Revista Ciencia en Desarrollo. Vol. 2 Num. 15. pp. 212 - 217
julio - diciembre 2024

https://doi.org/10.19053/01217488.v15.n2.2024.15663

Artículo de revisión

Modelo oculto de Markov la piedra angular de la proteómica moderna

Hidden Markov model the cornerstone of modern proteomics

Isabel Cristina Castellanos Cuellar[1] ✉

[1]MS. Ciencias Bioquímica, Departamento Ciencias Básicas, Facultad de Ingeniería, Universidad EAN, Bogotá Colombia.

Recepción: 22-feb-2023 **Aceptado:** 09-marzo-2024 **Publicado:** 23-jul-2024
Cómo citar: Castellanos, I. C. (2024). Modelo Oculto De Markov La Piedra Angular De La Proteómica Moderna. Ciencia En Desarrollo, 15(2). https://doi.org/10.19053/01217488.v15.n2.2024.15663

Resumen

El modelo oculto de Markov (HMM) se ha convertido en una de las herramientas más utilizadas en el análisis de secuencias biológicas, ya que proporcionan un sólido marco matemático para modelar y analizar secuencias biológicas. En este documento, presentamos una revisión del concepto básico de los HMM y cómo es posible usar de manera efectiva el HMM para la representación de secuencias biológicas en la identificación de secuencias de proteínas evolutivamente distantes.

Palabras Clave: Modelo oculto de Markov (HMM), bioinformática, dominios, proteínas.

Abstract

Hidden Markov Model (HMM) have become one of the most widely used tools in biological sequence analysis, as they provide a robust mathematical framework for the modelling and analysis of biological sequences. In this paper we review the basic concept of HMMs and how it is possible to effectively use HMMs to represent biological sequences when identifying evolutionarily distant protein sequences.

Keywords: Hidden Markov Model (HMM), bioinformatics, domain, proteins.

E-ISSN: 2462-7658

1. Introducción

La proteómica está relacionada con la identificación y cuantificación del contenido total de proteínas presentes en una célula, tejido u organismo a través de la aplicación de tecnologías [1] que nos permiten el estudio de estas biomoléculas y su interacción, en un organismo, sistema, o cualquier otro contexto biológico donde ellas están presentes. El término "proteoma"se refiere a todas las proteínas expresadas por un genoma, por ello con la finalización del Proyecto Genoma Humano a inicios de siglo [2] y el avance acelerado de las técnicas de secuenciación de ADN, la posibilidad de estudiar a gran escala los proteomas desde el punto de vista bioinformático es una oportunidad única para el entendimiento de los sistemas biológicos. La proteómica actualmente abarca interrogantes biológicos que pueden ser estudiados con la identificación y análisis de secuencias de proteínas homólogas; que son aquellas secuencias que comparten en términos evolutivos un ancestro común y pueden ser definidas matemáticamente desde la medida de similaridad de sus secuencias de aminoácidos (estructura primaria) desde bases de datos. No obstante, existen secuencias de proteínas homólogas entre sí, que, aunque realizan una función celular conservada entre organismos; poseen una secuencia de aminoácidos tan divergente, que su identificación en bases de datos por técnicas de alineamientos locales, como herramienta matemática, no es certera.

Para este conjunto de proteínas, el HMM se ha convertido en la piedra angular para el avance de la proteómica moderna. Algunos ejemplos son los estudios realizados por Melo et al, (2008) [3] que analiza la posibilidad de una reproducción de tipo sexual en una eucariota ancestral, o el estudio de la regulación de la degradación y abundancia de las proteínas en los organismos [4] o el análisis de los procesos que potencian y regulan el movimiento en las células, [5] y como estas proteínas interactúan entre sí en los procesos invasivos de patógenos de interés médico [6], o en la identificación de blancos para el diseño de pruebas de detección o tratamientos terapéuticos [7]. Entre otros muchos posibles estudios que han sido posibles en proteómica con la combinación de excelentes herramientas bioinformáticas como las implementaciones del HMM para proteínas de baja similaridad y métodos heurísticos para alineamientos locales para la identificación de proteínas de alta similaridad con tiempos de ejecución viables para las tecnologías computacionales actuales. Dentro de los métodos heurísticos para alineamientos locales BLAST *(Basic Local Alignment Search Tool)* es; una de las alternativas actuales más destacadas [8], por su alto rendimiento para tratamiento masivo de datos y la identificación certera de secuencias de proteínas con altos porcentajes de identidad entre homólogos.

