Métodos de inferencia estadística para entrenamiento de modelos ocultos de Markov

Ricardo Antonio Mendoza León*

Politécnico Grancolombiano

Fecha de recepción: 28 de abril de 2010
Fecha de aprobación: 9 de junio de 2010

Resumen Este documento presenta una revisión general de las diferentes aproximaciones y métodos en inferencia estadística, aplicados al problema de entrenamiento o ajuste de parámetros en Modelos Ocultos de Markov. Se tratarán los algoritmos EM *(Expectation Maximization)* y GEM *(Generalized Expectation Maximization)*, el marco de modelos gráficos y sus algoritmos ML *(Maximum Likelihood)* y MAP *(Maximum a Posteriori)*, así como modelos de conjunto, variacionales y métodos de muestreo MCMC *(Markov Chain Montecarlo)*.

Abstract This paper presents an overview of the different approaches and methods in statistical inference, applied to the problem of training or parameter adjustment of Hidden Markov Models. We review the EM *(Expectation Maximization)* and GEM algorithms, the graphical models framework includding the ML *(Maximum Likelihood)* and MAP *(Maximum a Posteriori)* algorithms, ensemble and variational models, and Markov Chain Montecarlo *(MCMC)* sampling methods.

Palabras Clave: inferencia estadística, modelos ocultos de Markov, algoritmo EM, modelos gráficos, modelos variacionales, muestreo MCMC.

Keywords: statistical inference, hidden Markov models, EM algorithm, graphical models, variational models, MCMC sampling.

* Ingeniero de Sistemas de la Universidad de los Andes, Bogotá, Colombia. Docente de cátedra del Politécnico Grancolombiano en las asignaturas de Programación de Computadores y Redes II. Entre sus intereses académicos se encuentran la Bioinformática, *Machine Learning*, Seguridad Informática y la Compresión de Datos. `ramendoza@poli.edu.co`. El proyecto de investigación del cual es producto este documento ha sido financiado por la Fundación Politécnico Grancolombiano, mediante el contrato de investigación No: 2010-D4-FICB-C5-BC-14, del 14 de diciembre de 2009.

1. Introducción

Durante los últimos 30 años, los Modelos Ocultos de Markov (HMM), se han transformado en una herramienta de amplio uso en la comunidad científica. Dado lo anterior, se han desarrollado valiosos aportes a diferentes problemáticas como reconocimiento de voz [40], Bioinformática [2,30,41], Finanzas [20] y Control estocástico [14].

El aprendizaje en HMM, es un complejo problema de optimización estocástica multi paramétrica, dada su alta complejidad dimensional, la existencia de múltiples óptimos locales y la limitada cantidad de muestras empleadas en el entrenamiento. Obtener un HMM que represente adecuadamente las características del espacio poblacional de interés, requiere del análisis profundo del espacio poblacional y la selección e implementación de un método de aprendizaje adecuado para la topología del HMM.

Este documento presenta una revisión global de los diferentes métodos disponibles en la inferencia estadística, aplicados al problema de la estimación de los parámetros del HMM [10].

La segunda sección provee de una breve introducción a los HMM, características relevantes de su topología y los tres problemas básicos en su uso. En la sección tres se presentará el algoritmo clásico para el ajuste de parámetros Baum-Welch [3,39], extendiendo en aspectos de su derivación, para el caso discreto bajo el marco del algoritmo EM [13,38]. Adicionalmente se introducirá el algoritmo Baldi-Chauvin [1] como propuesta alternativa al algoritmo Baum-Welch bajo el marco del algoritmo GEM [38].

En la sección cuatro, se presentará un recuento de metodologías alternativas en inferencia estadística, aplicadas al HMM y la inferencia de sus parámetros, incluyendo un recuento de la literatura a la fecha, de aplicaciones prácticas en diversas problemáticas.

2. Modelos ocultos de Markov

Los HMM nos permiten modelar la dinámica de un sistema (oculto), al cual no podemos acceder (observar) de forma directa; por el contrario de forma indirecta mediante la observación de eventos externos, suponemos que están correlacionados con dicho sistema y su estado. Existen diversas razones por los cuales el sistema no es accesible de forma directa, como la imposibilidad física o la presencia de ruido en la medición [39,13].

De forma general definimos un HMM, como un modelo probabilístico, utilizado para representar la probabilidad conjunta de un conjunto de variables aleatorias [6]. En este conjunto de variables aleatorias distinguimos dos tipos. El primero corresponde a los posibles eventos o símbolos observables O_t, que pueden presentarse al realizar una observación indirecta del sistema oculto. El segundo corresponde al estado en el cual se encuentra el sistema oculto Q_t durante una observación.

