnf(Y,\theta) = \ln f(X|\theta) - \ln f(Z|Y,\theta)$$

Utilizando las definiciones dadas anteriormente podemos sustituir $f(X|\theta)$ por $L(\theta,X)$ y además $\ln L(\theta,X)$ por $l(\theta,X)$, de esta manera la expresión obtenida será

$$l(\theta|Y) = \ln f(X|\theta) - \ln f(Z|Y,\theta)$$

Tenemos que encontrar un valor de θ que maximice $f(Y|\theta)$, para poder resolver esto tenemos que calcular esperanzas, luego

$$l(\theta|Y) = \int \ln f(Y,Z|\theta) f(Z|Y,\theta) dZ - \int \ln f(Z|Y,\theta) f(Z|Y,\theta) dZ \tag{2.5}$$

A partir de (2.5) definimos las siguientes funciones:

$$Q(\theta,\theta^{(k)}) = E\{\ln f(X|\theta)|Y,\theta^{(k)}\} = \int \ln f(X|\theta) f(Z|Y,\theta^{(k)}) dZ$$

$$H(\theta,\theta^{(k)}) = E\{\ln f(Z|Y,\theta)|Y,\theta^{(k)}\} = \int \ln f(Z|Y,\theta) f(Z|Y,\theta) dZ$$

y tendremos que

$$l(\theta,Y) = Q(\theta,\theta^{(k)}) - H(\theta,\theta^{(k)}) \tag{2.6}$$

Observación 4: Notar que con $\theta^{(k)}$ nos referimos a la estimación en el paso (k) de θ.

2.2.2. Formulación del Algoritmo

La formulación de este algoritmo la podemos desarrollar ahora como una serie de pasos:

1. Elegimos un punto arbitrario $\theta^{(0)}$ en el espacio de los parámetros.
2. **Paso E** Calculamos $Q(\theta,\theta^{(k)})$.
3. **Paso M** Maximizamos la función $Q(\theta,\theta^{(k)})$ respecto a θ eligiendo $\theta^{(k+1)}$.
4. Repetimos los pasos 2 y 3 hasta la convergencia (utilizaremos un criterio de parada).

Criterio de parada

Los pasos E y M se repiten hasta que se cumple el siguiente criterio de parada. Para detenernos podemos considerar la diferencia absoluta

$$|L(\theta^{(k+1)}) - L(\theta^{(k)})|$$

y parar cuando esta sea menor que un parámetro suficientemente pequeño ε en el caso de la convergencia de la secuencia de log-verosimilitud. También podemos utilizar la magnitud del cambio entre los parámetros estimados en cada iteración, es decir, si la diferencia

$$|\theta^{(k+1)} - \theta^{(k)}|$$

es menor que un valor suficientemente pequeño ε, el algoritmo finaliza y por lo tanto el valor de θ será el encontrado en esa iteración.

2.3. Estimación de la frecuencia de los Alelos

Consideramos un problema multinomial con probabilidad de celdas dependientes de dos parámetros.
Los dos sistemas de clasificación más comunes para describir grupos sanguíneos en humanos son los antígenos (el sistema ABO) y el factor Rh.
El sistema ABO muestra el antígeno A, el antígeno B y sin antígenos O y sus cuatro fenotipos observables A, B, AB y O. Estos fenotipos surgen porque heredamos dos alelos, uno de nuestro padre y uno de nuestra madre. Los alelos A y B son dominantes respecto al alelo O.
Es decir, el fenotipo A surge con dos alelos A o un alelo A y otro O (el caso del fenotipo B es igual), en el caso del fenotipo AB será un alelo A y otro B y por último el caso del fenotipo O será cuando los dos alelos son O.
Vamos a tomar un ejemplo concreto ([9]), con una muestra de la población donante de sangre para el ejército en el año 1937 en Barcelona. Es una muestra con n=4280 observaciones, de los cuales n_A=2034 con fenotipo A, n_B=334 con fenotipo B, n_{AB}=136 con fenotipo AB y n_O=1776 con fenotipo O.
Si queremos estimar las frecuencias p_A, p_B y p_O para cada uno de nuestros antígenos necesitamos emplear el Algoritmo EM con los cuatro fenotipos observados (Y) y los subyacentes 6 genotipos. Notar lo siguiente

Categoría	Probabilidad	Frecuencia Observada
O	p_O^2	$n_O = 1776$
A	$p_A^2 + 2p_A p_O$	$n_A = 2034$
B	$p_B^2 + 2p_B p_O$	$n_B = 334$
AB	$2p_A p_B$	$n_{AB} = 136$

Cuadro 2.1: Datos multinomiales observados del ejemplo

Nota: Las probabilidades elegidas vienen dadas por la Ley de Hardy-Weinberg (A) Los datos observados están dados por el vector de frecuencias

$$y = (n_O, n_A, n_B, n_{AB})^T$$

Además tenemos que $p_0 + p_A + p_B$=1 . Por todo lo anterior no podemos aplicar el algoritmo directamente, si no que tenemos que dividir más las categorías.

Categoría	Probabilidad	Frecuencia Observada
O	p_O^2	n_O
AA	p_A^2	n_{AA}
AO	$2p_A p_O$	n_{AO}
BB	p_B^2	n_{BB}
BO	$2p_B p_O$	n_{BO}
AB	$2p_A p_B$	n_{AB}

Cuadro 2.2: Estructura de datos completa del ejemplo

La opción más natural para elegir el vector de los datos completos es

$$X = (n_O, Z^T)^T$$

donde

$$Z = (n_{AA}, n_{AO}, n_{BB}, n_{BO})^T$$

representa el vector de los datos no observados y denotamos como el vector de parámetros que queremos calcular a

$$\theta = (p_A, p_B)^T$$

ya que $p_O = 1 - p_A - p_B$.
Con todos estos datos, la función de log-verosimilitud de los datos completos será:

$$L(\theta, x) = logf(X|\theta) = n_{AA} ln p_A^2 + n_{AO} ln(2p_A p_O) + n_{BB} ln p_B^2 + n_{BO} ln(2p_B p_O) + \\ + n_{AB} ln(2p_A p_B) + n_O ln(p_O^2) + ln\begin{pmatrix} n \\ n_{AA} n_{AO} n_{BB} n_{BO} n_{AB} n_O \end{pmatrix}$$

El **paso E** requiere únicamente el cálculo de la esperanza de (2.6) condicionada a los datos observados de la nuestra n_A, n_B, n_{AB} y n_O y al vector de parámetros obtenido en cada iteración del algoritmo

$$\theta^{(m)} = (p_{(m),A}, p_{(m),B})^T$$

Es obvio que

$$E(n_{AB}|Y, \theta^{(m)}) = n_{AB}$$
$$E(n_O|Y, \theta^{(m)}) = n_O$$

Nos fijamos en el primer elemento de Z que es n_{AA}, por lo explicado en una sección anterior (**) es fácil verificar que este elemento tiene una distribución binomial con tamaño de la muestra n_A y parámetro de probabilidad

$$\frac{p_{(m)A}^2}{p_{(m)A}^2 + 2p_{(m)A}p_{(m)O}}$$

con $\theta^{(m)}$ vector de parámetros desconocidos que usamos en lugar de θ en la iteración (m+1) del algortimo. Luego obtenemos que

$$n_{(m)AA} = E(n_{AA}|Y, \theta^{(m)}) = n_A \frac{p_{(m)A}^2}{p_{(m)A}^2 + 2p_{(m)A}p_{(m)O}} \tag{2.7}$$

$$n_{(m)mAO} = E(n_{AO}|Y, \theta^{(m)}) = n_A \frac{2p_{(m)A}p_{(m)O}}{p_{(m)A}^2 + 2p_{(m)A}p_{(m)O}}$$

$$n_{(m)BB} = E(n_{BB}|Y, \theta^{(m)}) = n_B \frac{p_{(m)B}^2}{p_{(m)B}^2 + 2p_{(m)B}p_{(m)O}}$$

$$n_{(m)BO} = E(n_{BO}|Y, \theta^{(m)}) = n_B \frac{2p_{(m)B}p_{(m)O}}{p_{(m)B}^2 + 2p_{(m)B}p_{(m)O}}$$

El **paso M** del Algoritmo EM maximiza $Q(\theta|\theta^{(m)})$, función que viene de (2.8) remplazando los términos con las fórmulas anteriores. Esta maximización podemos obtenerla introduciendo un multiplicador de Lagrange λ y buscando un punto de la función sin restricciones estacionarias.

$$H(\theta, \lambda) = Q(\theta, \theta^{(m)}) + \lambda(p_A + p_B + p_O - 1)$$

Calculamos las ecuaciones de verosimilitud

$$\frac{\partial H(\theta, \lambda)}{\partial p_A} = \frac{2n_{(m)AA}}{p_A} + \frac{n_{(m)AO}}{p_A} + \frac{n_{AB}}{p_A} + \lambda = 0$$

$$\frac{\partial H(\theta,\lambda)}{\partial p_B} = \frac{2n_{(m)BB}}{p_B} + \frac{n_{(m)BO}}{p_B} + \frac{n_{AB}}{p_B} + \lambda = 0$$

$$\frac{\partial H(\theta,\lambda)}{\partial p_O} = \frac{n_{(m)AO}}{p_O} + \frac{n_{(m)BO}}{p_O} + \frac{2n_O}{p_O} + \lambda = 0$$

$$\frac{\partial H(\theta,\lambda)}{\partial \lambda} = p_A + p_B + p_O - 1 = 0$$

así obtendremos el punto estacionario que queremos calcular.
La solución resultante viene dada por

$$p_{(m+1),A} = \frac{2n_{(m)AA} + n_{(m)AO} + n_{AB}}{2n}$$

$$p_{(m+1),B} = \frac{2n_{(m)BB} + n_{(m)BO} + n_{AB}}{2n}$$

$$p_{(m+1),O} = \frac{n_{(m)AO} + n_{(m)BO} + 2n_O}{2n}$$

En la siguiente tabla muestra el progreso de las iteraciones del Algoritmo EM comenzando por p_{0A}=0,3, p_{0B}=0,2 y p_{0O}=0,5.