Las aplicaciones informáticas del Modelo Oculto de Markov (HMM) no solo cuentan con la capacidad de detectar y categorizar secuencias completas de proteínas que están vinculadas evolutivamente y muestran similitudes en niveles bajos, sino que también pueden identificar los dominios proteicos. Estos dominios, considerados como las unidades fundamentales de las proteínas según el campo de la Proteómica, presentan estructuras tridimensionales específicas que les permiten operar y evolucionar de manera independiente con respecto al resto de la cadena de aminoácidos. En otras palabras, los dominios son secuencias codificadas que persisten en diversos contextos genéticos y han sido preservadas a lo largo de la evolución en términos de secuencias de aminoácidos. La capacidad de las implementaciones HMM de discernir estos dominios brinda una perspectiva invaluable para comprender mejor la relación entre la estructura y la función de las proteínas y su evolución a lo largo del tiempo.

Bajo este panorama, esta revisión plantea los aspectos generales del fundamento de la identificación de secuencias de proteínas homólogas: El HMM para la identificación de proteínas desde bases de datos

2. Alineamiento local de secuencias de proteínas: BLAST

Las secuencias de proteínas homólogas son aquellas que comparten en términos evolutivos un ancestro común y pueden ser definidas matemáticamente desde la medida de similaridad en sus secuencias. BLAST [9] es un algoritmo diseñado para identificar coincidencias iniciales entre secuencias biológicas en una base de datos. BLAST fue lanzada en 1990, y desde ese momento el artículo que la describía se convirtió en uno de los más citados de la ciencia [10]. La razón: BLAST es un *software Front End* (disponible también con entornos gráficos) y disponible *on—line*, aspectos que resultan atractivos para muchos usuarios, se ha demostrado como una herramienta de investigación certera para el encuentro de secuencias homólogas con identidades mayores al 25 %, optimizada para acelerar el tiempo de respuesta de búsquedas de secuencias.

BLAST es una heurística [9], y como tal, es capaz de encontrar la mayoría de las coincidencias buscadas, aunque puede pasar por alto algunas (falsos negativos) o informar otras (falsos positivos), es capaz de encontrar coincidencias iniciales que luego se amplían con un algoritmo determinista, que como tal, puede encontrar exactamente el conjunto de aciertos en la consulta que de forma aproximada coincide dentro de un umbral especificado al unir métodos heurísticos con modelos deterministas BLAST permite que sus usuarios cuenten con un rendimiento en tiempos de ejecución superior. Para simplificar la búsqueda, BLAST, antes de comenzar, se realiza la partición de la secuencia consulta en palabras superpuestas de longitud k (k-mers) y genera un índice k-mer. Para iniciar la búsqueda de cada palabra en el vecindario en una tabla hash para encontrar la ubicación en la base de datos donde aparece cada palabra para la construcción de la colección de semillas, S. Las semillas desde la colección S se extienden hasta que la puntuación de la alineación descienda por debajo de algún umbral X. para finalmente reportar las coincidencias con las puntuaciones más altas.

Pese a todas las ventajas expuestas, BLAST es una herramienta poco eficiente para la identificación de secuencias con bajos porcentajes de identidad con sus homólogos. El *software* BLAST es desarrollado implementando el algoritmo de *Smith-Waterman* [8] el cual basa su uso en algoritmos de programación dinámica [11] para alineamientos locales que determinan los aciertos con respecto a un sistema de puntajes llamado “Matrices de sustitución” que cuantifican la relación de una secuencia con otra.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	B	Z	X	*
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0	-2	-1	0	-4
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3	-1	0	-1	-4
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3	3	0	-1	-4
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3	4	1	-1	-4
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1	-3	-3	-2	-4
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2	0	3	-1	-4
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2	1	4	-1	-4
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3	-1	-2	-1	-4
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3	0	0	-1	-4
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3	-3	-3	-1	-4
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1	-4	-3	-1	-4
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2	0	1	-1	-4
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1	-3	-1	-1	-4
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1	-3	-3	-1	-4
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2	-2	-1	-2	-4
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2	0	0	0	-4
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0	-1	-1	0	-4
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3	-4	-3	-2	-4
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1	-3	-2	-1	-4
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4	-3	-2	-1	-4
B	-2	-1	3	4	-3	0	1	-1	0	-3	-4	0	-3	-3	-2	0	-1	-4	-3	-3	4	1	-1	-4
Z	-1	0	0	1	-3	3	4	-2	0	-3	-3	1	-1	-3	-1	0	-1	-3	-2	-2	1	4	-1	-4
X	0	-1	-1	-1	-2	-1	-1	-1	-1	-1	-1	-1	-1	-1	-2	0	0	-2	-1	-1	-1	-1	-1	-4
*	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	1

