Tenemos una nueva secuencia de ADN, ¿y ahora qué?

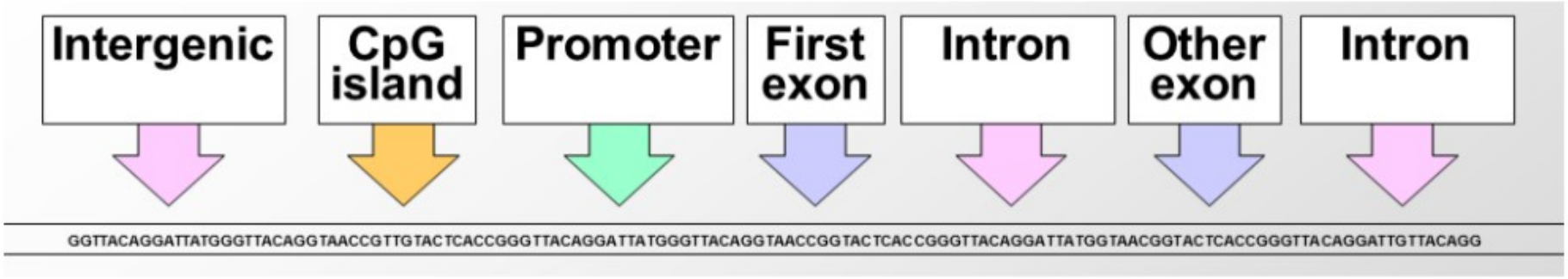
1. Alinearla:
• con cosas que conocemos (búsqueda en bases de datos).
• con cosas desconocidas (ensamblar / agrupar (clustering))
2. Visualizarla: "Regla genómica n. ° 1": ¡Mire sus datos!
• Buscar composiciones de nucleótidos no estándar.
• Busque las frecuencias k-mer (todas las subsecuencias de long. k) que están asociadas con las regiones de codificación de proteínas, los datos recurrentes, las altas en contenido de GC, etc.
• Busque motivos, firmas evolutivas.
• Traducir y buscar marcos de lectura abiertos, stop codones, etc.
• Buscar Patrones y despues desarrollar herramientas para determinar modelos probabilisticos razonables

Tenemos una nueva secuencia de ADN, ¿y ahora qué?

Modelar:

- Hacer una hipótesis.
- Construir un modelo generativo para describir la hipótesis.
- Usar ese modelo para encontrar secuencias de tipo similar.

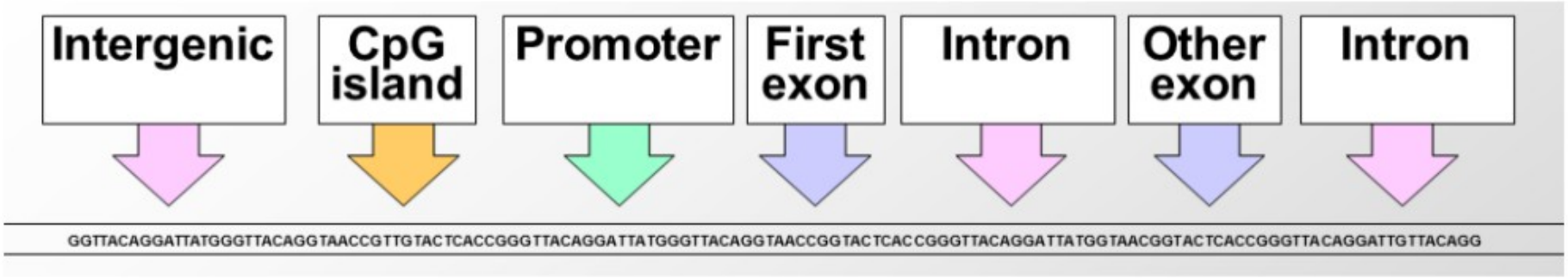
No buscamos secuencias que necesariamente tengan antepasados comunes, sino que nos interesan las que tengan propiedades similares. En realidad, no sabemos cómo modelar genomas completos, pero podemos modelar pequeños aspectos de genomas. La tarea requiere comprender todas las propiedades de las regiones del genoma y computacionalmente, construir modelos generativos para representar hipótesis. Para una secuencia dada, queremos anotar las regiones si son intrones, exones, intergénicos, promotores o regiones clasificables de otro modo.



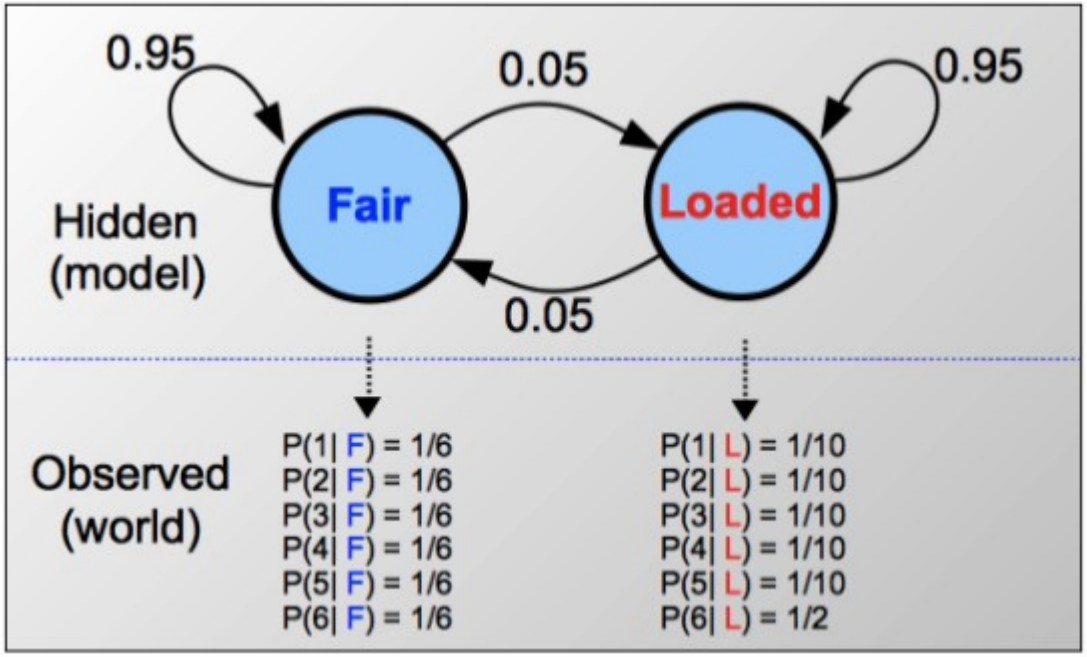
Tenemos una nueva secuencia de ADN, ¿y ahora qué?

¿Por qué el modelado de secuencias probabilísticas?

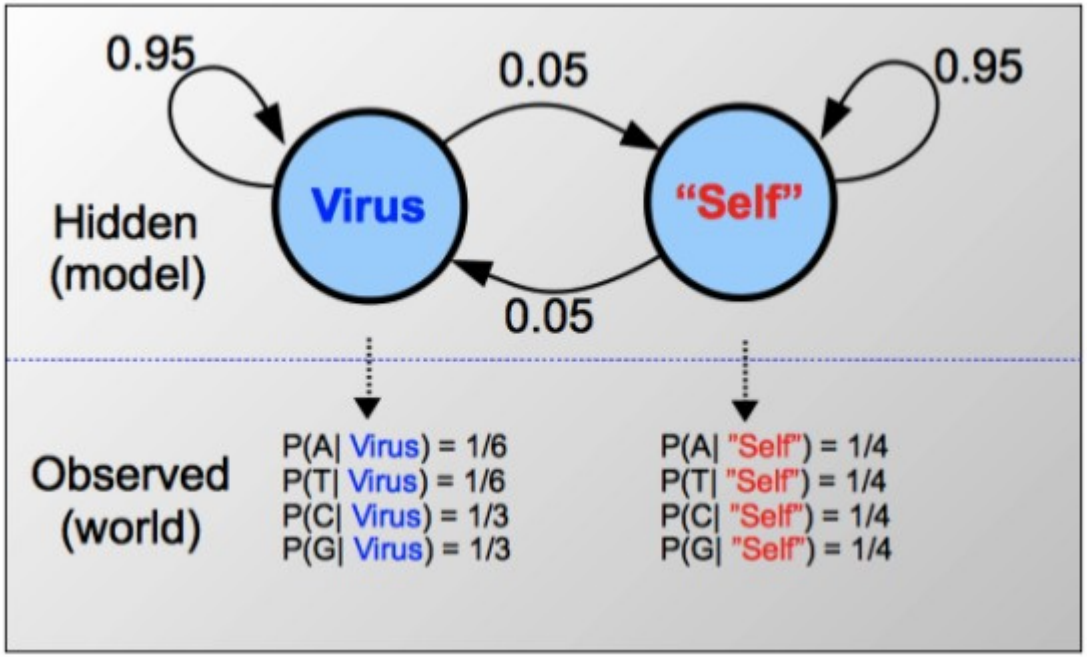
- Los datos biológicos son ruidosos.
- La probabilidad proporciona un cálculo para manipular modelos.
- No se limita a las respuestas sí / no, puede proporcionar grados de certidumbre.
- Muchas herramientas computacionales comunes se basan en modelos probabilísticos.
- Nuestras herramientas: Markov Chains y HMM.

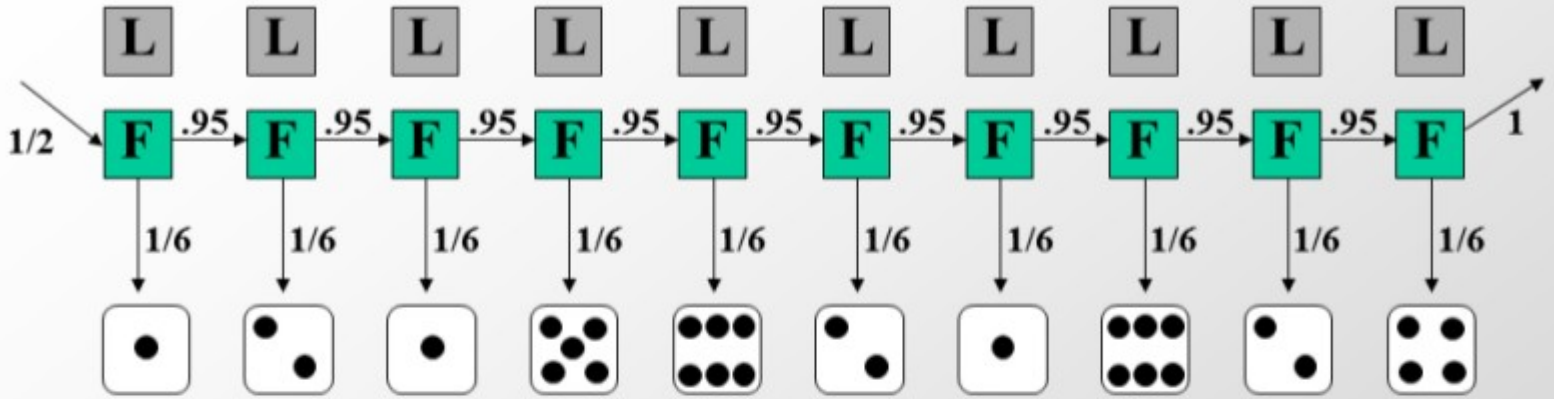


El problema del casino deshonesto



Un modelo biológico similar





Joint probability of observing x and a specific path π, where:

