Initiation à la Bioinformatique

Daniel Gautheret
ESIL, Université de la Méditerranée

V.2005.3 http://www.esil.univ-mrs.fr/~dgaut/Cours

Bioinformatique

Deux définitions possibles

- ★ Applications de l'informatique à la biologie (en anglais: *computational biology*)
- ★ Analyse de l'information biologique (en anglais: *bioinformatics*)

C'est cette bioinformatique que nous abordons ici.

L'information est:

- ★ La séquence
- ★ La structure
- ★ La fonction, les interactions etc.

Pour quoi faire?

La bioinformatique est d'abord utilisée pour identifier les gènes, étudier leur fonction et leur évolution.

- ★ "Fonction" peut être entendu dans un sens général (ATPase, RNA-Polymérase, etc.) ou dans un sens beaucoup plus précis, avec identification des résidus essentiels, éléments structuraux, sites de fixation aux ligands, site catalytiques, etc.

Par exemple…

- ★ Pour chercher chez un organisme modèle un gène homologue à un gène humain d'intérêt
- ★ Pour rechercher des gènes liés à la pathogénicité
- ★ Pour concevoir une expérience de mutagénèse dirigée sur une protéine
- ★ Pour trouver tous les gènes présents sur un chromosome/ génome/contig nouvellement séquencé

En fait, la liste des questions est illimitée

La déduction par homologie, ou le « dogme central » de la bioinformatique

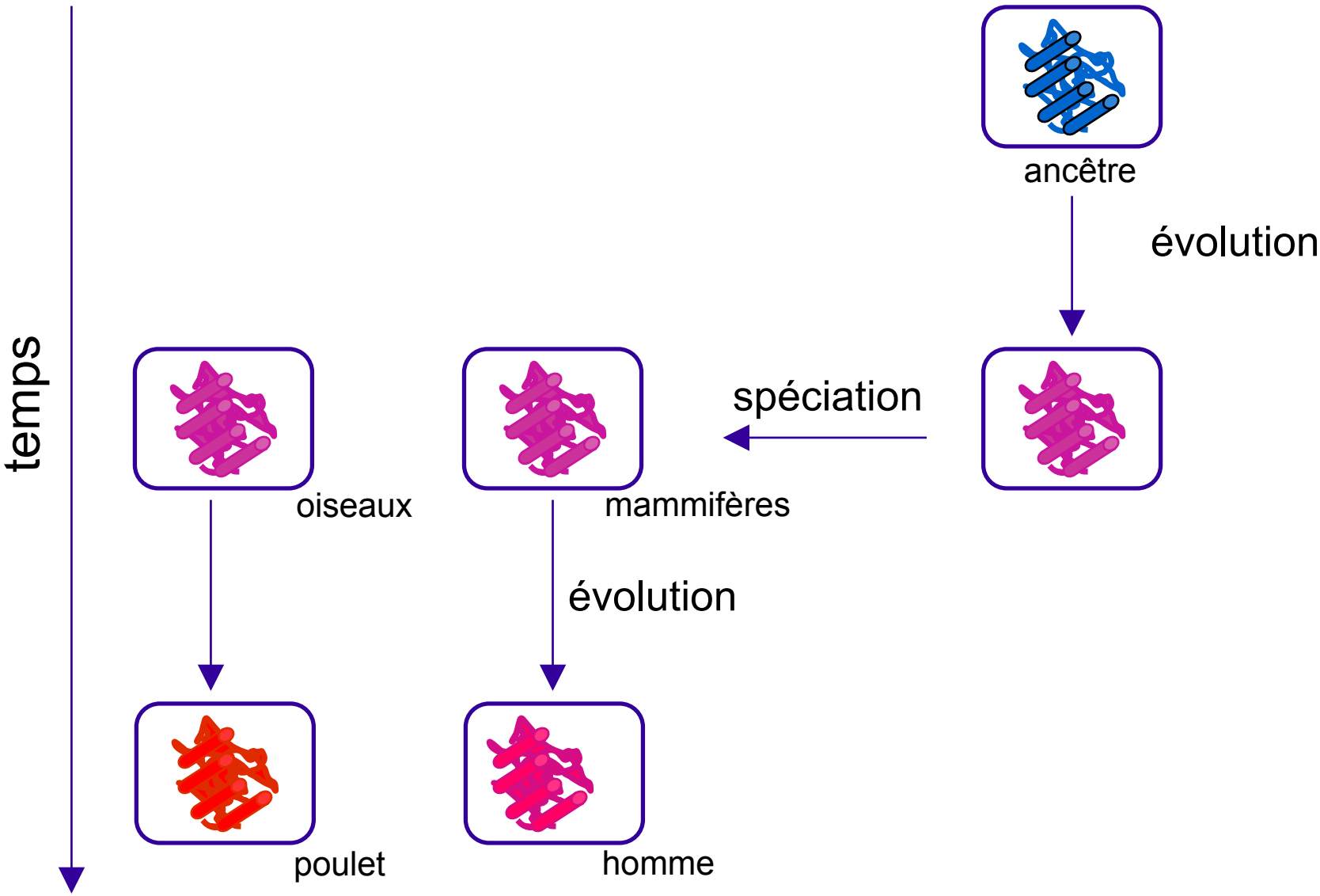
- ★ Si la bioinformatique « marche », c'est parce que l'évolution des gènes laisse une trace parfaitement visible lorsque l'on compare leur séquence
  - Evolution des gènes=mutations, insertions, délétion
  - Les gènes des organismes modernes sont issus de remaniement de gènes ancestraux: on peut donc déduire la fonction de la plupart des gènes par comparaison avec les gènes « homologues » d'autres espèces.
  - Les régions fonctionnelles des gènes (sites catalytique, de fixation, etc.) sont soumises à sélection. Elles sont relativement préservées par l'évolution car des mutations trop radicales sont désavantageuses.
  - Les régions non fonctionnelles ne subissent aucune pression de sélection et divergent rapidement à mesure que s'accumulent les mutations.

L'homologie de séquence

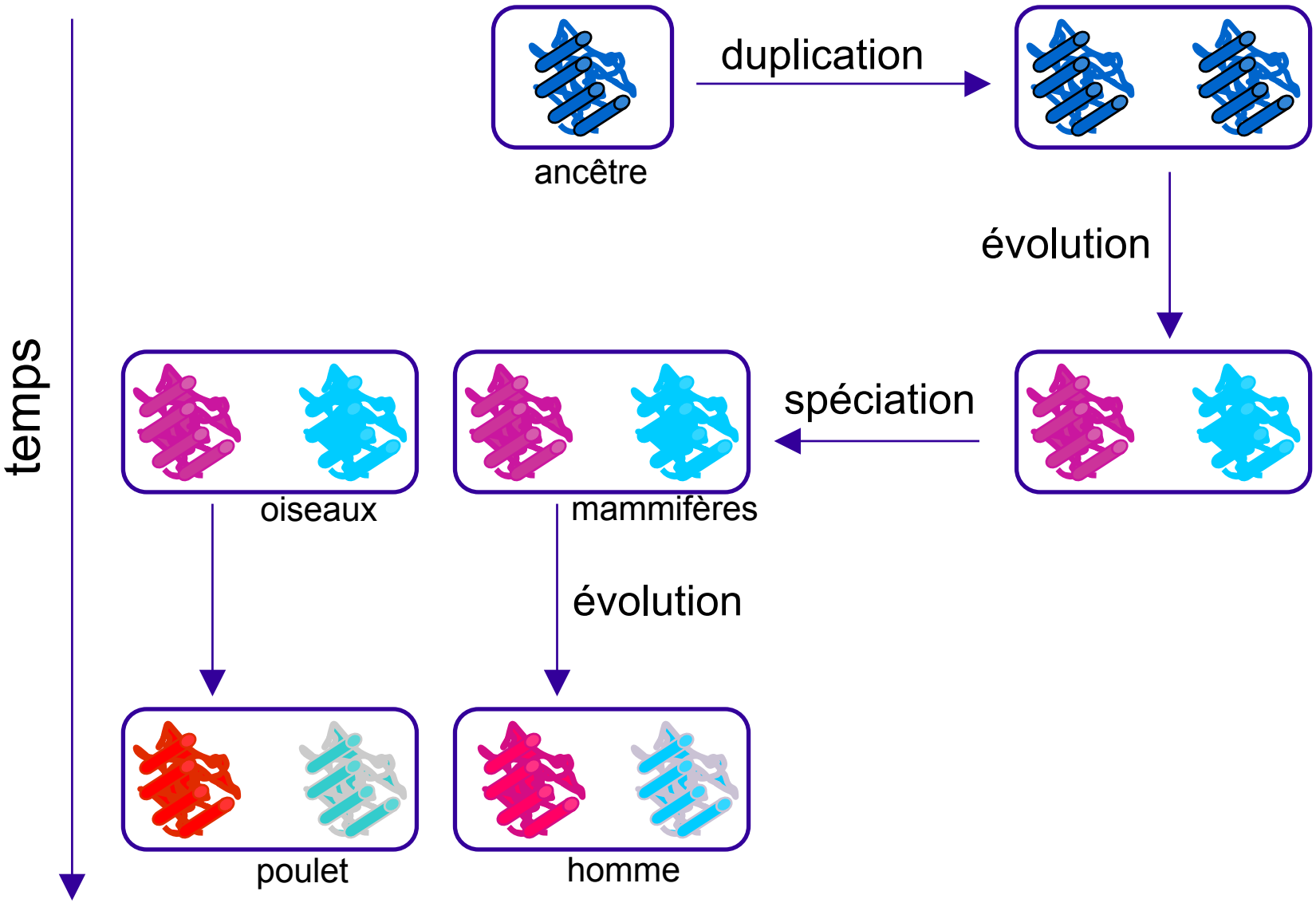
En bioinformatique: Homologie = parenté = ancêtre commun

- ★ Le bras humain est homologue à l'aile de l'oiseau
- ★ Le bras humain n'est pas homologue à l'aile de la mouche
- ★ On est homologue ou on ne l'est pas.
- ★ Donc on ne dit pas: "très homologue", "faible homologie", « 28% d'homologie », etc.
- ★ Pour une notion quantitative, on parle de similitude ("très similaire", etc.) ou d'identité (28% d'indentité)

Evolution d'un gène au cours de l'évolution

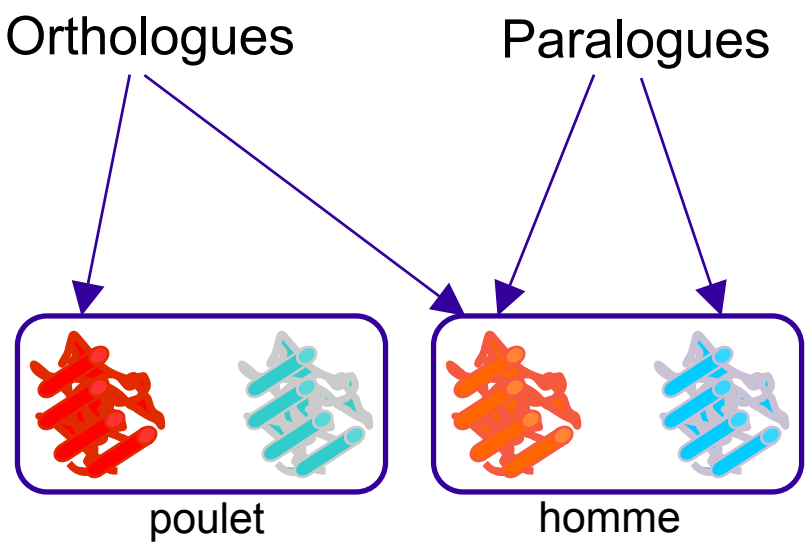


Apparition de nouveaux gènes par duplication



Paralogues et orthologues (Fitsh, 1970)

- ★ Homologues: gènes provenant d'un ancêtre commun
- ★ Paralogues: gènes homologues issus d'un phénomène de duplication
- ★ Orthologues: gènes homologues issus de la spéciation
- ★ Transfert horizontal: par endosymbiontes, etc. Fitsch a aussi introduit "xénologue" pour évoquer ce cas.