Iteración	$p_{(m),A}$	$p_{(m),B}$	$p_{(m),O}$
0	0,3	0,2	0,5
1	0,3083	0,0614	0,6303
2	0,3002	0,0567	0,6431
3	0,2985	0,0566	0,6450
4	0,2982	0,0565	0,6453
5	0,2981	0,0565	0,6454
6	0,2981	0,0565	0,6454

Cuadro 2.3: Resultados en las iteraciones del Algortimo EM en el ejemplo de los Alelos

Luego nuestro vector de parámetros obtenido en este ejemplo será:

$$\hat{\theta} = (0,2981 \quad 0,0565)$$

por lo que $\hat{p}_O = 1 - \hat{p}_A - \hat{p}_B$ será 0,6454.

Vamos a estudiar si las frecuencias genotípicas se desvían significativamente de las que obtendríamos en equilibrio Hardy-Weinberg.

Las valores esperados gracias a la Ley de Hardy-Weinberg son las siguientes:

$$n_A = 2027 \qquad n_B = 326 \qquad n_{AB} = 144 \qquad n_O = 1783$$

Aplicamos un test χ^2 definido como

$$c^2 = \sum_{i=1}^{n} \frac{(observados_i - esperados_i)^2}{esperados_i}$$

El valor crítico de chi-cuadrado para un grado de libertad y p = 0.05 es 3.841. En nuestro caso el valor obtenido al realizar el test es 0,3291005 que es mucho menor. Por tanto, aceptamos la hipótesis nula, y consideramos que nuestra población se encuentra en equilibro de Hardy-Weinberg.

2.3.1. Algoritmo EM Generalizado (GEM)

En algunas ocasiones, la solución del paso M existe en forma cerrada. En estos casos donde no lo hace, puede que no sea posible encontrar el valor de θ que maximiza la función $Q(\theta, \theta^{(k)})$. Para este tipo de situaciones (DLR) definieron un Algoritmo EM Generalizado (Algortimo GEM) en el cual para la elección de $\theta^{(k+1)}$ debemos que tener en cuenta que se cumpla

$$Q(\theta^{(k+1)}, \theta^{(k)}) \geq Q(\theta^{(k)}, \theta^{(k)})$$

Es decir, se elige $\theta^{(k+1)}$ para incrementar el valor de la Q-función $Q(\theta, \theta^{(k)})$ sobre su valor en $\theta = \theta^{(k)}$ en lugar de maximizarla en todo $\theta \in \Theta$. Por una propiedad que veremos en el siguiente apartado la condición anterior en $\theta^{(k+1)}$ es suficiente para asegurar que

$$L(\theta^{(k+1)}) \geq L(\theta^{(k)})$$

Por la tanto la esperanza $L(\theta)$ no disminuye después de una iteración GEM, y así una secuencia de los valores esperados GEM debe converger si está acotado superiormente.

2.3.2. Propiedades del Algoritmo

Nota: Recordemos que una función dos veces derivable $h(w)$ es convexa en un intervalo (a,b) si y solo si $h''(w) \geqslant 0 \quad \forall \in (a,b)$. Si la desigualdad es estricta diremos que es estrictamente convexa.

Proposición: Desigualdad de Jensen. Sea W una variable aleatoria arbitraria en un intervalo (a,b). Si $h(w)$ es convexa en (a,b), entonces E[$h(W)$]$\geqslant h$[E(W)], si estas dos esperanzas existen. Si la función $h(w)$ es estrictamente convexa, entonces, la desigualdad de Jensen se mantiene si y solo si W=E(W) es casi seguro.

Demostración. Para simplificar la demostración asumimos que $h(w)$ es derivable. Si tomamos u=E(W) es claro que u pertenece a (a,b). Para un w en (a,b) tenemos:

$$h(w) = h(u) + h'(u)(w-u) + h''(v)\frac{(w-u)^2}{2} \geqslant h(u) + h'(u)(w-u)$$

para algún v entre u y w (notar que v esta en (a,b)). Sustituimos el valor aleatorio W por el punto w y tomamos esperanzas.

$$E[h(W)] \geqslant h(u) + h'(u)[E(W) - u] = h(u) = h(E(W))$$

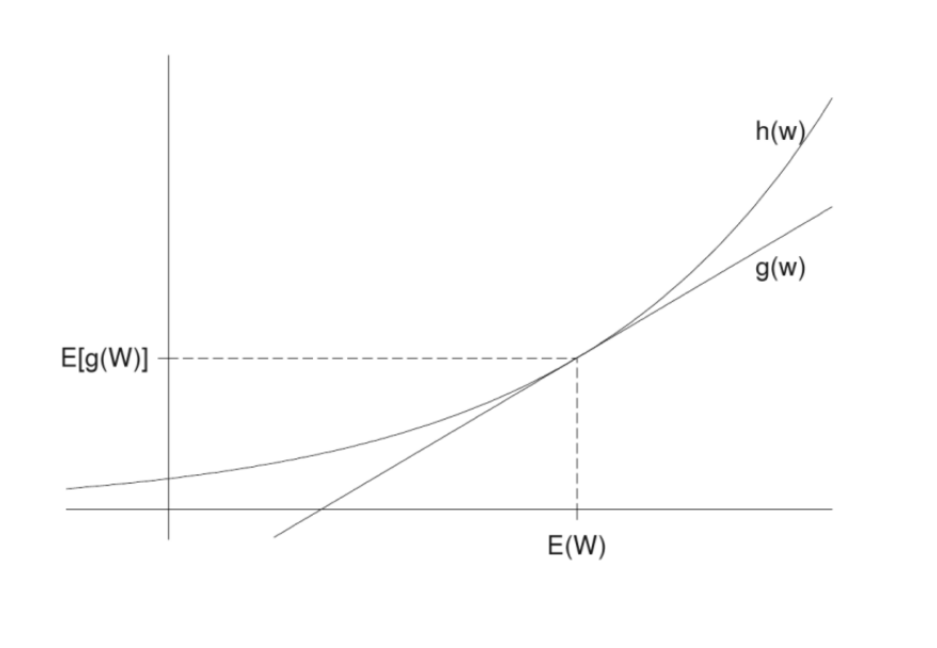
Si $h(w)$ es estrictamente convexa, entonces el término $\frac{(w-u)^2}{2}$ es positivo donde $w \neq u$. Una demostración alternativa que no depende de la existencia de h"(w) es la siguiente: □

Una demostración geométrica alternativa que no depende de la existencia de h"(w) es la siguiente:

Demostración. La función $g(w)$ es tangente a la función convexa $h(w)$ en $w = E(W)$. Por convexidad, $h(w) \geq g(w)$ para todo w y así E[$h(W)$]≥E[$g(W)$]. Sin embargo la linealidad de $g(w)$ implica que E[$g(W)$]=g[E(W)]=h[E(W)]

□

Proposición: Desigualdad de la información. Sean f y g dos densidades respecto a una medida μ. Suponer f>0 y g>0 en casi todo punto respecto a μ. Si denotamos E_f como la esperanza respecto a la densidad $fd\mu$, entonces E_f(ln f) $\geq E_f$(ln g). La igualdad solo se cumple si $f = g$ en casi todo punto respecto a μ.



Demostración. Ya que -ln(w) es una función estrictamente convexa en $(0,\infty)$, la desigualdad de Jensen aplicada a la variable aleatoria $\frac{g}{f}$ implica

$$\begin{aligned} E_f(lnf) - E_f(lng) &= E_f(-ln\frac{g}{f}) \\ &\geq -lnE_f(\frac{g}{f}) \\ &= -ln\int \frac{g}{f} f \partial\mu \\ &= -ln\int g \partial\mu \\ &= 0 \end{aligned}$$

La desigualdad se mantiene solo si $\frac{g}{f} = E_f(\frac{g}{f})$ en casi todo punto respecto μ. Pero $E_f(\frac{g}{f})$=1.

□

Propiedad de convergencia del Algortimo EM (Monotonía)

Dempster, Laird y Rubin demostraron que la función de verosimilitud $L(\theta)$ no decrece después de cada iteración del Algortimo EM, es decir

$$L(\theta^{(k+1)}) \geq L(\theta^{(k)}) \tag{2.8}$$

para k=0,1,2...