Figura 1: Matriz de sustitución BLOSON62. Fuente: [12]

Una matriz de sustitución de aminoácidos figura 1 es la matriz BLOSUM 62 [12]. En términos generales las matrices BLOSUM se obtienen utilizando bloques de secuencias de aminoácidos similares como datos para a través de la aplicación métodos estadísticos a los datos obtener las puntuaciones de similitud como los valores

proporcionales a la tasa a la que un aminoácido *i* cambia a un aminoácido *j* para todos los pares de aminoácidos posibles.

El tipo de matriz usada es determinante para los resultados que se obtendrán ya que cumple la función de asignar una puntuación a cada residuo emparejado. En principio, el uso de una matriz incorrecta puede llevar a calificar erróneamente los alineamientos y por lo tanto obtener resultados equivocados. No obstante, [13] reportó que la matriz BLOSUM 62 usada por defecto para los cálculos de BLAST contenía errores en el código fuente del software utilizado para crearla, además de diferencias a la descripción exacta del algoritmo descrito por [14] para las matrices BLOSUM. Este caso es digno de mención por tres razones: Primero, estas matrices BLOSUM se usan en infinidad de herramientas en biología computacional; segundo, estos errores pasaron desapercibidos durante 15 años; y tercero, las matrices "erróneas" funcionan mejor que las matrices que se obtienen usando exactamente el algoritmo descrito por Henikoff y Henikoff.

Aun así, es ineludible; tener en cuenta que, BLAST no dispone de matrices de sustitución específicas para cada una de las secuencias y organismos existentes, siendo esta la primera limitante, ya que las posiciones y los residuos específicos no necesariamente tienen los mismos patrones de conservación en diferentes contextos. El hecho de no poder contar con matrices específicas para el análisis de cada conjunto de proteínas ha hecho que soluciones computacionales alternativas sean propuestas en busca de algoritmos inteligentes capaces de aprender a fin de evitar la dependencia de parámetros fijos y generales como las matrices de sustitución. NCBI; por ejemplo, propone el software PSI-BLAST como alternativa [15], es un software capaz de crear nuevas matrices de sustitución a partir de alineamientos múltiples de secuencias obtenidos por búsquedas locales con BLASTp, facilitando el encuentro de secuencias con porcentajes de identidad inferiores al 25 % con respecto a sus ortólogos, sin embargo este software no permite la manipulación estricta por parte del usuario para controlar las secuencias que contribuyen a la construcción de la matriz, lo que puede resultar en encuentros erróneos como lo muestra la figura 2.

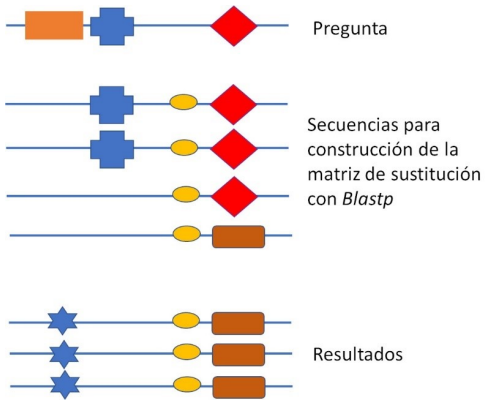

Figura 2: Ejemplo de falsos positivos obtenidos con el software PSI–BLAST.

Otra limitante de los métodos de perfiles calificados por matrices de sustitución es que dependen en gran medida de los parámetros del método. En particular la penalización por *gap*, que corresponde a un valor fijo sin tener en cuenta que un conjunto de secuencias posee regiones muy conservadas (sitios catalíticos) y regiones variables, penalizando la aparición de *gaps* de igual forma en todas las regiones; las sustituciones, inserciones o deleciones en una región conservada deberían idealmente penalizarse más que en regiones variables, de igual forma, algunas clases de sustituciones deberían penalizarse con diferente relación y valor en una posición u otra. Estas limitantes se ven superadas con el uso del modelo estadístico HMM que denota un alineamiento múltiple variable con penalizaciones por *gap* dependientes de la posición; ya que en el las penalizaciones son definidas en términos de probabilidad de acuerdo a la información suministrada (secuencias de entrenamiento).