$$\pi = \mathrm{F},\ \mathrm{F},\ \mathrm{F},\ \mathrm{F},\ \mathrm{F},\ \mathrm{F},\ \mathrm{F},\ \mathrm{F},\ \mathrm{F},\ \mathrm{F}$$

$$x = 1,\ 2,\ 1,\ 5,\ 6,\ 2,\ 1,\ 6,\ 2,\ 4$$

Joint probability: $P(x,\pi) = P(x \mid \pi)P(\pi) = P(emissions \mid path) \times P(path)$

$$P = \frac{1}{2} \times \overset{emission}{P(1 \mid F)}\, \overset{transition}{P(F_{i+1} \mid F_i)}\, \overset{emission}{P(2 \mid F)}\, \overset{transition}{P(F \mid F)} \ldots \overset{emission}{P(4 \mid F)}$$

$$= \frac{1}{2} \times \left(\frac{1}{6}\right)^{10} \times (0.95)^{9}$$

$$= 5.2 \times 10^{-9}$$

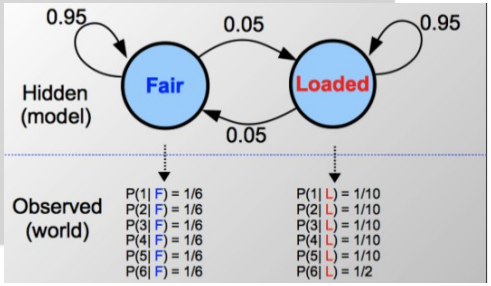
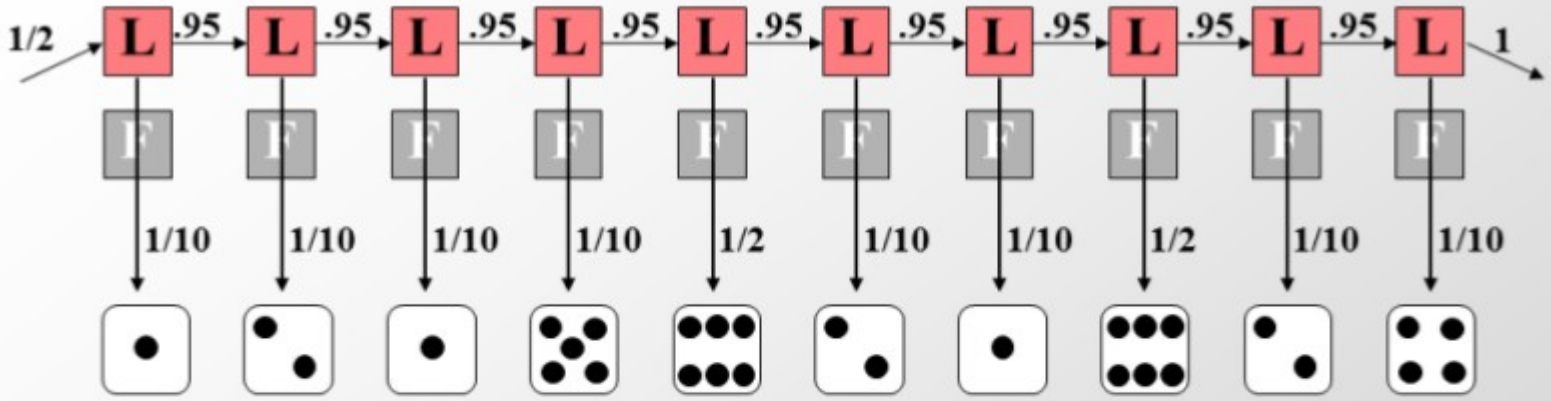


Figure 4: Running the model: probability of a sequence, given path consists of all fair dice



Joint probability of observing x and a specific path π, where:

$$\pi = \mathrm{L},\ \mathrm{L},\ \mathrm{L},\ \mathrm{L},\ \mathrm{L},\ \mathrm{L},\ \mathrm{L},\ \mathrm{L},\ \mathrm{L},\ \mathrm{L}$$

$$x = 1,\ 2,\ 1,\ 5,\ 6,\ 2,\ 1,\ 6,\ 2,\ 4$$

$$P = \frac{1}{2} \times \overset{emission}{P(1 \mid L)}\, \overset{transition}{P(L_{i+1} \mid L_i)}\, \overset{emission}{P(2 \mid L)}\, \overset{transition}{P(L \mid L)} \ldots \overset{emission}{P(4 \mid L)}$$

$$= \frac{1}{2} \times \left(\frac{1}{10}\right)^8 \times \left(\frac{1}{2}\right)^2 \times (0.95)^9$$

$$= 7.9 \times 10^{-10}$$

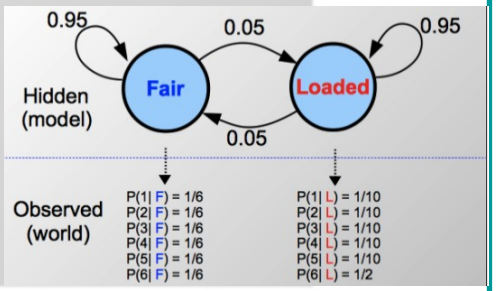
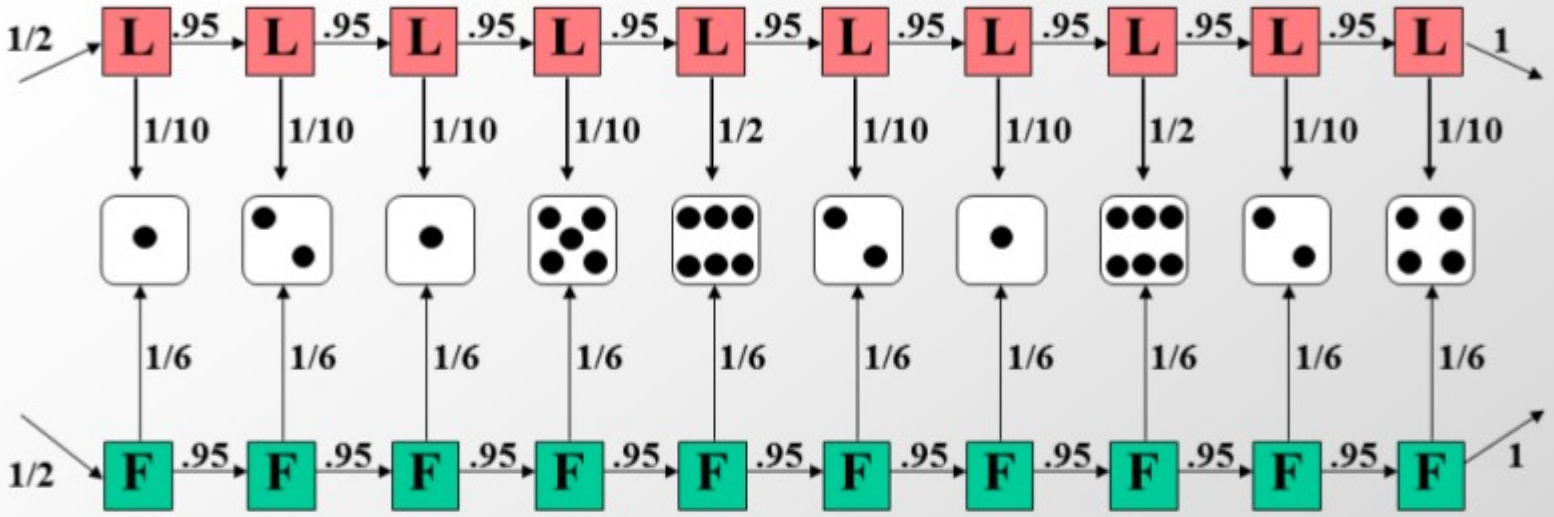


Figure 5: Running the model: probability of a sequence, given path consists of all loaded dice

Comparing the two paths



Two sequence paths:

$$P(x, \text{all - Fair}) = 5.2 \times 10^{-9}$$

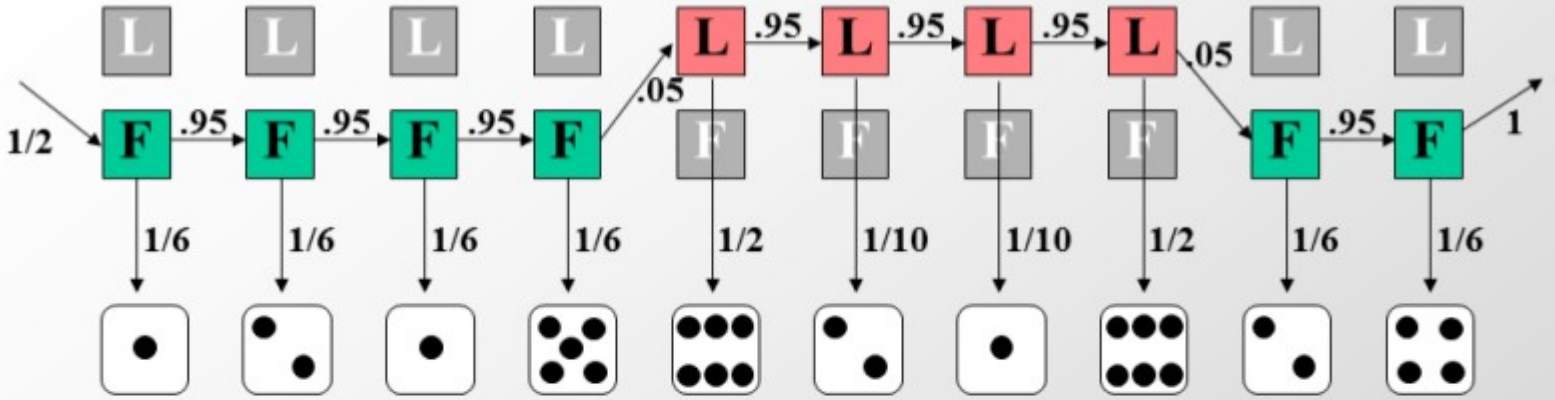
$$P(x, \text{all - Loaded}) = 7.9 \times 10^{-10}$$

Likelihood ratio:

P(x, all-Fair) is 6.58 times more likely than P(x, all-Loaded)

It is 6.58 times more likely that the die is fair all the way, than loaded all the way.