Las variables aleatorias de observación, puede ser bien discretas $O = o \in V = \{1, 2, \ldots, L\}$, o continuas. La medida de probabilidad en cada caso estará

definida, bien por una función de masa de probabilidad (pmf) o por una función de densidad de probabilidad (pdf) de tipo *gaussiano* generalmente [6,14,39,13]. Las variables aleatorias de estado oculto son discretas y finitas $Q_t \in \{1, 2, \ldots, N\}$, pero variantes como los HMM infinitos [4], permiten superar esta restricción.

Con base en estos dos tipos, son construidas secuencias de variables aleatorias tanto de observaciones $O = \{O_1 = o_1, O_2 = o_2, \ldots, O_t = o_t\}$ como de estados ocultos del sistema $Q = \{Q_1, Q_2, \ldots, Q_t\}$. De esta forma, el par (O, Q), representa la posible historia dinámica del sistema oculto.

La probabilidad de una determinada secuencia de estados ocultos $q = \{q_1, q_2, \ldots, q_L\}$ es calculada empleando probabilidades de transición entre los estados, siguiendo un proceso de Markov [15,45]. En este proceso la probabilidad de transición de estado, asume invariancia en el tiempo $P\left(q_i^{t=k} \left| q_{i-1}^{t=k}\right.\right) = P\left(q_i^{t=l} \left| q_{i-1}^{t=l}\right.\right)$ y dependencia, únicamente frente a los k estados anteriores $P(q_i|q_{i-1}, q_{i-2}, \ldots, q_{i-k}, \ldots, q_1) = P(q_i|q_{i-1}, q_{i-2}, \ldots, q_{i-k})$.

Para el caso $k = 1$, tenemos un HMM de primer orden y su proceso es descrito mediante cadenas de Markov condicionales. Los HMM de primer orden son tradicionalmente los más usados. La razón de esto, deriva en la simplificación de los cálculos, mediante el empleo de técnicas de programación dinámica, explotadas por algoritmos como *forward*, *backward*, Viterbi y Baum-Welch.

Definimos un HMM de primer orden, mediante la tripla $\lambda = (\pi, A, B)$ donde $\pi_i = P(Q_1 = q_i)$ es el vector de probabilidad inicial para los estados ocultos, $A = \{a_{ij}\} = P(Q_t = j \,|Q_{t-1} = i\,)$, es la matriz de probabilidad de transición de estados y $B = \{b_i(O_t)\} = P(O_t = o_t \,|Q_t = i\,)$, es la matriz de probabilidad de difusión para las observaciones, dado el estado oculto actual.

2.1. Tres problemas básicos en HMM

Existen tres problemas básicos al emplear HMM [39], los cuales son:

1. Problema de la evaluación: dada una secuencia de observaciones $O = (o_1, o_2, \ldots o_L)$ y un HMM $\lambda = (\pi, A, B)$, determinar $P(O|\lambda)$. Los algoritmos: *forward* o *backward* son comúnmente utilizados en su solución.
2. Problema de la decodificación: dada una secuencia de observaciones $O = (o_1, o_2, \ldots o_L)$, y un HMM $\lambda = (\pi, A, B)$, encontrar la secuencia de estados ocultos $Q^k = \left\{q_1^k, q_2^k, \ldots q_L^k\right\}$, tal que:

$$Q^k = \max_{Q^i} P\left(Q^i|\lambda, O\right) \tag{1}$$

   Su solución se obtiene mediante el algoritmo de Viterbi [47].
3. Problema del aprendizaje: dada una secuencia de observaciones $O = (o_1, o_2, \ldots o_L)$ determinar los parámetros del modelo $\lambda^* = (\pi, A, B)$, tal que:

$$\lambda^* = \max_{\lambda^i \in \Omega} P\left(O|\lambda^i\right) \tag{2}$$

   Donde Ω corresponde al espacio de parámetros en la topología del HMM particular. El algoritmo empleado tradicionalmente para su solución es el algoritmo Baum-Welch o también conocido como *forward-backward*.

2.2. Topología del HMM

La topología del HMM, hace referencia a la cantidad de parámetros y restricciones sobre los mismos. La cantidad de estados dependerá del conocimiento a priori de las características del sistema oculto en estudio y como éste se manifiesta. Sin embargo no existen limitaciones sobre el número máximo de estados ocultos y símbolos observables que pueden representar un HMM.

También es posible definir restricciones sobre las matrices de transición, difusión y estado inicial. Comúnmente estas restricciones son creadas empleando probabilidades con valor cero.

Las restricciones aportan significado semántico a los estados del modelo oculto y sus transiciones, creando un mecanismo para la formulación de hipótesis estructurales sobre el sistema oculto. Las topologías no ergódicas de izquierda a derecha o *left to right*, de uso común en reconocimiento de voz [42,40], y topologías cíclicas con restricciones parciales de transición empleadas en Bioinformática son algunos ejemplos [41,46,9].