Los HMM descritos por Andréi Markov en 1906 [16], son una representación estadística extremadamente versátil que se puede utilizar para modelar cualquier conjunto de datos de símbolos discretos unidimensionales. Un HMM describe una serie de observaciones a través de un proceso estocástico "oculto" que cumple con la propiedad de Markov: —Dado un evento i la probabilidad de ocurrencia del siguiente evento j solo depende de la probabilidad condicional $P_{(i|j)}$.

En 1989 Rabiner [17] aplicó este modelo estadístico en técnicas de reconocimiento de voz y solo hasta 1994 Krogh y sus colaboradores [18] aplican el HMM en técnicas bioinformáticas para proteómica realizando una analogía a su aplicación en las técnicas de reconocimiento de voz.

En proteómica el "modelamiento de proteínas" puede ser descrito por el HMM: En esta aplicación las observaciones son representadas por los 20 aminoácidos formadores de una proteína y el modelo HMM es el que por un proceso aleatorio y "oculto" genera las secuencias de aminoácidos, entonces el modelo puede definir una probabilidad de distribución sobre las posibles secuencias de aminoácidos que genere. Un buen modelo de proteínas es aquel que asigna una alta probabilidad de distribución a las secuencias que pertenecen al conjunto de secuencias que modela. En términos biológicos los HMM construidos para familias, dominios o motivos de proteínas describen la estructura primaria de las secuencias con los elementos básicos que caracterizan las moléculas homólogas. Pero, ¿Cómo se construye un HMM para una familia de proteínas?.

3. Arquitectura HMM para proteínas

Consideramos una familia de proteínas con una función celular en común como la actividad deubiquitinadora de las DUBs (*deubiquitinating enzyme*) UCH (*ubiquitin C-terminal hydrolase*): La función deubiquitinadora de la familia de secuencias DUBs UCH puede ser caracterizada en función de una secuencia de posiciones en el espacio *A1....A5* (figura 3) donde se ubican los aminoácidos —indicados en la figura— con su estructura dentro de la cadena polipeptídica, en la parte superior de la figura 3 y como letras en el alineamiento —parte inferior de la figura— obedeciendo a una probabilidad de distribución sobre los 20 aminoácidos para cada posición en la proteína. En términos generales esta es la definición de un perfil [19].

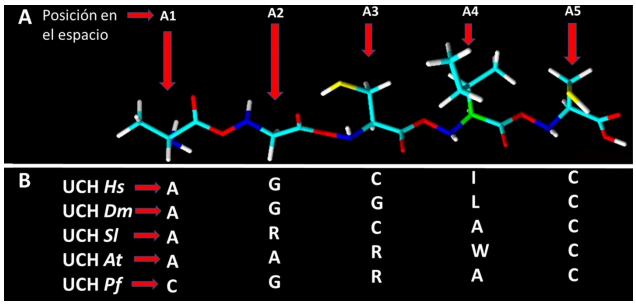

Figura 3: **Sitio catalítico Cys de la enzima UCH. A** *Estructura 3D de fragmento UCH de H. sapienes.* **B** *Alineamiento múltiple de secuencias fragmento UCH. * H. sapiens* (Hs), *D. melanogaster* (Dm), *S. licopersycum* (Sl), *A. thaliana* (At) y *P. falciparum* (Pf).

La estructura de un HMM para una familia de proteínas es en términos generales es similar a la de un perfil de secuencias. La línea principal de un HMM contiene una secuencia de M estados los cuales

son llamados estados de *match*, que corresponden a las posiciones de los aminoácidos en proteína o a las columnas en un alineamiento múltiple, $M = 4$ en el ejemplo de la figura 4. Cada uno de estos estados puede generar una letra x desde un alfabeto de 20 letras (que representan los 20 aminoácidos) de acuerdo a la probabilidad de distribución $P(x|mk)$ donde k es un contador $k = 1....M$. De la notación $P(x|mk)$ se infiere que cada estado de *match* (mk) tiene una probabilidad de distribución distinta.

Por cada estado de *match* (mk) existe un estado de deleción dk que no produce ningún aminoácido y que es un estado alterno para los eventos en los que no hay transición hacia un estado de *match* (mk) , figura 4.

