Capítulo 4

Procedimientos clásicos de aprendizaje

Surely it is enough that the likes of you and I at least try to make our small contribution count for something true and worthy.
Kazuo Ishiguro, The remains of the day.

Existe una larga tradición de aplicación de los métodos estocásticos al aprendizaje. En este capítulo revisaremos aquellos que presentan una relación más estrecha con los algoritmos que vamos a desarrollar. En particular, los modelos de Markov ocultos, entrenados frecuentemente con el método de Baum-Welch, y los métodos bayesianos que determinan automáticamente el número de estados del modelo mediante mecanismos de fusión.

4.1 Modelos de Markov ocultos

Los modelos de Markov constituyen modelos probabilísticos para sucesos que ocurren secuencialmente en el tiempo. Por este motivo, han

sido utilizados con frecuencia en problemas como los de reconocimiento del habla (Rabiner 1989) y los de modelización del lenguaje natural (Brown *et al.* 1992, Suen 1979, Stolcke & Segal 1994).

Una *cadena de Markov* es un suceso aleatorio x cuya probabilidad sólo depende de un número predeterminado n de sucesos anteriores (Chung 1967), es decir:

$$\begin{aligned} p(x_t = o_t | x_{t-1} = o_{t-1}, ..., x_1 = o_1) \quad = \\ p(x_t = o_t | x_{t-1} = o_{t-1}, ..., x_{t-n} = o_{t-n}) \end{aligned} \tag{4.1}$$

Este tipo de modelos ha sido utilizado con frecuencia en el tratamiento de lenguaje natural, donde son denominados modelos de n-gramas. En ellos la probabilidad de aparición de una palabra w_k está determinada por las $n-1$ palabras anteriores. Habitualmente, se eligen los modelos basados en bigramas ($n = 2$), trigramas ($n = 3$) o combinaciones de ambos (Charniak 1994). Por ejemplo, en el caso de los trigramas, la probabilidad $p_3(w_k)$ de la palabra w_k viene dada por:

$$p_3(w_k) = p(w_k | w_{k-2} w_{k-1}) \tag{4.2}$$

Para que las probabilidades de w_1 y w_2 puedan ser determinadas mediante la fórmula anterior, es necesario suponer que el texto viene precedido de unas pseudo-palabras w_{-1} y w_0. No es este el único caso en el que la falta de datos puede originar problemas: normalmente los datos están muy dispersos y hasta el 25% de los trigramas en un texto pueden no haber aparecido durante la etapa de entrenamiento (Charniak 94). Por todos estos motivos, frecuentemente se toma una combinación de modelos del tipo:

$$\hat{p}(w_k) = \lambda_1 p_1(w_k) + \lambda_2 p_2(w_k) + \lambda_3 p_3(w_k) \tag{4.3}$$

donde los parámetros λ_n deben ser elegidos automáticamente y pueden depender de la cantidad de cada tipo de n-gramas observados en la fase de entrenamiento. La interpolación entre los modelos es una cuestión aún controvertida (Kneser & Ney 1995).

Una forma de soslayar este tipo de dificultades es utilizar modelos de Markov ocultos, una generalización de las cadenas de Markov. En ellos, dada una observación previa, más de un estado (en el sentido de conjuntos distintos de probabilidades para las palabras siguientes) es posible y de ahí el nombre de modelos ocultos. Formalmente, un *modelo de Markov oculto* es una cuádrupla $M = (Q, I, \mathcal{A}, T)$ compuesta por:

- un conjunto finito de estados que denotaremos mediante su número de orden $Q = \{1, 2, ..., N\}$;
- un estado $I \in Q$ que es marcado como estado inicial;
- un conjunto finito $\mathcal{A}$ de símbolos de salida;
- un conjunto T de transiciones.