3. Inferencia por máxima verosimilitud

3.1. Algoritmo Baum-Welch

El algoritmo Baum-Welch es un método de reestimación iterativa de los parámetros del HMM. En cada iteración, Baum-Welch calcula un nuevo conjunto de parámetros $\lambda^* = (\pi^*, A^*, B^*)$ con base en los parámetros actuales $\lambda = (\pi, A, B)$, de forma tal que el modelo ajuste mejor la muestra:

$$P\left(O|\lambda^*\right) \geq P\left(O|\lambda\right) \tag{3}$$

La convergencia del algoritmo a un óptimo local está garantizada (Wu, 1983), siempre y cuando el espacio de parámetros sea continuo y no presente restricciones de transición.

3.2. Derivación EM del algoritmo Baum-Welch

El algoritmo Baum-Welch es un caso especial del algoritmo de Maximización de Expectativa (EM) sobre el modelo estocástico representado por los HMM [13,38,49,6].

EM es un método iterativo general de ajuste de máxima verosimilitud, en distribuciones sobre variables aleatorias ocultas. EM, asume la existencia de un conjunto de "información completa" $Z = (X, Y)$ así como la existencia de una densidad de probabilidad conjunta sobre éste:

$$P\left(Z|\Theta\right) = P\left(X, Y|\Theta\right) = P\left(Y|X, \Theta\right) P\left(X|Y, \Theta\right) \tag{4}$$

Donde X y Y, son denominadas: la información incompleta, proveniente de variables observables y la información oculta, proveniente de variables no observables, respectivamente.

La distribución de la densidad de probabilidad conjunta, sobre la información completa e incompleta, es definida mediante la construcción de relaciones entre los valores X y Y. Con base en esto, se define la función de verosimilitud sobre los parámetros Θ, de la distribución conjunta de información completa:

$$\mathcal{L}(\Theta|Z) = \mathcal{L}(\Theta|X,Y) = P(X,Y|\Theta) \tag{5}$$

El algoritmo EM, calcula el valor esperado del logaritmo de la función de verosimilitud de información completa, para los nuevos parámetros Θ , dados los parámetros actuales Θ^{i-1}:

$$Q\left(\Theta,\Theta^{i-1}\right) = E\left(\log P(X,Y|\Theta)\,|X,\Theta^{i-1}\right) \tag{6}$$

La anterior expresión se conoce como el paso de expectativa (E). El segundo paso, de maximización (M), el cual busca determinar el conjunto de parámetros Θ^i que maximice la función Q:

$$\Theta^i = \underset{\Theta_j\in\Omega}{\text{máx}}\, Q\left(\Theta_j,\Theta^{i-1}\right) \tag{7}$$

En el caso del algoritmo Baum-Welch, los pasos de expectativa y maximización son realizados de forma simultánea. Inicialmente se define la función de expectativa Q para el HMM:

$$\begin{aligned} Q(\lambda,\lambda') &= \sum_q \log\left(P(O,q|\lambda)\right)P(O,q|\lambda') \\ &= \sum_q \log\left(\pi_{q_0}\prod_{t=1}^{|O|=L} a_{q_{t-1},q_t} b_{q_t}(o_t)\right)\pi'_{q_0}\prod_{t=1}^{|O|=L} a'_{q_{t-1},q_t} b'_{q_t}(o_t) \end{aligned} \tag{8}$$

De lo anterior se obtiene el siguiente resultado ordenando los términos:

$$\begin{aligned} Q(\lambda,\lambda') = \sum_q \log\pi_{q_0} P(O,q|\lambda') &+ \sum_q\left(\sum_{t=1}^{L}\log b_{q_t}(o_t)\right)P(O,q|\lambda') \\ &+ \sum_q\left(\sum_{t=1}^{L}\log a_{q_{t-1},q_t}\right)P(O,q|\lambda') \end{aligned} \tag{9}$$

Para obtener los parámetros λ, se procede optimizando cada término independiente en la expresión anterior, mediante el método de multiplicadores de Langrange, agregando las restricciones estocásticas $\sum_{i=1}^{N}\pi_1 = 1$, $\sum_{j=1}^{N} a_{i,j} = 1$ y $\sum_{j=1}^{M} b_i(o(j)) = 1$. Resolviendo y despejando los parámetros del modelo, se obtienen los términos de reestimación del algoritmo Baum-Welch:

$$\pi_i = \frac{\alpha_i(1)\,\beta_i(1)}{\sum_{j=1}^{N} \alpha_j(1)\,\beta_j(1)} \tag{10}$$

$$a_{i,j} = \frac{\sum_{t=1}^{L-1} \xi_{i,j}(t)}{\sum_{t=1}^{L-1} \gamma_i(t)} \tag{11}$$

$$b_i(k) = \frac{\sum_{t=1}^{L-1} \delta_{o_t,v_k}\gamma_i(t)}{\sum_{t=1}^{L-1} \gamma_i(t)} \tag{12}$$