Fonction et homologie

- Homologie n'implique pas même fonction: par exemple l'aile de l'oiseau et le bras humain n'ont pas la même fonction
- Des orthologues rapprochés (p. ex. homme/souris) ont le plus souvent la même fonction dans l'organisme.
- Des orthologues distants (p. ex. homme/mouche) ont plus rarement le même rôle *phénotypique*, mais peuvent exercer le même rôle dans une *voie* donnée.
- Les paralogues acquièrent rapidement des fonctions différentes

A quel point des séquences homologues se ressemblent-elles?

- ★ De 100% à quelques nucléotides/aminoacides en commun.
- ★ Il n'y a pas vraiment de limite, mais en dessous d'un certain niveau d'identité (*twilight zone*), il devient difficile de distinguer une homologie d'une ressemblance fortuite. 2 séquences d'ADN prises au hasard ont 25% de nt communs.
- ★ Des séquence sans ressemblance apparente peuvent parfaitement être homologues (on le retrouve par ex. au niveau 3D)
- ★ Par contre, étant donné la dimension de l'espace des séquences possibles, une ressemblance importante est généralement interprétée comme une homologie, et non pas comme une évolution convergente.

Comment détecter une homologie?

Principe: comparaison de séquences

- ★ L'alignement des séquences est la principale méthode de comparaison. Elle permet d'identifier des régions conservées. On en déduit l'homologie.
- ★ D'autres méthodes existent:
  - Analyse statistique des « mots » contenus dans la séquence
  - Recherche de domaines ou motifs communs

Alignement de séquences

- Comparer des séquences serait relativement simple si elles avaient toutes la même longueur. Comme ce n'est pas le cas, il faut les aligner, c'est à dire trouver où se trouvent les insertions et délétions, représentées par des « indels » (« gaps »)

★ Distance d'édition

- Selon ce concept, le bon alignement est celui qui minimise les opérations à réaliser pour passer d'un séquence à l'autre.
- Opérations: conservation, remplacement/mutation, délétion, insertion. Une pénalité peut être affectée à chaque opération, par exemple c=0, m=1, d=2, i=2. La distance finale entre les deux séquences (distance d'édition) est la somme de ces pénalités.

	Seq 1	**CAGTGGT-GC**
	Seq 2	**CA-TCGTAGC**
	distance	**ccicmccdcc = 0+0+2+0+1+0+0+2+0+0 = 5**
Ou, variante:	ressemblance	**ccicmccdcc = 2+2-1+2-1+2+2-1+2+2 = 11**

c=0, m=1, d=2, i=2.

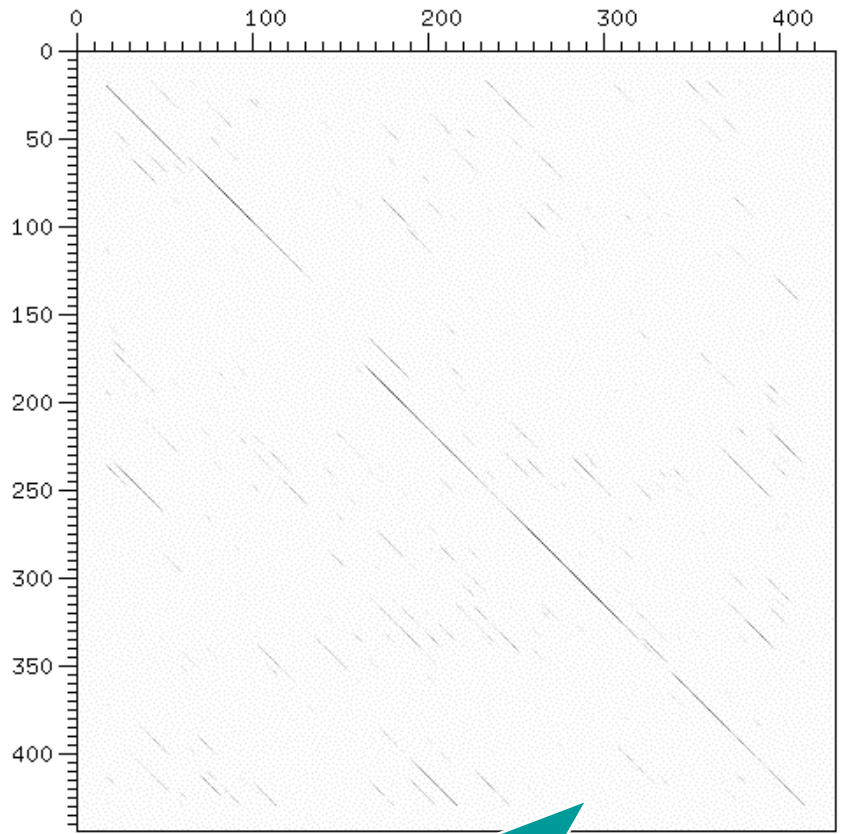
c=2, m=-1, d=-1, i=-1.

★ Comment trouver le meilleur alignement?

- Le nombre d'alignements possibles est trop élevé: on ne peut pas les essayer tous pour trouver celui qui minimise la distance (ou maximise la ressemblance).

Les « dot plots »

- Deux séquences à comparer sont représentées (ici 2 gènes de globine), une horizontalement, l'autre verticalement. On dessine ensuite un point dans la matrice lorsque les deux positions correspondantes sont identiques. Lorsque des régions se ressemblent, on voit apparaître une diagonale. Les décalages entre les diagonales correspondent à des insertions ou délétions. Plusieurs diagonales parallèles indiquent une répétition.
- Pour "nettoyer" le dot plot, on utilise souvent non pas un point par base, mais un point lorsque n bases sont identiques, ou n bases identiques dans une fenêtre de N. Cela réduit considérablement le nombre de points.
- Les dot plots sur des génomes complets permettent de visualiser les évènements à grande échelle, la synthénie, etc.



Alignement: trouver le meilleur chemin dans ce graphe

Algorithme de programmation dynamique

Needleman & Wunsh (1970).

Etape 1: Remplissage de la matrice

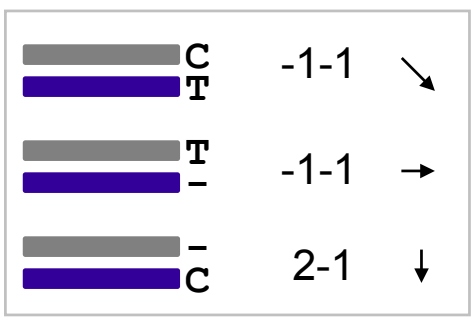
- ★ On veut aligner les séquences CAGTG et ACTCGT.
- ★ Une matrice de scores est remplie:

```
On définit les coût, par ex:
  v([-],[x]) = -1 [ins ou dél]
  v([x],[y]) = -1 (x ≠ y) [mut]
  v([x],[x]) = +2 [cons]

v ([0->i], [0->j]) = max {
  v ([0->i-1], [0->j-1]) + v ([i],[j])
  v ([0->i-1], [0->j ]) + v ([i],[-])
  v ([0->i ], [0->j-1]) + v ([-],[j]) }

L'amorçage se fait avec: v([],[])=0
et v([-],[x])=-1
```

Par ex. l'entrée colorée est obtenue par max (-4-1, -1-1, -5-1), c'est à dire -2.



i / j		C	A	G	T	G
	0	-1	-2	-3	-4	-5
A	-1	-1	1	0	-1	-2
C	-2	1	0	0	-1	-2
T	-3	0	0	-1	2	1
C	-4	-1	-1	-1	1	1
G	-5	-2	-2	1	0	3
T	-6	-3	-3	0	3	2

Exemple tiré du cours Bioinformatics & Computational Genomics du Weizmann Institute

Programmation dynamique

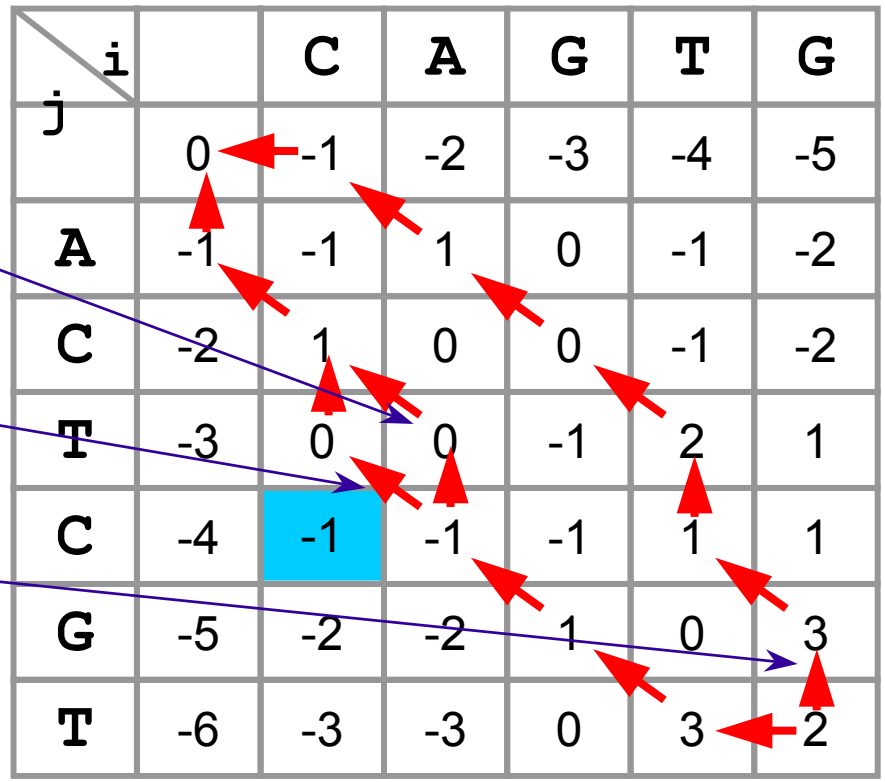
Etape 2: Reconstitution de l'alignement

- ★ Le chemin (alignement) optimal est déterminé par l'algorithme:
- ➢ en partant de la cellule (m,n), retrouver quelle cellule était responsable de la cellule courante (chemin suivi représenté par les traits rouges).
- ➢ Ici 3 Solutions:

```
ACTCGT-
-CA-GTG

ACTCGT-
-C-AGTG

-ACTCGT
CAGT-G-
```



i / j		C	A	G	T	G
	0	-1	-2	-3	-4	-5
A	-1	-1	1	0	-1	-2
C	-2	1	0	0	-1	-2
T	-3	0	0	-1	2	1
C	-4	-1	-1	-1	1	1
G	-5	-2	-2	1	0	3
T	-6	-3	-3	0	3	2

Matrices de Substitution

- Matrice 4X4 (nt) ou 20x20 (aa) décrivant la distance ou la similitude entre résidus.
- Estiment le coût ou le taux de remplacement d'1 résidu par un autre (distance).
- Le choix d'une matrice affecte fortement le résultat de l'analyse. Chaque matrice de score représente implicitement une théorie évolutive donnée

Matrices DNA

	A	C	G	T
A	2	-1	-1	-1
C	-1	2	-1	-1
G	-1	-1	2	-1
T	-1	-1	-1	2

Matrice identité

	A	C	G	T
A	3	-1	1	-1
C	-1	3	-1	1
G	1	-1	3	-1
T	-1	1	-1	3

Matrice transition/transversion

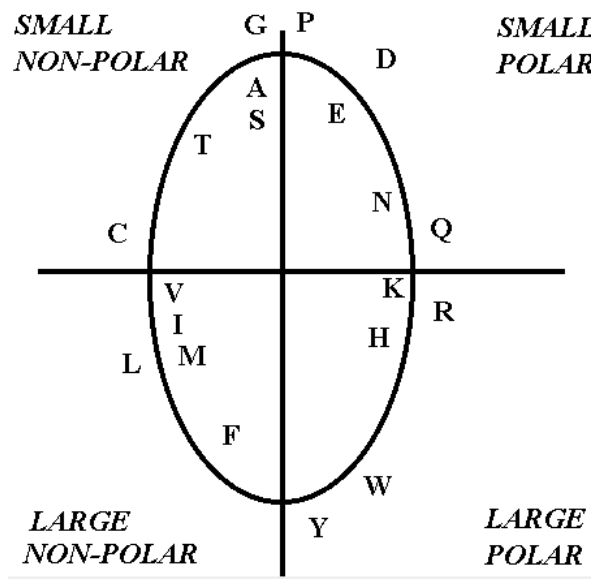
Matrices de Substitution

Matrices fondées sur le code génétique

- ★ Les scores sont déterminés en fonction du nombre commun de nucléotides présents dans les codons des acides aminés, ce qui revient à considérer le minimum de changements nécessaires en bases pour convertir un acide aminé en un autre.