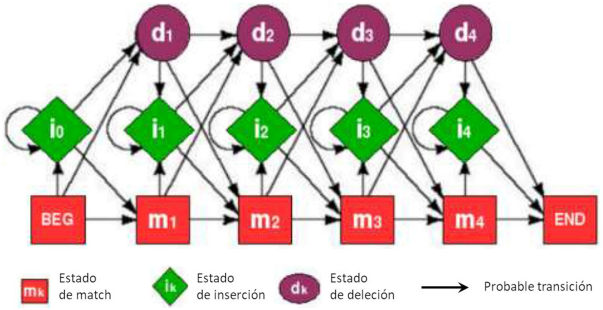

Figura 4: Representación gráfica (grafo) de un perfil HMM para cuatro para secuencias de proteínas.

Existen también $m + 1$ estados de inserción $i_k - 1$ los cuales generan aminoácidos del mismo modo que el estado de match pero con una distribución de probabilidad asociada $P(x|i_k)$. Finalmente, por convención se adiciona un estado de inicio "begin" y un estado final "end" denotados en la figura 4 como BEG y END los cuales no producen ningún aminoácido. Desde cada uno de estos estados, son posibles las transiciones indicadas con flechas a otros estados. Las transiciones entre estados de match o deleción siempre se mueven hacia adelante en el modelo mientras que las transiciones desde un estado de inserción pueden registrar retornos sobre el mismo estado de inserción; en razón a que múltiples inserciones pueden ocurrir en un alineamiento múltiple. La probabilidad de transición desde un estado q a un estado r es denotada como $T(r|q)$.

Con esta estructura del modelo una secuencia puede ser generada evolucionando de estado a estado en el contador k de acuerdo al modelo (figura 4) de la siguiente forma: Se comienza en un estado de inicio (BEG-begin) donde se puede elegir una transición a un estado m_1, d_1 o i_0 aleatoriamente de acuerdo a las probabilidades T(m_1|BEG), T(d_1|BEG) y T (i_0|BEG). Si la transición es hacia m_1 se generará el primer aminoácido x_1 con una probabilidad de distribución $P(x|m_1)$ y una transición al próximo estado acorde con una T ($^*|m_1$) donde * indica el posible próximo estado. Si el próximo estado es el estado de inserción i_1 entonces se generará un aminoácido x_2 acorde a $P(x|i_1)$ y se selecciona el próximo estado de acuerdo a $T(^*|i_1)$. Si el próximo paso es al estado de deleción d_2, no se genera ningún aminoácido y la transición al siguiente estado será de acuerdo a $T(^*|d_2)$. Continuando de esta manera llegaremos al estado final END generando una secuencia de aminoácidos $x_1, x_2, x_3, ...x_i$ por la secuencia de estados $q_0, q_1, q_2, q_3...q_N, q_{N+1}$ de acuerdo al modelo. Donde $q_0 = m_0$ (estado inicial-BEG) y $q_{N+1} = m_{M+1}$ (estado final-END).

La longitud de N es igual a la longitud de la secuencia de proteína ya que los estados de deleción no producen aminoácidos. Si q_i es un estado de match o inserción, se define $l_{(i)}$ como el índice de la secuencia $x_1...x_L$ de los aminoácidos producidos en el estado q_i. Solamente los aminoácidos emitidos por el proceso q son observables, pero no la ruta o secuencia de estados q, de ahí el calificativo de Modelo "Oculto" de Markov, ya que el proceso por el cual son generadas las cadenas de Markov no son visibles al observador. El proceso de generación de una cadena de Markov dentro del HMM pude ser descrito como la probabilidad de la secuencia de eventos $q_0....q_{N+1}$ deducido desde la secuencia $x_1...x_L$.

$$ProbX_1 \ldots X_{L,N+1}|modelo = (T(m_{N+1}|q_N) \times \prod_{(i=1)}^{N} T(q_i|q_{i-1})\ P(x_{l(i)}|q_i) \quad (1)$$

Donde el conjunto de $P(x_{l(i)}|q_i) = 1$ en los casos en que q_i sea un estado de deleción q_k. para una descripción formal del HMM [17]. Esta expresión (Ec. 1) tiene tres limitantes debido a la complejidad de los cálculos:

Primero: Calcular eficientemente la probabilidad de la secuencia de aminoácidos x en un modelo - P(x|*modelo*) de acuerdo la secuencia observada $x = x_1x_2....x_N$; es decir la suma de las probabilidades de todas las posibles trayectorias que podrían producir esa secuencia. La limitante radica que el número de rutas en la arquitectura del HMM es exponencial haciéndolo un cálculo de alta complejidad. Ya que en cada tiempo $k, k = 1, 2...M$ donde se tienen N estados posibles alcanzables son necesarias N^M operaciones para todas las posibles iteraciones. Sin embargo, este cálculo es posible realizarlo con el uso de una técnica de programación dinámica implementando el algoritmo de avance (*for-ward algorithm*) que realiza a una velocidad aceptable este cálculo. El resultado de este cálculo es expresado como -*log*P(*secuencia*|*modelo*), que resume la posibilidad de una secuencia dada en el modelo es decir el score de una secuencia para un modelo dado.

Segundo: Encontrar la trayectoria de estados q más probable $q = (q_1q_2...q_{N+1})$ de acuerdo a un modelo para las observaciones $x = (x_1x_2....x_N)$, ya que el número de cálculos es exponencial -NM- para la determinación de las probabilidades al igual que en el caso anterior. Para este cálculo se usa el algoritmo de retroceso (*backward algorithm*).

Tercero: Maximizar Prob($x_1...x_L, q_0...q_{N+1}$|*modelo*) ajustando la matriz de transición $(A)A = (a_{ij})_{N\times N}$ donde a_{ij} es la probabilidad de transición del estado i al j, el vector de probabilidad de emisión de aminoácidos (B)- uno por cada estado y el vector de probabilidad del estado inicial π logrando con esto que el modelo aprenda; es decir que logre describir de la forma más próxima a la realidad un conjunto de observaciones (el conjunto de secuencias de una familia de proteínas), dominio o motivo de proteínas. Este cálculo se realiza utilizando el algoritmo de *Viterbi* (Para ver una descripción detallada del desarrollo, matemático de cada uno de estos algoritmos [17, 20]).

4. Ventajas del HMM en proteómica

La ventaja de usar HMMs en proteómica es que tienen una base probabilística formal; donde se puede utilizar la teoría de probabilidad para dirigir los parámetros de anotación de una secuencia. Esta base probabilística permite realizar determinaciones que con los métodos heurísticos comunes —descritos anteriormente— no son permitidas. De esta forma el HMM a diferencia de otros métodos tienen en cuenta en la identificación de secuencias los siguientes parámetros [18], que desde el punto de vista biológico son relevantes:

1. Algunas posiciones en la secuencia de aminoácidos de una proteína muestran un alto grado de conservación de residuos específicos —el caso de los sitios catalíticos— mientras otras posiciones pueden mostrar considerables variaciones.

2. En determinadas posiciones de la secuencia algunos aminoácidos pueden no estar presentes en algunas proteínas sin que esto afecte la función de la proteína.

3. Las inserciones de aminoácidos en una secuencia de proteína pueden ser permitidas solo en algunas zonas de la proteína mientras que en otras no. Por esta razón los HMM, llamados perfiles HMM

en bioinformática, son una fuente muy importante de información de las familias de proteínas [18]. Actualmente, y gracias a que los HMM poseen parámetros estadísticos definidos lo que permite que sean construidos computacionalmente, sin necesidad de ninguna intervención o curación manual; se han creado bibliotecas de centenares de perfiles HMM que pueden ser usados en la búsqueda de secuencias no anotadas como aproximación de las relaciones funcionales y/o estructurales de una familia de proteínas.

Una de las bases de datos públicas más robustas de perfiles HMM y constantemente actualizada es la base de datos de PFAM (Protein Family Data Base) [21, 22, 23]. La base de datos de PFAM ha sido construida a partir de alineamientos múltiples de secuencias de familias de proteínas utilizando el software CLUSTALW los cuales han sido usados para la construcción de los perfiles HMM con el paquete de software HMMER [24, 25] los investigadores del Centro Sanger son los realizadores de esta colección [21]. El siguiente paso después de construido el perfil HMM para un conjunto de secuencias (secuencias de entrenamiento), es la búsqueda de secuencias dentro de grandes bases de datos que tengan una alta probabilidad de haber sido generadas por el modelo. Para tal fin es necesario determinar la probabilidad que tiene una secuencia para la trayectoria recorrida a través del modelo para su generación. Sin embargo, para las secuencias que no hacen parte del conjunto de entrenamiento del modelo, es decir para las secuencias que se encuentran en las bases de datos donde se quiere evaluar el HMM, esta trayectoria no se conoce por ello se construye un alineamiento entre la secuencia a evaluar y el modelo HMM —de forma similar a un alineamiento entre dos secuencias— como acercamiento al encuentro de la trayectoria más probable.