What about partial runs and die switching



What is the likelihood of

$$\pi = \text{F, F, F, F, L, L, L, L, F, F}$$

$$x = 1,\ 2,\ 1,\ 5,\ 6,\ 2,\ 1,\ 6,\ 2,\ 4$$

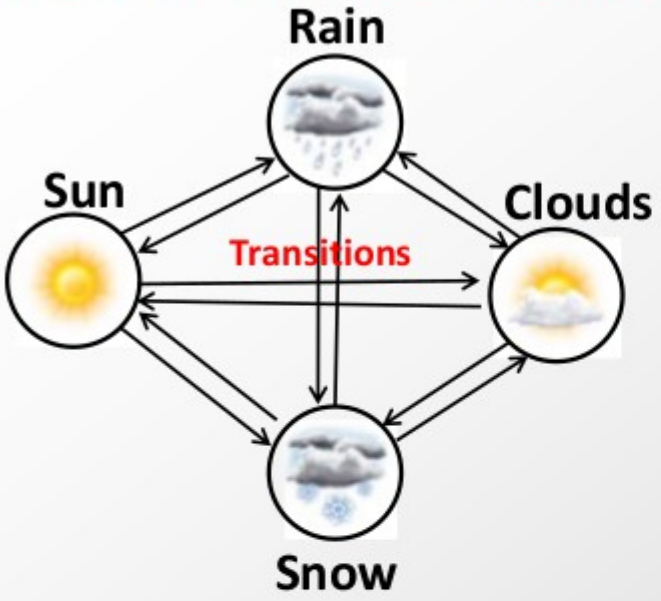
$$P = \frac{1}{2} \times \overset{\text{emission}}{P(1 \mid F)}\, \overset{\text{transition}}{P(F_{i+1} \mid F_i)}\, \overset{\text{emission}}{P(2 \mid F)}\, \overset{\text{transition}}{P(F \mid F)} \ldots \overset{\text{emission}}{P(4 \mid F)}$$

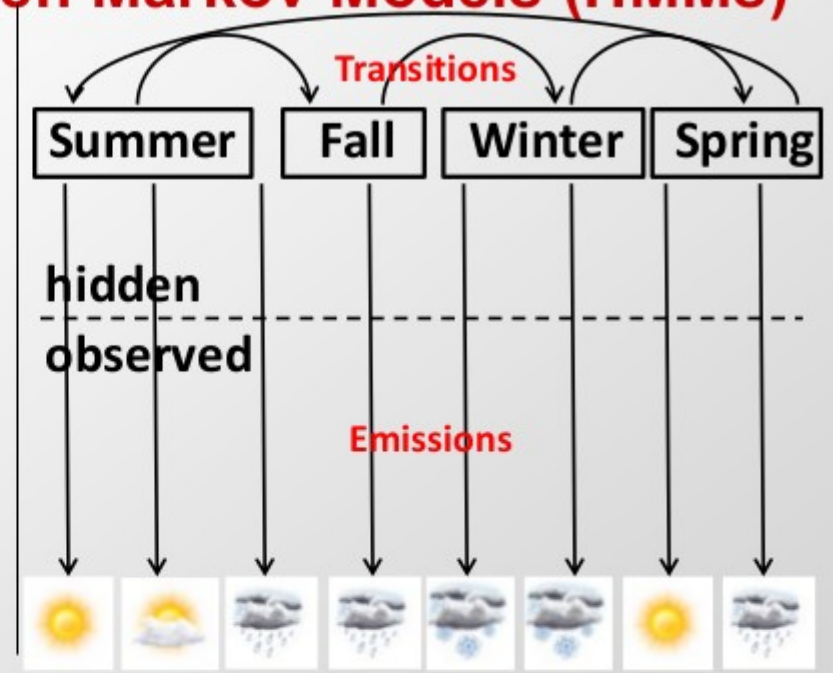
$$= \frac{1}{2} \times \left(\frac{1}{10}\right)^2 \times \left(\frac{1}{2}\right)^2 \times \left(\frac{1}{6}\right)^5 \times (0.95)^7 \times (0.95)^2$$

$$= 2.8 \times 10^{-10}$$

Much less likely, due to high cost of transitions

Markov chains and Hidden Markov Models (HMMs)





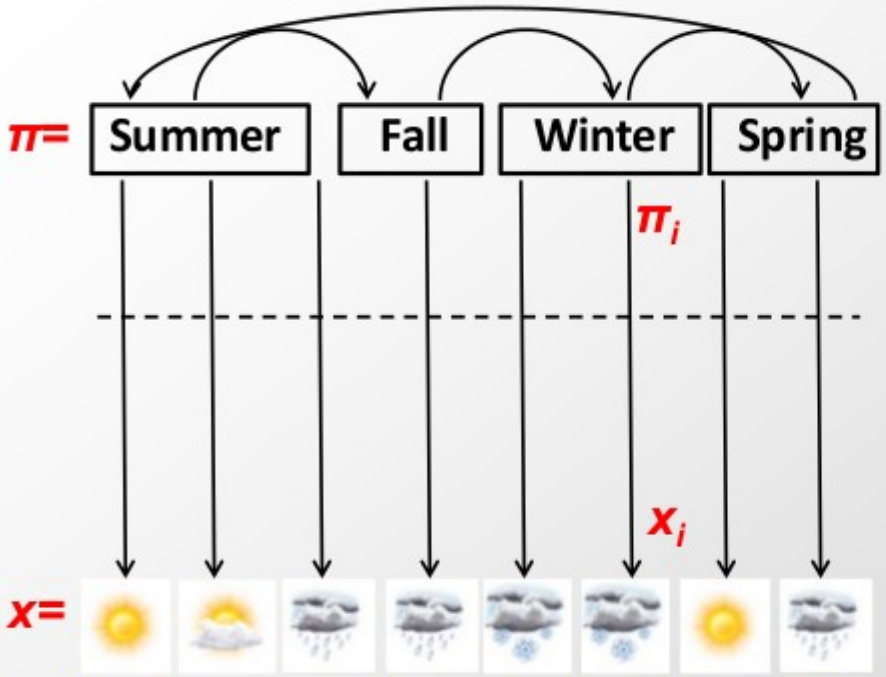
All observed

- Markov Chain
  - Q: states
  - p: initial state probabilities
  - A: transition probabilities
- What you see is what you get: next state only depends on current state (no memory)

- HMM
  - Q: states, p: initial, A: transitions
  - V: observations
  - E: emission probabilities
- Hidden state of the world determines emission probabilities
- State transitions are a Markov chain

5

A Markov chain is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.



Transitions: $a_{kl}=P(\pi_i=l|\pi_{i-1}=k)$

Transition probability from state k to state l

Emissions: $e_k(x_i)=P(x_i|\pi_i=k)$

Emission probability of symbol x_i from state k

- Vector x = Sequence of observations
- Vector π = Hidden path (sequence of hidden states)
- Transition matrix A=a_{kl}=probability of $k \rightarrow l$ state transition
- Emission vector $E=e_k(x_i)$ = prob. of observing x_i from state k
- Bayes's rule: Use $P(x_i|\pi_i=k)$ to estimate $P(\pi_i=k|x_i)$

Examples of HMMs for genome annotation

Application	Detection of GC-rich regions	Detection of conserved regions	Detection of protein-coding exons	Detection of protein-coding conservation	Detection of protein-coding gene structures	Detection of chromatin states
Topology / Transitions	2 states, different nucleotide composition	2 states, different conservation levels	2 states, different tri-nucleotide composition	2 states, different evolutionary signatures	~20 states, different composition/ conservation, specific structure	40 states, different chromatin mark combinations
Hidden States / Annotation	GC-rich / AT-rich	Conserved / non-conserved	Coding exon / non-coding (intron or intergenic)	Coding exon / non-coding (intron or intergenic)	First/last/middle coding exon,UTRs, intron1/2/3, intergenic, *(+/- strand)	Enhancer / promoter / transcribed / repressed / repetitive
Emissions / Observations	Nucleotides	Level of conservation	Triplets of nucleotides	Nucleotide triplets, conservation levels	Codons, nucleotides, splice sites, start/stop codons	Vector of chromatin mark frequencies

The main questions on HMMs

1. Scoring x, one path = Joint probability of a sequence and a path, given the model

- GIVEN a HMM M, a path π, and a sequence x,
- FIND Prob[x, π | M]

➔ "Running the model", simply multiply emission and transition probabilities

➔ Application: "all promoter" vs. "all backgorund" comparisons

2. Scoring x, all paths = total probability of a sequence, summed across all paths

- GIVEN a HMM M, a sequence x
- FIND the total probability P[x | M] summed across all paths

➔ Forward algorithm, sum score over all paths (same result as backward)

SCORING

3. Viterbi decoding = parsing a sequence into the optimal series of hidden states

- GIVEN a HMM M, and a sequence x,
- FIND the sequence π^* of states that maximizes P[x, π | M]

➔ Viterbi algorithm, dynamic programming, max score over all paths, trace pointers find path

4. Posterior decoding = total prob that emission x_i came from state k, across all paths

- GIVEN a HMM M, a sequence x
- FIND the total probability P[π_i = k | x, M)

➔ Posterior decoding: run forward & backward algorithms to & from state π_i =k

PARSING

5. Supervised learning = optimize parameters of a model given training data

- GIVEN a HMM M, with unspecified transition/emission probs., labeled sequence x,
- FIND parameters $\theta = (e_i, a_{ij})$ that maximize P[x | θ]

➔ Simply count frequency of each emission and transition observed in the training data

6. Unsupervised learning = optimize parameters of a model given training data

- GIVEN a HMM M, with unspecified transition/emission probs., unlabeled sequence x,
- FIND parameters $\theta = (e_i, a_{ij})$ that maximize P[x | θ]

➔ Viterbi training: guess parameters, find optimal Viterbi path (#2), update parameters (#5), iterate

➔ Baum-Welch training: guess, sum over all emissions/transitions (#4), update (#5), iterate

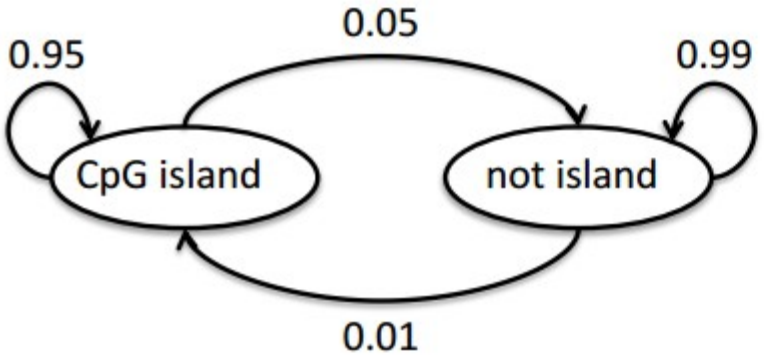
LEARNING

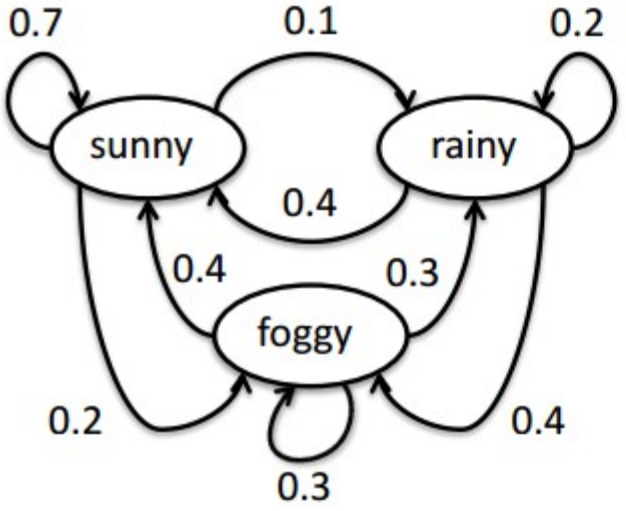
Left: CpG sites at 1/10 nucleotides, constituting a CpG island. The sample is of a gene-promoter, the highlighted ATG consitutes the start codon.

Right: CpG sites present at every 1/100 nucleotides, consituting a more normal example of the genome, or a region of the genome that is commonly methylated.

Graphical Representations

- can be represented graphically by drawing circles for states, and arrows to indicate transitions between states





arrow weights indicate probability of that transition

each hidden state "emits" an observable variable whose distribution depends on the state – what can we actually observe from the CpG island model?