Matrices fondées sur les propriétés physicochimiques

- ★ Les plus courantes sont celles basées sur le caractère hydrophile ou hydrophobe des protéines. Ces matrices sont peu utilisées.



Une représentation bidimensionnelle des propriétés des aa calculée d'après la matrice de Dayhoff par G. Vriend, Centre for Molecular and Biomolecular Informatics, University of Nijmegen

Matrices de Dayoff ou PAM

Margaret Dayhoff, 1978

- **PAM = Percentage of Accepted point Mutation**
- Probabilité d'observer la mutation X->Y après un temps évolutif donné.
  Basé sur alignement de protéines conservées à + de 85%.

Chaque case représente la probabilité de voir ces deux résidus remplacés l'un par l'autre dans un alignement. (matrice lod-score, de "log-odds" ou "log des chances").

•Un exemple de lod-score est:
S = log (Fij / (Fi x Fj))
Où Fij est la fréquence de remplacement du résidu i par j, et Fi et Fj sont les fréquences respectives des résidus i et j.

•Dans cette matrice de similitude, plus la valeur est négative, plus la probabilité est faible, plus le remplacement est rare.

•La table est valable pour une certaine distance évolutive.

•La distance est mesurée en PAM: nbre de mutations ponctuelles par 100 aa.

•2 Séquences séparées par une unité PAM: 1 mutation par 100 aa.

•Les valeurs sont déterminées initialement pour des protéines séparées de 6 à 100 PAM, puis extrapolées pour 150, 250 PAM, etc.

•Pour des protéines éloignées, on ne pourrait pas directement extrapoler à partir de valeurs tirées par ex. de PAM 10, car la *nature* des mutations change avec la distance évolutive. Le code génétique, par exemple, influence les mutations permises sur une courte durée, mais pas sur une longue durée.

Matrice de Dayoff (1979)

A	B	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y	Z	
0.4	0.0	-0.4	0.0	0.0	-0.8	0.2	-0.2	-0.2	-0.2	-0.4	-0.2	0.0	0.2	0.0	-0.4	0.2	0.2	0.0	-1.2	-0.6	0.0	A
	0.5	-0.9	0.6	0.4	-1.0	0.1	0.3	-0.4	0.1	-0.7	-0.5	0.4	-0.2	0.3	-0.1	0.1	0.0	-0.4	-1.1	-0.6	0.4	B
		2.4	-1.0	-1.0	-0.8	-0.6	-0.6	-0.4	-1.0	-1.2	-1.0	-0.8	-0.6	-1.0	-0.8	0.0	-0.4	-0.4	-1.6	0.0	-1.0	C
			0.8	0.6	-1.2	0.2	0.2	-0.4	0.0	-0.8	-0.6	0.4	-0.2	0.4	-0.2	0.0	0.0	-0.4	-1.4	-0.8	0.5	D
				0.8	-1.0	0.0	0.2	-0.4	0.0	-0.6	-0.4	0.2	-0.2	0.4	-0.2	0.0	0.0	-0.4	-1.4	-0.8	0.6	E
					1.8	-1.0	-0.4	0.2	-1.0	0.4	0.0	-0.8	-1.0	-1.0	-0.8	-0.6	-0.6	-0.2	0.0	1.4	-1.0	F
						1.0	-0.4	-0.6	-0.4	-0.8	-0.6	0.0	-0.2	-0.2	-0.6	0.2	0.0	-0.2	-1.4	-1.0	-0.1	G
							1.2	-0.4	0.0	-0.4	-0.4	0.4	0.0	0.6	0.4	-0.2	-0.2	-0.4	-0.6	0.0	-0.4	H
								1.0	-0.4	0.4	0.4	-0.4	-0.4	-0.4	-0.4	-0.2	0.0	0.8	-1.0	-0.2	-0.4	I
									1.0	-0.6	0.0	0.2	-0.2	0.2	0.6	0.0	0.0	-0.4	-0.6	-0.8	0.1	K
										1.2	0.8	-0.6	-0.6	-0.4	-0.6	-0.6	-0.4	0.4	-0.4	-0.2	-0.5	L
											1.2	-0.4	-0.4	-0.2	0.0	-0.4	-0.2	0.4	-0.8	-0.4	-0.3	M
												0.4	-0.2	0.2	0.0	0.2	0.0	-0.4	-0.8	-0.4	0.2	N
													1.2	0.0	0.0	0.2	0.0	-0.2	-1.2	-1.0	-0.1	P
														0.8	0.2	-0.2	-0.2	-0.4	-1.0	-0.8	0.6	Q
															1.2	0.0	-0.2	-0.4	0.4	-0.8	0.6	R
																0.4	0.2	-0.2	-0.4	-0.6	-0.1	S
																	0.6	0.0	-1.0	-0.6	-0.1	T
																		0.8	-1.2	-0.4	-0.4	V
																			3.4	0.0	-1.2	W
																				2.0	-0.8	Y
																					0.6	Z

W=Tryprophane (Cyclique)
C= Cysteine (Soufre)

Autres matrices de substitution

BLOSUM

- ★ Le but est de détecter des relations entre protéines plus éloignées.
- ★ Avec les matrices PAM, les valeurs pour des protéines éloignées sont extrapolées. Avec BLOSUM, ces valeurs sont obtenues en comparant des blocs facilement alignables (sans gaps) dans des familles de protéines très éloignées.
- ★ Ces matrices sont reconnues pour mettre en valeur les similitudes biologiquement importantes (celles qui sont présentes dans les régions alignées sans gaps).
- ★ BLOSUM62: faite à partir d'un alignement de séquences ayant 62% de similitude, BLOSUM45: 45%, etc.

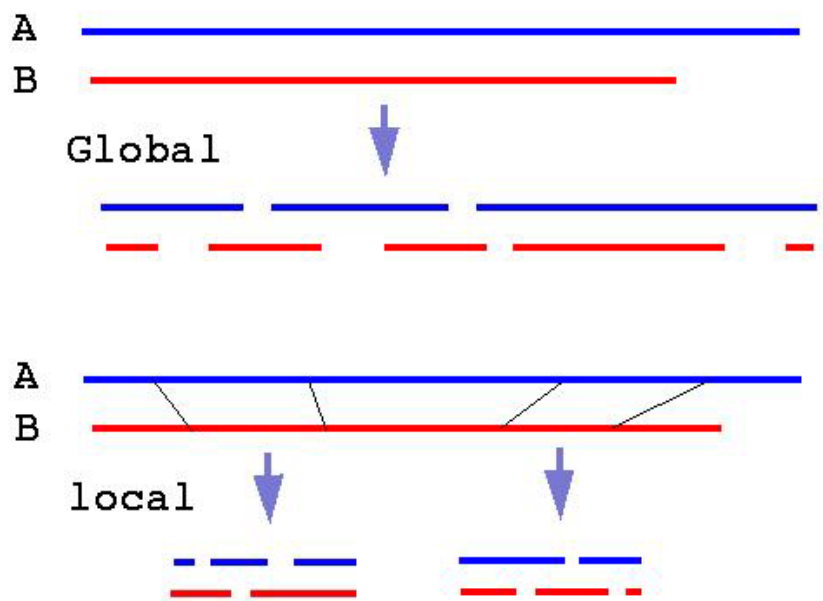
Matrices d'après alignement 3D

- ★ Basées sur la structure secondaire ou tertiaire. Evaluent la propension d'un acide aminé à adopter une certaine conformation. Fiables car fondées sur le meilleur alignement possible. Encore incomplètes en raison de la taille des banques de données 3D.

Alignement local ou global

Des finalités très différentes:

- ★ l'alignement global est conçu pour comparer des séquences homologues sur toute leur longueur.
- ★ L'alignement local est conçu pour rechercher des régions semblables entre A et B.



Les programmes d'alignement global

- ★ Méthode employée pour aligner des séquences dont on soupçonne l'homologie. L'alignement est optimisé sur toute la longueur des séquences. L'algorithme de référence est celui de Needleman & Wunsh (1970).
- ★ Utilisé principalement aujourd'hui dans le cadre de l'alignement multiple (voir plus loin)

Les programmes d'alignement local

- Aligne seulement les régions dont le score est supérieur à un seuil donné. Utilisé lorsque l'on veut aligner deux séquences de taille très différente. (par ex. dans une recherche de sous-séquence). Beaucoup plus rapide que l'alignement global.

★ **Smith-Waterman**

- Programmation dynamique avec arrêt de la procédure quand le score devient trop faible. Selection du meilleur alignement local.

★ **Fasta (Lipman & Pearson, 1985)**

- Heuristique: recherche d'abord des segments de longueur k exactement semblables (k-mots), raccorde ces segments si présents sur une même diagonale où sur des diagonales proches, puis réaligne la région par programmation dynamique. Une seule solution par couple de séquences comparées.

Blast (Lipman, Karlin, Altschul, 1990)

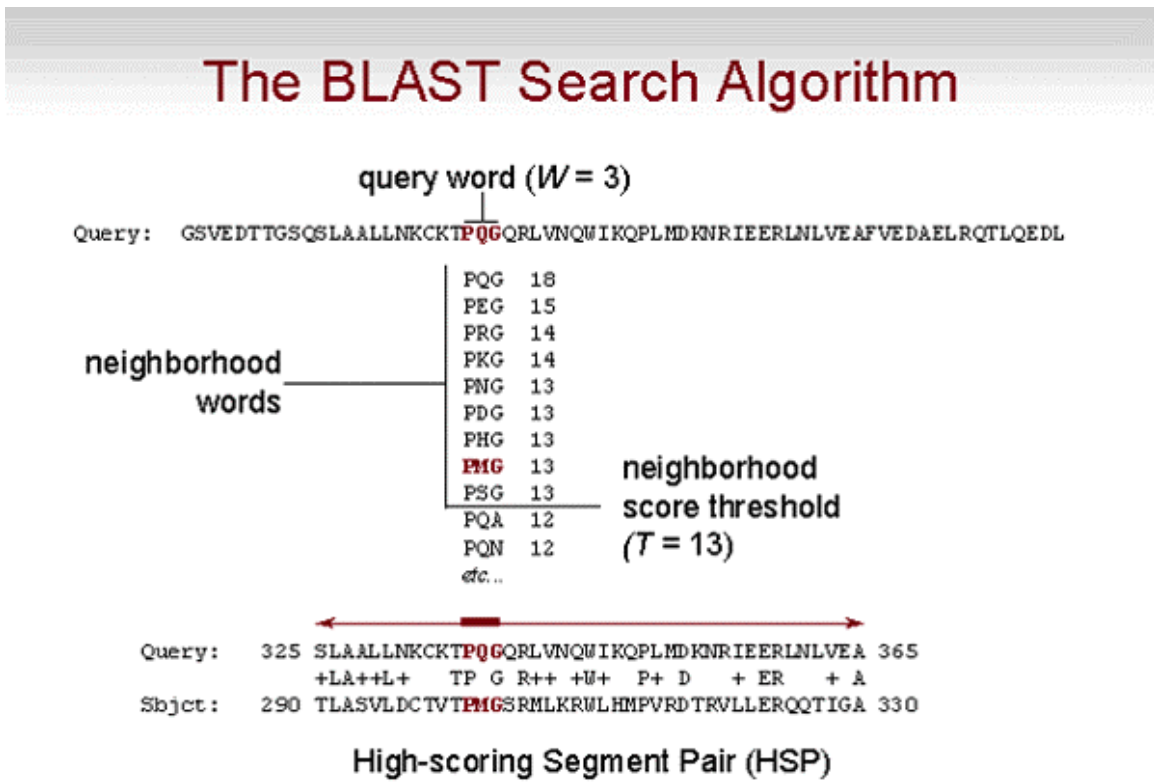
★ Le plus utilisé des programmes d'alignement local

- k-mots également, mots approchés permis au dessus d'un certain score.
- Pré-codage de la base de données et de la requête pour recherche plus rapide des k-mots.
- Version 1: sans Gaps
- Version 2: avec Gaps

★Points forts

•Rapidité

•Calcul de la valeur statistique des scores.

