Alignement multiple, profiles, modèles de Markov caché

Alignement multiple

alignement de deux séquences $\Rightarrow$ alignement de plusieurs séquences

: (extension naturelle pour l'informaticien)

: une philosophie différente pour le biologiste

2 séquences : est-ce qu'elles ont reliées ?

K séquences (reliées) : quels sont les traits communs ?

Applications

– représentation de familles de protéines
– identification et représentation de caractéristiques préservés et leur correlation à la structure/fonction
– inférence de l'Évolution

un problème difficile . . .

Un alignement multiple

```
Lycopersi CGAGCGCGTT GTTGGAGAAA  AAGATCAATA TATTGCTTAT
Convolvul ---GCGCGTT ATTGGAGAAA  AAGATCAATT TATTGCTTAT
Ipomoea   CATGCGCGTT GTTGGAGAAA  AAGATCAATA TATTGCTTAT
Borago    CGATGCCGTT CCGGGAGAAG  AAAATCAATA TATATGTTAT
Heliotrop CGATCCCGTT CCTGGAGACG  AAGATCAATA TATTGCTTAT
Hydrophyl CGATCCCGTT CTTGGAGAAG  AAGATCAATA TATTGCTTAT
Eriodicty CGATCCCGTT CCTGGAGAAG  AAGATCAATA TATTTGTTAT
Digitalis TGAGCCCGTT CCTGGAGAAG  CAGATCAATA TATCTGTTAT
Buddleja  CGAGCCCGTT CCTGGAGAAA  CAGATCAATA TATCTGTTAT
```

Valeur de l'alignement ?

Scores

On a K séquences de longueur n ...

Problème : comment spécifier la valeur d'une colonne dans l'alignement ?

Approche 1 : Spécifier la valeur de toutes les colonnes possibles :
$\left(\Sigma \cup \{-\}\right)^K$ si on a K séquences.

Approche 2 : fonction SP «sum of pairs» — considérer l'alignement de chaque paire de séquences :
$\mathsf{score} = \sum_{i<j} \mathsf{score}(\text{ali de } S_i, S_j)$

Calcul

Solution par programmation dynamique :
matrice de taille $O(n^K)$, récurrences avec $2^K - 1$ termes ...

$$A(i_1, i_2, \ldots, i_K) = \max\left\{ \begin{array}{l} A(i_1 - 1, i_2, \ldots, i_K) + \mathsf{score}\begin{pmatrix} S_1[i_1] \\ - \\ \cdots \\ - \end{pmatrix}, \\ \ldots, \\ A(i_1 - 1, i_2 - 1, \ldots, i_K - 1) + \mathsf{score}\begin{pmatrix} S_1[i_1] \\ S_2[i_2] \\ \cdots \\ S_K[i_K] \end{pmatrix} \end{array} \right\}$$

PD prend temps exponentiel ... est-ce qu'il y a une meilleure méthode ?

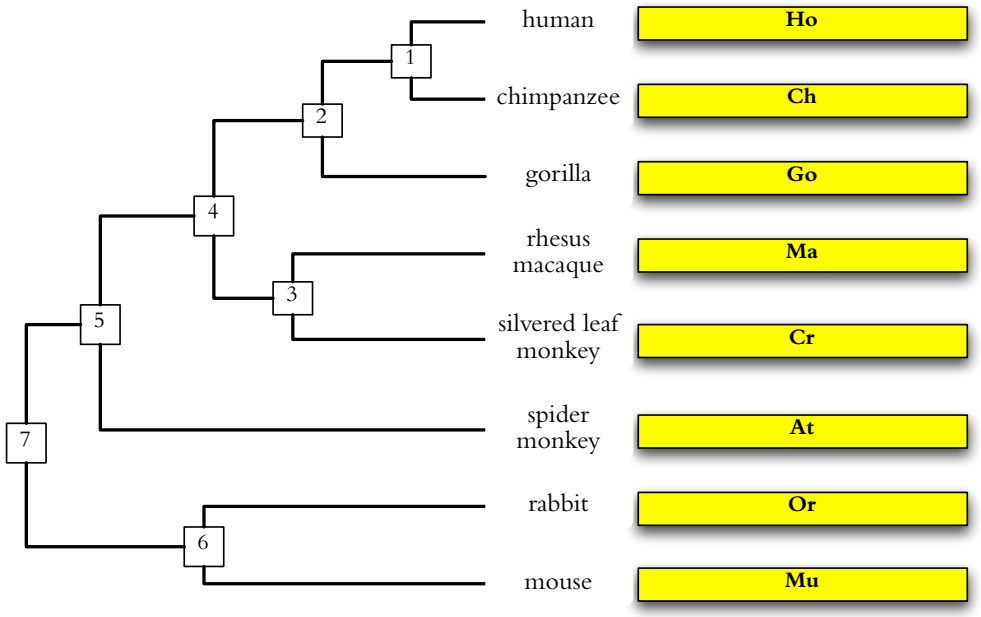
Thm. Il est NP-difficile de trouver l'alignement multiple qui est optimale selon SP.

D'autres définitions d'alignement

Séquence de consensus S^*, $\mathsf{score} = \sum_i \mathsf{score}(\text{ali de } S_i, S^*)$
(NP-difficile de trouver S^*)

Alignement à un arbre
(aussi NP-difficile)

Alignement à un arbre



Idée : calculer les alignements à partir des feuilles vers la racine

— il faut aligner des alignements aux noeuds internes

Principe : les colonnes d'un alignmenent ne changent pas quand l'alignment est aligné à sa soeur

Alignement à un arbre 2

D'autres héuristiques dans ce contexte

: ajouter une séquence à la fois — arbre de «peigne»

: alignement au consensus — arbre «étoile»

: arbre «guide» — ClusterX, MUSCLE

Représentation de l'information dans l'alignement

0. séquence de consensus

1. signature («patterns»)

2. profile [trou OK] ou PSSM [sans trou] «position-specific scoring matrix»

3. modèle de Markov caché : HMM «Hidden Markov Model»

problème : alignement à une famille

Signatures — PROSITE

entrée PS00028 : class I zinc-finger pattern

(un motif important dans facteurs de transcription)

cca. 600 séquences

```
ID   ZINC_FINGER_C2H2_1; PATTERN.
AC   PS00028;
DT   APR-1990 (CREATED);
DT   JUN-1994 (DATA UPDATE);
DT   JUL-1998 (INFO UPDATE).
DE   Zinc finger, C2H2 type, domain signature.
PA   C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H.
NR   /RELEASE=38,80000;
NR   /TOTAL=2189(453); /POSITIVE=2147(412);
NR   /UNKNOWN=6(6); /FALSE_POS=36(35);
NR   /FALSE_NEG=3; /PARTIAL=2;
```

SYNTAXE DE PROSITE

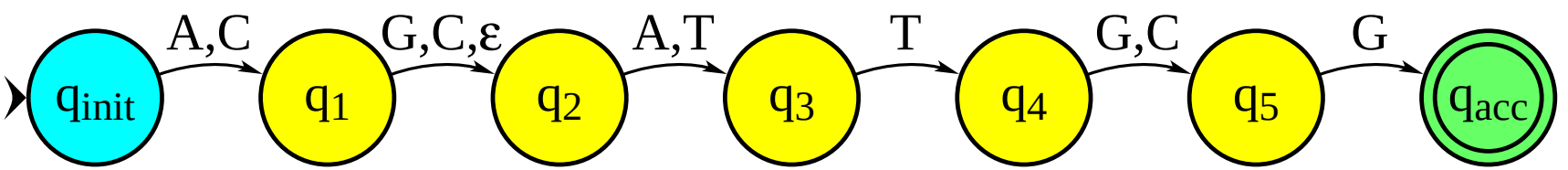
- lettres pour les acides aminés ; `X` acide arbitraire
- $[\ldots]$: choix altérnatifs dans une position
- $\{\ldots\}$: choix exclus dans une position
- (i, j) : répété i–j fois
- $-$ séparateur entre les positions

zinc-finger motif : `C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H`
déscription de cca. 10^{27} séquences possibles

AUTOMATES FINIS ET ALIGNEMENTS

Un alignement multiple peut être representé par un automate fini (ou expression régulière) :

```
CGATCG
C-ATGG
ACTTCG
```



on a perdu l'info sur les fréquences des lettres . . .