Demostración. Recordemos que

$$k(x|y;\theta) = \frac{g_c(x;\theta)}{g(y;\theta)}$$

es la densidad de X condicionada a una Y dada. Entonces la función de log-verosimilitud viene dada por

$$\begin{aligned} logL(\theta) &= log(g(y;\theta)) = \\ &= log(g_c(x;\theta)) - logk(x|y;\theta) = \\ &= logL_c(\theta) - logk(x|y;\theta) \end{aligned} \tag{2.9}$$

Tomando esperanzas en ambos lados de la igualdad (2.9) respecto a la distribución condicional de X dada $Y = y$, usando la aproximación $\theta^{(k)}$ para θ, tenemos entonces

$$\begin{aligned} logL(\theta) &= E_{\theta^{(k)}}\{logL_c(\theta)|y\} - E_{\theta^{(k)}}\{logk(X|y;\theta)|y\} \\ &= Q(\theta;\theta^{(k)}) - H(\theta;\theta^{(k)}) \end{aligned} \tag{2.10}$$

donde

$$H(\theta;\theta^{(k)}) = E_{\theta^{(k)}}\{logk(X|y;\theta)|y\}$$

De (2.10) tenemos

$$logL(\theta^{(k+1)}) - logL(\theta^{(k)}) = \{Q(\theta^{(k+1)};\theta^{(k)}) - Q(\theta^{(k)};\theta^{(k)})\} - \{H(\theta^{(k+1)};\theta^{(k)}) - H(\theta^{(k)};\theta^{(k)})\} \tag{2.11}$$

La primera diferencia en el lado derecho de (2.11) es no negativo ya que $\theta^{(k+1)}$ se elige de modo que

$$Q(\theta^{(k+1)};\theta^{(k)}) \geq Q(\theta;\theta^{(k)}) \qquad \forall\theta \in \Theta \tag{2.12}$$

Por lo tanto, (2.8) se mantiene si la segunda diferencia en el lado derecho de (2.11) es no positiva; es decir, si

$$H(\theta^{(k+1)};\theta^{(k)}) - H(\theta^{(k)};\theta^{(k)}) \leq 0 \tag{2.13}$$

Para cualquier θ

$$\begin{aligned} H(\theta;\theta^{(k)}) - H(\theta^{(k)};\theta^{(k)}) &= E_{\theta^{(k)}}[log\{k(X|y;\theta)/k(X|y;\theta^{(k)}\}|y) \\ &\leq log[E_{\theta^{(k)}}\{k(X|y;\theta)/k(X|y;\theta^{(k)}\}|y)] \\ &= log\int_{X(y)} k(x|y;\theta)dx \\ &= 0 \end{aligned} \tag{2.14}$$

donde la desigualdad en (2.14) es consecuencia de la Desigualdad de Jensen y la concavidad de la función logaritmo.
Esto establece (2.13) y por lo tanto la desigualdad (2.8), demostrando que la función de verosimilitud $L(\theta)$ no disminuye después de una iteración del algoritmo EM. La verosimilitud aumentará si la desigualdad (2.12) es estricta. Así, para una secuencia limitada de valores de log-verosimilitud $\{L(\theta^{(k)})\}$, $L(\theta^{(k)})$ converge monótonamente a algún L^*.
Una consecuencia de (2.8) es la auto-consistencia del algoritmo EM. Si la estimación mediante máxima verosimilitud de θ maximiza globalmente $L(\theta)$, debe satisfacer

$$Q(\hat{\theta};\hat{\theta}) \geq Q(\theta;\hat{\theta}) \tag{2.15}$$

para todo θ. Por otro lado

$$Q(\hat{\theta};\hat{\theta}) < Q(\theta_0;\hat{\theta})$$

para algún θ_0, implica que

$$L(\theta_0) > L(\hat{\theta})$$

lo que contradice el hecho de que $\hat{\theta}$ sea el máximo global de $L(\theta)$.
Se verá que la forma diferencial de (2.15) es que $\hat{\theta}$ es una raíz de la ecuación

$$\left[\frac{\partial Q(\theta;\hat{\theta})}{\partial\theta}\right]_{\theta=\hat{\theta}} = 0$$

□

Propiedad de convergencia del Algortimo EM Generalizado (Monotonía)

En este algoritmo, $\theta^{(k+1)}$ no se elige como el máximo global $Q(\theta;\theta^{(k)})$ con respecto a θ aunque satisface

$$Q(\theta^{(k+1)};\theta^{(k)}) \geq Q(\theta^{(k)};\theta^{(k)})$$

Hemos visto en el apartado anterior que esta condición es suficiente para asegurar que (2.8) se mantiene para una secuencia iterativa $\{\theta^{(k)}\}$. Por lo tanto la función de verosimilitud no disminuye después de una iteración GEM.

Score statistics (Valor puntuación)

Denotamos

$$S(y;\theta) = \frac{\partial logL(\theta)}{\partial \theta}$$

como el vector gradiente de la función de log-verosimilitud $L(\theta)$; esto es el resultado estadístico basado en los datos observados y. El vector gradiente de la función de log-verosimilitud de los datos completos viene dada por

$$S_c(x;\theta) = \frac{\partial logL_c(\theta)}{\partial \theta}$$

El resultado estadístico de los datos incompletos $S(y;\theta)$ lo podemos expresar como la esperanza condicionada a y del resultado estadístico de los datos completos, es decir

$$S(y;\theta) = E_\theta\{S_c(X;\theta)|y\}$$

Para ver esto, observamos que

$$\begin{aligned} S(y;\theta) &= \frac{\partial logL(\theta)}{\partial \theta} \\ &= \frac{\partial logg(y;\theta)}{\partial \theta} \\ &= \frac{g'(y;\theta)}{g(y;\theta)} \\ &= \frac{\int_{X(y)} g'_c(x;\theta)dx}{g(y;\theta)} \end{aligned} \tag{2.16}$$

donde la $'$ denota diferenciación respecto a θ. Multiplicando y dividiendo por $g_c(x;\theta)$ en (2.16), tenemos que

$$\begin{aligned} S(y;\theta) &= \int_{X(y)} \{\partial logg_c(x;\theta)/\partial\theta\}\{g_c(x;\theta)/g(y;\theta)\}dx \\ &= \int_{X(y)} \{\partial logL_c(\theta)\}k(x|y;\theta)dx \\ &= E_\theta\{\partial logL_c(\theta)/\partial\theta|y\} \\ &= E_\theta\{S_c(X;\theta)|y\} \end{aligned} \tag{2.17}$$

Tasa de convergencia del Algortimo EM

Vamos a interpretar el Algoritmo EM como una sucesión de iteraciones, que alternan un paso E con un paso M, lo podemos ver como una aplicación

$$M : \theta \longrightarrow \theta$$
$$\theta^{(k+1)} = M(\theta^{(k)}) \qquad k = 0, 1, 2, ..$$

Si $\theta^{(k)}$ converge a algún punto θ^* y $M(\theta)$, entonces θ^* es un punto fijo del algoritmo que cumple

$$\theta^* = M(\theta^*)$$

Gracias a la serie de Taylor de $\theta^{(k+1)} = M(\theta^{(k)})$ sobre el punto $\theta^{(k)} = \theta^*$, tenemos en un entorno de θ^* que

$$\theta^{(k+1)} - \theta^* \approx J(\theta^*)(\theta^{(k)} - \theta^*)$$

donde $J(\theta)$ es la matriz Jacobiana dxd de $M(\theta) = (M_1(\theta), M_2(\theta), ..., M_d(\theta))^T$, donde el elemento (i, j) de $J_{ij}(\theta)$ es igual a

$$J_{ij}(\theta) = \frac{\partial M_i(\theta)}{\partial \Theta_j} \tag{2.18}$$

donde $\Theta_j = (\theta)_j$ Esto, en un entorno de θ^*, el Algoritmo EM es esencialmente una iteración lineal con matriz tasa $J(\theta^*)$, ya que $J(\theta^*)$ es típicamente distinto de cero. Por esta razón, $J(\theta^*)$ se nombra a menudo como la matriz de tasa de convergencia, o simplemente, la tasa de convergencia. Para el vector Θ, una medida de la tasa real de convergencia observada es la tasa global de convergencia, que se define como

$$r = \lim_{k \to \infty} \frac{||\theta^{(k+1)} - \theta^*||}{||\theta^{(k)} - \theta^*||}$$

donde $||\cdot||$ es cualquier norma d-dimensional del espacio euclideo $\mathbb{R}^d$. Sabemos que, en ciertas condiciones de regularidad,

$$r = \lambda_{max} \equiv \text{el mayor valor propio de} \quad J(\theta^*)$$

Debemos tener en cuenta que un gran valor de r implica una convergencia lenta. Para ser coherente con la noción común de que cuanto mayor sea el valor de la medida, más rápida será la velocidad a la que el algoritmo converge, Meng ([22]) definió s = 1-r como la velocidad global de la convergencia. Por lo tanto s es el valor propio más pequeño de

$$S = I_d - J(\theta^*)$$

que puede ser llamado la (matriz) velocidad de convergencia donde S se refiere a menudo como la matriz de iteración en la literatura de optimización.
Podemos definir la matriz $J(\theta^*)$ en términos de la tasa de convergencia tenemos

$$J(\theta^*) = \mathscr{I}_c^{-1}(\theta^*; y)\mathscr{I}_m(\theta^*; y)\} \tag{2.19}$$

2.3.3. Elección de los valores iniciales

La elección de los valores iniciales tiene una gran importancia ya que puede influir en la velocidad de convergencia del algoritmo y en su capacidad para encontrar el máximo global. Distintos autores han trabajado sobre este tema, como Laird ([8])que propuso una rejilla de búsqueda para establecer los valores iniciales, más adelante Leroux ([14]) sugirió el uso de información complementaria con el fin de formar grupos cuyos recursos fueron utilizados como valores iniciales. Además McLachlan ([21]) propuso el uso de análisis de componentes principales para la selección de valores iniciales para el caso de mezclas multivariantes.
Aunque muchos autores han tratado sobre este tema nunca se ha llegado a un consenso sobre la estrategia de inicialización, por lo tanto, se recomienda probar varias y elegir la que nos de mejores resultados.