Una transición $t = (i, j, a, p_{ij}(a)) \in T$ contiene un estado i de partida, un estado j de llegada, un símbolo $a \in \mathcal{A}$ y una probabilidad $p_{ij}(a)$ de que a partir del estado i se efectúe una transición al estado j y se genere un símbolo a (si utilizamos el modelo para asignar una probabilidad a cada cadena, diremos más bien que se procesa el símbolo a de la cadena de entrada). Si bien no se admiten transiciones redundantes (es decir, transiciones con el mismo valor para los estados y el símbolo), no hay, en cambio, ninguna restricción sobre el número de transiciones que con el mismo símbolo a parten de un estado dado i. Con frecuencia, se parte de la suposición de que existen transiciones desde cada estado y con cada símbolo a todos los estados, aunque la probabilidad de algunas de ellas puede ser cero posteriormente. En este sentido, los modelos ocultos de Markov son más generales que los autómatas finitos deterministas estocásticos definidos anteriormente, y se corresponden más bien con los autómatas indeterministas estocásticos[1] una clase de

[1] Esto es, autómatas finitos indeterministas (Hopcroft & Ullman 1979) que incorporan una función de probabilidad de transición al estilo de la definida en la sección 2.4.

autómatas que no se puede reducir a la de los autómatas deterministas estocásticos, aunque la incluye propiamente.

La probabilidad de generación de una cadena $w = a_0 a_2 ... a_{L-1}$ de longitud L por el modelo M es

$$p(w|M) = \sum_{c(I,L)} \prod_{k=0}^{L-1} p_{i_k i_{k+1}}(a_k) \tag{4.4}$$

donde $c(I, L)$ representa todas las secuencias $i_0 i_2 ... i_L$ que describen caminos de longitud L entre los estados de Q que comienzan en $i_0 = I$. Estas probabilidades, así como el camino más probable para una cadena dada, pueden ser computadas de forma eficiente utilizando el algoritmo de Viterbi (1967).

Los modelos de Markov ocultos pueden ser entrenados utilizando el algoritmo de Baum y Welch (Baum 1972). Este procedimiento recalcula iterativamente las probabilidades de transición que efectúa el modelo M al analizar una cadena $w = a_0 a_2 ... a_{L-1}$. Para describir este algoritmo necesitamos definir previamente las siguientes magnitudes:

1. La *probabilidad hacia adelante* $\alpha_i(t)$ es la probabilidad de que M genere la subcadena $a_0 a_1 a_2 ... a_{t-1}$ terminando en el estado i, y está definida para $t = 0, 1, ..., L$. Las probabilidades $\alpha_i(t)$ pueden ser calculadas en un tiempo lineal en función de L de la siguiente forma. Para $t = 0$:

   $$\alpha_i(t = 0) = \delta_{iI}, \tag{4.5}$$

   y procediendo iterativamente, para $t = 1, ...L$:

   $$\alpha_i(t) = \sum_j \alpha_j(t-1) p_{ji}(a_{t-1}). \tag{4.6}$$

   Por ejemplo, $\alpha_i(t = 1) = \sum_j \delta_{jI} p_{ji}(a_0) = p_{Ii}(a_0)$, tal y como cabe esperar.

2. La *probabilidad hacia atrás* $\beta_i(t)$ se define como la probabilidad de que M genere la cadena $a_t a_{t+1} ... a_{L-1}$ partiendo del estado

i. Estas probabilidades pueden calcularse también en tiempo lineal, pues para $t = L$

$$\beta_i(t = L) = 1, \tag{4.7}$$

y para $t = 0, 1, ..., L - 1$

$$\beta_i(t) = \sum_j \beta_j(t + 1) p_{ij}(a_t). \tag{4.8}$$

Por ejemplo, $\beta_i(L - 1) = \sum_j p_{ij}(a_{L-1})$.