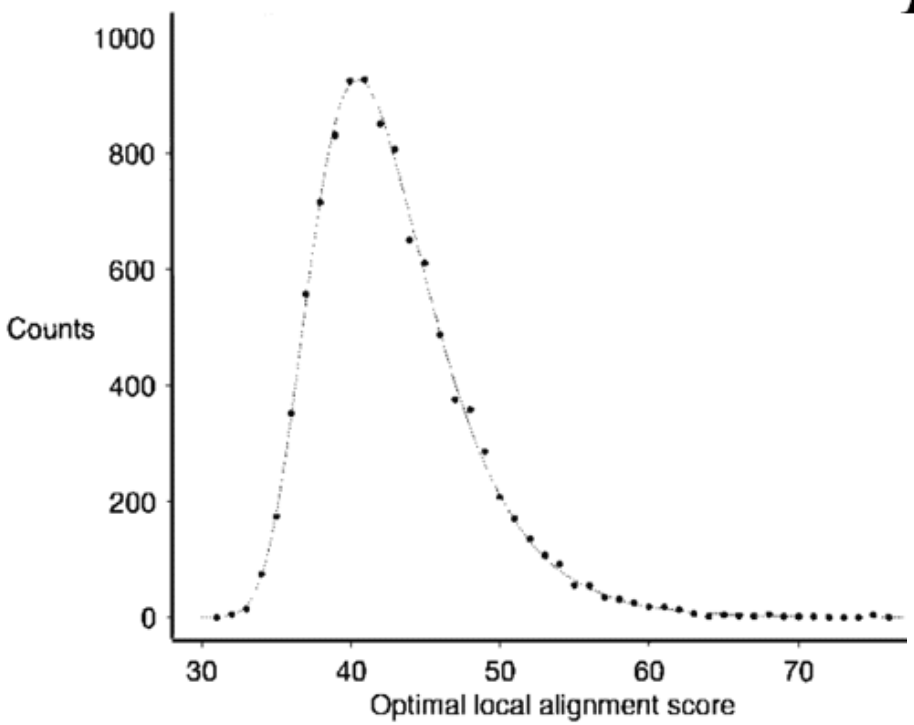


Statistiques de Blast

Expectation value (E-value): nombre de solutions attendues par chance avec un score *S* ou plus

- En raison de l'algorithme d'alignement, qui recherche le meilleur score possible pour une position donnée, les scores des HSP ne suivent pas une distribution normale, mais une **distribution des valeurs extrêmes**. En comparant 2 séquences de longueurs n et m, le nombre attendu de HSP ayant un score S ou plus est défini par:

$$E = Kmne^{-\lambda S}$$



où K et lambda sont des paramètres statistiques dépendant du système de score et de la composition de fonds en aminoacides. Blast estime ces paramètres a priori pour les différents systèmes de score (BLOSUM62, etc.). Pour un alignement sans gaps, K et lambda peuvent être calculés. Pour un alignement avec gaps, il a fallu recourir à des simulations sur un grand nombre de séquences aléatoires.

exemple de tracé de scores d'alignement optimaux

Statistiques de Blast

P value

- La probabilité de trouver au moins un HSP de score >= S est:

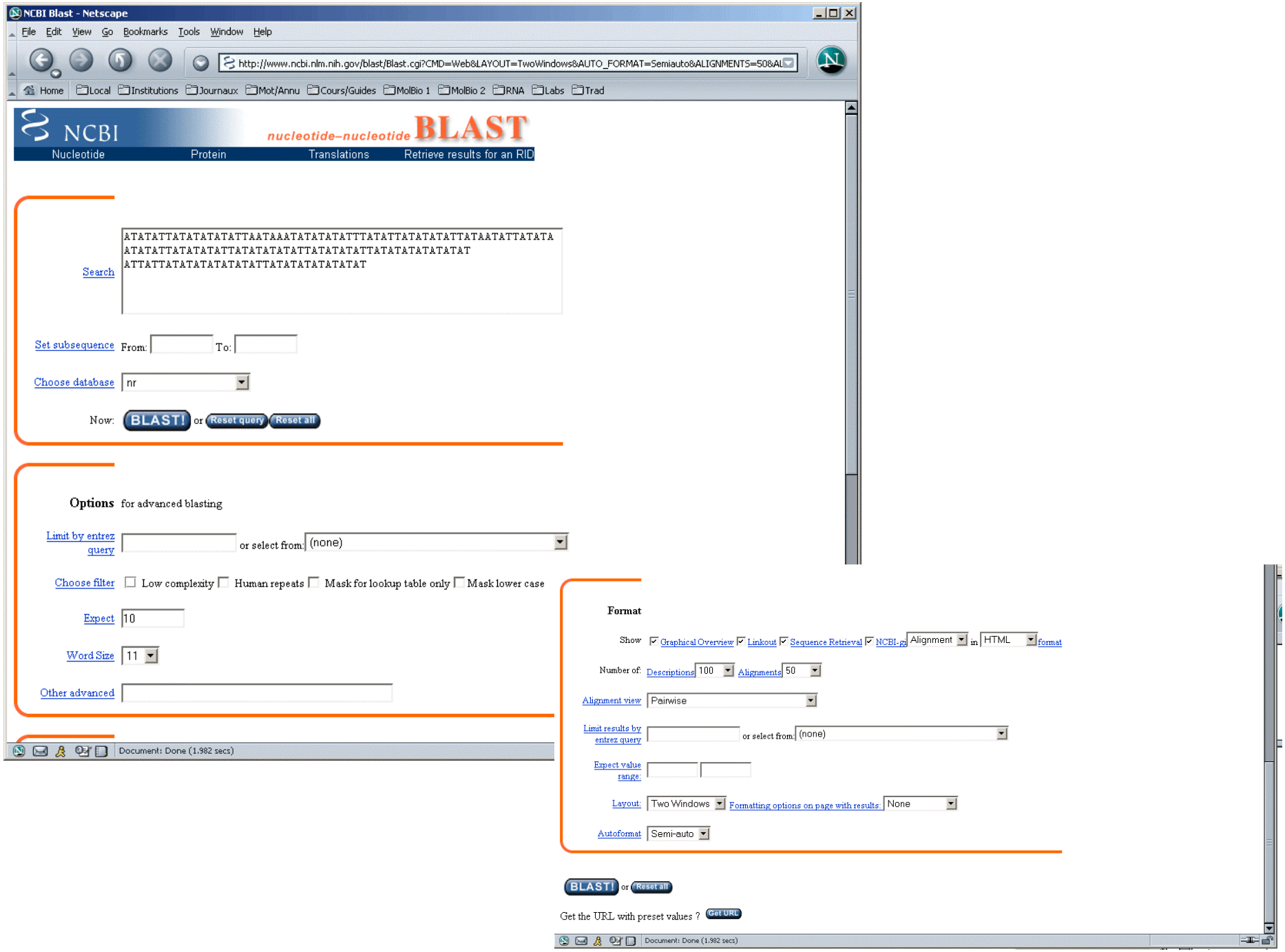
$$P = 1 - e^{-E} = 1 - \exp(-Kmn\, e^{-\lambda x})$$

Recherche dans les banques

- Les équations de E et P s'appliquent à la comparaison de 2 séquences. Si l'on compare une séquence à une banque en contenant un grand nombre, les chances d'obtenir un certain score sont bien sûr plus élevées. Blast fait comme si la recherche s'effectuait dans une longue séquence de longueur N (longueur totale de la banque), en tenant compte en outre des effets de bordure (en raison de leur longueur, les séquences "requêtes" ne peuvent arriver trop près des bords des séquences de la base).

Blast en pratique

- ★ Visiter Le serveur Blast du NCBI:
- ★ Programs: *blastn:* AN contre AN *blastp:* Prot contre Prot *blastx:* AN 6 cadres contre prot *tblastn:* Prot contre AN 6 cadres *tblastx:* AN 6 cadres contre AN 6 cadres
- ★ Database: nr (non redondant) est automatiquement selectionné en version "protéines" ou "acides nucléiques" selon qu'on utilise blastp ou blastn.
- ★ Masquage: les régions risquant de produire des solutions non spécifiques peuvent-être remplacées par des X
  - Régions de basse complexité
  - Séquences répétées eucaryotiques


NCBI Blast - Netscape
File Edit View Go Bookmarks Tools Window Help
http://www.ncbi.nlm.nih.gov/blast/Blast.cgi?CMD=Web&LAYOUT=TwoWindows&AUTO_FORMAT=Semiauto&ALIGNMENTS=50&AL
Home Local Institutions Journaux Mot/Annu Cours/Guides MolBio 1 MolBio 2 RNA Labs Trad
NCBI
nucleotide–nucleotide BLAST
Nucleotide Protein Translations Retrieve results for an RID
Search
ATATATTATATATATATTAATAAATATATATATTTATATTATATATATTATAATATTATATA
ATATATTATATATATTATATATATATTATATATATTATATATATATATAT
ATTATTATATATATATATATTATATATATATAT
Set subsequence From: To:
Choose database nr
Now: BLAST! or Reset query Reset all
Options for advanced blasting
Limit by entrez query or select from: (none)
Choose filter Low complexity Human repeats Mask for lookup table only Mask lower case
Expect 10
Word Size 11
Other advanced
Document: Done (1.982 secs)
Format
Show Graphical Overview Linkout Sequence Retrieval NCBI-gi Alignment in HTML format
Number of: Descriptions 100 Alignments 50
Alignment view Pairwise
Limit results by entrez query or select from: (none)
Expect value range:
Layout: Two Windows Formatting options on page with results: None
Autoformat Semi-auto
BLAST! or Reset all
Get the URL with preset values ? Get URL
Document: Done (1.982 secs)

Sortie de Blast

Query=
(734 letters)

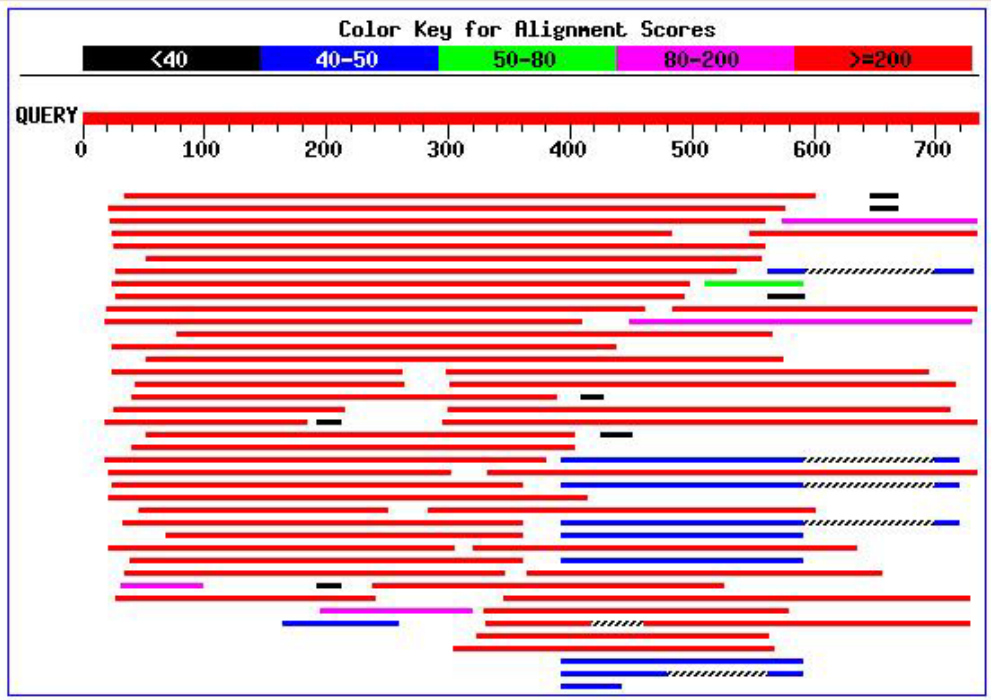
Database: Non-redundant Database of GenBank EST Division
1,938,225 sequences; 736,227,809 total letters