2.4. Estimación de los errores estándar

2.4.1. Extracción de la matriz de la información observada en términos de la función de log-verosimilitud de los datos completos

Louis (1982) probó que la matriz de los valores perdidos $\mathscr{I}_m(\theta;y)$ puede ser expresada de la siguiente forma:

$$\begin{aligned}\mathscr{I}_m(\theta;y) &= cov_\theta\{S_c(X;\theta)|y\} \\ &= E_\theta\{S_c(X;\theta)S_c^T(X;\theta)|y\} - S(y;\theta)S^T(y;\theta)\end{aligned} \tag{2.20}$$

ya que

$$S(y;\theta) = E_\theta\{S_c(X;\theta)|y)\} \tag{2.21}$$

Sustituyendo (2.20) y (2.21) en (2.3), tenemos que

$$\begin{aligned}I(\theta;y) = \mathscr{I}_c(\theta;y) - \mathscr{I}_m(\theta;y) &= \mathscr{I}_c(\theta;y) - cov_\theta\{S_c(X;\theta)|y\} \\ &= \mathscr{I}_c(\theta;y) - E_\theta\{S_c(X;\theta)S_c^T(X;\theta)|y\} + S(y;\theta)S^T(y;\theta)\end{aligned} \tag{2.22}$$

Louis establece (2.22) trabajando con $I(\theta;y)$ de la siguiente manera.
Por (2.16) podemos escribir

$$I(\theta;y) = -\frac{\partial S(y;\theta)}{\partial\theta} \tag{2.23}$$

$$= -\partial[\{\int_{X(y)} g_c'(x;\theta)dx\}/g(y;\theta)]\partial\theta \tag{2.24}$$

$$= -\{\int_{X(y)} \partial^2 g_c(x;\theta)/\partial\theta\partial\theta^T dx\}/\{g(y;\theta)\} + \{\int_{X(y)} g_c'(x;\theta)dx\}\{\int_{X(y)} g_c'(x;\theta)dx\}^T/\{g(y;\theta)\}^2$$

Vamos a derivar (2.17) y llegamos a que

$$I(\theta;y) = -\{\int_{X(y)} \partial^2 g_c(x;\theta)/\partial\theta\partial\theta^T dx\}/g(y;\theta) + S(y;\theta)S^T(y;\theta) \tag{2.25}$$

y usando (2.16) para el último término de (2.23).
El primer término en el lado derecho de (2.25) lo podemos escribir de la siguiente manera

$$-\{\int_{X(y)} \partial^2 g_c(x;\theta)/\partial\theta\partial\theta^T dx\}/g(y;\theta) = \tag{2.26}$$

$$\begin{aligned}&= -\int_{X(y)} [\{\partial^2 log g_c(x;\theta)/\partial\theta\partial\theta^T\}\{g_c(x;\theta)/g(y;\theta)\}]dx \\ &- \int_{X(y)} \{g_c'(x;\theta)/g_c(x;\theta)\}\{g_c'(x;\theta)/g_c(x;\theta)\}^T\{g_c(x;\theta)/g(y;\theta)\}dx \\ &= -\int_{X(y)} I_c(\theta;x)k(x|y;\theta)dx - \int_{X(y)} S_c(x;\theta)S_c^T(x;\theta)k(x|y;\theta)dx \\ &= E_\theta\{I_c(\theta;X)|y\} - E_\theta\{S_c(X;\theta)S_c^T(X;\theta)|y\} \\ &= \mathscr{I}_c(\theta;y) - E_\theta\{S_c(X;\theta)S_c^T(X;\theta)|y\}\end{aligned}$$

Sustituyendo (2.26) en (2.25) obtenemos la expresión (2.22) para $I(\hat{\theta})$.
De (2.23), vemos que la matriz de la información observada $I(\hat{\theta})$ puede ser calculada como

$$\begin{aligned}I(\hat{\theta};y) = \mathscr{I}_c(\hat{\theta};y) - \mathscr{I}_m(\hat{\theta};y) &= \mathscr{I}_c(\hat{\theta};y) - [cov_\theta\{S_c(X;\theta)|y\}]_{\theta=\hat{\theta}} \\ &= \mathscr{I}_c(\hat{\theta};y) - [E_\theta\{S_c(X;\theta)S_c^T(X;\theta)\}|y]_{\theta=\hat{\theta}'}\end{aligned} \tag{2.27}$$

si el último término de (2.22) es cero entonces se cumple

$$S(y;\theta) = 0$$

Nota:el símbolo $'$ denota la derivada respecto θ.
Nota:En toda esta sección hemos asumido que tenemos condiciones de regularidad para el intercambio de las operaciones de diferenciación e integración en caso de ser necesario.

2.4.2. Algoritmo EM suplementado (SEM)

Meng y Rubin ([22]) definieron un procedimiento que obtiene estimaciones numéricas estables de la matriz de covarianzas asintótica obtenida mediante el Algoritmo EM. Usando solamente el código para obtener la matriz de covarianzas de los datos completos, el propio código del algoritmo EM y el código de operaciones para las matrices. En particular, ni verosimilitudes, ni derivadas parciales de los logaritmos de las verosimilitudes tienen que ser evaluados.
Básicamente consiste en usar el hecho de que la velocidad de convergencia se rige por la fracción de los datos desconocidos para encontrar un aumento de la variabilidad debido a la información que añadimos en la matriz de covarianza de los datos completos.
Meng y Rubin ([22]) se refieren al algoritmo EM con su modificación para la provisión de la matriz de covarianza asintótica como el Algoritmo EM suplementado (SEM).
Respecto a esto, Smith (1977) señala la posibilidad de obtener la varianza asintótica en los casos en los que los parámetros sean individuales mediante el uso de la tasa de convergencia r del Algoritmo EM. Él da la siguiente expresión

$$v = \frac{v_c}{1-r}$$

donde v y v_c denotan la varianza asintótica del estimador de máxima verosimilitud basado en la observación de datos incompletos y datos completos, respectivamente.
Esta expresión la podemos escribir de la siguiente manera

$$v = v_c + \Delta v \tag{2.28}$$

donde

$$\Delta v = \frac{r}{1-r} v_c \tag{2.29}$$

es el crecimiento de la varianza debido a no observar los datos Z. Meng y Rubin ([22]) extendieron este resultado para el caso multivariante con $d > 1$ parámetros. Denotamos V como la matriz de covarianzas asintótica de la estimación de máxima verosimilitud $\hat{\theta}$. Análogamente a (2.28), probaron que

$$I^{-1}(\hat{\theta};y) = \mathscr{I}_c^{-1}(\hat{\theta};y) + \Delta V \tag{2.30}$$

donde

$$\Delta V = \{I_d - J(\hat{\theta})\}^{-1} J(\hat{\theta}) \mathscr{I}_c^{-1}(\hat{\theta};y) \tag{2.31}$$

y $J(\theta)$ (definida en (2.18)). Por lo tanto los elementos diagonales de ΔV dan los aumentos en las varianzas asintóticas de las componentes de $\hat{\theta}$ debido a los datos que faltan. Teniendo en cuenta (2.3) obtenemos

$$I(\theta;y) = \mathscr{I}_c(\theta;y)\{I_d - \mathscr{I}_c^{-1}(\theta;y)\mathscr{I}_m(\theta;y)\}$$

De (2.19) llegamos a que

$$J(\hat{\theta}) = \mathscr{I}_c^{-1}(\hat{\theta}^*;y)\mathscr{I}_m(\hat{\theta}^*;y)$$

para una secuencia EM que satisface que

$$\frac{\partial Q(\theta : \theta^{(k)})}{\partial \theta} = 0$$

De ahí que la matriz de información observada $I(\hat{\theta};y)$ puede expresarse como

$$I(\hat{\theta};y) = \mathscr{I}_c(\hat{\theta};y)\{I_d - J(\hat{\theta})\}$$

que calculando su inversa, nos lleva a

$$\begin{aligned} I(\hat{\theta};y) = \{I_d - J(\hat{\theta})\}^{-1}\mathscr{I}_c(\hat{\theta};y)^{-1} &= [I_d + \{I_d - J(\hat{\theta})\}^{-1}J(\hat{\theta})]\mathscr{I}_c(\hat{\theta};y)^{-1} \\ &= \mathscr{I}_c(\hat{\theta};y)^{-1} + \{I_d - J(\hat{\theta})\}^{-1}J(\hat{\theta})\mathscr{I}_c(\hat{\theta};y)^{-1} \end{aligned}$$

estableciendo así (2.31).