Para una secuencia en particular, un alineamiento con el modelo HMM (o con la trayectoria de eventos) se realiza con la asignación de estados para cada residuo en la secuencia, llevándonos un gran número de posibles alineamientos para una secuencia dada [24]. A modo de ejemplo, una secuencia de aminoácidos representada como x_1, x_2, x_3.... y un HMM representado como m_1, m_2, m_3... para los estados de *match* y i_0, i_1, i_2... para los estados de inserción puede tener un alineamiento de la siguiente forma: Un aminoácido x_1 en estado de *match* m_1, dos aminoácidos x_2, x_3 en el estado de inserción i_1, x_4 en el estado de match m_2, x_5 en el estado de match m_6 (después de pasar por tres estados de deleción) y así sucesivamente hasta alinear la cadena completa. Para cada posible alineamiento es decir para cada posible trayectoria de estados para formar la secuencia x_1, x_2, x_3... se debe calcular la probabilidad de la secuencia o el *score* para entonces encontrar el mejor alineamiento, el que tienen el mayor *score* o probabilidad de cumplir el perfil HMM. Aunque son numerosos los alineamientos que puede tener una secuencia que se evalúa con el modelo HMM es posible calcularlos con algoritmos de programación dinámica como el algoritmo de avance *(forward algorithm)*. El *score* calculado por el algoritmo de avance *(forward algorithm)* y normalizado para evitar su dependencia del tamaño de la secuencia encuestada —de forma similar a la función del *score* en BLAST— es útil para clasificar los encuentros resultantes de la búsqueda de secuencias homólogas con un perfil HMM determinado [18].

El modelo HMM es muy eficiente para familias de proteínas con bajos porcentajes de identidad y que contienen características funcionales y/o estructurales definidas [26], ya que nos permite cuantificar la forma —para nosotros "oculta"– en que se relacionan las secuencias de una familia de proteínas, desde sus secuencias de aminoácidos—para nosotros la información conocida—.

En conclusión, el método de Modelos Ocultos de Markov (HMM) emerge como una piedra angular fundamental en el campo de la proteómica debido a su capacidad excepcional para modelar patrones y relaciones en secuencias de proteínas. La versatilidad de los HMMs radica en su habilidad para capturar tanto las características globales como las sutilezas locales en las secuencias, permitiendo la identificación de dominios funcionales, la detección de motivos conservados y la predicción de estructuras secundarias. Su aplicación no solo optimiza la anotación y clasificación de proteínas, la evolución molecular y la función biológica. En última instancia, la adopción y desarrollo continuo de los HMMs promete seguir impulsando avances trascendentales en la investigación proteómica y el entendimiento de los procesos celulares y moleculares.

Referencias

[1] B. Aslam, M. Basit, M. A. Nisar, M. Khurshid, and M. H. Rasool, "Proteomics: Technologies and their applications," Journal of Chromatographic Science, vol. 55, no. 2. 2017. doi: 10.1093/chromsci/bmw167.

[2] D. R. Bentley, "The human genome project - An overview," Medicinal Research Reviews, vol. 20, no. 3. 2000. doi: 10.1002/(sici)1098-1128(200005)20:3<189::aid-med2>3.0.co;2-%23.

[3] S. P. Melo et al., "Transcription of meiotic-like-pathway genes in Giardia intestinalis," Mem Inst Oswaldo Cruz, vol. 103, no. 4, 2008, doi: 10.1590/S0074-02762008000400006.

[4] I. C. Castellanos, E. P. Calvo, and M. Wasserman, "A new gene inventory of the ubiquitin and ubiquitin-like conjugation pathways in giardia intestinalis," Mem Inst Oswaldo Cruz, vol. 115, 2020, doi: 10.1590/0074-02760190242.

[5] D. I. Resnicow, J. C. Deacon, H. M. Warrick, J. A. Spudich, and L. A. Leinwand, "Functional diversity among a family of human skeletal muscle myosin motors," Proc Natl Acad Sci U S A, vol. 107, no. 3, 2010, doi: 10.1073/pnas.0913527107.