3. La probabilidad de que M genere $w = a_0 a_1 ... a_{L-1}$ es:

$$p(w|M) = \beta_I(0), \tag{4.9}$$

y también

$$p(w|M) = \sum_i \alpha_i(L). \tag{4.10}$$

Las dos ecuaciones anteriores no son más que un caso particular de la expresión general:

$$p(w|M) = \sum_i \alpha_i(t)\beta_i(t). \tag{4.11}$$

Si en la expresión anterior sustituimos $\beta_i(t)$ siguiendo (4.8), obtenemos

$$p(w|M) = \sum_{ij} \alpha_i(t) p_{ij}(a_t) \beta_j(t + 1), \tag{4.12}$$

expresión que nos resultará útil más adelante.

4. La probabilidad $\gamma_i(t)$ de que, al generar w, el t-ésimo estado en la secuencia de $L + 1$ estados sea i es proporcional a $\alpha_i(t)\beta_i(t)$. Teniendo en cuenta (4.11) para la normalización, obtenemos:

$$\gamma_i(t) = \frac{1}{p(w|M)} \alpha_i(t)\beta_i(t). \tag{4.13}$$

5. La probabilidad $\xi_{ij}(t)$ de que, al generar w, la transición causada por a_t ocurra entre los estados i y j es proporcional a $\alpha_i(t)p_{ij}(a_t)\beta_j(t+1)$. De nuevo, teniendo en cuenta (4.12) a efectos de normalización, obtenemos:

$$\xi_{ij}(t) = \frac{1}{p(w|M)}\alpha_i(t)p_{ij}(a_t)\beta_j(t+1) \tag{4.14}$$

Con estas definiciones, resulta sencillo describir el algoritmo de entrenamiento de Baum y Welch. En este procedimiento se recalculan iterativamente las probabilidades de transición de la siguiente forma:

$$p'_{ij}(a) = \frac{\sum_{t=0}^{L-1}\xi_{ij}(t)\delta_{a_t a}}{\sum_{t=0}^{L-1}\gamma_i(t)} \tag{4.15}$$

hasta que se llegue a las proximidades de un punto fijo. En el caso de que se disponga de una muestra $S = \{w_1, w_2, ..., w_n\}$ para el entrenamiento compuesta de numerosas cadenas, los sumatorios, tanto en el numerador como en el denominador de la ecuación (4.15), deben sumar también sobre las diferentes cadenas w_k de la muestra.

Este procedimiento garantiza que las nuevas probabilidades mejoran la verosimilitud de la muestra, $p(S|M)$, con lo que finalmente debe alcanzarse un mínimo. No existe garantía sin embargo, de que este mínimo sea global y no local. Esto constituye la mayor dificultad del procedimiento, pues los experimentos prueban que su buen rendimiento requiere una estimación del tamaño del modelo que se debe inferir (al menos, de su tamaño máximo) y alguna información sobre la estructura de este modelo para evitar que los valores iniciales de las probabilidades $p_{ij}(a)$ sean elegidos meramente al azar. Nótese que, si existe más de una representación posible del problema, el método de Baum-Welch no tiene ninguna razón para optar por la representación más sencilla. Otra dificultad importante en el aprendizaje de lenguajes estocásticos ha sido puesta de manifiesto por Abe y Warmuth (1992): tanto los autómatas probabilísticos como los modelos ocultos de Markov no pueden ser entrenados en un tiempo polinómico en función del tamaño del alfabeto $\mathcal{A}$.

Por último, merece la pena destacar que en los modelos ocultos, a diferencia de las cadenas de Markov, la probabilidad de un suceso puede depender de símbolos anteriores muy alejados en el tiempo. Por ejemplo, una cadena de Markov no puede describir un comportamiento tan sencillo como el siguiente:

$$p(w_k) = \begin{cases} p_1 & \text{si } k \text{ es par} \\ p_2 & \text{en caso contrario} \end{cases} \tag{4.16}$$

Este tipo de comportamientos, sin embargo, se puede describir adecuadamente utilizando los modelos de Markov ocultos o modelos basados en autómatas finitos estocásticos como los descritos en el capítulo 2.