2.4.3. Métodos para el cálculo de error estándar

Método Louis

El método de Louis requiere solo la primera y segunda derivada de la función de log-verosimilitud de los datos completos, los cuales, en general, son más fáciles de resolver que los derivados correspondientes de la función de log-verosimilitud de los datos incompletos. De la fórmula de Louis (2.27) para la matriz de los datos observados $I(\hat{\theta};y)$ tenemos

$$I(\hat{\theta};y) = \mathscr{I}_c(\hat{\theta};y) - [cov_\theta\{S_c(X;\theta)|y\}]_{\theta=\hat{\theta}} \tag{2.32}$$

Podemos expresar también la matriz

$$\begin{aligned} \mathscr{I}_c(\theta;y) &= E_\theta\{-\partial^2 logL_c(\theta)/\partial\theta\partial\theta^T|y\} \\ &= -[\partial^2 Q(\theta;\theta_0)/\partial\theta\partial\theta^T]_{\theta_0=\theta} \end{aligned}$$

Esto implica que podemos intercambiar el orden de diferenciación e integración. Sea supuesto en el resto de esta sección. Podemos utilizar (2.32) para calcular la matriz de información observada, que en la inversión, da una estimación de la matriz de covarianza de la estimación de máxima verosimilitud. La fórmula se simplifica en el caso del modelo multinomial, pero, lamentablemente, sólo en los casos en que las segundas derivadas de las frecuencias esperadas de las casillas sean todas cero. Vamos a aplicar este método a nuestro ejemplo ([9]).

Ejemplo

Vamos a utilizar la función de log-verosimilitud de los datos completos pero, esta vez, utilizaremos los siguientes datos:

$$n_A^+ = n_{AA} + \frac{1}{2}n_{AO} + \frac{1}{2}n_{AB} = n_A - \frac{1}{2}n_{AO} + \frac{1}{2}n_{AB}$$

$$n_B^+ = n_{BB} + \frac{1}{2}n_{BO} + \frac{1}{2}n_{AB} = n_B - \frac{1}{2}n_{BO} + \frac{1}{2}n_{AB}$$

$$n_O^+ = n_O + \frac{1}{2}n_{AO} + \frac{1}{2}n_{BO}$$

Denotamos como n_A^*, n_B^* y n_o^* como las esperanzas condicionadas, dados los datos observados usando $\theta = \hat{\theta}$. Tenemos que:

$$Q(\theta;\hat{\theta}) = 2\{n_A^* log(p_A) + n_B^* log(p_B) + n_O^* log(p_O)\}$$

de lo cual obtenemos

$$\frac{\partial Q(\theta;\hat{\theta})}{\partial\theta} = 2\begin{pmatrix} \frac{n_A^*}{p_A} - \frac{n_O^*}{p_O} \\ \frac{n_B^*}{p_B} - \frac{n_O^*}{p_O} \end{pmatrix}$$

y

$$-\left[\frac{\partial^2 Q(\theta;\hat{\theta})}{\partial\theta\partial\theta^T}\right]_{\theta=\hat{\theta}} = 2\begin{pmatrix} \frac{n_A^*}{\hat{p_A}^2}+\frac{n_O^*}{\hat{p_O}^2} & \frac{n_O^*}{\hat{p_O}^2} \\ \frac{n_B^*}{\hat{p_B}^2}+\frac{n_O^*}{\hat{p_O}^2} & \frac{n_O^*}{\hat{p_O}^2} \end{pmatrix} \tag{2.33}$$

Si tenemos en cuenta que n_{AO} y n_{BO} tienen distribuciones binomiales independientes, vemos que $cov_\theta\{S_c(X;\theta)|y\}$ viene dado por

$$2\begin{pmatrix} n_A\frac{(p_A+p_O)^2}{p_Ap_O(p_A+2p_O)^2}+n_B\frac{p_B}{p_O(p_B+2p_O)^2} & n_A\frac{(p_A+p_O)}{p_O(p_A+2p_O)^2}+n_B\frac{p_B+p_O}{p_O(p_B+2p_O)^2} \\ n_A\frac{(p_A+p_O)}{p_O(p_A+2p_O)^2}+n_B\frac{p_B+p_O}{p_O(p_B+2p_O)^2} & n_B\frac{(p_B+p_O)^2}{p_Bp_O(p_B+2p_O)^2}+n_A\frac{p_A}{p_O(p_A+2p_O)^2} \end{pmatrix}$$

Esta última expresión evaluada en $\theta=\hat{\theta}$ nos da la matriz de los datos observados $I(\hat{\theta};y)$. Al invertir esta matriz obtenemos la matriz de covarianzas de la estimación mediante máxima verosimilitud:

$$\begin{pmatrix} 2,957\times e^{-5} & -1,956\times e^{-6} \\ -1,956\times e^{-6} & 6,416\times e^{-6} \end{pmatrix}$$

Método Baker

Para poder explicar este método tenemos que definir antes algunos términos.

Definición. Sean A y B matrices y v un vector. Vamos a definir las siguientes operaciones que claramente serán validas cuando los órdenes de las matrices sean los adecuados.
A.B: multiplicación elemento por elemento de las matrices A y B.
A/B: división elementos por elemento de las matrices A y B.
1: vector columna con todo unos.
diag(v):matriz diagonal con elementos v en la diagonal.
bl(A,B): una matriz con A y B formando un bloque diagonal con ceros en los demás lugares.
hc(A,B): concatenación horizontal de A y B.
vc(A,B):concatenación vertical de A y B.
Estas tres últimas operaciones pueden tener más de dos elementos, un ejemplo de como se denota en estos casos sería $bl_{|k}matriz(k)$. Los datos siguen un modelo de Poisson, multinomial o producto multinomial con parámetro un d-vector θ. Sea $y=(y_1,...y_M)^T$ el elemento de los datos incompletos y $x=(x_1,...x_N)^T$ el elemento de los datos completos con N≥M.
Las expresiones $U(\theta)$ y $V(\theta)$ denotan E(Y) y E(X) respectivamente, donde U y V están relacionadas por $U(\theta)=CV(\theta)$.
Escribimos $V(\theta)$ como

$$V(\theta)\propto exp\left[\sum_{k=1}^{K}G^kT^k\right]$$

donde $T^k=t^k(X^k\theta^k;Z^k)$ es un m-vector, θ^k es un d^k-vector de un subconjunto de parámetros con $\sum_{k=1}^K d^k=d$, X^k es una $m\times d^k$ matriz, Z^k es un m-vector y G^k es una $N\times m$ matriz.La función t^k opera en cada elemento de sus argumentos vectoriales.

Denotamos T' y T'' al vector de la primera y segunda derivada, respectivamente de las componentes de T^k con respecto a los correspondientes elementos de $(X_q^{(k)}\theta^k)$.

Sean

$$R = 1 - C^T(Y/U)$$
$$S = hc_{|k}G^k diagT'^{(k)}X^k$$
$$I_1 = bl_{|k}X^{(k)^T} diagT''^{(k)}.(G^{(k)^T}(R.V))X^k$$
$$I_2 = S^T diag(R.V)S$$
$$I_3 = S^T diag(V)C^T diag(Y/(U.U))Cdiag(V)S$$

Entonces, la matriz de la información observada I, se obtiene

$$I(\hat{\theta};y) = I_1 + I_2 + I_3$$

La matriz de la información esperada se obtiene

$$\mathscr{I}(\hat{\theta}) = S^T diag(V)C^T diag(1/Y)Cdiag(V)S$$

Aplicamos este método a nuestro ejemplo ([9]):
Ejemplo

En nuestro ejemplo tenemos que N=6, M=4, n=4280, θ= $\begin{pmatrix} p_A \\ p_B \end{pmatrix}$ Notar que $p_O = 1 - p_A - p_B$.

$$U(\theta) = N\begin{pmatrix} p_O^2 \\ p_A^2 + 2p_A p_O \\ p_B^2 + 2p_B p_O \\ 2p_A p_B \end{pmatrix}, V(\theta) = N\begin{pmatrix} p_O \\ p_A^2 \\ 2p_A p_O \\ p_B^2 \\ 2p_B p_O \\ 2p_A p_B \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Podemos ver entonces que

$$V = nexp(GT), X = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ -1 & -1 \end{pmatrix}, Z = \begin{pmatrix} e \\ 0 \\ 0 \\ 1 \end{pmatrix}, T = log(Z + X\theta), C = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 0 & 2 & 0 & 0 \\ log(2) & 1 & 0 & 1 \\ 0 & 0 & 2 & 0 \\ log(2) & 0 & 1 & 1 \\ log(2) & 1 & 1 & 0 \end{pmatrix}$$

La matriz de varianzas y covarianzas para este método es:

$$\begin{pmatrix} 2,958 \times e^{-5} & -1,956 \times e^{-6} \\ -1,956 \times e^{-6} & 6,421 \times e^{-6} \end{pmatrix}$$

Notar que las dos matrices obtenidas con los métodos son casi iguales.

Capítulo 3

Extensiones Algoritmo EM

En este capítulo, vamos a ver algunas extensiones del Algortimo EM. Nos vamos a centrar en el Algoritmo ECM y en dosde las versiones de Monte Carlo del Algoritmo EM, la que lleva el propio nombre de estos métodos (Monte Carlo EM) y la EM Bayesiana.