[6] P. C. Hernández, L. Morales, I. C. Castellanos, M. Wasserman, and J. Chaparro-Olaya, "Myosin B of Plasmodium falciparum (PfMyoB): in silico prediction of its three-dimensional structure and its possible interaction with MTIP," Parasitol Res, vol. 116, no. 4, 2017, doi: 10.1007/s00436-017-5417-y.

[7] S. Yoodee and V. Thongboonkerd, "Bioinformatics and computational analyses of kidney stone modulatory proteins lead to solid experimental evidence and therapeutic potential," Biomedicine & Pharmacotherapy, vol. 159, p. 114217, 2023.

[8] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," J Mol Biol, vol. 147, no. 1, pp. 195–197, 1981.

[9] G. Myers, "What's Behind Blast," 2013.

[10] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," J Mol Biol, vol. 215, no. 3, 1990, doi: 10.1016/S0022-2836(05)80360-2.

[11] R. Giegerich, "A systematic approach to dynamic programming in bioinformatics," Bioinformatics, vol. 16, no. 8. 2000. doi: 10.1093/bioinformatics/16.8.665.

[12] ncbi, "Entries for the BLOSUM62 matrix at a scale of ln(2)/2.0," ftp://ftp.ncbi.nlm.nih.gov/blast/matrices, Feb. 22, 2023.

[13] M. P. Styczynski, K. L. Jensen, I. Rigoutsos, and G. Stephanopoulos, "BLOSUM62 miscalculations improve search performance," Nature Biotechnology, vol. 26, no. 3. 2008. doi: 10.1038/nbt0308-274.

[14] S. Henikoff and J. G. Henikoff, "Amino acid substitution matrices from protein blocks," Proc Natl Acad Sci U S A, vol. 89, no. 22, 1992, doi: 10.1073/pnas.89.22.10915.

[15] S. F. Altschul et al., "Gapped BLAST and PSI-BLAST: A new generation of protein database search programs," Nucleic Acids Research, vol. 25, no. 17. 1997. doi: 10.1093/nar/25.17.3389.

[16] A. A. Markov, "Rasprostranenie zakona bol'shih chisel na velichiny, zavisyaschie drug ot druga," Izvestiya Fiziko-matematicheskogo obschestva pri Kazanskom universitete, vol. 2-ya seriy, 1906.

[17] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proceedings of the IEEE, vol. 77, no. 2, 1989, doi: 10.1109/5.18626.

[18] A. Krogh, M. Brown, I. S. Mian, K. Sjölander, and D. Haussler, "Hidden Markov Models in computational biology applications to protein modeling," J Mol Biol, vol. 235, no. 5, 1994, doi: 10.1006/jmbi.1994.1104.

[19] M. Gribskov, R. Lothy, and D. Eisenberg, "Profile analysis," Methods Enzymol, vol. 183, no. C, 1990, doi: 10.1016/0076-6879(90)83011-W.

[20] G. D. Forney, "The Viterbi Algorithm," Proceedings of the IEEE, vol. 61, no. 3, 1973, doi: 10.1109/PROC.1973.9030.

[21] R. D. Finn et al., "Pfam: clans, web tools and services.," Nucleic Acids Res, vol. 34, no. Database issue, 2006, doi: 10.1093/nar/gkj149.

[22] S. El-Gebali et al., "The Pfam protein families database in 2019," Nucleic Acids Res, vol. 47, no. D1, 2019, doi: 10.1093/nar/gky995.

[23] J. Mistry et al., "Pfam: The protein families database in 2021," Nucleic Acids Res, vol. 49, no. D1, 2021, doi: 10.1093/nar/gkaa913.

[24] S. R. Eddy, "What is a hidden Markov model?," Nature Biotechnology, vol. 22, no. 10. 2004. doi: 10.1038/nbt1004-1315.

[25] S. C. Potter, A. Luciani, S. R. Eddy, Y. Park, R. Lopez, and R. D. Finn, "HMMER web server: 2018 update," Nucleic Acids Res, vol. 46, no. W1, 2018, doi: 10.1093/nar/gky448.

[26] K. Karplus, C. Barrett, and R. Hughey, "Hidden Markov models for detecting remote protein homologies," Bioinformatics, vol. 14, no. 10, 1998, doi: 10.1093/bioinformatics/14.10.846.