4.2 Modelos bayesianos y fusión de estados

Supongamos que existe $\{M_0, M_1, M_2, ...\}$, un espacio de hipótesis o modelos mutuamente excluyentes (como mucho uno es correcto) y exhaustivas (al menos uno es correcto). Dado un conjunto de datos observado X, existen, al menos, dos criterios basados en la teoría de probabilidades para elegir la mejor hipótesis:

- verosimilitud máxima (VM), esto es, elegir el modelo M tal que la probabilidad $p(X|M)$ es máxima;
- probabilidad *a posteriori* máxima (PAM), es decir, elegir el modelo M que hace máxima la probabilidad $p(M|X)$.

Para estudiar la relación entre ambos criterios, supongamos que existe alguna forma de asignar una probabilidad independiente de los datos experimentales a cada modelo M. Esta distribución $p(M)$ es lo que se denomina la probabilidad *a priori*. Entonces, la probabilidad de que M sea el modelo correcto, dado que como resultado de un experimento hemos obtenido la muestra finita X, es

$$p(M|X) = \frac{p(X|M)p(M)}{p(X)} \tag{4.17}$$

expresión que se conoce con el nombre de *teorema de Bayes* y que es consecuencia directa de la definición de probabilidad condicionada

$$p(X|Y) = \frac{p(X \cap Y)}{p(Y)} \; . \tag{4.18}$$

Dado que el denominador $p(X)$ no depende del modelo elegido, el criterio bayesiano de inferencia (PAM) es una generalización del criterio de verosimilitud máxima (VM). En particular, éste último se corresponde con el caso en que la probabilidad $p(M)$ es constante, es decir, todos los modelos M son *a priori* equiprobables.

Las probabilidades *a priori* $p(M)$ permiten introducir de forma natural en el proceso de aprendizaje el grado de creencia asociado a los diferentes modelos. De esta manera se puede incorporar, por ejemplo, la preferencia por los modelos más sencillos. En efecto, tomando logaritmos en la expresión (4.17), se observa que maximizar $p(M|X)$ es equivalente a minimizar

$$-\log p(M|X) = -\log p(M) - \log p(X|M) \tag{4.19}$$

procedimiento que puede interpretarse como la minimización de la longitud de la descripción de los datos X conjuntamente con la del modelo de codificación M. De hecho $-\log p(M)$ es la longitud óptima de codificación del modelo dada la distribución *a priori* mientras que $-\log p(X|M)$ se corresponde con la longitud del código óptimo para los datos X usando M como modelo probabilístico. A la inversa, cualquier esquema de codificación que asigna a M un código de longitud $\lambda(M)$ puede ser utilizado para generar una distribución *a priori* de los modelos:

$$p(M) \propto \exp^{-\lambda(M)} \tag{4.20}$$

Por ejemplo, la forma más natural de codificar un modelo de Markov es simplemente enumerar sus transiciones, por lo que podemos tomar $\lambda(M)$ como el número de transiciones del modelo.

En el caso de los autómatas finitos estocásticos (o modelos de Markov ocultos), la verosimilitud máxima se consigue cuando el modelo

predice como probabilidad para cada cadena la frecuencia relativa observada en el experimento. En efecto, si el número de cadenas de tipo w en X es n_w, y la probabilidad asignada por M a w es p_w, hay que minimizar

$$-\log p(X|M) \propto -\sum_w n_w \log p_w \tag{4.21}$$

sometido a la restricción $\sum_w p_w = 1$, lo que corresponde a una minimización sobre una variedad lineal. Utilizando un multiplicador de Lagrange $\lambda \sum_w \Delta p_w = 0$, la variación de $p(X|M)$ cerca del mínimo es

$$\Delta \log p(X|M) = \sum_w (\frac{n_w}{p_w} - \lambda)\Delta p_w = 0 \tag{4.22}$$

donde, gracias al multiplicador de Lagrange, todas las variaciones Δp_w son independientes, lo que implica que cada término se anula, y por lo tanto

$$p_w = \frac{n_w}{\lambda} \, . \tag{4.23}$$

Es decir, cada probabilidad es proporcional al número de observaciones de esa cadena, y λ es el número total de cadenas en la muestra para que la normalización de la probabilidad sea la correcta.