Searching.....done

If you have any problems or questions with the results of this search please refer to the **BLAST FAQs**

Distribution of 76 Blast Hits on the Query Sequence

Mouse-over to show defline and scores. Click to show alignments



Sortie de Blast

```
gb|AA421570|AA421570  zu25d04.s1 Soares NhHMPu S1 Homo sapiens c...  765  0.0
gb|AA670218|AA670218  ad19g11.s1 Soares NbHFB Homo sapiens cDNA ...  739  0.0
gb|AA628778|AA628778  af42c05.s1 Soares total fetus Nb2HF8 9w Ho...  722  0.0
gb|AI246150|AI246150  qi29a04.x1 Soares_NhHMPu_S1 Homo sapiens c...  704  0.0
gb|AA588195|AA588195  no23h07.s1 NCI_CGAP_Pr22 Homo sapiens cDNA...  690  0.0
gb|AI143969|AI143969  qe01c10.x1 Soares_testis_NHT Homo sapiens ...  690  0.0
gb|AA977500|AA977500  on60d06.s1 Soares_NFL_T_GBC_S1 Homo sapien...  680  0.0
gb|AA779757|AA779757  af44e01.s1 Soares total fetus Nb2HF8 9w Ho...  642  0.0
gb|AA487274|AA487274  aa94e08.s1 Stratagene fetal retina 937202 ...  624  e-177
gb|AA953515|AA953515  on80a09.s1 Soares_NFL_T_GBC_S1 Homo sapien...  624  e-177
gb|AA129683|AA129683  zn91b03.s1 Stratagene lung carcinoma 93721...  617  e-175
gb|AA938880|AA938880  op74c01.s1 Soares_NFL_T_GBC_S1 Homo sapien...  585  e-165
ares_NFL_T_GBC_S1 Homo sapien...  585  e-165
ares parathyroid tumor NbHPA ...  575  e-162
ares_NFL_T_GBC_S1 Homo sapien...  533  e-150
I_CGAP_Thy1 Homo sapiens cDNA...  398  e-109
apiens cDNA clone 286077 3'.      274  1e-71
ratagene fetal retina 937202 ...  202  3e-50
ares NhHMPu S1 Homo sapiens c...  200  1e-49
ares NhHMPu S1 Homo sapiens c...  200  1e-49
ratagene lung carcinoma 93721...  151  1e-34

gb|N92166|N92166 yz89b07.r1 Homo sapiens cDNA clone 290197 5'.
          Length = 479

 Score = 52.0 bits (26), Expect = 7e-05
 Identities = 89/105 (84%), Positives = 89/105 (84%), Gaps = 9/105 (8%)


Query: 7   ccctaatttgtagccagtccaacccttcattccttggaggatttagttttgggataaaa 66
           ||| |||||||||||||| |  |||||||| | |||||||||||| |||||||  |||
Sbjct: 384 cccnaatttgtagccagtcaa--cctttcatnc-ttggaggatttag-tttggga-naaa 438


Query: 67  attttggtcccttgggcacagagacattccactattaatgaagta 111
           || ||||   ||||||||||||||||   |||||||||||||
Sbjct: 439 atnttgg--ncttgggcacagagacat--nactattaatgaagta 479


 gb|AI044766|AI044766 UI-R-C1-kb-g-03-0-UI.s1 UI-R-C1 Rattus norvegicus cDNA
          UI-R-C1-kb-g-03-0-UI 3'
          Length = 432

 Score = 44.1 bits (22), Expect = 0.017
 Identities = 31/34 (91%), Positives = 31/34 (91%)


Query: 243 gaaaaaatttttggtaaacagtattgtgtaaaat 276
           |||||| ||| ||||||||||||| ||||||||||
Sbjct: 295 gaaaaagtttatggtaaacagtaatgtgtaaaat 262


 Score = 40.1 bits (20), Expect = 0.27
 Identities = 20/20 (100%), Positives = 20/20 (100%)


Query: 391 tgtgtaaatttaatataaca 410
           ||||||||||||||||||||
Sbjct: 135 tgtgtaaatttaatataaca 116
```

A connaître:

- Score
- Identities
- Expect
- Query
- Subject

Exemples usage Blast (suivre les liens)

- Blastx cDNA / Prot
- Blastn cDNA / Gb
- Blastn cDNA / dbEST
- Pieges: vecteurs
- Pièges: basse complex.
- Pièges: Alu
- Pièges: transmembrane

- Caractériser le gène correspondant à un cDNA
- Trouver d'autres cDNA ou mRNA semblables
- Trouver des EST correspondant à un cDNA (p.ex. pour profil d'expression)
- Attention: il reste souvent des séquences de vecteurs attachées aux ARNm
- Séquences riches en AT, riches en résidus hydrophobes, etc.
- Séquences répétées=40% du génome humain
- Séquences transmembranaires (hydrophobes): même comportement que basse complexité

Vaut-il mieux comparer les protéines ou l'ADN pour rechercher des homologues d'une séquence?

- La meilleure façon de détecter des similitudes entre séquences est généralement la *comparaison au niveau protéique.*
  1. Il existe 20 aa contre 4 bases. La probabilité de trouver une "lettre" donnée par hasard est donc plus importante pour les bases.
  2. Plusieurs codons produisent le même aa. 134 / 549 substitutions de bases sont synonymes. Les séquences protéiques sont plus informatives.
  3. La raison principale est en fait l'existence d'outils de comparaison plus puissants pour les aa: utilisation des propriétés physicochimiques ou des substitutions observées dans l'évolution. Même lorsque les aa sont différents, on est capable de retrouver des similitudes. On en est tout à fait incapable au niveau des bases.
- Il existe en fait des cas où la séquence d'ADN est plus conservée que la séquence protéique, ce qui enlève du poids à l'argument 1
- Les comparaisons avec les séquences protéiques ne permettent de détecter que les régions codantes. Evidemment, on utilisera toujours la séquence ADN/ARN pour analyser ce qui n'est pas traduit!

Ensembles de séquences / Alignement multiple

Pourquoi analyser des ensembles de séquences?

L'ensemble de séquence (représenté généralement par un alignement multiple) révèle des aspects que l'on ne pouvait pas visualiser en comparant 2 séquences

- ★ Identifier les acides aminés essentiels,
- ★ Identifier les domaines
- ★ Etablir les signatures de chaque famille de protéines
- ★ Etablir la phylogénie des séquences, et même parfois des organismes, Distinguer paralogues et orthologues
- ★ Comme une aide à la modelisation: Les algorithmes de prediction de structures secondaires exploitent beaucoup mieux les alignements multiples. Connaitre les aminoacides permis à telle ou telle position facilite l'inférence 3D.

Représenter un ensemble de séquences

- ★ Lorsqu'on travaille sur un ensemble de séquences homologues (contenant un motif ou un domaine conservé), on cherche souvent à décrire rigoureusement ce qui est important dans ces séquences (pour déterminer les résidus fonctionnels, pour rechercher d'autres instances de la fonction, etc.)
- ★ Mais comment représenter ce qui est important?

Consensus

- ★ A partir d'un alignement, on détermine les résidus les plus fréquents à chaque position. Si la fréquence dépasse un certain seuil: séquence inclue dans le consensus. P. ex. consensus 90%:
- ★ Faible spécificité / sensibilité

Représenter un ensemble de séquences

Expression régulière

- ★ Un chaine de caractères décrivant un ensemble des séquences, avec des alternatives possibles à chaque position. c'est la méthode utilisée dans PROSITE. Exemple de descripteur PROSITE:
- ★ [AC]-x-V-x(4)-{ED} x: N'importe quel aa
  []: choix entre plusieurs aa
  {}: Tous, sauf les aa mentionnés
  (x,y): Répétition x à y fois
- ★ Semblables en principe, les langages utilisés dans Prosite et dans les expressions régulières Unix diffèrent dans les détails.

Expressions régulières Unix:

^ Le début d'une ligne
. Tout caractère (sauf newline)
$ La fin d'une ligne
| Choix. A|B: A ou B
() groupement
[] Classe de caracteres. [AGUC]: A,G,U ou C
\ Avant un caractère spécial
* 0 fois ou plus
+ une fois ou plus
? une fois ou zero
{n} exactement n fois
{n,} au moins n fois
{n,m} de n a m fois

Représenter un ensemble de séquences

Profil ou Matrice score-position (*Position Weight Matrix*)

- Plus subtil que les consensus: Pour chaque position de l'alignement, on détermine la fréquence d'observation des différents résidus.
- Ceci est résumé dans un tableau qui donne pour chaque position les fréquences des 20 a.a. (ou 4 bases)
- Une matrice de score est calculée à partir du tableau, selon la formule:
  $S_{b,i} = log(F_{b,i} / F_b)$ (Fb est la fréquence observée dans le génome analysé)
- La recherche est effectuée en faisant glisser une fenêtre sur la séquence à analyser et en calculant le score total à chaque position de la fenêtre.
- Une banque de profils interrogeable est disponible à l'ISREC: Profilescan
- La banque de motifs Prosite est également distribuée sous forme de profils.

Représenter un ensemble de séquences

Profils…

	A	G	G	A	T	C	T	C	T
	A	A	C	C	A	C	G	G	A
	A	A	C	G	T	C	G	C	A
	A	G	G	T	A	C	T	G	T
	A	A	C	A	T	C	A	A	T
	A	A	G	T	T	C	T	C	T
A	6	4	0	2	2	0	1	1	2
C	0	0	3	1	0	6	0	3	0
G	0	2	3	1	0	0	2	2	0
T	0	0	0	2	4	0	3	0	4

log(0): remplacé par pénalité fixe ou selon modèle

log(Fb,i /Fb)

0,6	0,43	####	0,12	0,12	####	-0,18	-0,18	0,125
####	####	0,3	-0,18	####	0,6	####	0,3	######
####	0,12	0,3	-0,18	####	####	0,12	0,12	######
####	####	####	0,12	0,43	####	0,3	####	0,426

A A C C A C G G A A A C C A C G G A A A C C

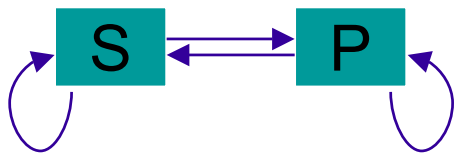
+0,6 -10 +0,3 -0,18 +0,12 -10 -0,18 +0,3 -10

Score=-29

Chaînes de Markov (MM: Markov Models)

★ **Imaginons un monde binaire**
- **P=Pluie**
- **S=Soleil**

★ **S P S S P P P S P P P ?**
- **Quel est le temps le plus probable demain?**
- **Solution: mesurer les probabilités de transition sur un ensemble d'entraînement, puis les appliquer à la séquence observée**
- **Ordre 1: P -> ?**
- **Ordre 5: P S P P P -> ?**



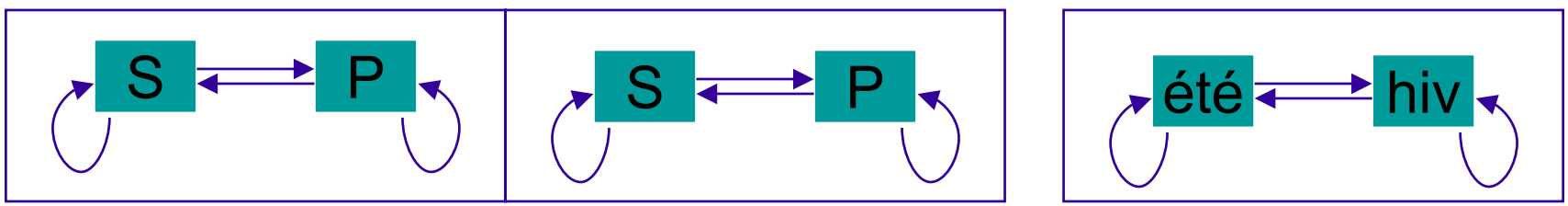
★ Une *chaine de Markov* est une collection d'ETATS correspondant chacun à une observation, où le passage d'un état à l'autre (flèches) est associé à une probabilité.