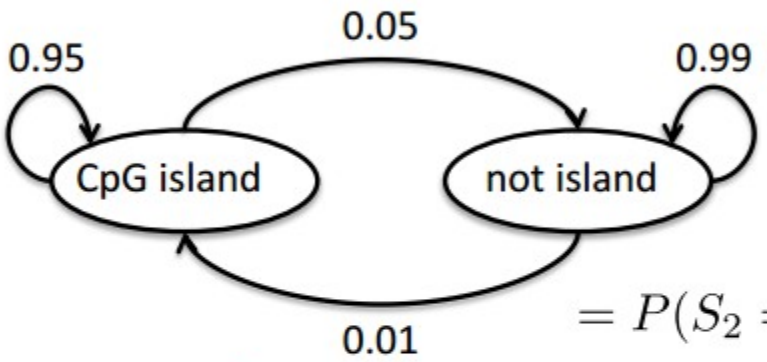
we observe the bases A, T, G, C, where observing a G or C is more likely in a CpG island

what might we observe to infer the state in this "weather" model? (pretend you can't see the weather because you're toiling away in a basement lab with no windows)

we could use whether or not people brought their umbrellas to lab

Graphical Representations

- can be represented graphically by drawing circles for states, and arrows to indicate transitions between states



Given that we're currently in a CpG island, what is the probability that the next two states are CpG island (I) and not island (G), respectively?

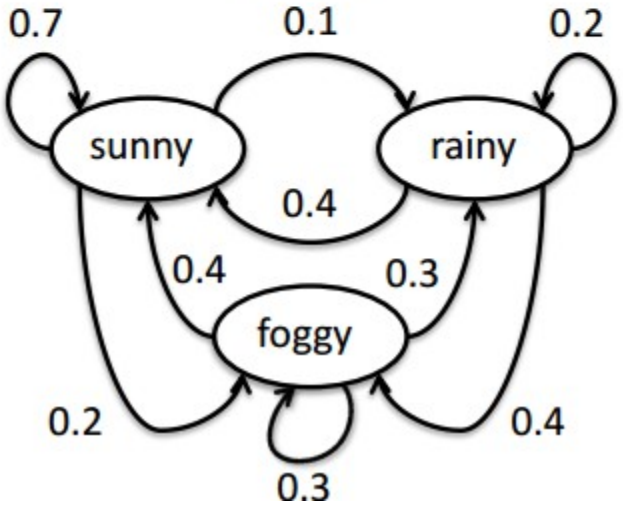
$$P(S_2 = I, S_3 = G | S_1 = I)$$

$$= P(S_2 = I | S_1 = I) * P(S_3 = G | S_1 = I, S_2 = I)$$

markov property →

$$= P(S_2 = I | S_1 = I) * P(S_3 = G | S_2 = I)$$

$$= 0.95 * 0.05 = 0.0475$$



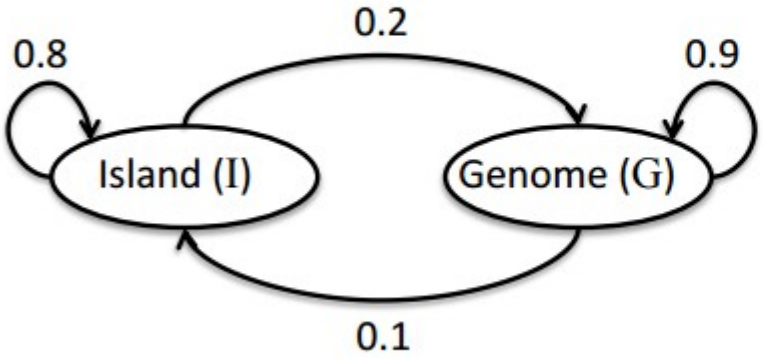
If it's currently rainy, what's the probability that it will be rainy 2 days from now?

$$P(S_3 = R | S_1 = R)$$

Need to sum the probabilities over the 3 possible paths RRR, RSR, RFR:

$$= (0.2)(0.2) + (0.4)(0.1) + (0.4)(0.3) = 0.2$$

HMMs continued



What information do we need in order to fully specify one of these models?

(1) $P_1(S)$ = probability of starting in a particular state S (vector with dimension = # of states)

$$P_1(S) = \begin{matrix} \text{I} & \text{G} \end{matrix} \begin{bmatrix} 0.1 & 0.9 \end{bmatrix}$$

(2) probability of transitioning from one state to another (square matrix w/ each dimension = # of states, usually called the transition matrix, T)

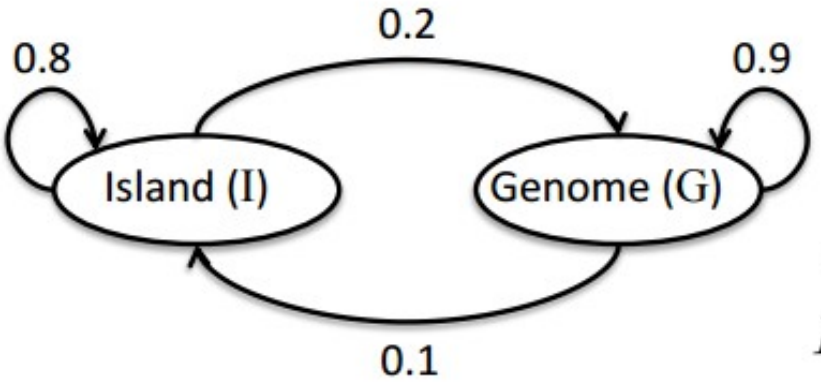
$$T = \begin{matrix} \\ \text{I} \\ \text{G} \end{matrix} \begin{matrix} \begin{matrix} \text{I} & \text{G} \end{matrix} \\ \begin{bmatrix} 0.8 & 0.2 \\ 0.1 & 0.9 \end{bmatrix} \end{matrix}$$

(3) $P_E(X|S)$ = probability of emitting X given current state S

$$P(X|S = I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \end{matrix} \begin{bmatrix} 0.4 & 0.4 & 0.1 & 0.1 \end{bmatrix}$$

$$P(X|S = G) = \begin{bmatrix} 0.1 & 0.1 & 0.4 & 0.4 \end{bmatrix}$$

Using HMMs as generative models



$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ \end{matrix} \begin{bmatrix} 0.1 & 0.9 \end{bmatrix}$$

$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ \end{matrix} \begin{bmatrix} 0.4 & 0.4 & 0.1 & 0.1 \end{bmatrix}$$

$$P(X|S=G) = \begin{bmatrix} 0.1 & 0.1 & 0.4 & 0.4 \end{bmatrix}$$

We want to generate a DNA sequence of length L that could be observed from this model

(1) choose initial state from $P_1(S)$

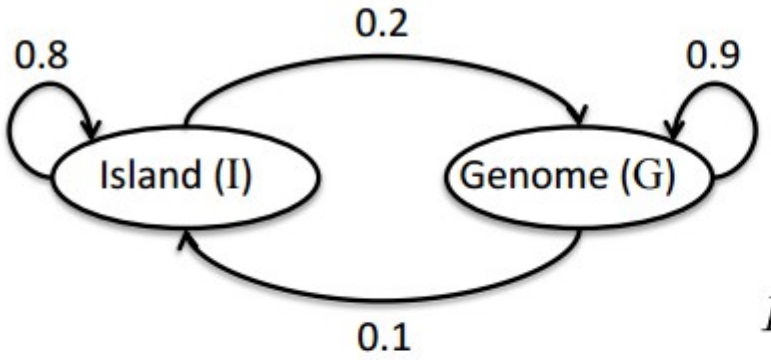
(2) emit first base of sequence according to current state and $P_E(X|S)$

for $1 < i < L$:

(3) choose state at position i according to transition matrix and state at position $i-1$, e.g. using $P_T(S_i|S_{i-1})$

(4) emit base of sequence according to current state S_i and $P_E(X|S_i)$

The Viterbi Algorithm



$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ [0.1 & 0.9] \end{matrix}$$

$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ [0.4 & 0.4 & 0.1 & 0.1] \end{matrix}$$

$$P(X|S=G) = [0.1 \quad 0.1 \quad 0.4 \quad 0.4]$$

Often, we want to infer the most likely sequence of hidden states S for a particular sequence of observed values O (e.g. bases); in other words, find

$S^{opt} = s_1^{opt}, s_2^{opt}, \ldots$ that maximizes $P(S = s_1, \ldots, s_n, O = o_1, \ldots, o_n)$

-what is the optimal parse for the following sequence? GTGCCTA

-we're going to find this recursively, e.g. we find optimal parse of the first two bases GT in terms of paths up to the first base, G

What is the optimal parse for the first base, G?

- if first state is I?

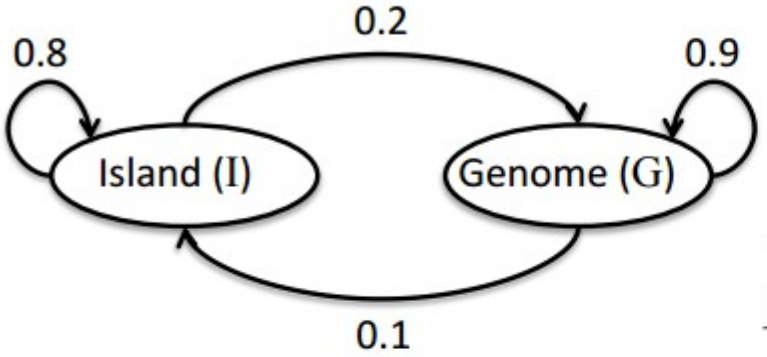
$P(X_1 = G \mid S_1 = I) = P_1(I) * P_E(G \mid S=I) = (0.1)*(0.4) = 0.04$

- if first state is G?

$P(X_1 = G \mid S_1 = G) = P_1(G) * P_E(G \mid S=G) = (0.9)*(0.1) = 0.09$

Therefore, the optimal parse for the first base is state G (note this doesn't yet consider the rest of the sequence!)

Using HMMs as generative models



$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ \end{matrix} \begin{bmatrix} 0.1 & 0.9 \end{bmatrix}$$

$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ \end{matrix} \begin{bmatrix} 0.4 & 0.4 & 0.1 & 0.1 \end{bmatrix}$$

$$P(X|S=G) = \begin{bmatrix} 0.1 & 0.1 & 0.4 & 0.4 \end{bmatrix}$$

What is the most likely parse for the following sequence? GTGCCTA

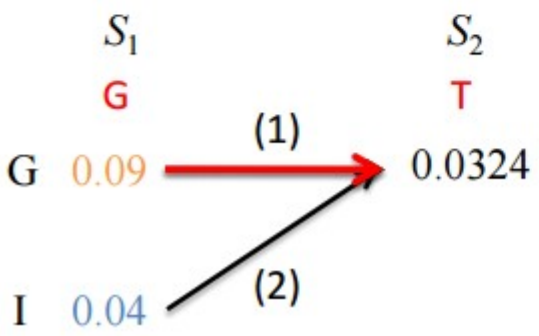
Two possible ways of being in state G in position 2:

prob of optimal sequence of hidden states ending with state G at pos. 1

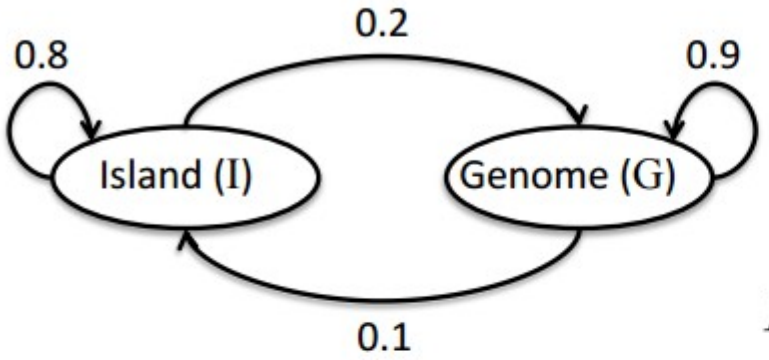
(1) $S_1 = \text{G}$: $P(S_1, S_2, X_1, X_2) = 0.09 * P_T(\text{G} \mid \text{G}) * P_E(\text{T} \mid \text{G})$
$= 0.09 * 0.9 * 0.4 = 0.0324$

prob of optimal sequence of hidden states ending with state I at pos. 1

(2) $S_1 = \text{I}$: $P(S_1, S_2, X_1, X_2) = 0.04 * P_T(\text{G} \mid \text{I}) * P_E(\text{T} \mid \text{G})$
$= 0.04 * 0.2 * 0.4 = 0.0032$