PROFILE

$\rightarrow$ enregistrer la fréquence de symbols dans l'alignement multiple

Problème : trouver le profile dans une séquence

alignement de longueur n sur k séquences
calculer $p_j(a)$: fréquence/probabilité de caractère a dans colonne j,
où $a \in \Sigma$ et $j = 1, \ldots, n$

Alignement à un profile

Valeur d'un alignement d'une séquence S à un profile P

1. $S[i]$ aligné à colonne j de P

$$\text{score}(S[i], P[j]) = \sum_{a \in \Sigma} s(S[i], a) p_j(a) = \text{score}_j(S[i])$$

2. trou de longueur i dans P — insertion en colonne j : $-\alpha_j - \beta_j(i-1)$

3. trou de longueur i dans S — suppression de colonnes $j, \ldots, j+i-1$: $-\gamma_j - \delta_{j+1} - \delta_{j+2} - \cdots - \delta_{j+i-1}$

Alignement à un profile 2

PD : $G(i,j)$ score de l'alignement optimal entre $S[1..i]$ et les premières j colonnes de P.

Récurrences avec E (trou en S), F (trou en P), et G (match) :

$$E(i,j) = \max\Big\{G(i,j-1) - \gamma_j, E(i,j-1) - \delta_j\Big\}$$
$$F(i,j) = \max\Big\{G(i-1,j) - \alpha_j, F(i-1,j) - \beta_j\Big\}$$
$$G(i,j) = \max\Big\{E(i,j), F(i,j), G(i-1,j-1) + \text{score}_j(S[i])\Big\}$$

Alignement à un profile 3

Initialisation :
$G(i,0) = E(i,0) = 0$ (alignement peut commencer dans le milieu de S),
$G(0,j) = F(0,j) = -\gamma_1 - \sum_{i=2}^{j} \delta_i$ (il commence au début du profile).

Score de l'alignement optimal : $\max\Big\{G(i,n) \colon i = 1,\ldots,|S|\Big\}$

Temps de calcul : $O(|S|n)$ (après le calcul de score_j)

PROFILE DE PROSITE

exemple : Profile de homeobox

spécifie entre autres la pondération score_j, α_j, β_j, γ_j, et δ_j

```
ID   HOMEOBOX_2; MATRIX.
...
                A   B   C   D   E   ...
...
/M: SY='E'; M= -5,  2,-25,  3, 11, ...
/M: SY='Q'; M= -3, -4,-25, -4, 12, ...
...
/I:         I=-8; MI=-8; IM=-8; DM=-15; MD=-15;
```

Modèles probabilistes

Modèles probabilistes des séquences

- modèle iid

- segmentation

- modèle de Markov caché

Segmentation

Modèle : séquence X de longueur n où $X[i] \in \{\mathtt{W}, \mathtt{S}\}$
($\mathtt{W} = \{\mathtt{A}, \mathtt{T}\}; \mathtt{S} = \{\mathtt{C}, \mathtt{G}\}$).

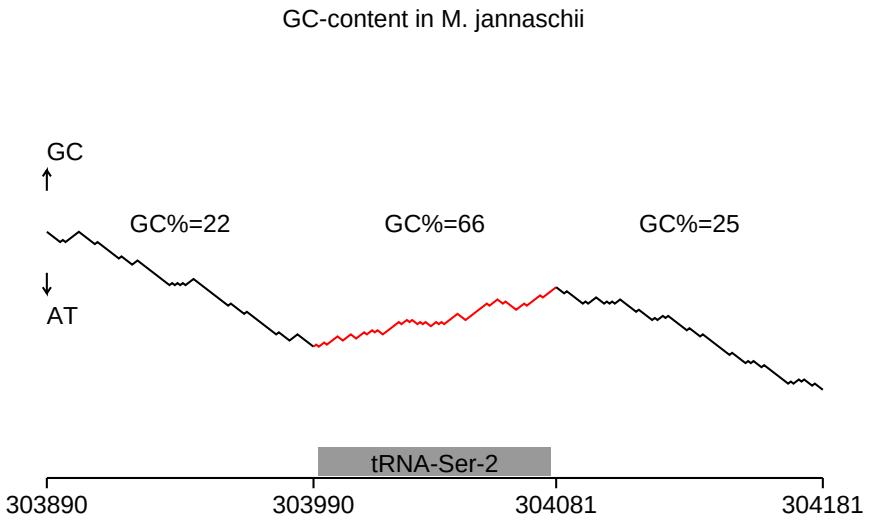
Classe de positions : séquence Z de longueur n où $Z[i] \in \{0, 1\}$;
$0 =$ riche en $\mathtt{AT}$; $1 =$ riche en $\mathtt{GC}$.

Caractère en position i dépend de $Z[i]$ seulement :

$$\mathbb{P}\Big\{ X[i] = x \;\Big|\; Z[i] = z \Big\} = p_z(x).$$

GÈNES ARN DANS DES THERMOPHILES

en procaryotes thermophiliques on peut identifier les gènes ARN par contenu de GC



M. jannaschii : $p_0(\mathtt{S}) = 0.22, p_1(\mathtt{S}) = 0.66$

Segmentation : trouver Z à partir de X

Gènes ARN

ARN de transfert

ARN ribosomale

d'autres (p.e., Ribonucléase P ARN)

acide nucléique à un brin — repliement par liaisons hydrogène

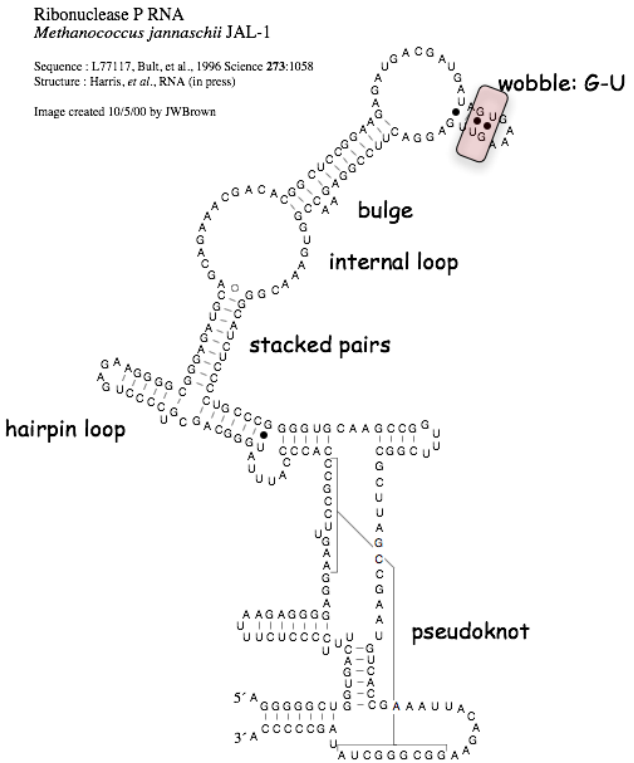
structure est importante

ARNt

(notez les hélices)

Wikipedia

Structure secondaire — RNase P RNA



Recherche de gènes ARN

- modèle de structure secondaire : scanner la séquence du génome

- exemple : `tRNAScan`

- on reviendra à cette question . . .