Por tanto, la verosimilitud máxima se consigue tomando como modelo un lenguaje finito que puede ser generado, por ejemplo, mediante un autómata con estructura de árbol en el que cada nodo corresponde a uno de los prefijos observados (véase la figura 4.1) y las probabilidades de transición se eligen iguales a las frecuencias relativas en la muestra de los símbolos que siguen a cada prefijo. Modelos más pequeños (es decir, autómatas con un número menor de estados) conducen necesariamente a valores menores de la verosimilitud. A pesar de ello, esta disminución puede resultar compensada por la predisposición hacia los modelos más sencillos contenida en las probabilidades *a priori*. De esta forma es posible optar por modelos más sencillos aunque sean de verosimilitud menor.

Un algoritmo para la inferencia de modelos de Markov ocultos basado en la fusión de estados y la utilización de probabilidades *a priori*

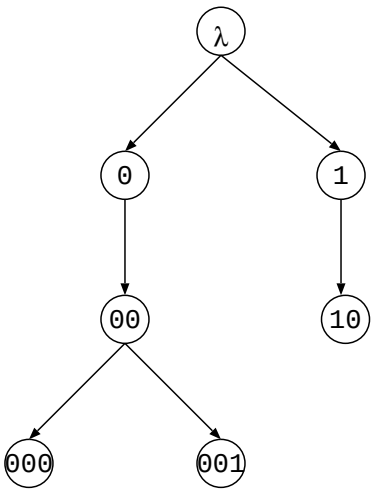

Figura 4.1: Árbol de prefijos para el conjunto $\{0, 10, 000, 001\}$

ha sido propuesto por Stolcke y Omohundro (1993). Su algoritmo puede describirse de la siguiente manera:

1. construir el modelo M_0 de verosimilitud máxima para el conjunto de datos X;
2. reiterar desde $i = 0$:
   (a) determinar el conjunto K de $|M_i|(|M_i|-1)$ posibles fusiones de estados en M_i;
   (b) para cada $k \in K$ calcular el nuevo modelo $k(M)$ y su probabilidad a posteriori $p(k(M_i)|X)$;
   (c) elegir la fusión k^* que maximiza $p(k(M_i)|X)$ y tomar como $M_{i+1} = k^*(M_i)$.
   (d) terminar si $p(M_i + 1|X) < p(M_i|X)$. En caso contrario continuar con $i = i + 1$.

Con este algoritmo, la disminución de verosimilitud provocada por la fusión de estados del modelo queda compensada por la preferencia expresada por la probabilidad *a priori* por los modelos más sencillos.

El algoritmo introduce también una probabilidad *a priori* para los parámetros del modelo (las probabilidades de transición) en forma de una distribución de Dirichlet. Esto es equivalente a admitir un conjunto de ejemplos virtuales para cada transición que se añaden a los reales a la hora de calcular los parámetros más probables. Los resultados descritos en Stolcke & Omohundro (1994) demuestran que el procedimiento proporciona buenos resultados para modelos sencillos y muestras pequeñas, comparables o mejores a los obtenidos con un entrenamiento del tipo Baum-Welch. El procedimiento no ha sido estudiado con muestras grandes y autómatas de tamaño grande debido a su elevado coste: nótese que cada iteración requiere $|M_i|^2$ evaluaciones de la verosimilitud y que $|M_i|$ puede ser lineal con el tamaño de X. Una diferencia fundamental entre este método y el que proponemos en el capítulo siguiente es que mientras el prime método no garantiza la identificación en el límite del lenguaje correcto (es decir, del conjunto de cadenas con probabilidad no nula), el nuevo método sí que presenta esta propiedad.