Using HMMs as generative models



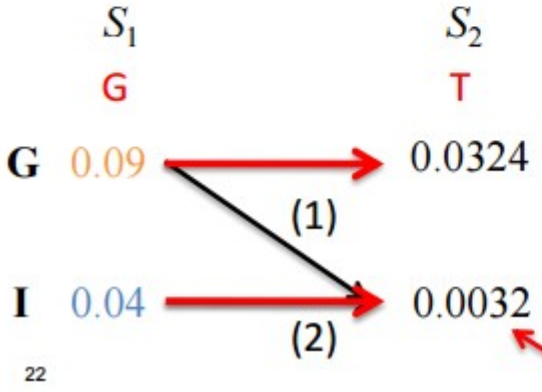
$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ [0.1 & 0.9] \end{matrix}$$

$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ [0.4 & 0.4 & 0.1 & 0.1] \end{matrix}$$

$$P(X|S=G) = [0.1 \quad 0.1 \quad 0.4 \quad 0.4]$$

What is the most likely parse for the following sequence? GTGCCTA

Now consider the possible ways of being in state I in position 2:

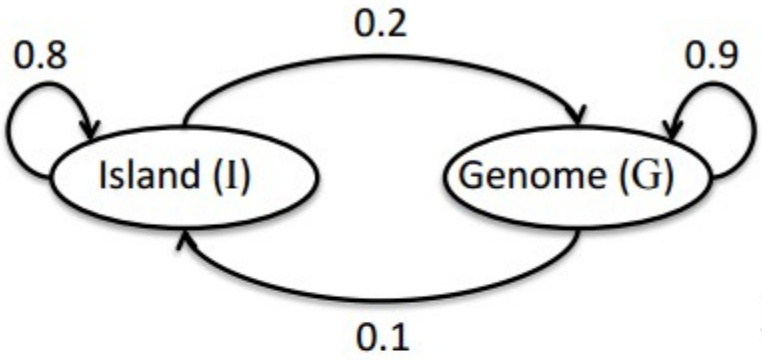


(1) S_1 = G: $P(S_1, S_2, X_1, X_2) = 0.09 * P_T(\text{I} \mid \text{G}) * P_E(\text{T} \mid \text{I})$
$= 0.09 * 0.1 * 0.1 = 0.0009$

(2) S_1 = I: $P(S_1, S_2, X_1, X_2) = 0.04 * P_T(\text{I} \mid \text{I}) * P_E(\text{T} \mid \text{I})$
$= 0.04 * 0.8 * 0.1 = 0.0032$

probability of the optimal parse *ending with state I at position 2 is* $\{S_1=\text{I}, S_2=\text{I}\}$

Using HMMs as generative models

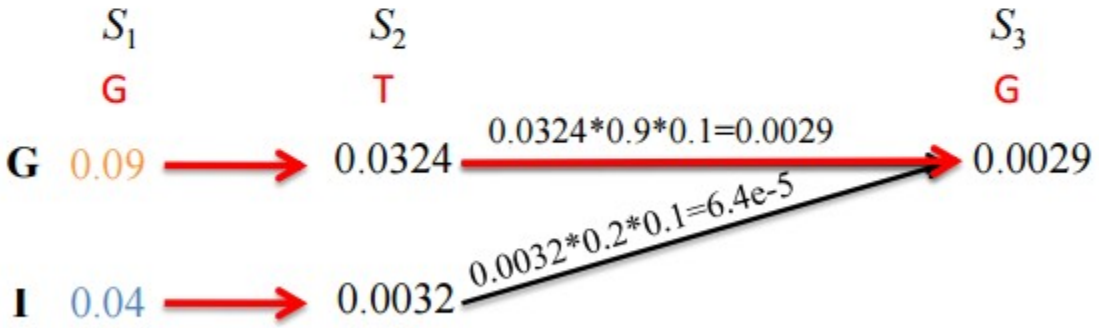


$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ [0.1 & 0.9] \end{matrix}$$

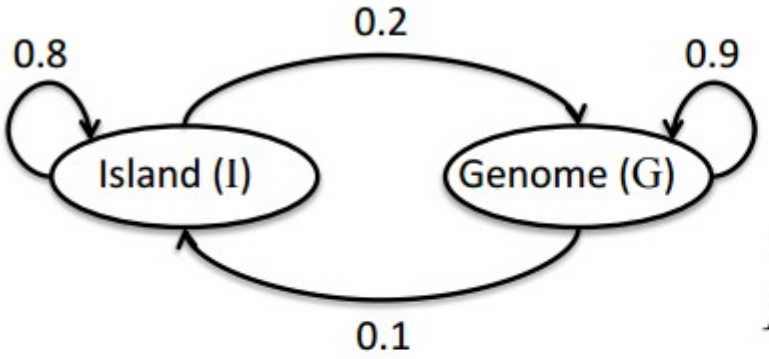
$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ [0.4 & 0.4 & 0.1 & 0.1] \end{matrix}$$

$$P(X|S=G) = [0.1 \quad 0.1 \quad 0.4 \quad 0.4]$$

What is the most likely parse for the following sequence? GTGCCTA



Using HMMs as generative models

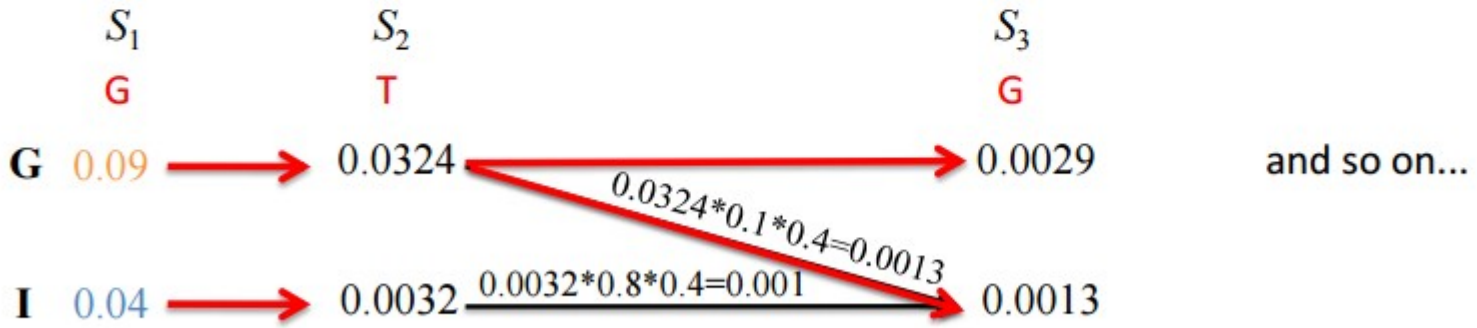


$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ [0.1 & 0.9] \end{matrix}$$

$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ [0.4 & 0.4 & 0.1 & 0.1] \end{matrix}$$

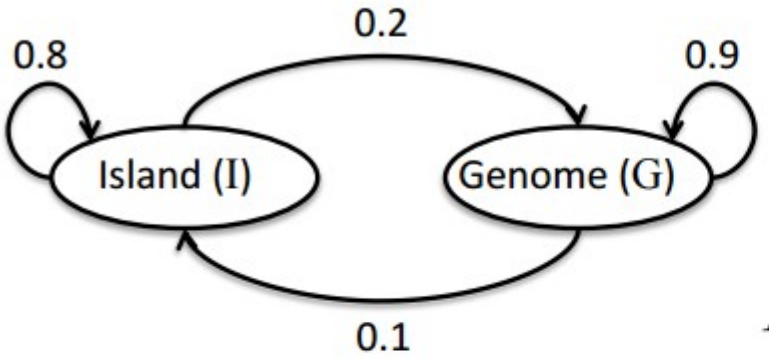
$$P(X|S=G) = [0.1 \quad 0.1 \quad 0.4 \quad 0.4]$$

What is the most likely parse for the following sequence? GTGCCTA



and so on...

Using HMMs as generative models



$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ [0.1 & 0.9] \end{matrix}$$

$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ [0.4 & 0.4 & 0.1 & 0.1] \end{matrix}$$

$$P(X|S=G) = [0.1 \quad 0.1 \quad 0.4 \quad 0.4]$$

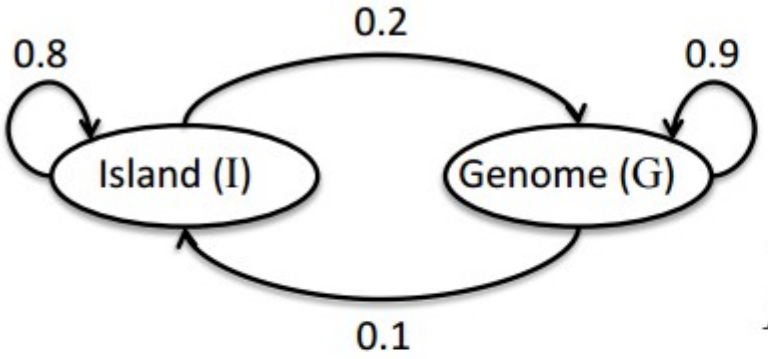
What is the most likely parse for the following sequence? GTGCCTA

	S_1	S_2	S_3	S_4	S_5	S_6	S_7
	G	T	G	C	C	T	A
G	0.09	0.0324	0.0029	2.61e-4	2.35e-5	1.06e-5	3.83e-6
I	0.04	0.0032	0.0013	4.16e-4	1.33e-4	1.06e-5	8.52e-7

Starting from highest final probability, traceback the path of hidden states:

G	G	I	I	I	G	G
G	T	G	C	C	T	A

Using HMMs as generative models



$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ [0.1 & 0.9] \end{matrix}$$

$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ [0.4 & 0.4 & 0.1 & 0.1] \end{matrix}$$

$$P(X|S=G) = [0.1 \quad 0.1 \quad 0.4 \quad 0.4]$$

What is the most likely parse for the following sequence? GTGCCTA

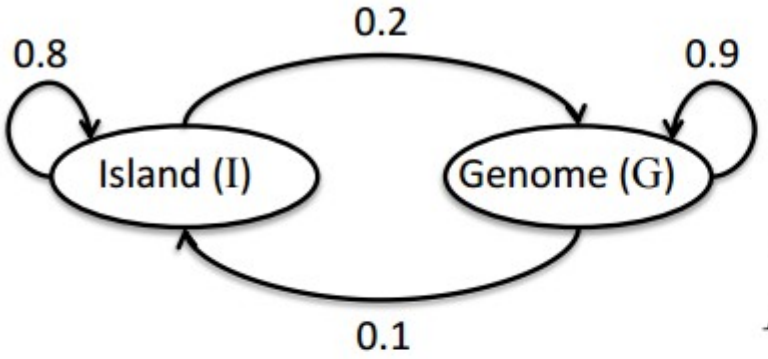
	S_1	S_2	S_3	S_4	S_5	S_6	S_7
	G	T	G	C	C	T	A
G	0.09	0.0324	0.0029	2.61e-4	2.35e-5	1.06e-5	3.83e-6
I	0.04	0.0032	0.0013	4.16e-4	1.33e-4	1.06e-5	8.52e-7

How many possible paths do we consider when advancing one position (from L-1 to L)?

$$\sum_{i=1}^{k}\sum_{j=1}^{k} P(\text{best path ending in } S_i \text{ at } L-1) * P(\text{transition from } S_i \to S_j \text{ and emit from } S_j \text{ at } L)$$

Answer: k^2. Therefore the run-time to obtain the optimal path up through pos. L is $O(k^2L)$.