VRAISEMBLANCE

Étant donné la séquence x :

- vraisemblance de z :

$$L(z) = \mathbb{P}\Big\{ X = x \;\Big|\; Z = z \Big\} = \prod_{i=1}^{n} p_{z[i]}(x[i]).$$

- log-vraisemblance

$$\ell(z) = \log L(z) = \sum_{i=1}^{n} \log p_{z[i]}(x[i])$$

$$= \underbrace{\sum_{i=1}^{n} \log p_0(x[i])}_{\text{vrais. de l'hypo } Z=0} + \underbrace{\sum_{i=1}^{n} \log \frac{p_{z[i]}(x[i])}{p_0(x[i])}}_{\text{LLR de l'hypo } z}.$$

Problème : le z qui maximise $\ell(z)$ n'est pas utile : $z[i] = \text{argmax}_z p_z(x[i])$
— trop de segments ($z[i] = 0$ si $x[i] = \texttt{W}$ et $z[i] = 1$ si $x[i] = \texttt{S}$)

Longueur de déscription

Comment choisir z ?

Principe de longueur de déscription minimale : le meilleur hypothèse est celui qui est le plus court à encoder (Rissanen 1983)

Encodage : données et modèle en même temps

Ici : encoder x et z :

$$\underbrace{\texttt{001010}\cdots\texttt{1}}_{x \text{ encodé en binaire}} \ \# \ \underbrace{C(z)}_{z \text{ encodé}}$$

Encodage

Encoder x : codage optimale en utilisant z
$-\lg p_{z[i]}(x[i])$ bits pour encoder $x[i]$.

Comment ? Encodage Huffmann de blocs de taille m

Exemple de l'avantage de Huffmann : $p(\texttt{W}) = 0.75, p(\texttt{S}) = 0.25$; $m = 2$
$\texttt{WW} \to \texttt{0}$, $\texttt{WS} \to \texttt{10}$, $\texttt{SW} \to \texttt{110}$, $\texttt{SS} \to \texttt{111}$

Nombre de bits par bloc en moyenne : $\frac{9}{16} \cdot 1 + \frac{3}{16} \cdot 2 + \frac{3}{16} \cdot 3 + \frac{1}{16} \cdot 3 = \frac{27}{16}$, donc 0.84 bits par caractère

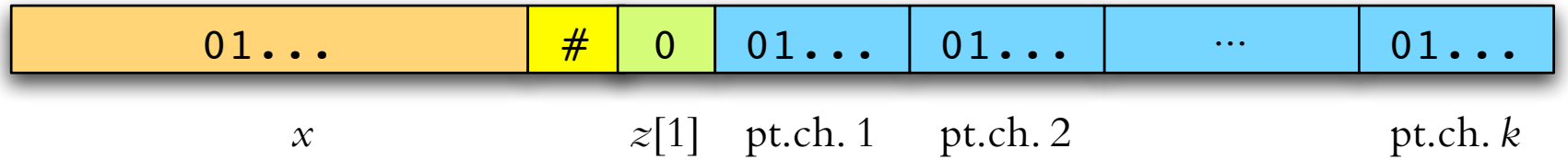
Encodage — segmentation

Segmentation z — comment l'encoder ?

Il y a beaucoup moins de changements $z[i] \neq z[i-1]$ que la longueur n de la séquence

Donc, on va juste donner la liste de *points de changement* ou $z[i] \neq z[i-1]$

1 bit pour encoder $z[0]$
chaque point de changement encodé en $\lg(n-1)$ bits (entier entre 2 et n)



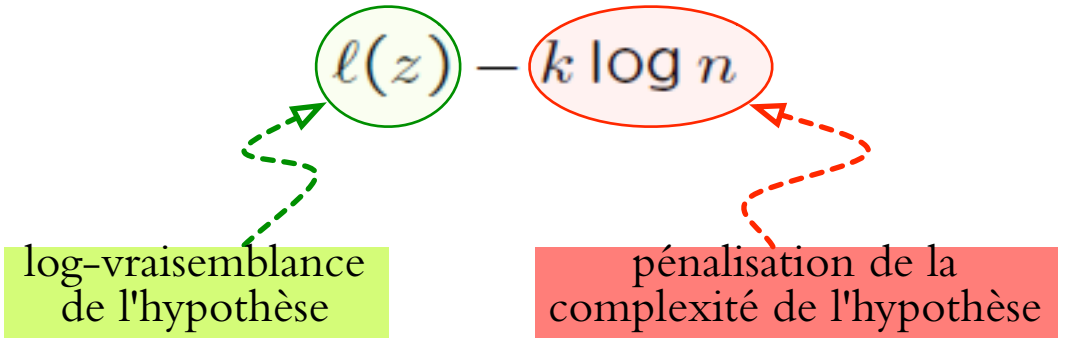
Encodage de z avec k points de changements : $1 + k \lg(n-1)$ bits

Encodage complète

Longueur de l'encodage complète x et z quand k points de changements :

$$\sum_{i=1}^{n}\Big(-\lg p_{z[i]}(x[i])\Big) + k\lg n + O(1) = -\lg L(z) + k\lg n + O(1)$$

Le meilleur choix de z minimise cette longueur — c-à-d il maximise



Principe universel de sélection de modèles probabilistes :
balancer la complexité du modèle et son accord avec les données

Algorithme

But : trouver z qui maximise $\mathrm{LLR}(z) - kh$

(car $\ell(z) = \ell(0) + \mathrm{LLR}(z)$; $h = \log n$)

Notation : $w(i) = \log \frac{p_1(x[i])}{p_0(x[i])}$ donc $\mathrm{LLR}(z) = \sum_{i:\ z[i]=1} w(i)$.

PD : $V_a(i)$ est le score de la meilleure segmentation de $1..i$ où $z[i] = a$.

$$
\begin{aligned}
V_0(i) &= \max\{V_0(i-1), V_1(i-1) - h\} && i > 1 \\
V_1(i) &= \max\{V_0(i-1) + w(i) - h, V_1(i-1) + w(i)\} && i > 1 \\
V_0(1) &= 0; V_1(1) = w(1)
\end{aligned}
$$

+ traceback sur les $\max$

Modèle de Markov caché

ensemble d'états $\mathcal{Q}$ de taille N

probabilités de transition $\tau\colon \mathcal{Q} \times \mathcal{Q} \mapsto [0,1]$ où $\sum_{q' \in \mathcal{Q}} \tau(q, q') = 1$ pour tout $q \in \mathcal{Q}$

probabilités d'émission $p\colon \mathcal{Q} \times \Sigma \mapsto [0,1]$ où $\sum_{c \in \Sigma} p(q, c) = 1$ pour tout $q \in \mathcal{Q}$.

probabilités d'état initial $\pi\colon \mathcal{Q} \mapsto [0,1]$ où $\sum_{q \in \mathcal{Q}} \pi(q) = 1$.

Génération d'une séquence $s = s_1 \cdots s_\ell$ au hasard :

```
1 Choisir l'état inital q_1 au hasard par les probabilités π.
2 pour i ← 1, ..., ℓ
3       Choisir s_i au hasard par les probabilités p(q_i, ·) : émettre s_i.
4       si i < ℓ alors Choisir q_{i+1} au hasard par τ(q, ·).
5 fin
```

Trois problèmes

Problème 1. Si on observe la séquence s, comment est-ce qu'on peut calculer $\mathbb{P}\left\{ s \;\middle|\; \mathcal{M} \right\}$, la probabilité que $\mathcal{M}$ engendre s ?

Problème 2. Si on observe la séquence s, comment est-ce qu'on peut trouver la séquence d'états $q_1 \cdots q_\ell$ qui y correspond le mieux ?