Donde $\alpha_i(t=l)$, es la probabilidad de observar la secuencia parcial $O = (o_1, o_2, \ldots o_{t=l})$ y finalizar en el estado oculto i, o probabilidad *forward* $\beta_i(t=l)$, es la probabilidad de continuar del estado oculto i en $t=l$, y luego observar la secuencia $O = (o_{t=l+1}, o_{l+2}, \ldots o_{t=T})$, o probabilidad *backward*:

$$\alpha_i(1) = \pi_i b_i(o_1) \qquad \alpha_i(t+1) = b_i(o_1)\left(\sum_{i=1}^{N} \alpha_i(t)\,a_{i,j}\right) \tag{13}$$

$$\beta_i(L) = 1 \qquad \beta_i(t) = \sum_{j=1}^{N} a_{i,j} b_j(o_{t+1})\,\beta_j(t+1) \tag{14}$$

La expresión $\gamma_i(t)$ es la probabilidad de encontrarse en el estado i en momento t, dada una secuencia de observaciones O. $\xi_{i,j}(t)$ es la probabilidad de continuar al estado oculto j, dado que se encuentra en el estado i en el instante $t=l$, observando la secuencia O.

$$\gamma_i(t) = \frac{\alpha_i(t)\,\beta_i(t)}{\sum_{j=1}^{N} \alpha_j(t)\,\beta_j(t)} \tag{15}$$

$$\xi_{i,j}(t) = \frac{\alpha_i(t)\,\alpha_{i,j} b_j(o_{t+1})\,\beta_i(t+1)}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i(t)\,\alpha_{i,j} b_j(o_{t+1})\,\beta_i(t+1)} \tag{16}$$

El proceso para los casos con múltiples secuencias de observación y secuencias de observación continuas es análogo. Estos se obtienen modificando la definición de función de expectativa de información completa.

En el caso de múltiples secuencias de observación, asumiendo que estas son independientes, se obtiene:

$$
\begin{aligned}
Q\left(\lambda,\lambda'\right) &= \sum_{q} \log\left(P\left(O^{(1\ldots m)},q|\lambda\right)\right) P\left(O^{(1\ldots m)},q|\lambda'\right) \\
&= \sum_{q} \log\left(\prod_{o^i \in O^{(1\ldots m)}} \pi_{q_0} \prod_{t=1}^{|O^i|=L^i} a_{q_{t-1},q_t} b_{q_t}\left(o_t^i\right)\right) \times \\
&\quad \prod_{o^i \in O^{(1\ldots m)}} \pi'_{q_0} \prod_{t=1}^{|O^i|=L^i} a'_{q_{t-1},q_t} b'_{q_t}\left(o_t^i\right) \qquad (17)
\end{aligned}
$$

Para el caso sobre espacios de observación continuos, las densidades condicionales de difusión suelen ser mixturas *gaussianas*, exponenciales o Dirichlet [39,6,14]. Más detalles de su derivación para el caso continuo pueden ser consultados en [6,13,39,38].

Las expresiones de reestimación del algoritmo Baum-Welch tienen una fuerte relación con los parámetros seleccionados al inicio del algoritmo. Lo anterior es una debilidad del método. Es de vital importancia seleccionar los mismos con especial atención, con el fin de obtener buenos resultados.

Otro problema son las probabilidades de transición iguales a cero. Baldi y Chauvin [1], puntualizan sobre las implicaciones de contar con probabilidades de transición o emisión iguales a cero. Dado su carácter absorbente que puede afectar la convergencia y ajuste del modelo.

3.3. Algoritmo Baldi-Chauvin

El algoritmo de Baldi-Chauvin [1], al igual que el algoritmo Baum-Welch, es un método iterativo, de ascenso de gradiente, que busca estimar los parámetros de máxima verosimilitud para el HMM, con base en las observaciones disponibles.

El método es una propuesta al aprendizaje tanto en línea *(Online Learning)* [33], como en lote, de los parámetros del HMM frente a una o múltiples secuencias. El aprendizaje en línea es un proceso empleado en problemas que requieren ajuste dinámico del modelo. Para cada ejemplo de entrenamiento, se ajusta y evalúa el nuevo modelo obtenido.

Los algoritmos en línea buscan principalmente minimizar el error de evaluación luego de cada ejemplo presentado [34]. El algoritmo Baum-Wech para múltiples secuencias emplea un proceso de aprendizaje en lote, en el cual todas las secuencias se evalúan a la vez, y por tanto no es aplicable para el aprendizaje en línea [1].

Baldi y Chauvin motivan su propuesta en problemas del algoritmo Baum-Welch, como los saltos abruptos del espacio de parámetros durante la reestimación, indeseables para el aprendizaje en línea. Sus debilidades frente a topologías izquierda - derecha y los efectos de las probabilidades de transición o emisión iguales a cero.