Using HMMs as generative models



$$P_1(S) = \begin{matrix} \text{I} & \text{G} \\ [0.1 & 0.9] \end{matrix}$$

$$P(X|S=I) = \begin{matrix} \text{C} & \text{G} & \text{A} & \text{T} \\ [0.4 & 0.4 & 0.1 & 0.1] \end{matrix}$$

$$P(X|S=G) = [0.1 \quad 0.1 \quad 0.4 \quad 0.4]$$

What is the most likely parse for the following sequence? GTGCCTA

	S_1	S_2	S_3	S_4	S_5	S_6	S_7
	G	T	G	C	C	T	A
G	0.09	0.0324	0.0029	2.61e-4	2.35e-5	1.06e-5	3.83e-6
I	0.04	0.0032	0.0013	4.16e-4	1.33e-4	1.06e-5	8.52e-7

What is optimal parse of the first 3 bases GTG?

G	G	G
G	T	G

We start at the highest probability for the last base, so we begin our traceback from the circled point above

Profile Analysis

El análisis de perfil : herramienta para encontrar y alinear secuencias relacionadas distantes y para identificar dominios de secuencias conocidos en nuevas secuencias.

Profile

Básicamente, un perfil es una descripción del Consenso de una alineación múltiple de secuencias. Utiliza un sistema de puntuación de posición específica para capturar Información sobre el grado de conservación en varias posiciones en la alineación múltiple.

Esta hace que sea un método mucho más sensible y específico para la búsqueda de bases de datos que los métodos de pares, como los utilizados por BLAST o FastA, que utilizan puntuación independiente de la posición.

Profile Analysis

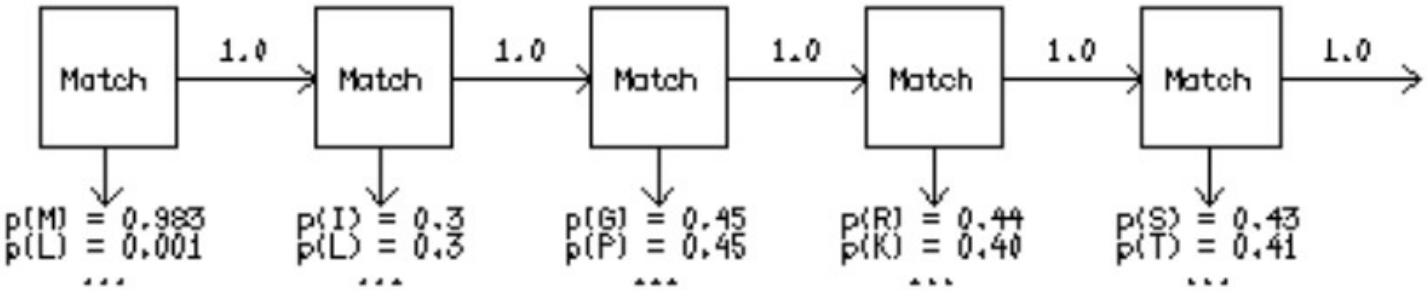
Profile hidden Markov models (HMMs) have several advantages over standard profiles. Profile HMMs have a formal probabilistic basis and have a consistant theory behind gap and insertion scores, in contrast to standard profile methods which use heuristic methods.

HMMs apply a statistical method to estimate the true frequency of a residue at a given position in the alignment from its observed
frequency while standard profiles use the observed frequency itself to assign the score for that residue.

Profile Analysis

What is a Profile HMM? - A Simplified Description

A profile HMM is a linear state machine consisting of a series of nodes, each of which corresponds roughly to a position (column) in the alignment from which it was built. If we ignore gaps, the correspondence is exact -- the profile HMM has a node for each column in the alignment, and each node can exist in one state, a match state. (The word "match" here implies that there is a position in the model for every position in the sequence to be aligned to the model.)



Pair HMM

HMM for pairwise sequence alignment, which incorporates affine gap scores.

"Hidden" States

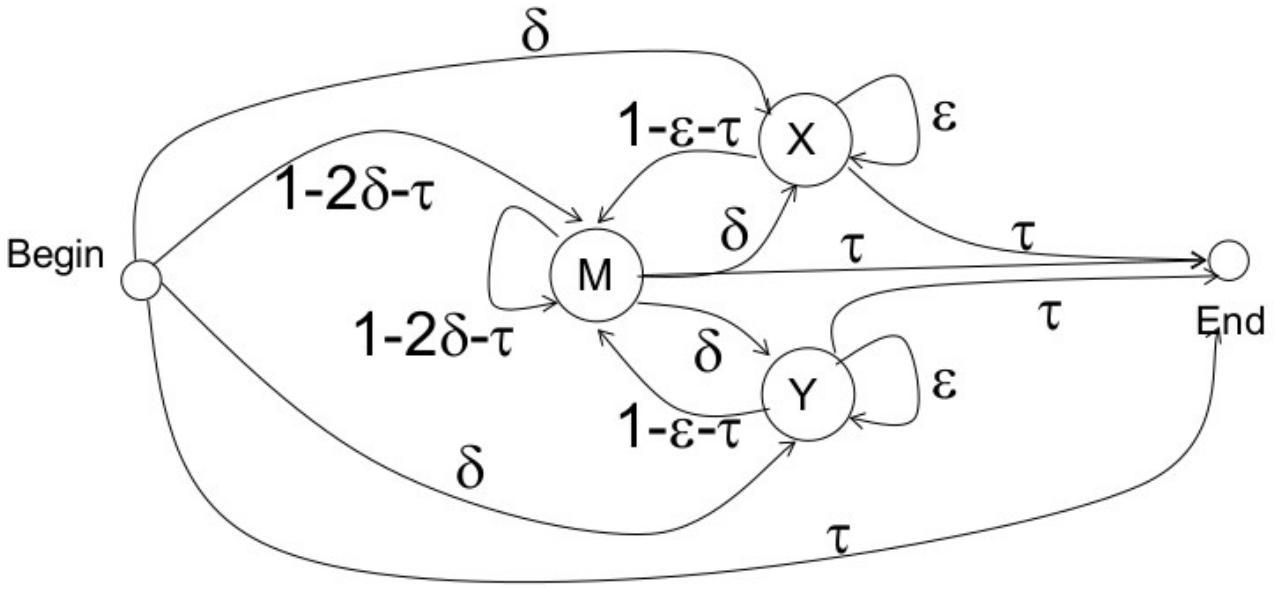
- Match (M)
- Insertion in *x* (X)
- insertion in *y* (Y)

Observation Symbols

- Match (M): $\{(a,b) | \ a,b \text{ in } \sum \}$.
- Insertion in *x* (X): $\{(a,-) | \ a \text{ in } \sum \}$.
- Insertion in *y* (Y): $\{(-,a) | \ a \text{ in } \sum \}$.

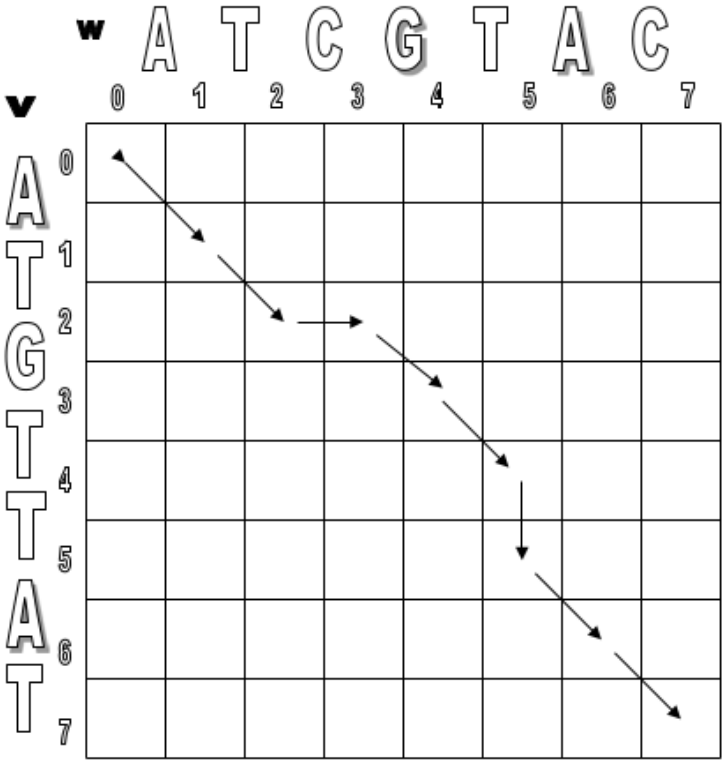
Profile Analysis

Pair HMMs



Profile Analysis

Alignment: a path → a hidden state sequence



A T - G T T A T
A T C G T - A C

M M Y M M X M M

Profile HMMs

- Models for (amino acid) *sequence families*
- Special *structure* (match, insert, and delete states; specific transition structure)
- Parameter estimation from given *multiple alignment*
- Can be used for examining the relation of new sequences to the family represented by the profile
- So-called motifs (PWMs, PSSMs) are a special case