3.1. Algoritmo ECM

3.1.1. Motivación

A pesar de que el Algoritmo EM es muy popular ya que el paso M sólo incluye una estimación máximo verosímil de los datos completos, lo cual hace que el problema tenga un cálculo simple, si esta estimación es un poco complicada, esto hace que este método sea poco atractivo computacionalmente.
Para resolver este problema Meng y Rubin ([22]) crearon el Algortimo ECM (esperanza-condicional maximización), el cual sustituye el paso M del Algoritmo EM con pasos CM que son computacionalmente más simples. Lo que hacemos en cada uno de estos pasos es maximizar la esperanza condicional de la función de log-verosimilitud completa que hemos calculado en el paso E.
A consecuencia de esto, normalmente, converge más lentamente que el Algortimo EM, en términos del número de iteraciones, pero es más rápido en computar.

3.1.2. Definición formal

En este algoritmo lo que vamos a hacer es sustituir el paso M del Algoritmo EM por $S > 1$ pasos. Denotamos $\theta^{(k+s/S)}$ como el valor de θ en cada paso CM que realicemos en la iteración (k+1), donde $\theta^{(k+s/S)}$ lo elegimos para maximizar

$$Q(\theta;\theta^{(k)})$$

sujeto a la restricción

$$g_s(\theta) = g_s(\theta^{(k+(s-1)/S)})$$

En este caso $C = (g_s(\theta), s = 1,2...S)$ es un conjunto de funciones (vectores) S preseleccionados. Así $\theta^{(k+s/S)}$ cumple

$$Q(\theta^{(k+s/S)};\theta^{(k)}) \geq Q(\theta;\theta^{(k)}) \text{ para todo } \theta \in \Theta_s(\theta^{(k+(s-1)/S)}) \tag{3.1}$$

donde

$$\Omega_s(\theta^{(k+(s-1)/S)}) \equiv \{\theta \in \Omega : g_s(\theta) = g_s(\theta^{(k+(s-1)/S)})\}$$

El valor de θ final del paso CM, $\theta^{(k+S/S)} = \theta^{(k+1)}$, lo introducimos en la iteración (k+2).
De (3.1) tenemos que

$$\begin{aligned} Q(\theta^{(k+1)};\theta^{(k)}) &\geq Q(\theta^{(k+((S-1)/S))};\theta^{(k)}) \\ &\geq Q(\theta^{(k+(S-2)/S)};\theta^{(k)}) \\ &\cdot \\ &\cdot \\ &\cdot \\ &\geq Q(\theta^{(k)};\theta^{(k)}) \end{aligned} \tag{3.2}$$

Esto demuestra que este algoritmo es un Algoritmo EM Generalizado y que también cumple las condiciones deseadas de convergencia antes explicadas, ya que con (3.2) hemos probado que era suficiente para que se cumpla

$$L(\theta^{(k+1)}) \geq L(\theta^{(k)})$$

Además ya que $g_s(\theta), s = 1,...S$ es diferenciable y el correspondiente vector gradiente $\nabla g_s(\theta)$ es de rango completo en $\theta^{(k)}, \forall\, k$, casi todas las propiedades del Algoritmo EM se mantienen. Pero tenemos que añadir una condición extra a la que llamamos "rellenar-espacio"

$$\bigcap_{s=1}^{S} G_s(\theta^{(k)}) = \{0\} \quad \forall k \tag{3.3}$$

donde $G_s(\theta)$ es el espacio columna de $\nabla g_s(\theta)$, es decir

$$G_s(\theta) = \{\nabla g_s(\theta)\eta : \eta \in \mathbb{R}^{d_s}\}$$

donde d_s es la dimensión del vector función $g_s(\theta)$. Tomando el complementario de ambos lados de la ecuación (3.3), esta condición equivale a decir que en cualquier θ^k, la envolvente convexa de todas las direcciones posibles determinada por los espacios de restricción $\Theta_s(\theta^{(k+(s-1)/S)})$ s=1,..S es todo el espacio euclídeo $\mathbb{R}^d$ y así, la maximización resultante es sobre todo el espacio de parámetros Θ y no un subespacio de él.
Nota: El Algoritmo EM es un caso particular del algoritmo ECM en el que S=1 y $g_1(\theta) \equiv constante$

3.2. Versiones Monte Carlo del Algoritmo EM

3.2.1. Motivación

A lo largo de las últimas décadas se han desarrollado muchos métodos basados en técnicas de simulación iterativas útiles especialmente en el cálculo de soluciones bayesianas para el tipo de problemas de datos incompletos. La mayoría de estos métodos estiman la densidad a posteriori y no encontrar la densidad del vector de parámetros θ. Como se destaca en Gelman y Rubin [13] , es aconsejable encontrar estimaciones MLE o MAP de θ antes de utilizar la simulación iterativa, debido a las dificultades en la evaluación de la convergencia de los métodos de simulación iterativa, particularmente cuando las regiones de alta densidad no son conocidas a priori.

3.2.2. Monte Carlo EM

Cuando utilizamos el Algoritmo EM nos encontramos ante un problema de implementación ya que la computación del paso E es bastante costosa. A causa de este problema se planteó la modificación de este paso utilizando una simulación por aproximaciones de Monte Carlo simulando los datos perdidos Z de una distribución condicional $k(z|y;\theta^{(k)})$ en la iteración $(k+1)$.

Lo que hacemos es maximizar la aproximación de la esperanza condicionada de la función de log-verosimilitud:

$$\hat{Q}(\theta;\theta^{(k)}) = \frac{1}{m}\sum_{j=1}^{m} logL_c(\theta;y,z_j)$$

Si aplicamos el limite cuando $x \to \infty$ obtendríamos lo mismo que en el Algoritmo EM.

Ejemplo

Vamos a aplicar a nuestro ejemplo [9] el Algoritmo Monte Carlo EM.
Lo que vamos a hacer es cambiar nuestro paso E antiguo por el paso del E del Algoritmo de Monte Carlo, pudiendo escribir $z_{11},...,z_{1m}$ y $z_{21},...,z_{2m}$ para cada distribución binomial independiente con una muestra de tamaño n_A y parámetro de probabilidad

$$\frac{p^{k^2}}{(p^{k^2}+2*p^k*r^k)}$$

y otra muestra de tamaño n_B

$$\frac{q^{k^2}}{(q^{k^2}+2*p^k*r^k)}$$

con $\theta^{(k)}$ usado en lugar del vector de parámetros obtenido en la iteración (k+1). Podemos usar la siguiente expresión en vez de la (2.7):

$$n_{AA}^{(k)} = \bar{z}_{1m} = \frac{1}{m}\sum_{j=1}^{m} z_{1i} \qquad n_{BB}^{(k)} = \bar{z}_{2m} = \frac{1}{m}\sum_{j=1}^{m} z_{2i}$$

3.2.3. EM Bayesiano

Podemos aplicar el Algoritmo EM para encontrar la distribución Bayesiana a posteriori.
Si imponemos un parámetro a priori $p(\theta)$ al parámetro θ, entonces

$$logL(\theta) + logp(\theta)$$

es la función logaritmo a posteriori. Su máximo lo obtenemos en el modo a posteriori. El paso E requiere el calculo de la esperanza condicional de la función de log-verosimilitud de los datos completos (funcion Q). El paso M se diferencia en que la función de maximización es igual a Q. La combinación de información previa y de una muestra proporciona una distribución a posteriori del parámetro en el que se basa la estimación. La función Q es ahora dada por

$$Q(\theta;\theta^{(k)}) = E_{\theta^k}\{logL_c(\theta)|y\} + logp(\theta)$$

La diferencia

$$L(\theta) + logp(\theta) - Q(\theta;\theta^k) = L(\theta) - E_{\theta^k}\{logL_c(\theta)|y\}$$

alcanza su máximo en $\theta = \theta^{(k)}$. Así, el paso M de maximizar $Q(\theta;\theta^{(k)})$ fuerza un aumento en la función logarítmica a posteriori.

Bibliografía

[1] S.G. BAKER, *A simple method for computing the observed information matrix when using the EM Algorithm*,Journal of Computational and Graphical Statistics, 1:63-76,1992

[2] T.R. BELIN, D.B. RUBIN, *A method for calibrating false-match rates in record linkage*, Journal of the American Statistical Asocciation, 90:694-707, 1995.

[3] W.R. BLINSCHKE, *Moment estimator fot the parameters of a mixture of two binomial distributions*, The Annals of Mathematical Statistics,33(2):444-454, 1962.

[4] A.COHEN, *Estimation in mixtures of two normal distributions*, Technometrics, 9(1):15-28, 1967.

[5] N. DAY, *Estimating the components of a mixture of normal distributions*, Biometrika, 56(3):463-474, 1969.

[6] P. DELMAR, S. ROBIN,R. TRONIK-LE, J.J DAUDIN, *Mixture model on the variance for the differential analysis of gene expression data*, Journal of the Royal Statistical Society, Serie C (Applied Statistics),54:31-50, 1977.

[7] R. DELGADO DE LA TORRE, *Probabilidad y Estadística para ciencias e ingenierías*, Delta Publicaciones, 2008.

[8] A. DEMPSTER, N. LAIRD, D. RUBIN, *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical Society, Serie B (Methodological),39(1):1-38, 1977.

[9] F. DURÁN JORDA, *El servicio de transfusión de sangre de Barcelona*, Revista de sanidad de guerra, Nº 8, Diciembre 1937.