La idea fundamental del método consiste en representar tanto las probabilidades de transición como emisión del HMM, mediante expresiones exponenciales normalizadas:

$$a_{i,j} = \frac{e^{\tau w_{ij}}}{\sum_{i=1}^{N} e^{\tau w_{ij}}} \tag{18}$$

$$b_i\left(o_t = j\right) = \frac{e^{\tau v_{ij}}}{\sum_{i=1}^{N} e^{\tau v_{ij}}} \tag{19}$$

Donde τ, es un factor de temperatura, que ajusta la reestimación del modelo. De esta forma, los parámetros a ajustar en el modelo corresponden a w_{ij} y v_{ij}. Esta representación elimina las probabilidades iguales a 0 y suaviza la variación de los parámetros en la reestimación. Los nuevos parámetros se obtienen mediante las diferencias:

$$\Delta w_{i,j} = \eta \frac{\left[n_{i,j}\left(O\right) - \left(\sum_{j=1}^{N} n_{i,j}\left(O\right)\right) a_{i,j}\right]}{\mathcal{L}\left(\lambda|O\right)} \tag{20}$$

$$\Delta v_{i,j} = \eta \frac{\left[m_{i,j}\left(O\right) - \left(\sum_{j=1}^{M} m_{i,j}\left(O\right)\right) b_i\left(o = j\right)\right]}{\mathcal{L}\left(\lambda|O\right)} \tag{21}$$

Donde η, es la tasa de aprendizaje, $\mathcal{L}\left(\lambda|O\right)$ es la función de verosimilitud para el modelo actual y $n_{i,j}$, $m_{i,j}$ la cantidad de $i \to j$ transiciones y emisiones, dada la observación y el modelo actual respectivamente.

De la reestimación anterior, se obtiene un incremento marginal monotónico no necesariamente máximo de la función de verosimilitud respecto al modelo actual. El método Chauvin-Baldi es un caso especial del algoritmo GEM [38,6,1,5] que a diferencia de EM, su convergencia solo exige:

$$Q\left(\Theta^i, \Theta^{i-1}\right) > Q\left(\Theta, \Theta^{i-1}\right) \tag{22}$$

4. Propuestas alternativas en inferencia de parámetros para HMM

4.1. Modelos de conjunto

La idea general de los modelos de conjunto o *ensembles* [12,36,18], se fundamenta en suponer que los modelos inferidos contienen información parcial sobre las características del sistema oculto real.

De lo anterior es posible construir un modelo consenso que sintetice la mayor cantidad de información sobre las características del modelo oculto real, contenida en modelos particulares.

Una de las ventajas de esta aproximación, a diferencia de los métodos EM y GEM, es poder contar con indicadores para la incertidumbre en los parámetros, como por ejemplo, los intervalos de confianza.

Los modelos particulares representan diferentes resultados del entrenamiento sobre una o múltiples observaciones individuales. Así proveen diferentes puntos de vista sobre el modelo real. Una fortaleza de estos métodos, es aprovechar la información contenida en modelos sobre estimados.

En los métodos *ensemble*, los parámetros del modelo son variables aleatorias, regidas por una familia de distribuciones caracterizada por un conjunto de hiperparámetros. En este sentido, los parámetros de cada modelo generado, constituyen una muestra sobre el espacio de parámetros del HMM.

El modelo consenso corresponde al valor esperado de los parámetros dada la familia de distribuciones definida. Los hiperparámetros son ajustados, empleando diversos métodos como máxima verosimilitud, momentos o EM.

La propuesta de McKay [36], define la familia de distribuciones como la distribución posterior en un modelo de inferencia bayesiano, donde las probabilidades *a priori* son evaluadas mediante distribuciones Dirichlet parametrizadas.

Otro ejemplo de modelo *ensemble* pondera conjuntos de parámetros para construir el modelo consenso [12].

Los modelos *ensemble* han sido aplicados en problemas como reconocimiento de escritura [27] y alineación de proteínas [44].

4.2. Modelos gráficos

Los modelos gráficos [28,43,24,26,22,10,7] son métodos de representación y análisis de modelos probabilísticos mediante el uso de grafos. En estos, los nodos corresponden a variables aleatorias y los arcos representan supuestos de independencia o dependencia condicional entre variables. De esta forma es posible analizar y obtener probabilidades marginales y condicionales de interés, mediante el empleo de técnicas y algoritmos derivados de la teoría de grafos.

Dada la naturaleza de las relaciones de dependencia e independencia condicional, presente en los HMM, es posible aplicar esta metodología y sus técnicas para plantear y solucionar diferentes problemáticas. Los HMM, de cualquier orden y topología, corresponden a casos particulares en el contexto de los modelos gráficos [28].

Existen dos clases de representación de modelos gráficos, los dirigidos y los no dirigidos, representados por grafos dirigidos acíclicos y grafos no dirigidos, respectivamente. Los HMM son expresables en forma natural mediante la representación dirigida de los supuestos de dependencia condicional entre parámetros y variables aleatorias.