Problème 3. Comment est-ce qu'on choisit les paramètres τ, p, π (apprentissage)

Problème 1

vraisemblance : probabilité que $\mathcal{M}$ émet $s = s_1 \cdots s_\ell$

$$L_{\mathcal{M}}(s) = \mathbb{P}\Big\{ s \;\Big|\; \mathcal{M} \Big\} = \sum_{q_1,\ldots,q_\ell} \pi(q_1) p(q_1, s_1) \tau(q_1, q_2) \cdots \tau(q_{\ell-1}, q_\ell) p(q_\ell, s_\ell).$$

$\rightarrow$ sommation sur tous les chemins : $O(N^\ell)$ termes dans la formule — trop

Programmation Dynamique !

Soit $\alpha_i(q) = \mathbb{P}\big\{ s_1 \cdots s_i, q_i = q \big\}$ (production du préfixe en arrivant à l'état q).

Alors $L(s) = \sum_{q \in \mathcal{Q}} \alpha_\ell(q)$.

Calcul de $\alpha_i(q)$:

Initialisation : $\alpha_1(q) = \pi(q) p(q, s_1)$.

Récurrence : $\alpha_{i+1}(q) = \Big(\sum_{q'} \alpha_i(q') \tau(q', q) \Big) p(q, s_{i+1})$.

Problème 1 — cont.

On peut calculer les α_i d'une façon très efficace :
en temps $O(N^2\ell)$, avec $O(N)$ espace

Une autre possibilité

Soit $\beta_i(q) = \mathbb{P}\{s_{i+1} \cdots s_\ell, q_i = q\}$ (production du suffixe à partir de l'état q).

Alors $L(s) = \sum_{q \in \mathcal{Q}} \beta_1(q)\pi(q)$.

Initialisation : $\beta_\ell(q) = 1$.
Récurrence : $\beta_{i-1}(q) = \sum_{q'} \tau(q, q')p(q', s_i)\beta_i(q')$.

De nouveau : temps $O(N^2\ell)$, espace $O(N)$.

Problème 2

Séquence d'états la plus probable — qu'est-ce que ça veut dire ?

L'état le plus probable pour le i-ème caractère :
soit $\gamma_i(q) = \mathbb{P}\{q_i = q\}$ (que le i-ème état est q).

$$\gamma_i(q) = \frac{\alpha_i(q)\beta_i(q)}{\sum_{q' \in \mathcal{Q}} \alpha_i(q')\beta_i(q')}.$$

Alors $q_i^* = \arg\max \gamma_i(q)$ est l'état le plus probable pour le i-ème caractère.

Le chemin le plus probable — algorithme de Viterbi

Algo de Viterbi

Soit $\delta_i(q) = \max_{q_1,\ldots,q_{i-1}} \mathbb{P}\Big\{q_1 \cdots q_{i-1} q_i = q, s_1 \cdots s_i\Big\}$ (meilleur chemin pour le préfixe).

Initialisation : $\delta_1(q) = \pi(q)p(q, s_1)$.

Récurrence : $\delta_{i+1}(q) = \Big(\max_{q'} \delta_i(q')\tau(q', q)\Big)p(q, s_{i+1})$