★ les probabilités de passage d'un état à l'autre sont appelées *probabilités de transition*.

★ Le système a besoin d'une phase d'*entrainement* pour déterminer les probabilités de transition.

Chaînes de Markov cachées (HMM)

- ★ **Cas où l'information que l'on cherche n'est pas un évènement de la chaîne.**
- ★ **Par exemple: S P S S P P P S P P P -> est-on en été ou en hiver?**
- ★ **Dans ce cas, il faut entraîner deux MM (été et hiver) et évaluer en plus les transitions été/hiver:**

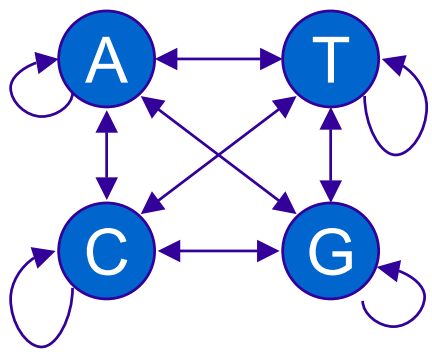


été hiver HMM

Modèles Markoviens et séquences biologiques

- Lorsqu'on sait que la succession des nt est importantes (par ex. dinucléotides (CpG), trinucléotides (codons), etc.), on veut un *modèle* dans lequel la probabilité d'une base dépende des bases précédentes.
- La base d'entraînement est constituée d'un ensemble de séquences de la même famille à reconnaître (par exemple: exons).
- Pour calculer la probabilité qu'une séquence appartienne à cette famille, il suffit d'observer les transitions apparaissant dans cette séquence, puis de se reporter au MM pour obtenir les probas. La probabilité finale est le produit des probabilités de transition.

Ici, chaine d'ordre 1. Dans une *Chaine d'ordre k* : l'état suivant dépend des k états précédents. Par exemple, ordre 5: probabilité d'observer un A après avoir vu AAUAA.



Illustr. adaptée de Durbin et al. Biological Sequence Analysis, Cambridge University Press.

HMM et séquences biologiques

- S’il y a plusieurs type d’objets à identifier (cf été/hiver, ou intron/exon/promoteur, etc.) il faut employer *plusieurs MM.*
- Lorsque qu’on a plusieurs modèles, il faut également détecter que l'on passe d'un modèle à l'autre, donc ajouter à chaque état d'un modèle une probabilité de passer à un état de l'autre modèle. Il n'y a plus de correspondance directe entre les bases et les états. Par ex. la base G peut se trouver dans un modèle ou dans l'autre. On dit alors que le modèle est *caché*
- Exemple: modélisation d’un gène complet (programme Genscan): MM pour intron/exon/intergénique, et transitions d’un MM à l’autre.

PSI-Blast (recherche itérative de profil)

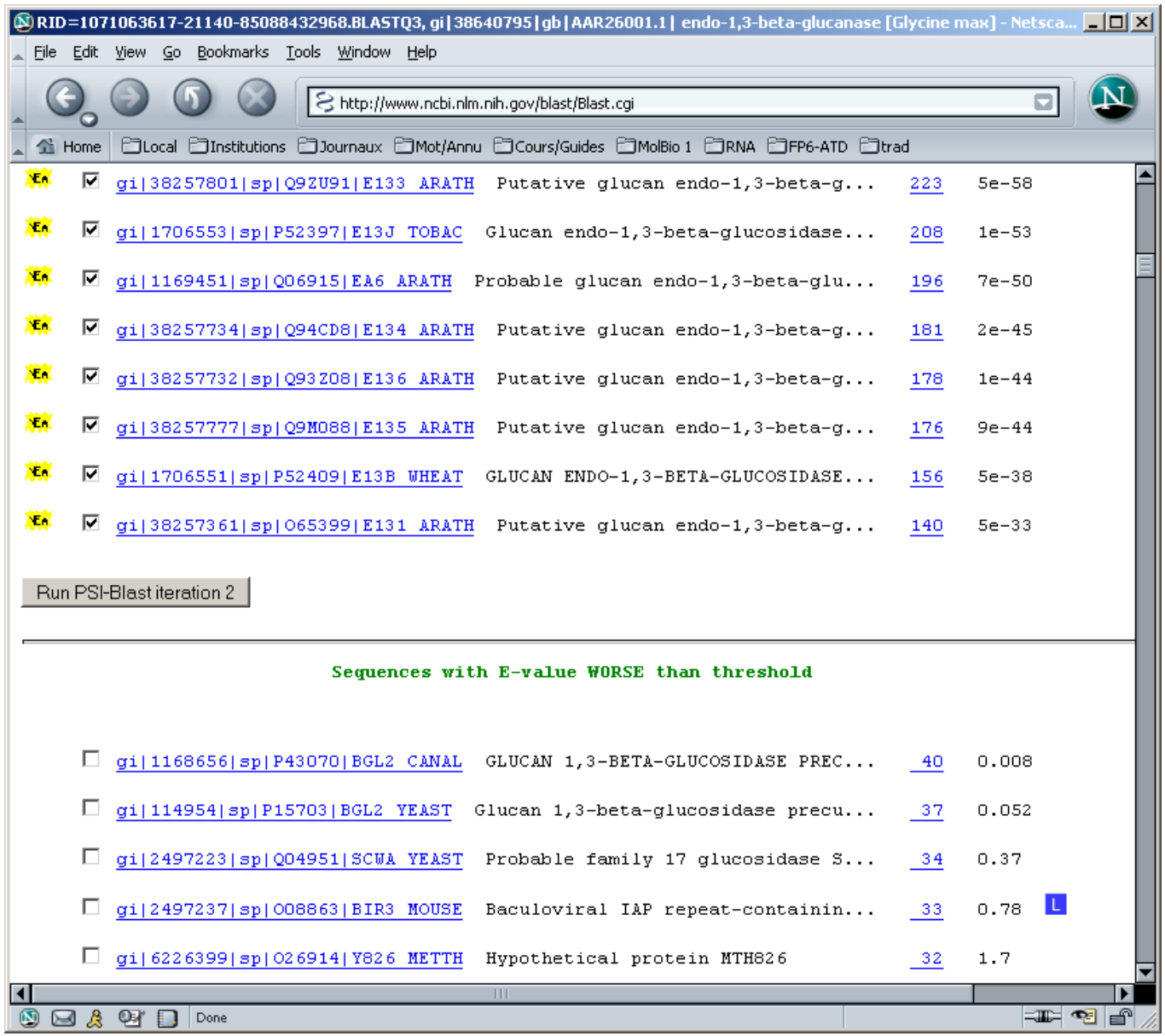
Principe

- ★ Une première séquence est recherchée dans une base de données
- ★ Les séquences similaires significatives sont alignées sur la séquence requête.
- ★ Un profil est construit
- ★ Ce profil est recherché dans la banque de donnée pour collecter des séquences supplémentaires, etc.

Avantages et inconvénients

- ★ Excellent pour la recherche d'homologues éloignés.
- ★ Si une séquence sans parenté avec la première est collectée accidentellement, celle-ci entraine tous ses homologues avec elle au tour suivant. Le profil perd son sens.
- ★ Les protéines multidomaines posent le même problème
- ★ N’existe que pour les protéines