[10] S.J. FINCH, N.R. MENDEL, THODE, H.C.JR, *Probabilistic measures of adequazy of a numerical search for a global maximum*, Journal of the American Statistical Association,84(408):1020-1023, 1989.

[11] J.H. FRIEDMAN, *Multivariate adaptive regression splines*, The Annals of Statistics,19:1-141, 1991.

[12] G.J. MCLACHLAN, T. KRISHNAN, *The EM Algorithm and Extensions*,Wiley-Interscience, New Jersey, 2008.

[13] A.GELMAN, D.B.RUBIN, *Inference from Iterative Simulation Using Multiple Sequences*, Statistical Science, vol 7 (4):457-472,Noviembre 1992.

[14] B.G. LEROUX, *Maximum-likelihood estimation for hidden Markov models*, Stochastic Processes and their Applications 40, 127-143,1992.

[15] V. HASSELBLAD, *Estimation of parameters for a mixture of normal distributions*,Technometrics, 8(3):312-444,1966

[16] M. JAMSHIDIAN, R.I. JENNRICH, *Standard errors for EM estimation*, Journal of the Royal Statistical Society, serie B, 62(2):257-270

[17] D. KARLIS, E. XEKALAKI, *Choosing initial values for the EM algorithm for finite mixtures*, Computational Statistics and Data Analysis, 41(3-4):577-590, 2003.

[18] R.J.A. LITTLE, D.B.RUBIN, *Statistical Analysis with Missing Data*, Wiley, New York, 1987.

[19] C.LIU, D. RUBIN, *The ECME algorithm: a simple extension of EM and ECM with faster monotone convergence*, Biometrika,81:633-648, 1994

[20] C.E. MCCULLOCH, *Review of .[EM] Algorithm and Extensions"*, Journal of the American Statistical Association 93:403-404,1998

[21] G.MCLACHLAN, D.PEEL, *Finite Mixture Models*, Wiley Series in Probability and Statistics, New York,2000

[22] X.L. MENG, D.B.RUBIN, *Using EM to obtain asymptotic variance-covariance matrix: the SEM algorithm*, Journal of the American Statistical Association,86(416):899-909, 1991.

[23] K.MENGERSE, C. ROBERT, D. TITTERINGTON, *Mixtures: Estimation and Applications*, Wiley Series in Probability and Mathematical Statistics, 2011.

[24] M. WATANABE, K. YAMAGUCHI, *The EM Algorithm and Related Statistical Models*, Marcel Dekker,Inc, New York, 2004.

[25] G.D. MURRAY, *Contribution to discussion of paper by A.P. Dempster, N.M. Laird and D.B. Rubin*, Journal of the Royal Statistical Society,Serie B, 39:23-24, 1977.

[26] R. OLIVA, J. ORIOLA, F. BALLESTA, J. CLÀRIA, *Genética médica*, Publicacions i Edicions Universitat de Barcelona, 2008.

[27] R. REDNER, W.F. HOMER, *Mixture densities, maximum likelihood and the EM Algorithm*, SIAM Review,26(2):195-239, 1984.

[28] M.R. SEGAL, P. BACCHETTI, N.P. JEWELL, *Variances for maximum penalized likelihood estiamtes obtained via the EM algorithm*, Journal of the Royal Statistical Society,Serie B, 56:345-352, 1994.

[29] C. WU, *On the convergence properties of the EM algorithm*, The Annals of Statistics,11(1):95-103, 1983.

[30] *Leyes de Mendel* `https://es.wikipedia.org/wiki/Leyes_de_Mendel`

Apéndice A

Ley de Hardy-Weinberg

La genética de poblaciones es una rama de la genética cuyo objetivo principal es descubrir la variación y distribución de la frecuencia de los alelos para explicar los fenómenos evolutivos.
Una de las leyes más importantes que utiliza es la *Ley De Hardy-Weinberg*. Esta ley proporciona un modelo matemático para el estudio de los cambios evolutivos en la frecuencia de los alelos dentro de una población.
En el año 1908, el matemático G. Hardy y el físico W. Weinberg, de manera independiente, plantearon la hipótesis de que *"la frecuencia de los alelos y de los genotipos se encuentra estable y en equilibrio genético (esto es, si la población reproductiva es grande, todos los individuos de la población tienen las mismas probabilidades de reproducirse y el cruce entre dos individuos se produce al azar, y no hay ni mutaciones, ni migración ni selección natural)"*.
Vamos a estudiar un población de 1000 individuos con dos alelos A y B codominantes con frecuencias alélicas p y q respectivamente. Con estos dos alelos tenemos tres tipos de genotipos: AA, AB Y BB y cada uno de ellos nos da un fenotipo diferente. Supongamos que obtenemos datos del fenotipo de la sangre de cada uno de ellos y los resultados de las frecuencias absolutas obtenidos son los siguientes :

$$AA = 520 \qquad AB = 160 \qquad BB = 320$$

Por lo tanto las frecuencias genotípicas obtenidas serían:

$$AA = 0{,}52 \qquad AB = 0{,}16 \qquad BB = 0{,}32$$

Las Leyes de Mendel nos dicen que:

- Si se cruzan dos razas puras (un homocigoto dominante con uno recesivo) para un determinado carácter, los descendientes de la primera generación serán todos iguales entre sí, fenotípica y genotípicamente, e iguales fenotípicamente a uno de los progenitores (de genotipo dominante), independientemente de la dirección del cruzamiento.
- Esta ley establece que durante la formación de los gametos, cada alelo de un par se separa del otro miembro para determinar la constitución genética del gameto filial.
- diferentes rasgos son heredados independientemente unos de otros, no existe relación entre ellos, por lo tanto el patrón de herencia de un rasgo no afectará al patrón de herencia de otro.

Aplicando esto a nuestra población, vemos que un individuo AA tiene dos alelos A, un individuo AB tiene un alelo A y otro B y un individuo BB tiene dos alelos B.
El número de alelos de cada tipo en esta población es:

$$A = 1200 \qquad B = 800$$

y sus frecuencias relativas son

$$\text{A} = 0{,}6 \qquad \text{B} = 0{,}4$$

Como los cruces se producen al azar, vamos a ver la probabilidad de cada cruce:

$$\begin{aligned}
AA \times AA &= 0,52 \times 0,52 = 0,2704 \\
AA \times AB &= 2 \times (0,52 \times 0,16) = 0,1664 \\
AA \times BB &= 2 \times (0,52 \times 0,32) = 0,3328 \\
AB \times AB &= 0,16 \times 0,16 = 0,0256 \\
AB \times BB &= 2 \times (0,16 \times 0,32) = 0,1024 \\
BB \times BB &= 0,32 \times 0,32 = 0,1024
\end{aligned}$$

Aplicando las Leyes de Mendel podemos calcular las frecuencias genotípicas de los descendientes de cada cruce:

$$\begin{aligned}
AA \times AA &\rightarrow AA(1) \\
AA \times AB &\rightarrow AA(\tfrac{1}{2}) \quad y \quad AB(\tfrac{1}{2}) \\
AA \times BB &\rightarrow AB(1) \\
AB \times AB &\rightarrow AA(\tfrac{1}{4}) \quad , \quad AB(\tfrac{1}{2}) \quad y \quad BB(\tfrac{1}{4}) \\
AB \times BB &\rightarrow AB(\tfrac{1}{2}) \quad y \quad BB(\tfrac{1}{2}) \\
BB \times BB &\rightarrow BB(1)
\end{aligned}$$

y teniendo en cuenta las frecuencias de cada cruce tenemos las siguientes frecuencias genotípicas para la primera generación filial:

$$\begin{aligned}
AA &= \frac{1}{2} \times 0,1664 + 1 \times 0,2704 + \frac{1}{4} \times 0,0256 = 0,36 \\
AB &= \frac{1}{2} \times 0,1664 + 1 \times 0,3328 + \frac{1}{2} \times 0,0256 + \frac{1}{2} \times 0,1024 = 0,48 \\
BB &= \frac{1}{4} \times 0,0256 + 1 \times 0,1024 + \frac{1}{2} \times 0,1024 = 0,16
\end{aligned}$$

Vamos a plantearnos ahora la relación que hay entre las frecuencias de los alelos A y B en la generación paterna y las frecuencias genotípicas que acabamos de calcular en la primera generación filial.
Si llamamos p a la frecuencia del alelo A y $q = 1 - p$ a la de B tenemos que:

$$\text{p} = 0{,}6 \qquad \text{q} = 0{,}4 \qquad \text{en este caso}$$

y las frecuencias genotípicas eran:

$$\text{AA} = 0{,}36 (= p^2) \qquad \text{AB} = 0{,}48 (= 2pq) \qquad \text{BB} = 0{,}16 (= q^2)$$

Su expresión en función de p y q no es casual, siempre se cumple cuando estamos en equilibrio genético y esto es lo que conocemos como la Ley de Hardy-Weinberg.

En situación de equilibrio genético, si las frecuencias de los alelos A y B en la generación paterna son, respectivamente, p y $q = 1 - p$, las frecuencias genotípicas en la primera generación filial son:

$$\text{AA: } p^2 \qquad \text{AB: } 2pq \qquad \text{BB: } p^2$$

Apéndice B

Programación algortimos en R

Órdenes para la estimación máximo verosímil de los parámetros de un modelo multinomial.

Modelo de frecuencias de alelos en grupos sanguíneos de acuerdo con el modelo de equilibrio de Hardy-Weinberg.