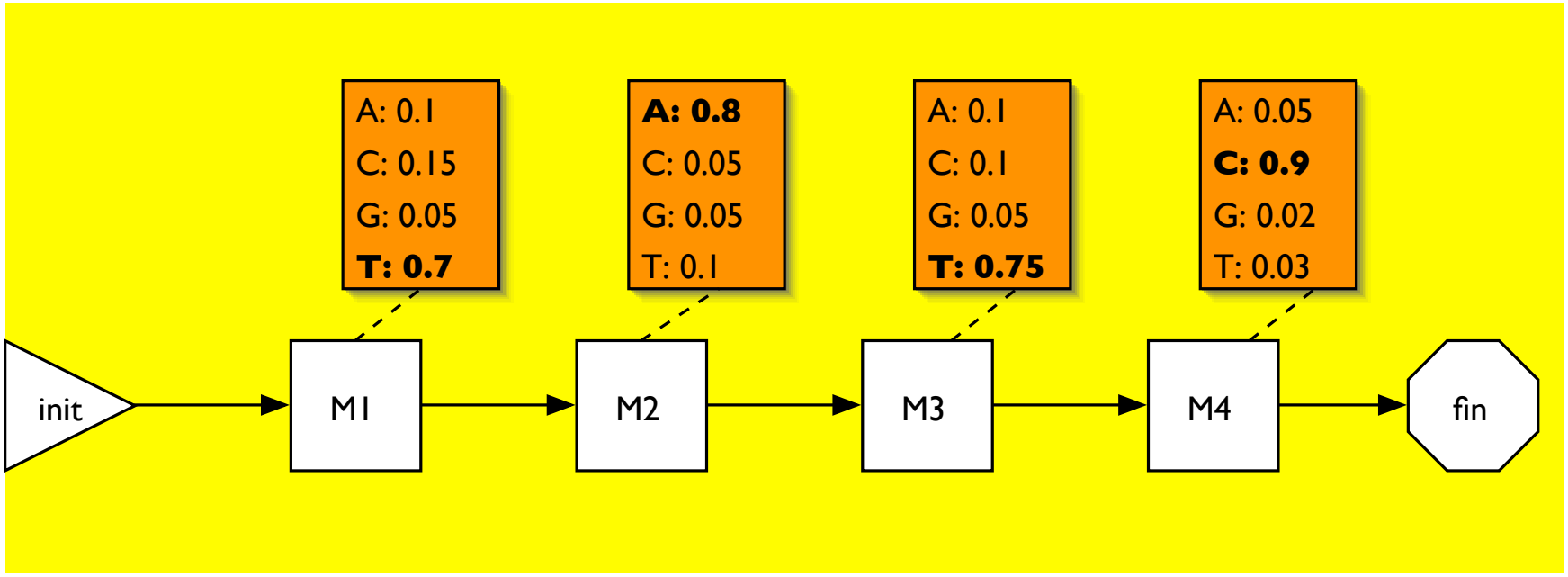
PD de nouveau

HMM POUR GÉNÉRATION D'UNE SÉQUENCE

Séquence d'ADN ou séquence protéique

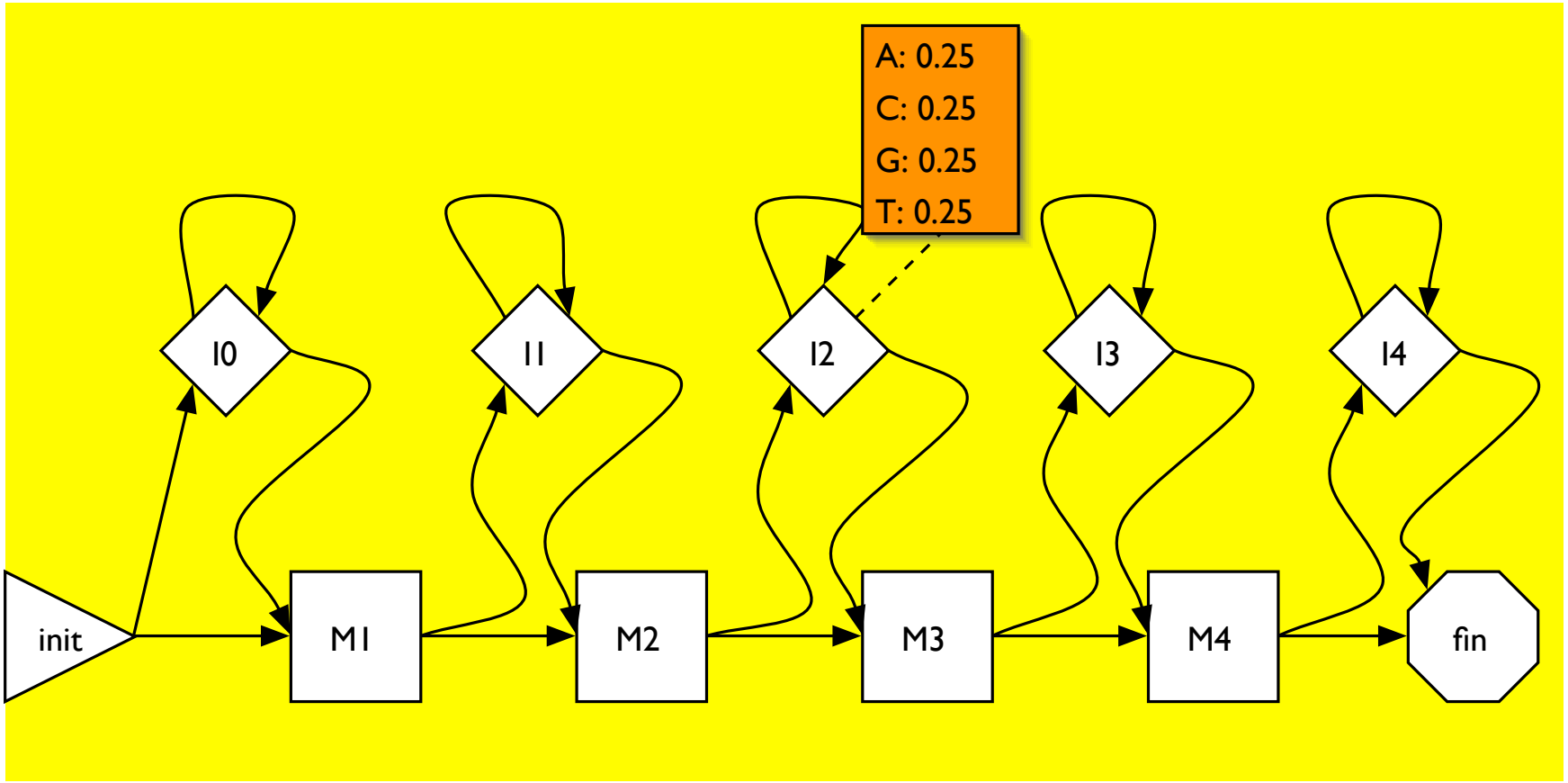
Construisons un modèle pour alignement

1. Substitutions



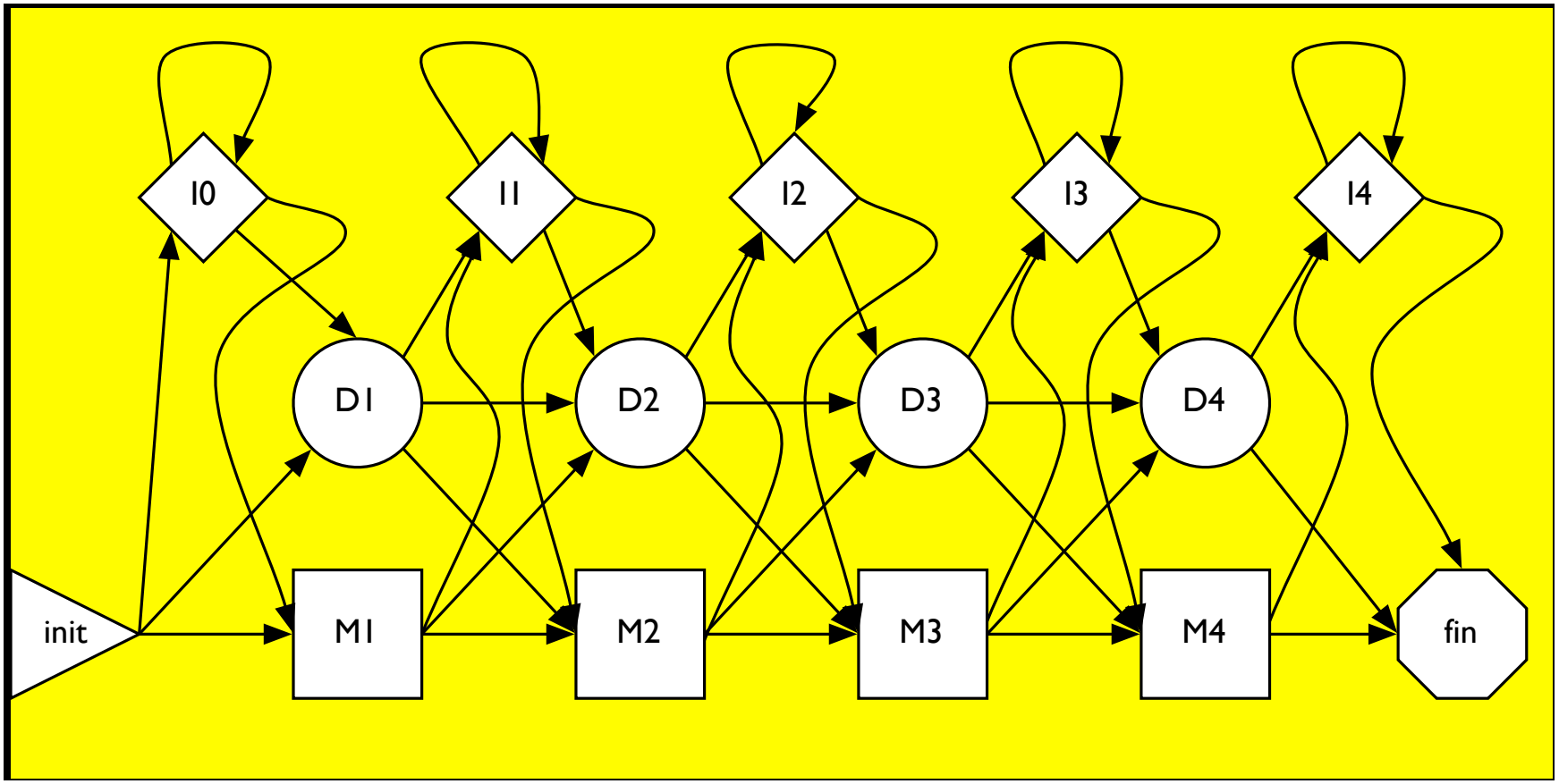
HMM de profile 2

2. Insertion de caractères



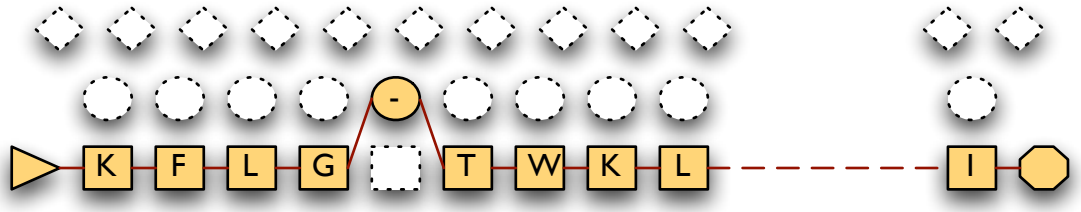
HMM de profile 3

3. Suppression de caractères



HMM DE PROFILE

Alignement à un HMM de profile : calculer le chemin Viterbi




```
Q9DAK4/4-132       QLQG.TWKSVSCDNF.ENYMKELGVGRASRK.LGCLAK.......PTVT
Q8QHA8/4-132       HFVG.TWKLLSSENF.EDYMKELGVGFATRK.MAGVAK.......PNLT
FABA_RAT/3-131     AFVG.TWKLVSSENF.DDYMKEVGVGFATRK.VAGMAK.......PNLI
MYP2_HUMAN/3-131   KFLG.TWKLVSSENF.DDYMKALGVGLATRK.LGNLAK.......PTVI
TLBP_MOUSE/4-132   PFLG.TWKLVSSENF.ENYVRELGVECEPRK.VACLIK.......PSVS
O57663/4-133       KFVG.TWKMISSDNF.DDYMKAIGVGFATRQ.VGNRTK.......PNLV
FABE_RAT/6-134     DLEG.KWRLVESHGF.EDYMKELGVGLALRK.MGAMAK.......PDCI
FABE_BOVIN/6-134   QLVG.RWRLVESKGF.DEYMKEVGVGMALRK.VGAMAK.......PDCI
```