Existen diversas técnicas de inferencia probabilística, análogas a los algoritmos: *forward*, *backward* y Viterbi, de carácter exacto y aproximado. Un ejemplo de estos últimos es el muestreo de Cadenas Markov Montecarlo o MCMC.

Los mecanismos de inferencia exacta, calculan probabilidades de interés mediante la evaluación eficiente y exhaustiva de las funciones de densidad o masa de probabilidad condicional –sobre las variables aleatorias presentes en el

grafo– explotando su topología. Ejemplos de métodos exactos son el *Junction Tree Algorithm* [26], *Factor Analysis* y *Component Analysis* [50].

En modelos gráficos, se dispone de diversas aproximaciones para el aprendizaje de los parámetros. Bajo información completa (todas las variables aleatorias son observables), el algoritmo de Maximización de Verosimilitud (ML) determina los parámetros, maximizando la función de verosimilitud sobre la información disponible.

$$\mathcal{L}(\theta) = \sum_{i=1}^{N} \log p(x_n|\theta) \tag{23}$$

El algoritmo Máximo *a Posteriori* (MAP) [16], incorpora conocimiento *a priori* disponible, mediante una distribución (generalmente Dirichlet), sobre los parámetros $p(\theta)$. Con base en esta distribución, MAP encuentra los parámetros que maximizan la distribución posterior $p(\theta|d)$, respecto la información d maximizando:

$$\mathcal{L}'(\theta) = \sum_{i=1}^{N} \mathcal{L}(\theta_i) + \log p(\theta_i) \tag{24}$$

En aprendizaje con información incompleta, se pueden extender los algoritmos ML y MAP, bajo el marco del algoritmo EM. Aquí tanto ML como MAP proveen las estadísticas necesarias sobre las relaciones entre variables observables y ocultas, requeridas por la función de expectativa de información completa. Más detalles de la derivación detallada de las expresiones de reestimación para ML y MAP, se pueden consultar en [43,16,8].

4.3. Modelos variacionales

Cuando no es posible o implica mucho trabajo aplicar métodos de optimización exacta sobre determinada función – objetivo; bien por la intratabilidad de las expresiones o el número de parámetros involucrados, los modelos variacionales [19,23,17,29,7] permiten estimar una distribución compleja, mediante el empleo de una distribución computable, más simple $q(\theta)$. Un ejemplo de esto, es aproximar la distribución marginal $p(x)$, con base en la distribución *a priori* $p(x, y|\theta)$ y la distribución *a posteriori* $p(x|\theta)$ respecto a los parámetros θ y variables ocultas y, mediante el empleo de una cota inferior:

$$p(x) = \iint p(x, y|\theta)\, d\theta dy = \iint p(y|x, \theta)\, p(x|\theta)\, d\theta dy \tag{25}$$

Tomando logaritmos, agregando la distribución desconocida $q(\theta, y)$ y aplicando la desigualdad de Jensen, se obtiene la cota inferior para $p(x)$:

$$\begin{aligned} p(x) &= \log \iint q(\theta, y) \frac{p(y|x, \theta)\, p(x|\theta)}{q(\theta, y)} d\theta dy \\ &\geq \iint q(\theta, y) \log \frac{p(y|x, \theta)\, p(x|\theta)}{q(\theta, y)} d\theta dy \end{aligned} \tag{26}$$

Si $q(\theta, y) = (y, x|\theta)$, entonces la desigualdad se transforma en una igualdad. Lo anterior ocurre si la divergencia Kullback-Leiber (KL) entre las dos distribuciones es igual a 0:

$$KL(q|p) = \iint q(\theta, y) \log \frac{p(x, \theta|y)}{q(\theta, y)} d\theta dy \tag{27}$$

De forma análoga a la anterior, es posible aplicar esta técnica en el contexto de los HMM, bien sobre los algoritmos EM, GEM o los algoritmos ML y MAP en modelos gráficos. Se invita al lector a consultar [24,19,16,18], para más detalles frente a la derivación de las aproximaciones para EM y modelos gráficos. Para una introducción en la derivación computacional de la distribución $q(\theta, y)$, mediante el método de elementos finitos, se puede consultar [19]. Ejemplos de aplicación del método variacional sobre HMM pueden consultarse en [21,35].

4.4. Métodos de muestreo

Las técnicas de muestreo son procedimientos utilizados para obtener conjuntos independientes de muestras, respecto a una distribución de probabilidad dada. Estas muestras, provenientes de la distribución de interés, se pueden emplear en procesos de inferencia estadística tanto de probabilidades como de parámetros [7].

El problema fundamental de estas técnicas radica en la obtención de muestras, en forma independiente para la distribución dada, ya que esto último no es trivial en todos los casos.

El muestreo de cadenas de Markov Montecarlo (MCMC) [37,7], es un conjunto general de técnicas de muestreo simulado, que emplean distribuciones construidas mediante cadenas de Markov, para generar muestras independientes con base en la distribución de interés.