Sortie PSI-Blast

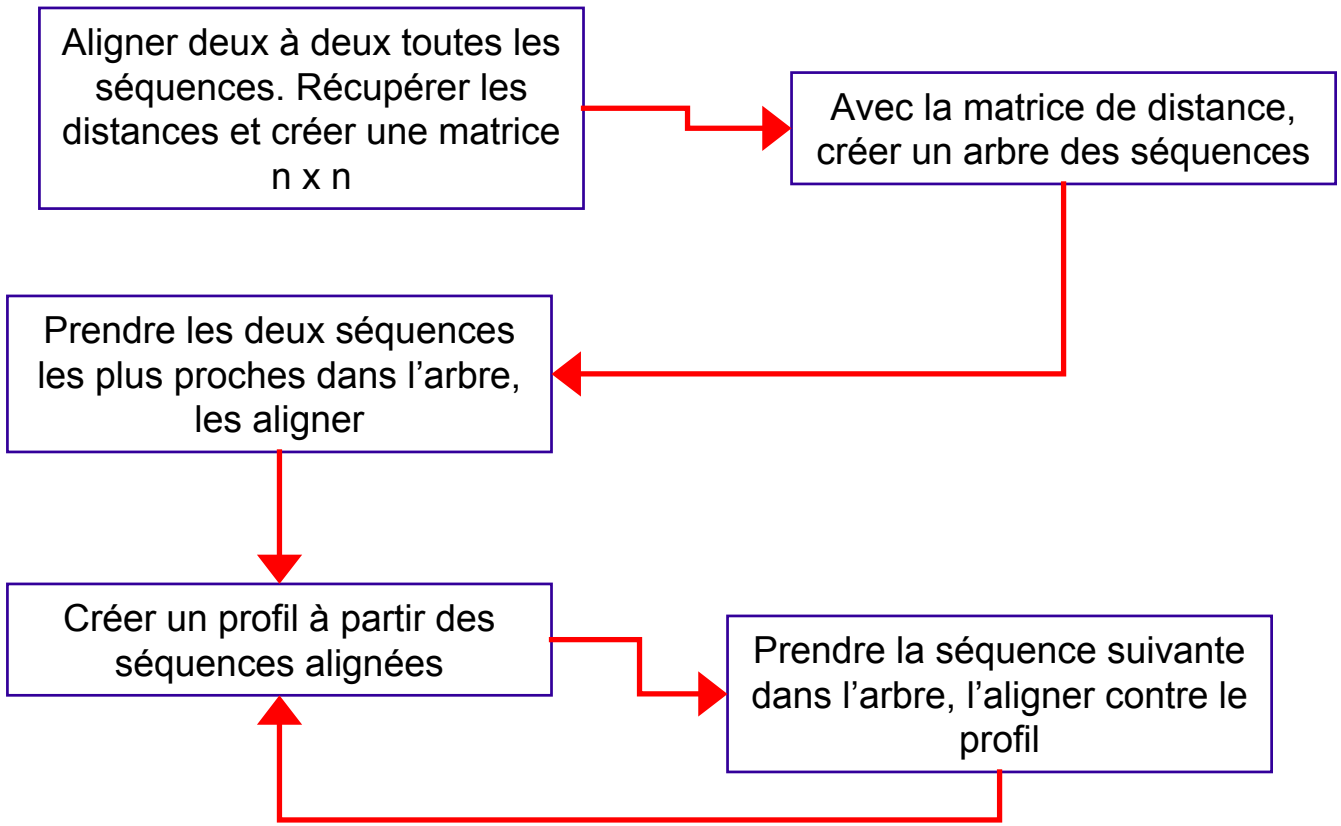


Clustal

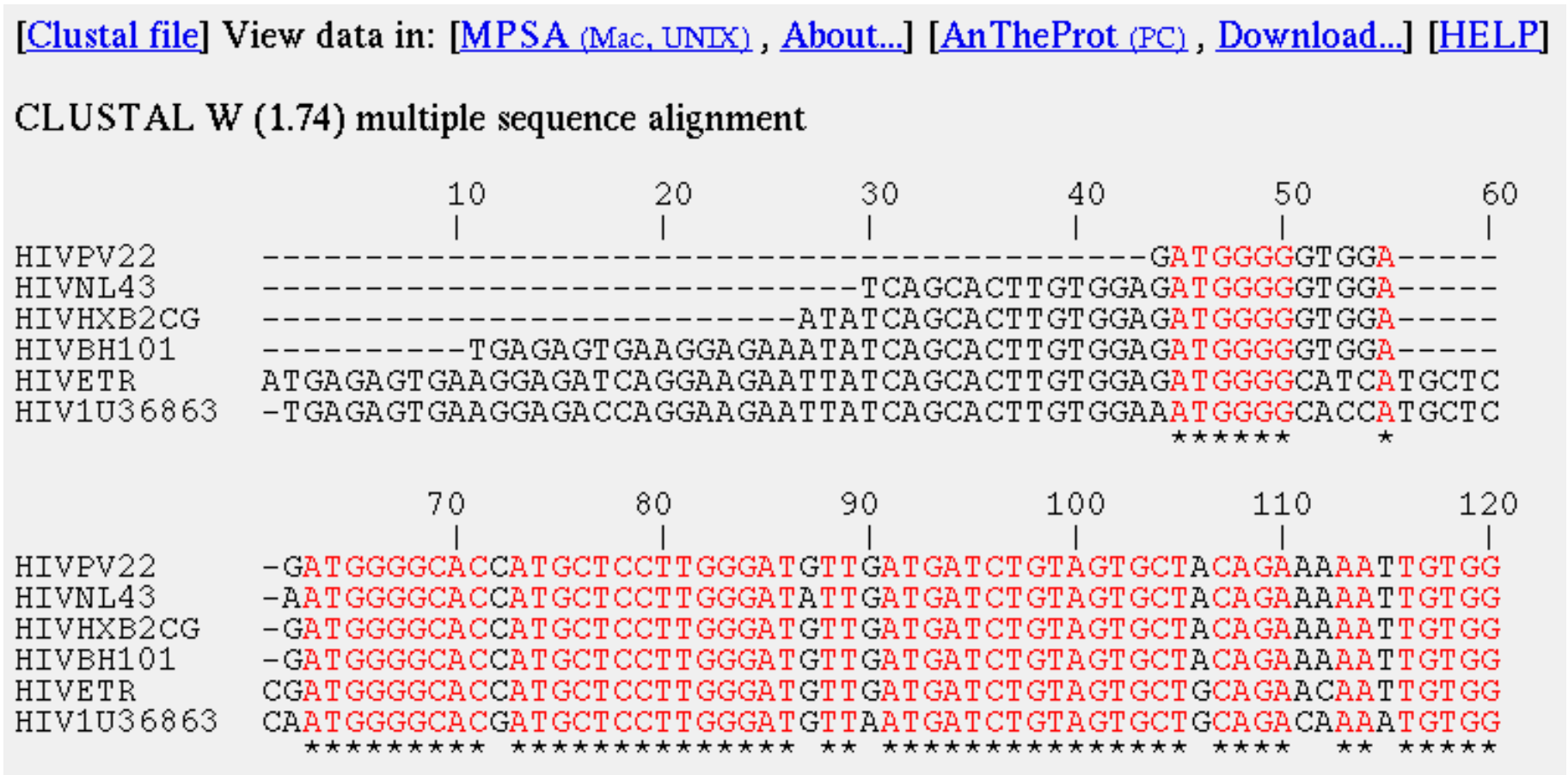
- Clustal (Des Higgins) est le programme d'alignement multiple le plus employé. Version actuelle: CLUSTALW
- Clustalw utilise les profils. Les séquences déjà alignées servent de profil pour diriger la suite de l'alignement.
- Chaque nouvelle séquence est alignée contre le profil des séquences déjà alignées.
- Pondération (Sequence weighting). Lorsque l'alignement contient plusieurs séquences très proches, celles-ci vont prendre plus d'importance dans le profil qu'une séquence isolée éloignée de celles-ci. Pour éviter que de tels groupes de séquences proches biaisent l'alignement, l'importance de chaque séquence dans le profil est pondérée en fonction du nombre d'homologues proches, tels que déterminés par un arbre.
- Clustal génère donc aussi un arbre, qu'on peut utiliser pour l'analyse phylogénétique

Algorithme de Clustal

Pour aligner n séquences



ClustalW


[Clustal file] View data in: [MPSA (Mac, UNIX) , About...] [AnTheProt (PC) , Download...] [HELP]
CLUSTAL W (1.74) multiple sequence alignment
10 20 30 40 50 60
HIVPV22 ----------GATGGGGGTGGA-----
HIVNL43 ----------TCAGCACTTGTGGAGATGGGGGTGGA-----
HIVHXB2CG ----------ATATCAGCACTTGTGGAGATGGGGGTGGA-----
HIVBH101 ----------TGAGAGTGAAGGAGAAATATCAGCACTTGTGGAGATGGGGGTGGA-----
HIVETR ATGAGAGTGAAGGAGATCAGGAAGAATTATCAGCACTTGTGGAGATGGGGCATCATGCTC
HIV1U36863 -TGAGAGTGAAGGAGACCAGGAAGAATTATCAGCACTTGTGGAAATGGGGCACCATGCTC
70 80 90 100 110 120
HIVPV22 -GATGGGGCACCATGCTCCTTGGGATGTTGATGATCTGTAGTGCTACAGAAAAATTGTGG
HIVNL43 -AATGGGGCACCATGCTCCTTGGGATATTGATGATCTGTAGTGCTACAGAAAAATTGTGG
HIVHXB2CG -GATGGGGCACCATGCTCCTTGGGATGTTGATGATCTGTAGTGCTACAGAAAAATTGTGG
HIVBH101 -GATGGGGCACCATGCTCCTTGGGATGTTGATGATCTGTAGTGCTACAGAAAAATTGTGG
HIVETR CGATGGGGCACCATGCTCCTTGGGATGTTGATGATCTGTAGTGCTGCAGAACAATTGTGG
HIV1U36863 CAATGGGGCACGATGCTCCTTGGGATGTTAATGATCTGTAGTGCTGCAGACAAAATGTGG

La Phylogénie Moléculaire

A quoi ça sert?

- ★ A retracer l'évolution des espèces. Les premiers arbres phylogénétiques réalisés à partir des ARN 16S et des cytochromes C ont avantageusement remplacés les phylogénies fondées sur l'étude des caractères qualitatifs. Elles ont permis notamment la découverte du domaine Archae, à coté des Bactéries et des Eucaryotes.
- ★ La phylogénie moléculaire permet également de comprendre l'évolution des gènes et d'en tirer des informations fonctionnelles très importantes: notamment paralogie et orthologie.

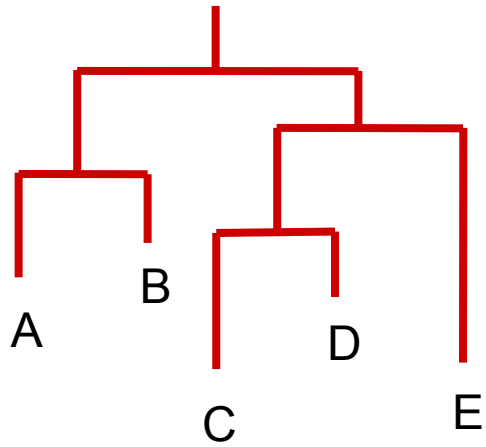
Les arbres

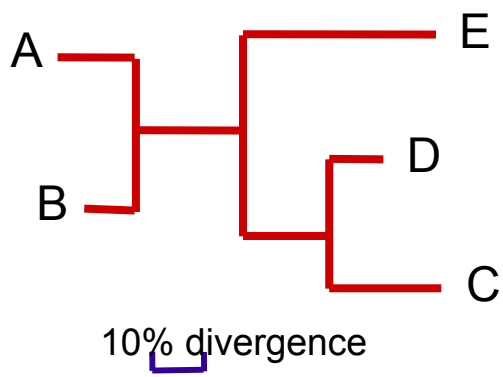
Arbres avec ou sans racine

- ★ Avec racine: montre les relations ancestrales.
- ★ Sans racine: montre les distances.

Structure d'un arbre

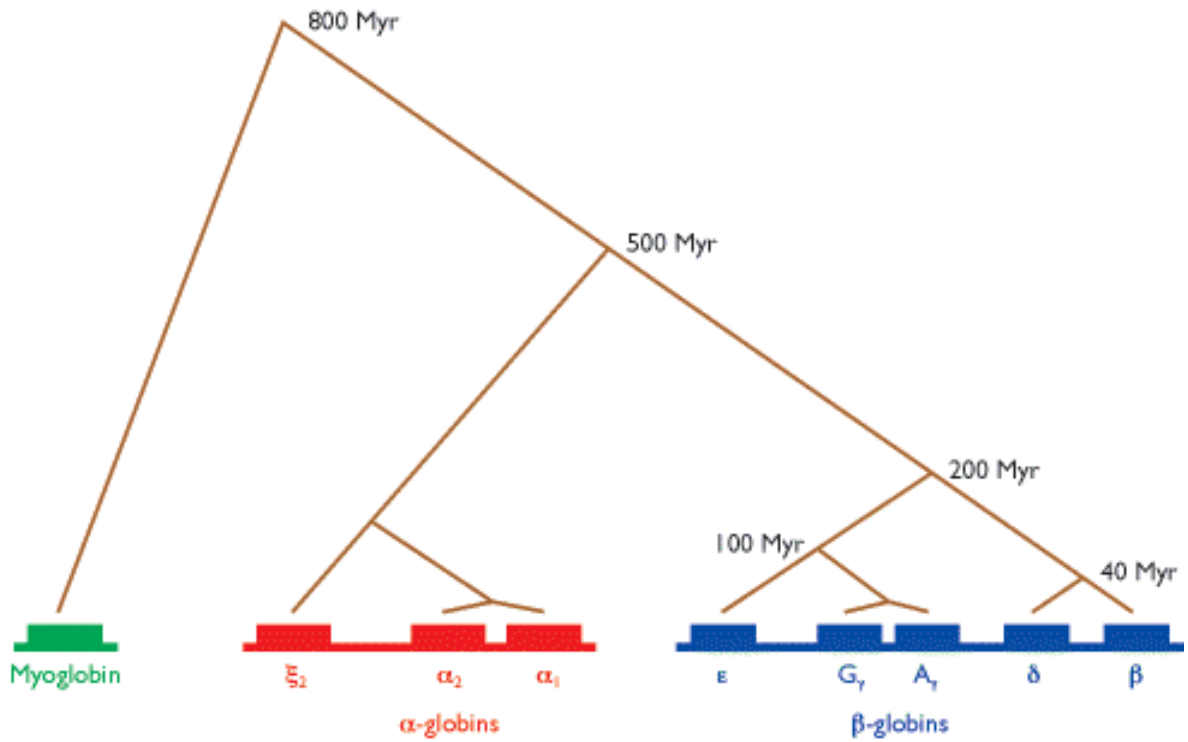
- ★ Feuilles, branches et noeuds
- ★ Un arbre représente mal les distances entre individus
- ★ Le meilleur arbre est celui qui minimise les distances et dont taille des branches respecte mieux les distances réelles





Exemple: les gènes de globine humains

★ Tous paralogues

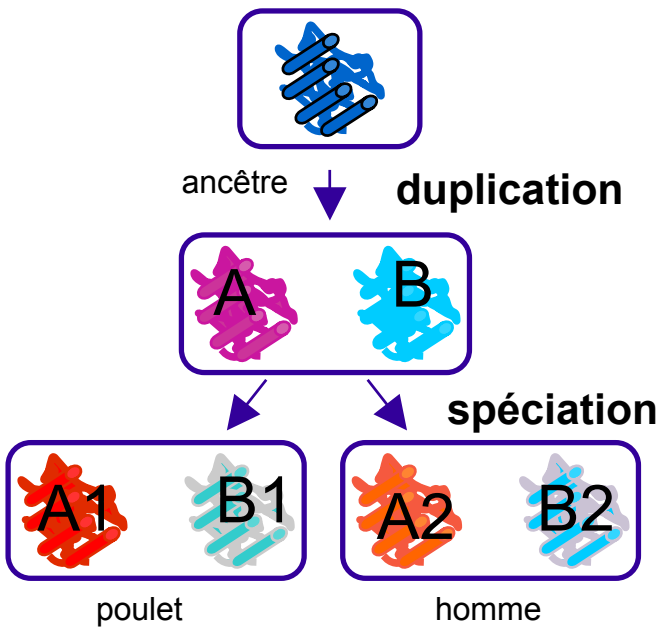


Les gènes se trouvent maintenant sur differents chromosomes: le gène de myoglobine est sur le chromosome 22, les gènes de α-globines sont sur le chromosome 16 et les gènes de β-globine sont sur le chromosome 11.