Viterbi : HMM de profile

Quel bonheur : un petit nombre de transitions, et pas de boucles !
Difficulté : états sans émission

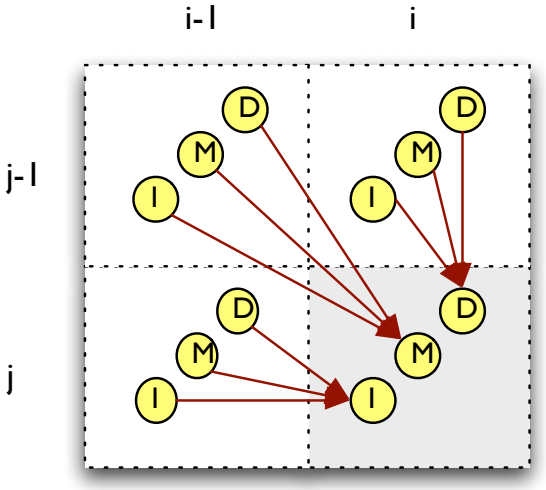
États de match : $M_j : j = 1, \ldots, n$
États d'insertion : $I_j : j = 0, \ldots, n$
États de suppression : $D_j : j = 1, \ldots, n$

Sous-problèmes pour PD : $\delta_j^{(M)}(i), \delta_j^{(I)}(i), \delta_j^{(D)}(i)$ pour émission de $s_1 \cdots s_i$ en finssant dans un état M_j, D_j, I_j.

Viterbi : HMM de profile 2

$$\delta_j^{(M)}(i) = p(M_j, s_i) \max\Big\{ \delta_{j-1}^{(M)}(i-1)\tau(M_{j-1}, M_j),$$
$$\delta_{j-1}^{(I)}(i-1)\tau(I_{j-1}, M_j),$$
$$\delta_{j-1}^{(D)}(i-1)\tau(D_{j-1}, M_j) \Big\}$$
$$\delta_j^{(I)}(i) = p(I_j, s_i) \max\Big\{ \delta_j^{(M)}(i-1)\tau(M_j, I_j),$$
$$\delta_j^{(I)}(i-1)\tau(I_j, I_j),$$
$$\delta_j^{(D)}(i-1)\tau(D_j, I_j) \Big\}$$
$$\delta_j^{(D)}(i) = \max\Big\{ \delta_{j-1}^{(M)}(i)\tau(M_{j-1}, M_j),$$
$$\delta_{j-1}^{(I)}(i)\tau(I_{j-1}, M_j),$$
$$\delta_{j-1}^{(D)}(i)\tau(d_{j-1}, M_j) \Big\}$$

VITERBI : HMM DE PROFILE 3



Cas de base (ali. global) : $\delta_0^{(I)}(0) = \tau(\mathsf{start}, I_0)$; $\delta_1^{(M)}(0) = \tau(\mathsf{start}, M_1)$; $\delta_1^{(D)}(0) = \tau(\mathsf{start}, D_1)$.

Fin : $\delta^{(\mathsf{fin})}(\ell) = \max\Big\{\delta_n^{(M)}(\ell)\tau(M_n, \mathsf{fin}), \delta_n^{(I)}(\ell)\tau(I_n, \mathsf{fin}), \delta_n^{(D)}(\ell)\tau(d_n, \mathsf{fin})\Big\}$.

Score de l'alignement à HMM

Logarithme de la probabilité pour générer une séquence $s_1 \cdots s_\ell$ sur un chemin $q_1 \cdots q_m$: c'est la somme des termes comme $\log(\tau(q_j, q_{j+1}))$ et $\log(p(q_j, s_i))$.

Contributions à $\log \delta_j(i)$:
$M_{j-1} \to M_j$: $\log \tau(M_{j-1}, M_j) + \log p(M_j, s_i)$ (substitution)
$M_j \to I_j$: $\log \tau(M_j, I_j) + \log p(I_j, s_i)$ (début d'un trou)
$I_j \to I_j$: $\log \tau(I_j, I_j) + \log p(I_j, s_i)$ (extension d'un trou)
etc.

$\log \tau$ et $\log p$ donnent les pondérations pour subsitutions ainsi que trou début, extension, et fin : tous les scores dépendent de la colonne j (comparez aux récurrences d'un alignement à un profile)

Score de l'alignement

Les probabilités définissent une pondération de l'alignement : alignement de score maximal (par log des probs) correspond au chemin Viterbi.

Avantage additionnel : évite l'underflow — probabilité pour un chemin long ne peut pas être représentée par une variable flottante p.e., $\epsilon_M = 10^{-324}$.

LODS

Hypothèse null : $s_1 \dots s_\ell$ est une séquence de caractères iid $\mathbb{P}\{s_i = \sigma\} = r_\sigma$ (souvent les mêmes probs d'émission aux états d'insertion).

Hypothèse altérnatif : $s_1 \dots s_\ell$ est générée par notre HMM de profile $\mathcal{M}$.

LODS (logarithme des chances) : comparer $p_0 = \mathbb{P}\left\{ s_1 \cdots s_\ell \;\middle|\; H_0 \right\}$ et $p_{\mathcal{M}} = \mathbb{P}\left\{ s_1 \cdots s_\ell \;\middle|\; H_1 \right\}$:

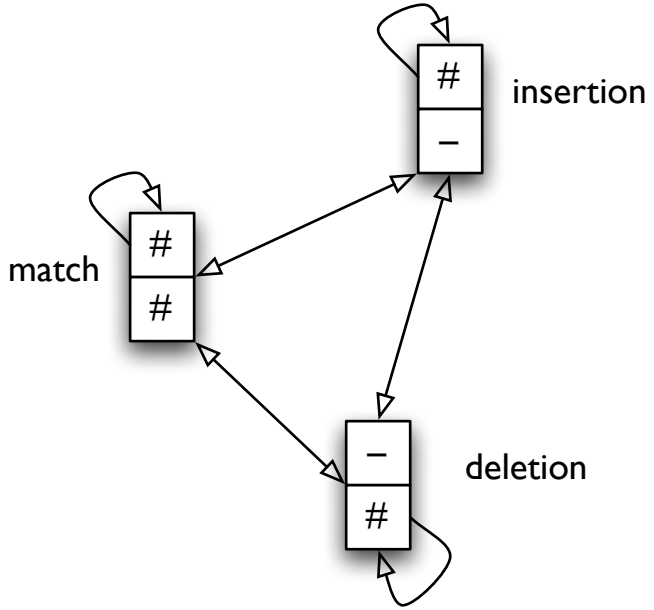
$$\mathsf{LODS} = \log \frac{p_{\mathcal{M}}}{p_0}$$

Calcul de LODS : par Viterbi, remplacer $p(q_j, s_i)$ par $\frac{p(q_j, s_i)}{r_{s_i}}$.