Ejemplos de técnicas MCMC son: el muestreo unidimensional como importancia, de rechazo y Metrópolis.

El muestreo de Gibbs [37,7], es una poderosa técnica de muestreo multidimensional empleado en la generación de muestras independientes. El muestreo simula cadenas de Markov cuyas probabilidades de transición se definen empleando la distribución de probabilidad condicional para cada variable en la distribución de interés.

La ventaja de utilizar distribuciones de probabilidad condicional sobre cada variable, se debe a que estas distribuciones son más simples de derivar, comparadas con la distribución de probabilidad conjunta de todas las variables.

Los procesos MCMC, mediante muestreo Gibbs, han sido empleados ampliamente en el problema de inferencia de parámetros, particularmente en el marco de los métodos gráficos [11,26] como alternativa a métodos exactos, como lo son los algoritmos ML, MAP; tanto para el caso de información completa como también para la incompleta. MCMC ha sido empleado con éxito, como método de aproximación en el marco del algoritmo EM, para la inferencia de los parámetros de HMM [45,21,10,31].

Una desventaja del método es el recurso de cómputo requerido para generar muestras suficientes y obtener buenas aproximaciones. Por otra parte, estos

métodos reducen sustancialmente el sobre ajuste de parámetros y la dependencia de un modelo inicial, problema muy común en el algoritmo Baum-Wech. El método MCMC sobre HMM, ha sido empleado en problemas como la predicción de terremotos [48] y ontología de genes [32].

Referencias

1. Baldi, P., Chauvin, Y.: Smooth On-Line Learning Algorithms for Hidden Markov Models. Neural Cornputation 6, 307-318. (1994)
2. Baldi, P., Brunak, S.: Bioinformatics: the machine learning approach. Boston: MIT Press. (2001)
3. Baum, L. E., Petrie, Soules, G., Weiss, N.: "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains" Ann. Math. Stat., vol. 41, no. 1, 164-171. (1970)
4. Beal, M. J., Ghahramani, Z., and Rasmussen, C. E.: The infinite hidden Markov model. Advances in Neural Information Processing Systems, volume 14. Cambridge: MIT Press. (2002)
5. Bengio, Y., Frasconi P.: Input/Output HMMs for sequence processing. IEEE Transactions on Neural Networks 7(5), 1231-1249. (1996)
6. Bilmes, J. A.: A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden markov models. Technical Report ICSI-TR-97-02, University of Berkeley. (1998)
7. Bishop, C. M.: Pattern Recognition and Machine Learning. New York: Springer. (2006)
8. Blanchet, J., Vignes, M.: A Model-Based Approach to Gene Clustering with Missing Observation Reconstruction in a Markov Random Field Framework. En Journal of Computational Biology, Vol 16, No 3. 475-486. (2009)
9. Boufounos, P., El-Difrawy, S., Ehrlich, D.: Hidden Markov Models for DNA Sequencing. Proceedings of Workshop on Genomic Signal Processing and Statistics (GENSIPS 2002), Raleigh, NC, USA. (2002)
10. Cappé, O., Moulines, E. and Rydén, T.: Inference in Hidden Markov Models. New York: Springer. (2005)
11. Chu, W., Ghahramani, Z., Wild, D.: A graphical model for protein secondary structure prediction. En Proc. 21st Ann. Intl. Conf. on Machine Learning (ICML), Banff, Canada. (2004)
12. Davis, R., Lovell, B. C.: Comparing and evaluating hmm ensemble training algorithms using train and test and condition number criteria. Pattern Anal Appl 6(4). 327-335. (2003)
13. Dempster, A. P, Laird, N. M., Rubin, D. B.: Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society. Series B (Methodological), Vol. 39, No. 1., 1-38. (1977)
14. Elliott, R. J., Aggoun, L., Moore, J. B.: Hidden Markov Models Estimation and Control., 3ed. New York: Springer. (2008)
15. Ephraim, Y., Neri Merhav, N.: Hidden Markov Processes. IEEE Transactions on Information Theory, Vol. 48, No. 6. (2006).
16. Ghahramani, Z.: Graphical models: parameter learning. En Arbib, M. A. (Ed). The Handbook of Brain Theory and Neural Networks. Cambridge, MA: MIT Press. (1995)