Parámetros de entrada $\rightarrow$ n_A,n_B, n_{AB} y n_O
Parámetros de salida $\rightarrow$ p_A,p_B, $p_O = 1 - p_A - p_B$

Modelo con datos completos: modelo multinomial con 6 celdas

$$(n_{AA}, n_{AO}, n_{BB}, n_{BO}, n_{AB}, n_{OO}) \sim multinomial(n, p_A^2, 2p_AP_O, p_B^2, 2p_BP_O, 2p_AP_B, p_O^2)$$

Modelo con datos incompletos: modelo multinomial con 4 celdas

$$(n_A, n_B, n_{AB}, n_{OO}) \sim multinomial(n, p_A^2 + 2p_AP_O, p_B^2 + 2p_BP_O, 2p_AP_B, p_O^2)$$

Algoritmo EM

```
nm <-c(naa,na0,nbb,nb0,nab,n00)
na<-2034
nb<-334
nab<-136
n0<-1776
n<-na+nb+nab+n0
n
[1] 4280 nm<-c(rep(0,6))
```

step 0

```
paold<- na/n
pbold<- nb/n
p0old<- 1-paold -pbold
pold<-c(paold, pbold, p0old)
pold
[1] 0.47523364 0.07803738 0.44672897
pnew<-c(1,0,0)
```

verr<-1.0

$while(verr >= 1{,}0e-10)\{$E-Step

$nm[1] < -na*(pold[1]^2)/(pold[1]^2+2*pold[1]*pold[3]);$
$nm[2] < -na*(2*pold[3]*pold[1])/(pold[1]^2+2*pold[1]*pold[3]);$
$nm[3] < -nb*(pold[2]^2)/(pold[2]^2+2*pold[2]*pold[3]);$
$nm[4] < -nb*(2*pold[3]*pold[2])/(pold[2]^2+2*pold[2]*pold[3]);$
$nm[5] < -nab;$
$nm[6] < -n0;$
$print(nm);$
$sum(nm)$
$naplus < -nm[1]+(nm[2]/2)+(nm[5]/2);$
$nbplus < -nm[3]+(nm[4]/2)+(nm[5]/2);$
$n0plus < -(nm[2]/2)+(nm[4]/2)+nm[6];$

$nplus < -c(naplus, nbplus, n0plus);$
$nplus;$
$sum(nplus);$
$logLc < -sum(nplus*log(pold))$
$logLc$

M-step

$panew < -naplus/n;$
$pbnew < -nbplus/n;$
$p0new < -n0plus/n;$
$pnew < -c(panew, pbnew, p0new);$
$logLcnew < -sum(nplus*log(pnew))$
$logLcnew$
$pnew;$
$verr < -sum((pold-pnew)^2);$
$print(verr);$
$pold < -pnew;$
$print(pold);$
$\}$

```
[1] 706.24036 1327.75964 26.82924 307.17076 136.00000 1776.00000
[1] 0.04513453
[1] 0.3360094 0.0580408 0.6059498
[1] 441.52666 1592.47334 15.26499 318.73501 136.00000 1776.00000
[1] 0.001999855
```

```
[1] 0.30508489 0.05668984 0.63822527
[1] 392.36722 1641.63278 14.20286 319.79714 136.00000 1776.00000
[1] 6.741833e-05
[1] 0.29934197 0.05656575 0.64409228
[1] 383.52843 1650.47157 14.04942 319.95058 136.00000 1776.00000
[1] 2.170058e-06
[1] 0.29830940 0.05654783 0.64514277
[1] 381.94841 1652.05159 14.02325 319.97675 136.00000 1776.00000
[1] 6.928854e-08
[1] 0.29812481 0.05654477 0.64533041
[1] 381.66626 1652.33374 14.01861 319.98139 136.00000 1776.00000
[1] 2.209182e-09
[1] 0.29809185 0.05654423 0.64536392
[1] 381.61588 1652.38412 14.01779 319.98221 136.00000 1776.00000
[1] 7.041925e-11
[1] 0.29808597 0.05654413 0.64536990
```

Datos esperados bajo el modelo H-W y contraste Chi-cuadrado de bondad de ajuste al modelo

nae<-$n*(pold[1]^2+2*pold[1]*pold[3])$
nbe<-$n*(pold[2]^2+2*pold[2]*pold[3])$
nabe<-$n*(2*pold[1]*pold[2])$
n0e<- $n*pold[3]^2$

chisqtest<-$sum((c(nae,nbe,nabe,n0e)-c(na,nb,nab,n0))^2/c(nae,nbe,nabe,n0e))$
$dchisq(chisqtest,df=1)$
```
[1] 0.3291005
```

Cálculo del estimador del error estándar: Método de Louis

var1<-matrix(0, 2,2)

var1[1,1]<-$nae/(pold[1]^2)+n0e/(pold[3]^2)$
var1[2,2]<-$nbe/(pold[2]^2)+n0e/(pold[3]^2)$
var1[1,2]<-$n0e/(pold[3]^2)$
var1[2,1]<-$n0e/(pold[3]^2)$

var1<-2*var1

var2<-matrix(0,2,2)

var2[1,1]<-$na*(pold[1]+pold[3])^2/(pold[1]*pold[3]*(pold[1]+2*pold[3])^2)+nb*pold[2]/(pold[3]*(pold[2]+2*pold[3])^2)$
[1] 16.12156$var2[1,2]<-na*(pold[1]+pold[3])/(pold[3]*(pold[1]+2*pold[3])^2)+nb*(pold[2]+pold[3])/(pold[3]*(pold[2]+2*pold[3])^2)$
$var2[2,1]<-var2[1,2]$
[1] 200.126$var2[2,2]<-nb*(pold[2]+pold[3])^2/(pold[2]*pold[3]*(pold[2]+2*pold[3])^2)+na*pold[1]/(pold[3]*(pold[1]+2*pold[3])^2)$
```
[1] 372.1613
```
var2<-2*var2

imat<-var1-var2
imat
```
[,1] [,2]
[1,] 46729.326 6204.184
[2,] 6204.184 207551.605
```

solve(imat) Matriz de varianzas y covarianzas estimadas
```
[,1] [,2]
[1,] 2.148511e-05 -6.422382e-07
[2,] -6.422382e-07 4.837277e-06
```

Cálculo del estimador del error estándar: Método de Baker

uvec<-c(n0e,nae,nbe,nabe)

$naae < -n * pold[1]^2$
$nbbe < -n * pold[2]^2$
$na0e < -n * 2 * pold[1] * pold[3]$
$nb0e < -n * 2 * pold[2] * pold[3]$

$vvec < -c(n0e, naae, na0e, nbbe, nb0e, nabe)$
$Cmat < -matrix(c(1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1), nrow = 4, ncol = 6, byrow = TRUE)$
$Xmat < -matrix(c(0,0,1,0,0,1,-1,-1), nrow = 4, ncol = 2, byrow = TRUE)$
$Zvec < -matrix(c(exp(1),0,0,1), nrow = 4, ncol = 1, byrow = TRUE)$
$Gmat < -matrix(c(0,0,0,2,0,2,0,0,log(2),1,0,1,0,0,2,0,log(2),0,1,1,log(2),1,1,0), nrow = 6, ncol = 4, byrow = TRUE)$

$theta < -matrix(c(pold[1], pold[2]), nrow = 2, ncol = 1)$

$Yobs < -c(n0, na, nb, nab)$

$Rvec < -c(1,1,1,1,1,1) - t(Cmat)$
$Tvec < -log(Zvec + Xmat$
$dTvec < -c(0, 1/pold[1], 1/pold[2], 1/pold[3])$
$d2Tvec < - - (dTvec^2)$

$Smat < -Gmat$
$I1mat < -t(Xmat)$
$I2mat < -t(Smat)$
$I3mat < -t(Smat)$
$Ibacker < -I1mat + I2mat + I3mat$

$solve(Ibacker)$ Matriz de varianzas y covarianzas estimadas
```
[,1] [,2]
[1,] 2.958844e-05 -1.956155e-06
[2,] -1.956155e-06 6.421127e-06
```

Matriz de información esperada

$Iexpect < -t(Smat)$

$solve(Iexpect)$
```
[,1] [,2]
[1,] 2.956693e-05 -1.956514e-06
[2,] -1.956514e-06 6.416492e-06
```

Cálculo formulas (2.1) y (2.2)

$Ifish < -4*n*(matrix(c(1,1,1,1),2,2)+matrix(c(pold[3]^2,-pold[1]*pold[3],$
$-pold[1]*pold[3],pold[1]^2),2,2)/(pold[1]^2+2*pold[1]*pold[3])+matrix(c(pold[2]^2,-pold[2]*$
$pold[3],$
$-pold[2]*pold[3],pold[3]^2),2,2)/(pold[2]^2+2*pold[2]*pold[3])+matrix(c(pold[2]^2,pold[1]*pold[2],$
$pold[1]*pold[2],pold[1]^2),2,2)/(2*pold[1]*pold[2]))$

Ifish
```
[,1] [,2]
[1,] 34518.04 10525.23
[2,] 10525.23 159057.75
```

solve(Ifish)
```
[,1] [,2]
[1,] 2.956693e-05 -1.956514e-06
[2,] -1.956514e-06 6.416492e-06
```