Pour des valeurs entières : $v = \left\lfloor \alpha \log_2 \frac{p}{r} \right\rfloor$ (r est la probabilité par H_0)

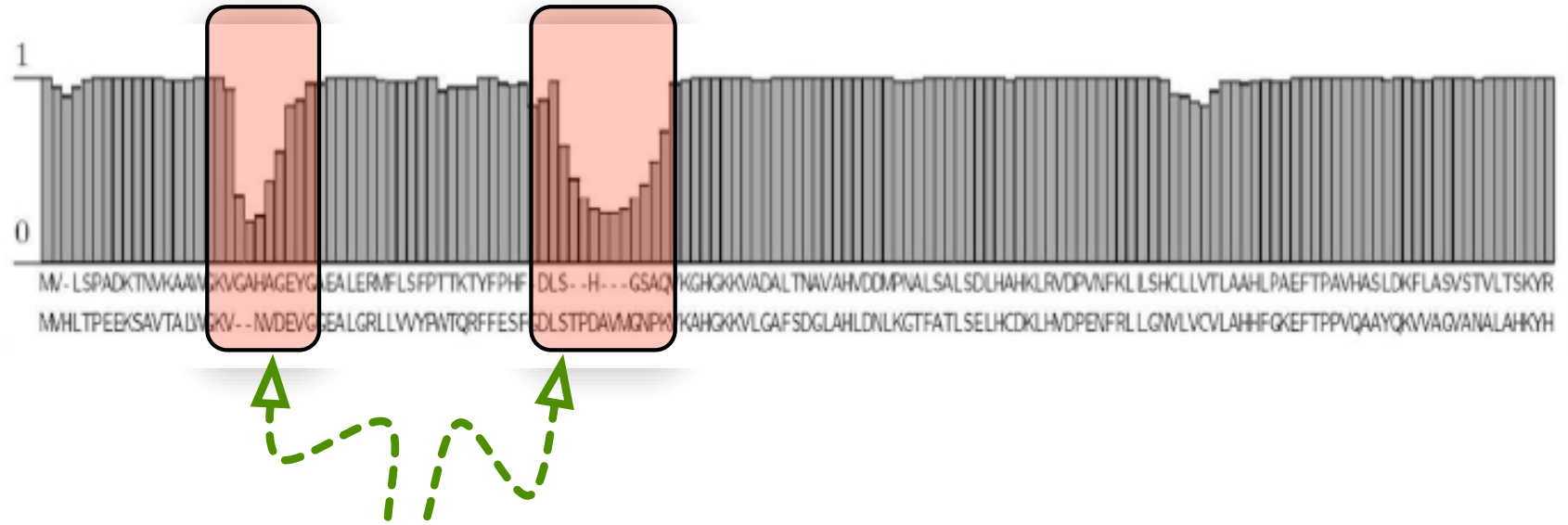
Machine d'alignement

HMM : émission de caractères (a, b) où $a, b \in \{\mathtt{A}, \mathtt{G}, \mathtt{T}, \mathtt{C}, -\}$ (sauf $a = b = -$)



Knudsen & Miyamoto *J. Mol. Biol.* 333 : 453 (2003)

Probabilité postérieure



Lunter et al., in Nielsen (ed.) *Statistical Methods in Molecular Evolution* (2005)

Construction d'un HMM de profile

- à partir de séquences : apprentissage

- à partir d'alignements (structuraux - très fiables) : utilisez les chemins [«seed alignments»] pour les paramètres

BD de HMMs de profile pour familles de protéines : Pfam

Apprentissage de paramètres

[Une approche de pseudo-compteurs.]

Si on a un alignement multiple, on identifie un chemin pour chaque séquence alignée.

- N nombre de séquences
- N_q nomre de chemins qui passent par q
- $N_{q \to q'}$ nombre de chemins avec une transition $q \to q'$
- $N_{q,\sigma}$ nombre de fois caractère σ est vu en état de match q
- x, y «pseudo-compteurs» (Laplace : $x = y = 1$)
- a_q nombre d'états q' avec $\tau(q, q') > 0$

$$\tau(q, q') = \frac{N_{q \to q'} + x}{N + a_q x} \qquad p(q, \sigma) = \frac{N_{q,\sigma} + y}{N_q + y|\Sigma|}$$

Émission dans un état d'insertion : selon les fréquences de «background»

Problème 3 — apprentissage

Sans alignements initiaux — c'est difficile

Échantillon de séquences : séquences à aligner

1 Initialiser τ, p and π.
2 **répéter**
3 calculer le chemin Viterbi pour les séquences de l'échantillon ;
4 recalculer τ, p, et π par les chemins Viterbi ;
5 **jusqu'à** l'optimum local est achevé.

+ détails de l'initialisation, méthodes numériques, etc.

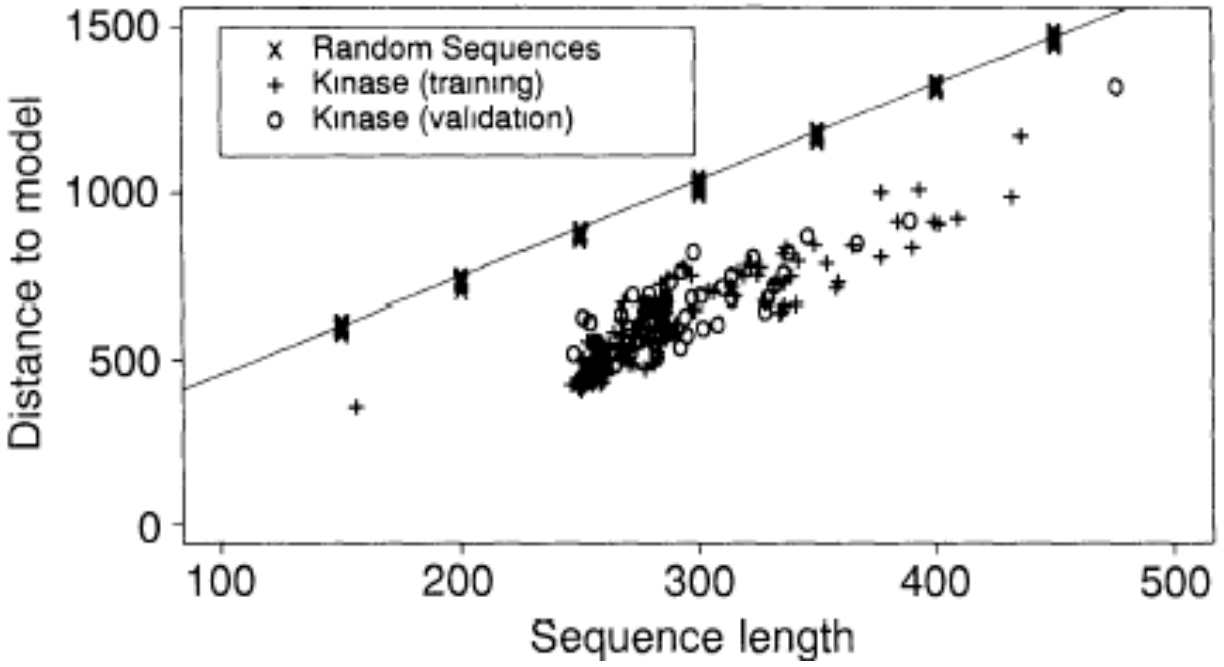
Baum-Welch : calculer tous les chemins pour toutes les séquences, les pondérer par leurs probabilités pour recalculation de τ, p, and π.

Classification

Pour une séquence s :
prendre profiles $\mathcal{M}_1, \ldots, \mathcal{M}_k$ et calculer les
vraisemblances $L_{\mathcal{M}_1}(s), \ldots, L_{\mathcal{M}_k}(s)$:
la meilleure* vraisemblance donne la classification de s.

* n'est pas aussi simple que ça
1. la vraisemblance dépend de la longueur - il faut normaliser
2. comparer les L des membres de la famille à celle d'autres protéines
pour avoir un seuil L_{famille} : les membres de la famille ont $L > L_{\text{famille}}$, autres ont $L < L_{\text{famille}}$.