17. Ghahramani, Z., M. Beal.: Graphical Models and Variational Methods. En M. Opper and D. Saad (Ed). Advanced Mean Field Methods - Theory and Practice. Cambridge, MA: MIT Press. (2001)
18. Heo, G., Woo, Y. W., Kim, K. B.: Properties of Ensemble Learning for Discrete Hidden Markov Models and Updating Prior Strategy. (2007)
19. Jaakkola, T. S.: Tutorial on variational approximation methods. En Advanced mean field methods. Cambridge, MA: MIT Press. (2001)
20. Jalen, L.: Some contributions to filtering theory with applications in financial modelling. Tesis Doctoral, Brunel University. (2009)
21. Jianfeng, G., Johnson, M.: A comparison of Bayesian estimators for unsupervised Hidden Markov Model POS taggers. En Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. 344-352. (2008)
22. Jiao, F.: Probabilistic Graphical Models and Algorithms for Protein Problems. Tesis Doctoral, University of Waterloo. (2007)
23. Jordan, M., Ghahramani, Z., Jaakkola, T. S., Saul, L.: An introduction to variational methods for graphical models. En Learning in graphical models. 105-161. Cambridge, MA: MIT Press. (1999)
24. Jordan, M. I.: (Ed). Learning in Graphical Models. Cambridge, MA: MIT Press. (1999)
25. Jordan, M. I.: Graphical models, exponential families, and variational inference. UC Berkeley Dept. of Statistics, Tech. Rep. 629. (2003)
26. Jordan, M. I.: Graphical Models. Statist. Sci., 19, 140-155. (2004)
27. Ko, A. H. R., Sabourin, R., Britto A. Jr.: Ensemble of HMM classifiers based on the Clustering Validity Index for a Handwritten Numeral Recognizer. Pattern Analysis and Applications Journal. (2008)
28. Lauritzen, S. L.: Graphical Models. Oxford Science Publications. (1996)
29. Lawrence, N. D.: Variational inference guide. Technical report, The University Of Sheffield Machine Learning Group. (2002)
30. Lesk, A. M.: Introduction to Bioinformatics. New York: Oxford University Press Inc. (2002)
31. Liang, K., Wang, X., Anastassiou, D.: Bayesian Basecalling for DNA Sequence Analysis Using Hidden Markov Models. En IEEE/ACM Transactions on Computational Biology and Bioinformatics 4, No. 3, 430-440. (2007)
32. Liang, K., Nettleton, D.: A Hidden Markov Model Approach to Testing Multiple Hypotheses on a Gene Ontology Graph. Dep. of Stat. Iowa State University. (2009)
33. Littlestone, N.: Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. Mach. Learning 2, 2, 285-318. (1988)
34. Littlestone, N., Warmuth, M. K.: The Weighted Majority algorithm. Information and Computation, 108, 212-261. (1994)
35. McGrory C. A., Titterington, D. M.: Variational Bayesian Analysis for Hidden Markov Models. En Australian & New Zealand J. of Stat. Vol, No 2, 227 - 244. (2009)
36. McKay, D. J. C.: Ensemble learning for hidden Markov models. Technical report, Cavendish Laboratory, University of Cambridge. (1997)
37. McKay, D. J. C.: Information Theory, Inference and Learning Algorithms. New York: Springer (2000)
38. McLachlan, G., Krishnan, T.: The EM Algorithm and Extension. New York: John Wiley and Sons. (1997)
39. Rabiner, L. R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE 77(2), 257-286. (1989)

40. Rabiner, L., Juang, B. H.: Fundamentals of Speech Recognition. Prentice Hall Signal Processing Series. New Jersey: Prentice Hall. (1993)
41. Seligmann, C.: Uso de Modelos Escondidos de Markov en Biología Molecular Computacional., Poliantea No 9, Bogotá: Politécnico Grancolombiano. (2009)
42. Shinozaki, T., Furui, S.: Hidden mode HMM using bayesian network for modeling speaking rate fluctuation. Proc. of ASRU, (US Virgin Islands), 417-422. (2003)
43. Smyth, P., Heckerman, D., Jordan, M. I.: Probabilistic independence networks for hidden Markov probability models. Neural Computation, 9(2), 227-269. (1997)
44. Song, J., Liu, C., Song, Y., Qu, J., Hura, G. S.: Alignment of multiple proteins with an ensemble of Hidden Markov Models. International Journal of Data Mining and Bioinformatics. Vol 4, No 1, 60-71. (2010)
45. Stroock, D. W.: An Introduction to Markov Processes, Berlin: Springer. (2005)
46. Tusnády, G. E., Simon, I.: Principles Governing Amino Acid Composition of Integral Membrane Proteins: Application to Topology Prediction. J. Mol. Biol. No 283, 489-506. (1998)
47. Viterbi, A. J.: Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. IEEE Transactions on Information Theory, Vol. 13, 260-269. (1967)
48. Werner M. J., Ide, K., Sornette, D.: Earthquake Forecasting Based on Data Assimilation: Sequential Monte Carlo Methods for Renewal Processes. (2009)
49. Wu, C. EJ.: On the convergence properties of the EM algorithm. Annals of Statistics, 11, 95-103. (1983).
50. Zhang, J., Ghahramani, Z., Yang, Y.: Learning Multiple Related Tasks using Latent Independent Component Analysis. Proceedings of NIPS 2005, Vancouver, Canada. (2005)