Multiple Alignment of Globins

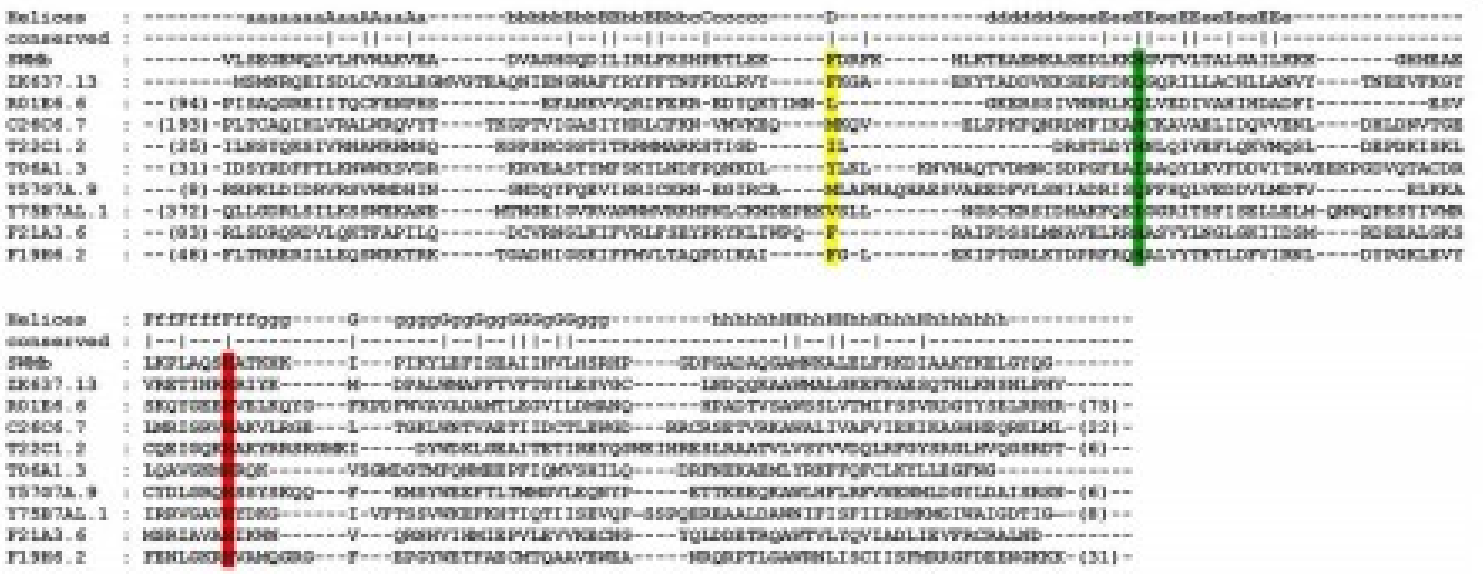


Fig. 1 Alignment of 9 representative globins and *Sperm whale* (SW) myoglobin. Eight alpha helices are shown as a-h above the alignment. Numbers between brackets indicate the number of amino acids preceding and following the globin domain. [Hoogewijs *et al. BMC Genomics* 2007 **8**:356]

Multiple sequence alignment (Globin family)

```
Helix                 AAAAAAAAAAAAAAAA   BBBBBBBBBBBBBBBBCCCCCCCCCCCC
HBA_HUMAN  ---------VLSPADKTNVKAAWGKVGA--HAGEYGAEALERMFLSFPTTKTYFPHF
HBB_HUMAN  --------VHLTPEEKSAVTALWGKV----NVDEVGGEALGRLLVVYPWTQRFFESF
MYG_PHYCA  ---------VLSEGEWQLVLHVWAKVEA--DVAGHGQDILIRLFKSHPETLEKFDRF
GLB3_CHITP ----------LSADQISTVQASFDKVKG------DPVGILYAVFKADPSIMAKFTQF
GLB5_PETMA PIVDTGSVAPLSAAEKTKIRSAWAPVYS--TYETSGVDILVKFFTSTPAAQEFFPKF
LGB2_LUPLU --------GALTESQAALVKSSWEEFNA--NIPKHTHRFFILVLEIAPAAKDLFS-F
GLB1_GLYDI ---------GLSAAQRQVIAATWKDIAGADNGAGVGKDCLIKFLSAHPQMAAVFG-F
Consensus            Ls....  v a W kv .  .    g . L.. f . P .   F  F

Helix          DDDDDDDEEEEEEEEEEEEEEEEEEE              FFFFFFFFFFFF
HBA_HUMAN  -DLS-----HGSAQVKGHGKKVADALTNAVAHV---D--DMPNALSALSDLHAHKL-
HBB_HUMAN  GDLSTPDAVMGNPKVKAHGKKVLGAFSDGLAHL---D--NLKGTFATLSELHCDKL-
MYG_PHYCA  KHLKTEAEMKASEDLKKHGVTVLTALGAILKK----K-GHHEAELKPLAQSHATKH-
GLB3_CHITP AG-KDLESIKGTAPFETHANRIVGFFSKIIGEL--P---NIEADVNTFVASHKPRG-
GLB5_PETMA KGLTTADQLKKSADVRWHAERIINAVNDAVASM--DDTEKMSMKLRDLSGKHAKSF-
LGB2_LUPLU LK-GTSEVPQNNPELQAHAGKVFKLVYEAAIQLQVTGVVVTDATLKNLGSVHVSKG-
GLB1_GLYDI SG----AS---DPGVAALGAKVLAQIGVAVSHL--GDEGKMVAQMKAVGVRHKGYGN
Consensus   .  t    .. . v..Hg kv. a   a...l   d   . a l. l   H  .

Helix        FFGGGGGGGGGGGGGGGGGG     HHHHHHHHHHHHHHHHHHHHHHHHHH
HBA_HUMAN  -RVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKYR------
HBB_HUMAN  -HVDPENFRLLGNVLVCVLAHHFGKEFTPPVQAAYQKVVAGVANALAHKYH------
MYG_PHYCA  -KIPIKYLEFISEAIIHVLHSRHPGDFGADAQGAMNKALELFRKDIAAKYKELGYQG
GLB3_CHITP --VTHDQLNNFRAGFVSYMKAHT--DFA-GAEAAWGATLDTFFGMIFSKM-------
GLB5_PETMA -QVDPQYFKVLAAVIADTVAAG---------DAGFEKLMSMICILLRSAY-------
LGB2_LUPLU --VADAHFPVVKEAILKTIKEVVGAKWSEELNSAWTIAYDELAIVIKKEMNDAA---
GLB1_GLYDI KHIKAQYFEPLGASLLSAMEHRIGGKMNAAAKDAWAAAYADISGALISGLQS-----
Consensus    v.   f  l . ..  ....    f   . aa. k. .     l sky
```

Ungapped score matrices

- For example, helix a in Fig. 1 is ungapped (16 columns)
- Associate with each (ungapped) position (column) i = 1, ..., L a probability distribution of symbols: $e_i(a)$ = position i has symbol a ε Σ with probability $e_i(a)$
- ML estimate:

  $e_i(a)$ = #a / #aligned_sequences

- Hence, assuming independence of positions (Bernoulli!)
  - the probability of sequence x is: $P(x) = \prod_i e_i(x_i)$
  - the log-odds with respect to random model (q_a) (= the background) is $S(x) = \sum_i \log(e_i(x_i)/q_{x(i)})$
- The resulting score matrix $(e_i(a)/q_a)_{a \varepsilon \Sigma, i=1...L}$ is called a *position specific score matrix* (PSSM) or a *position weight matrix* (PWM)

Count matrix for PSSM from multiple alignment

```
 aaaaaaaAaaAAaaAa
 -------|--||--|-
 SEGEWQLVLHVWAKVE
 SMNRQEISDLCVKSLE
 SAQGREIITQCFENPH
 TCAQIHLVRALWRQVY
 NSYQKSIVRNAWRHMS
 SYRDFFTLKNWWKSVD
 PKLDIDRVRSVWMDHI
 LGDRLSILKSSWEKAN
 SDRQRDVLQKTFAPIL
 TRRERILLEQSWRKTR
```

Multiple alignment of helix a of Fig. 1

$e_1(S) = 5/10 = 0.5$

$e_1(L) = 1/10 = 0.1$

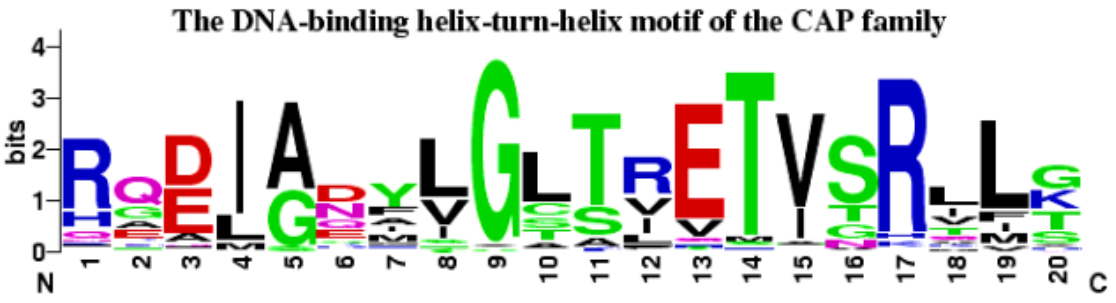
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
A	0															
R	0															
N	1															
D	0															
C	0															
Q	0															
E	0															
G	0															
H	0															
I	0															
L	1															
K	0															
M	0															
F	0															
P	1															
S	5	1	0	0	0	2	0	0	0	2	2	0	0	2	0	1
T	2															
W	0															
Y	0															
V	0															

Count matrix (fragment) of helix a of Fig. 1

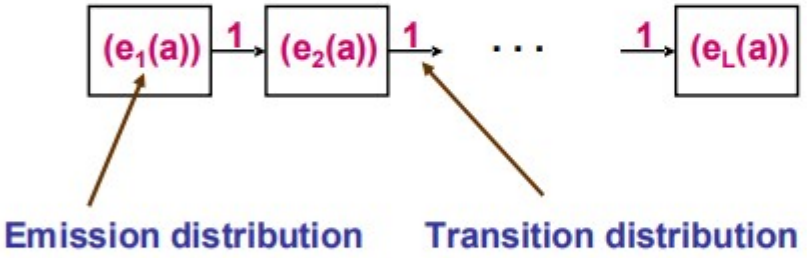
Profile / PSSM

- DNA / proteins Segments of the same length L;
- Often represented as Positional frequency matrix;

```
LTMTRGDIGNYLGLTVETISRLLGRFQKSGML
LTMTRGDIGNYLGLTIETISRLLGRFQKSGMI
LTMTRGDIGNYLGLTVETISRLLGRFQKSEIL
LTMTRGDIGNYLGLTVETISRLLGRLQKMGIL
LAMSRNEIGNYLGLAVETVSRVFSRFQQNELI
LAMSRNEIGNYLGLAVETVSRVFTRFQQNGLI
LPMSRNEIGNYLGLAVETVSRVFTRFQQNGLL
VRMSREEIGNYLGLTLETVSRLFSRFGREGLI
LRMSREEIGSYLGLKLETVSRTLSKFHQEGLI
LPMCRRDIGDYLGLTLETVSRALSQLHTQGIL
LPMSRRDIADYLGLTVETVSRAVSQLHTDGVL
LPMSRQDIADYLGLTIETVSRTFTKLERHGAI
```



($e_i(a)$) as a (trivial) HMM

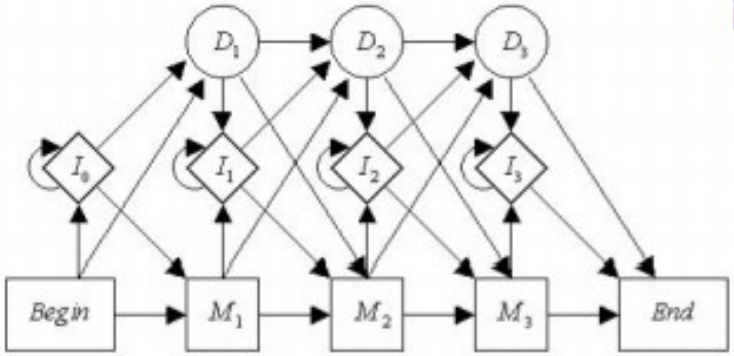


Alignments with gaps and the structure of profile HMMs

```
HBA_HUMAN    ...VGA--HAGEY...
HBB_HUMAN    ...V----NVDEV...
MYG_PHYCA    ...VEA--DVAGH...
GLB3_CHITP   ...VKG------D...
GLB5_PETMA   ...VYS--TYETS...
LGB2_LUPLU   ...FNA--NIPKH...
GLB1_GLYDI   ...IAGADNGAGV...
                ***  *****
```

'Backbone' = columns (*) that correspond to the conserved core of the sequence family to be modeled;