CLASSIFICATION : EXEMPLE

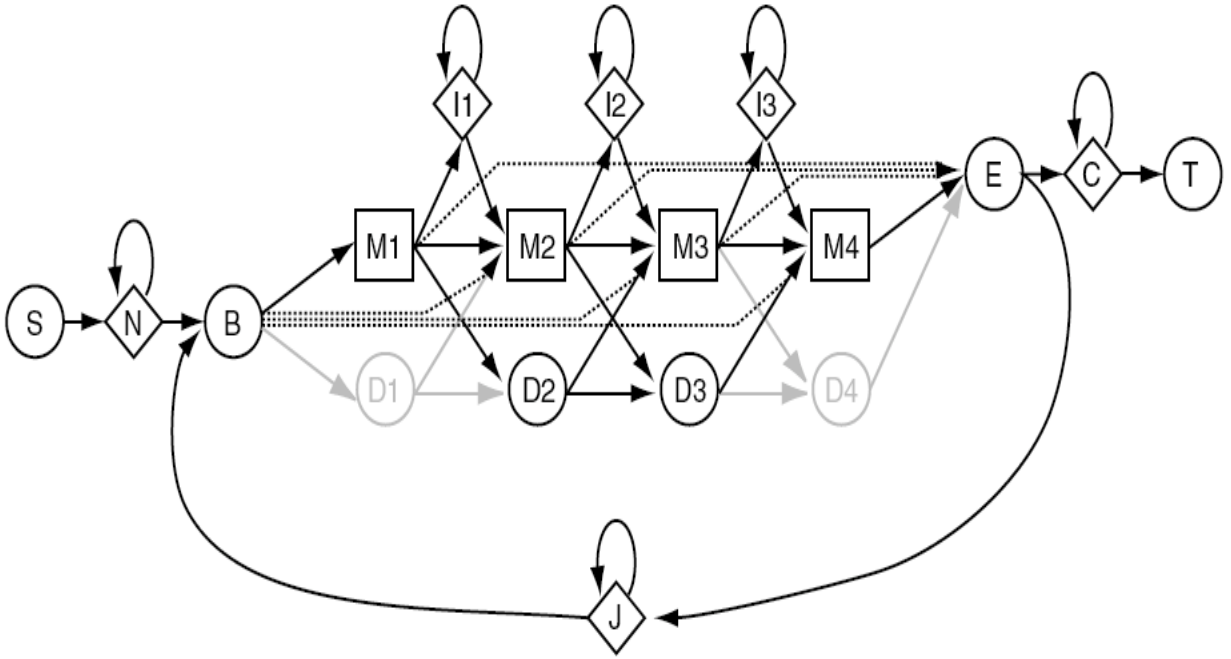


(Baldi et al. PNAS 91:1059)

(Y : – log vraisemblance, X : longueur de la séquence)

HMMs – EXTENSIONS

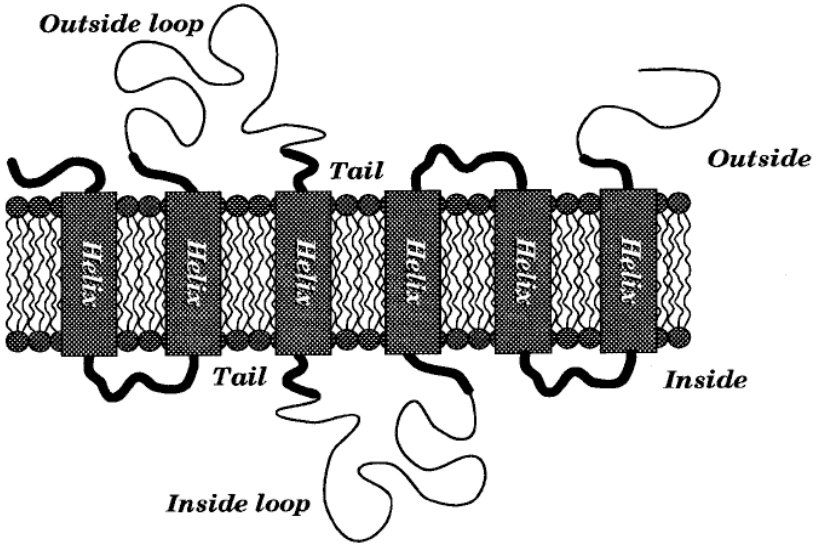
alignement local + plusieurs occurrences dans une séquence



(Eddy: HMMer manual v2.3.1)

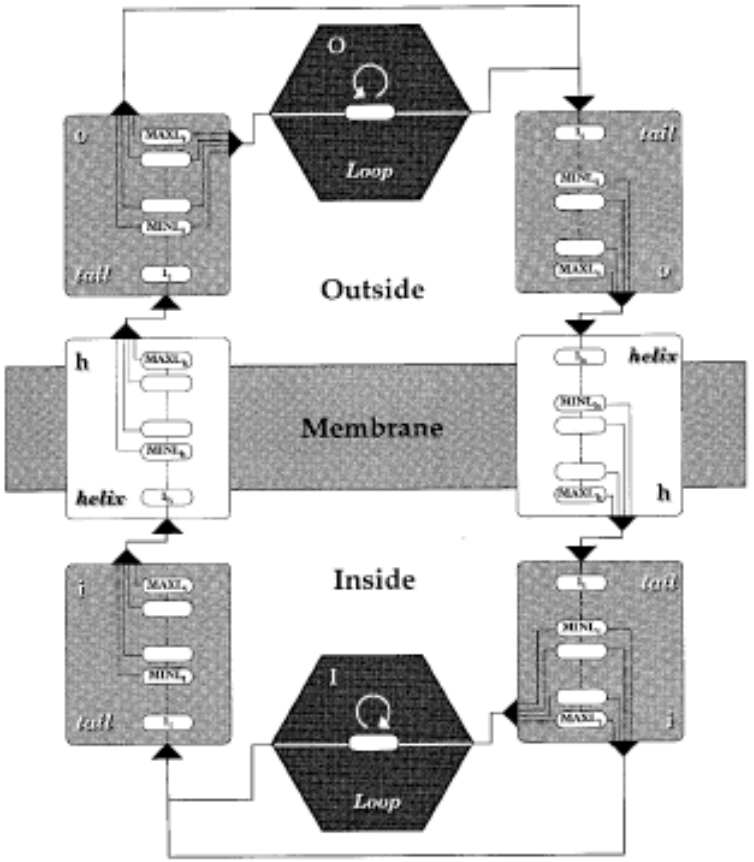
HMMs – architecture spécialisé

Exemple : protéines transmembranaires



Amino acid seq: MGDVCDTEFGILVA...SVALRPRKHGRWIV...FWVDNGTEQ...PEHMTKLHMM...
State seq: oooooooooohhhhh...hhhhiiiiiiihhh...hhhooooOO...OOOooooohhh...

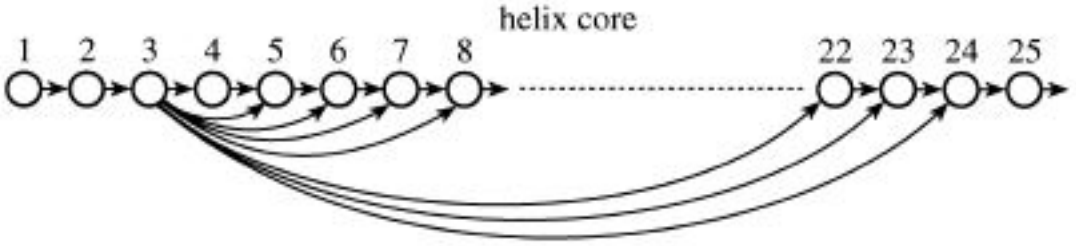
(Tusnády & Simon, J.Mol.Biol. 283:489)



(Tusnády & Simon, J.Mol.Biol. 283:489)

HMM – EXTENSIONS

duration de séjour dans un état : p.e., longueur du segment restreinte pour hélices dans le membrane



distribution générale : PD modélisant la duration de séjour explicitement (récurrence avec toutes les longueurs possibles)

Krogh et al. *J Mol Biol* 305 : 567 (2001)