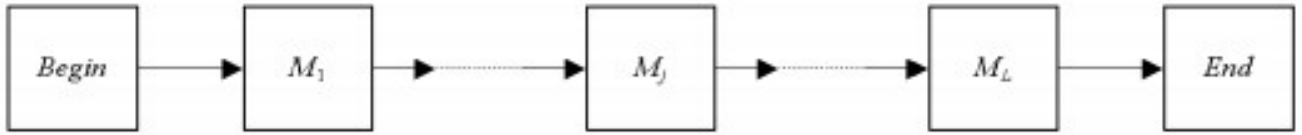
Other columns are needed to represent insertions



Transition structure of a profile HMM

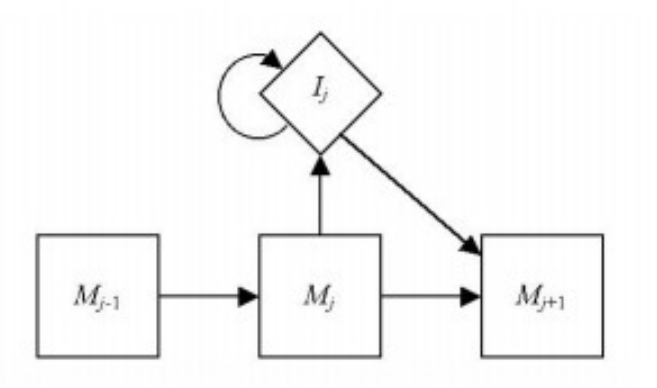
Backbone: match states

- Match states emit the symbols that belong to the 'backbone' of the model



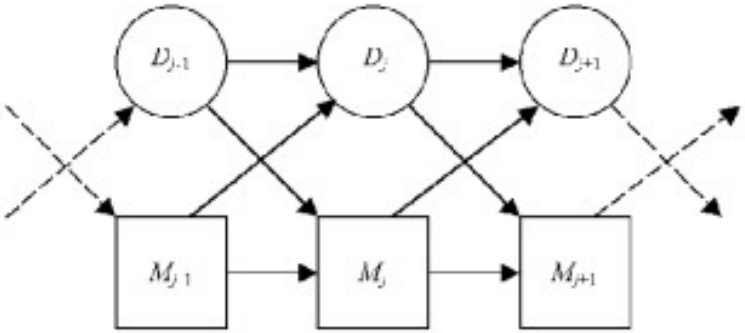
Insert states

- Each Insert state can emit between two match states any number of symbols that do not belong to the backbone model



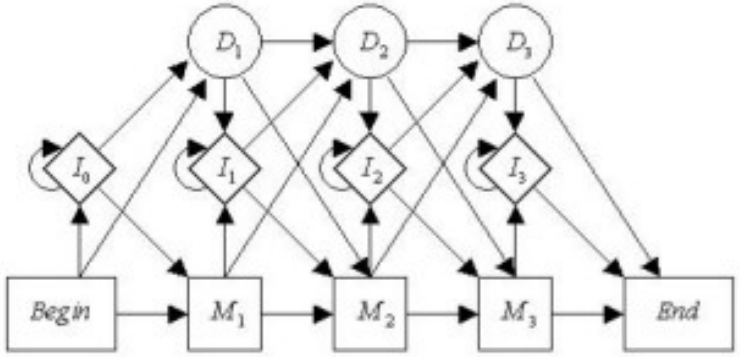
Delete states

- Delete states are needed to present 'jumps' (gaps) that pass some backbone states in an efficient way. This could be done with direct transitions but that would introduce a large number of parameters. Therefore the structure shown below is normally used.
- Delete states are *silent* (do not emit any symbol).



Profile HMM: standard structure

- All HMM algorithms (Viterbi, Forward, Backward, Baum-Welch training etc) can be adapted for the profile HMM



Profile HMM for global alignment

Learning profile HMMs from alignments

- **Input: Multiple alignment λ of some sample sequences from the sequence family to be modeled by the profile HMM**
- **1. Select some columns 1, ..., L of the alignment λ to the backbone; these will correspond to the match states M_1, ..., M_L of the profile HMM**
  - **Take the best conserved columns, with no gaps**
- **2. Estimate probabilities a_{kl}, $e_k(a)$**

$$a_{kl} = \frac{A_{kl}}{\sum_q A_{kq}} \qquad e_k(b) = \frac{E_k(b)}{\sum_\sigma E_k(\sigma)}$$

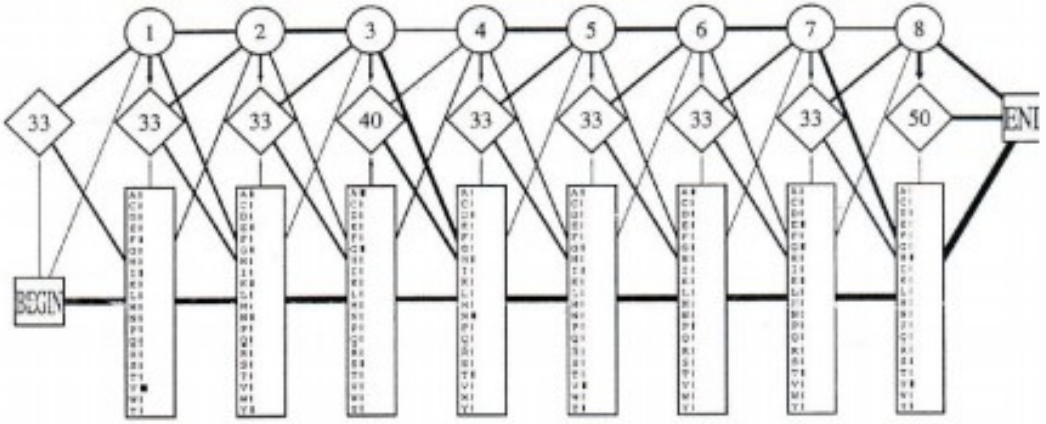
where

  - **A_{kl} = (the count of transitions k→l in λ) + 1 (= Laplace rule of pseudocounts)**
  - **$E_k(a)$ = (the count of emissions of a from state k in λ) + 1**

Learning a profile HMM: an example

```
HBA_HUMAN    ...VGA--HAGEY...
HBB_HUMAN    ...V----NVDEV...
MYG_PHYCA    ...VEA--DVAGH...
GLB3_CHITP   ...VKG------D...
GLB5_PETMA   ...VYS--TYETS...
LGB2_LUPLU   ...FNA--NIPKH...
GLB1_GLYDI   ...IAGADNGAGV...
                ***  *****
```

Ten columns from the multiple alignment of seven globin protein sequences. The starred columns are ones that will be treated as 'matches' in the profile HMM.



A HMM derived from the alignment using Laplace's rule (add pseudocount 1 to each count). Emission probabilities shown as bars opposite the different amino acids for each match state, transition probabilities indicated by the thickness of the lines. The $I \rightarrow I$ transition probabilities are shown as percentages in the insert states.

Multiple alignment with a known profile HMM

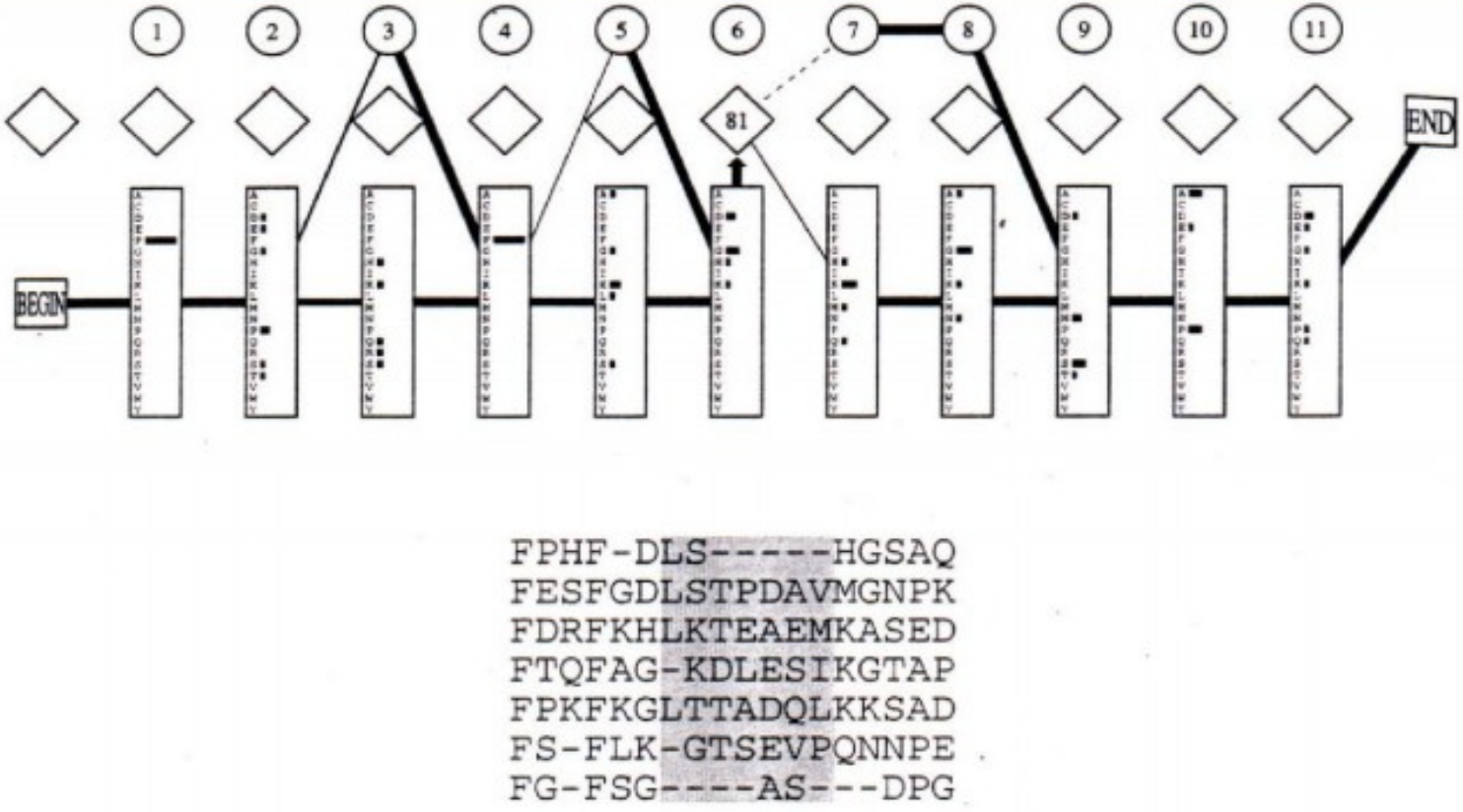
If the profile HMM *M* is known, the following procedure can be applied to generate multiple alignments:

- Align each sequence *S*(*i*) to the profile *M* separately (Viterbi path!)

- Accumulate the obtained alignments to a multiple alignment.

- Insert runs are not aligned, i.e. the choice of how to put the letters in the insert regions is arbitrary (Most profile HMM implementations simply left-justify insert regions, as in the following example).

Example: another profile HMM




```
FPHF-DLS-----HGSAQ
FESFGDLSTPDAVMGNPK
FDRFKHLKTEAEMKASED
FTQFAG-KDLESIKGTAP
FPKFKGLTTADQLKKSAD
FS-FLK-GTSEVPQNNPE
FG-FSG----AS---DPG
```

A model (top) estimated from an alignment (bottom). The columns in the shaded area of the alignment were treated as inserts