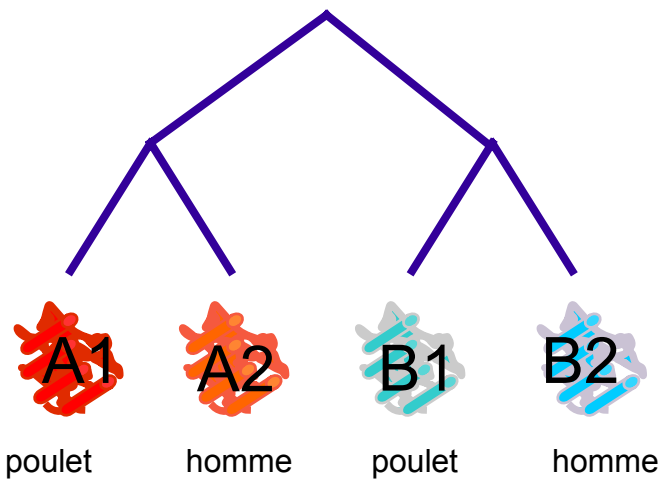
In « Genomes 2 », ©BIOS Scientific Publishers Ltd, 1999

Arbres avec paralogues et orthologues

- Admettons le schéma évolutif suivant (à gauche) ayant produit deux gènes paralogues présents chez tous les vertébrés.
- Etant donné que la duplication (ayant produit les paralogues) a eu lieu AVANT la spéciation (ayant produit les orthologues), les orthologues devraient être plus proches entre eux que les paralogues. L'arbre devrait donc ressembler au modèle de droite

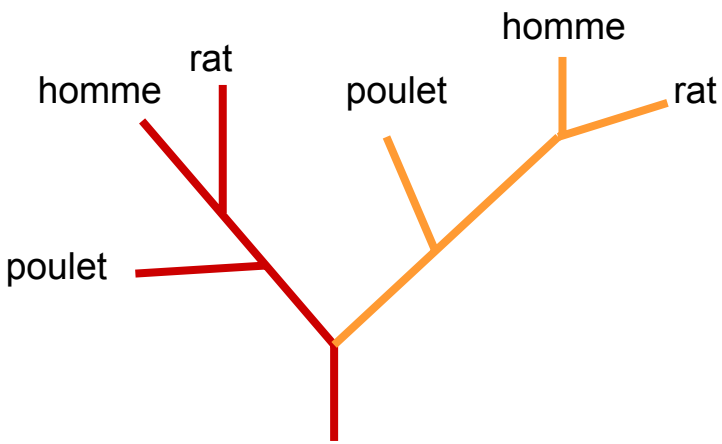


- Mais certains mécanismes d'évolution interfèrent avec ce schéma idéal:
  - Conversion: une partie d'un gène est recopiée dans l'autre copie de ce gène (advient lorsque les 2 copies sont physiquement proches). Ce mécanisme maintient une forte ressemblance entre les différentes copies d'un gène.
  - Transfert horizontal: par endosymbiontes, etc. Fitsch a aussi introduit "xénologue" pour évoquer ce cas.
  - On peut également avoir une organisation multidomaine qui provoque des erreurs sur les distances quand celles-ci sont calculées sur le gène entier.

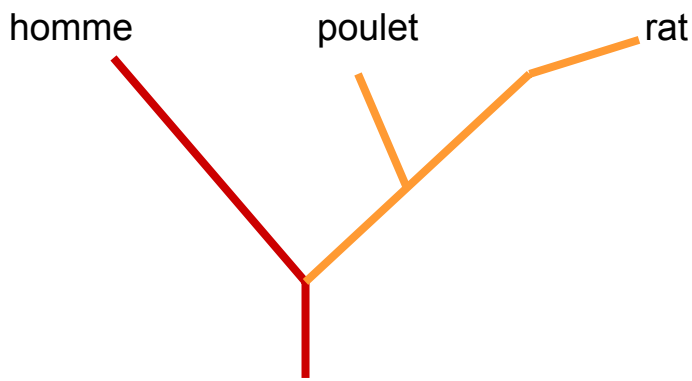


Exemples d'interprétation d'arbre

- 2 groupes d'orthologues (p.ex. globines alpha d'une part et beta d'autre part)



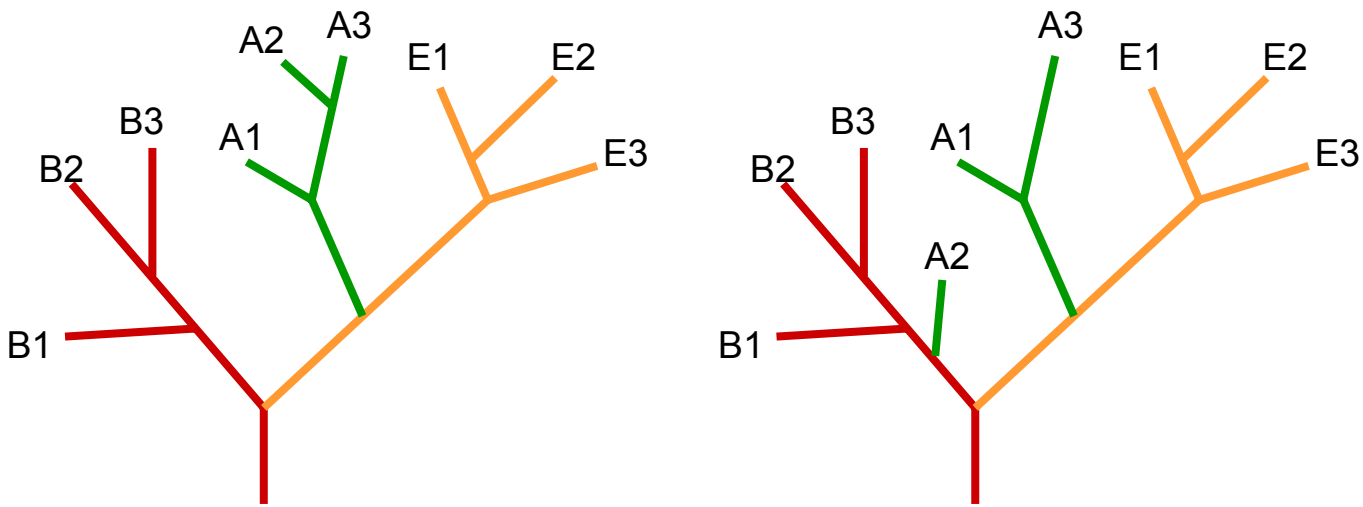
- Jeu de séquences incomplet (d'après cette structure, on soupçonne que la séquence humaine est paralogue aux autres)



Exemples d'interprétation d'arbre

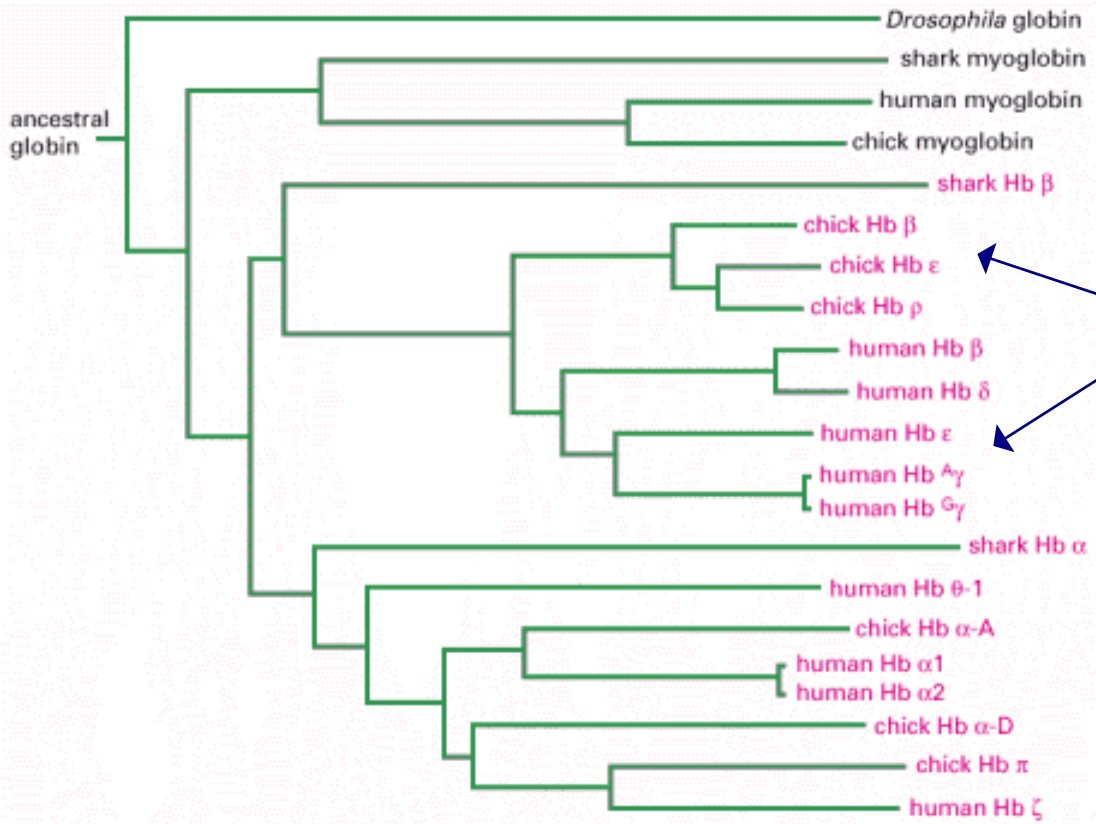
ARBRE SSU (rRNA Small Subunit)

Transfert latéral gène X d 'un ancêtre de B2 et B3 à un ancêtre de A2 et perte par A2 de l'homologue « résident » de X.



D 'après F. Doolitle, TIBS, 24, 1999, M5-M8.

Les gènes de globine chez # espèces



Myoglobines toutes orthologues.

Globines epsilon homme et poulet non orthologues.
=> Deux évènements de duplication

L’horloge moléculaire

- ★ L’hypothèse de l’horloge moléculaire pose que les substitutions dans une séquence s’effectuent à taux constant
- ★ Le nombre de substitutions observées permettrait donc de connaître la date de divergence entre deux espèces
- ★ Il faut toutefois pouvoir calibrer l’horloge (à partir de données classiques de fossiles: p.ex homme/orang-outang = 13My)
- ★ Mais les horloges moléculaires sont plus ou moins rapides selon les organismes, ou même à l’intérieur d’un organisme, sous l’influence de:
  - Temps de génération
  - Systèmes de réparation
  - Pression de sélection (p.ex: région codante ou non)

Position 1,2 (syn) ou 3 (non-syn) des codons

★ Substitutions synonymes et non synonymes

- Non-syn: mutation plus lentes que syn
- On considère que la trouisième position des codons n'est pas soumise à sélection, contrairement aux deux premières
- On peut utiliser la position 3 pour calibrer le nombre de mutations par unité de temps.

Les algorithmes de phylogénie moléculaire

Nombre d'arbres possibles

- Pour un arbre sans racine..

Espèces	Arbres
3	1
4	3
5	15
6	105
10	2 027 025
20	2.21 x 10e20

Vers 12 feuilles, il devient impossible d'évaluer tous les arbres pour trouver le meilleur

Algorithmes fondés sur les distances

- UPGMA
- Neighbor Joining (NJ)

Algorithmes fondés sur les caractères

- Parsimonie
- Maximum de vraissemblance

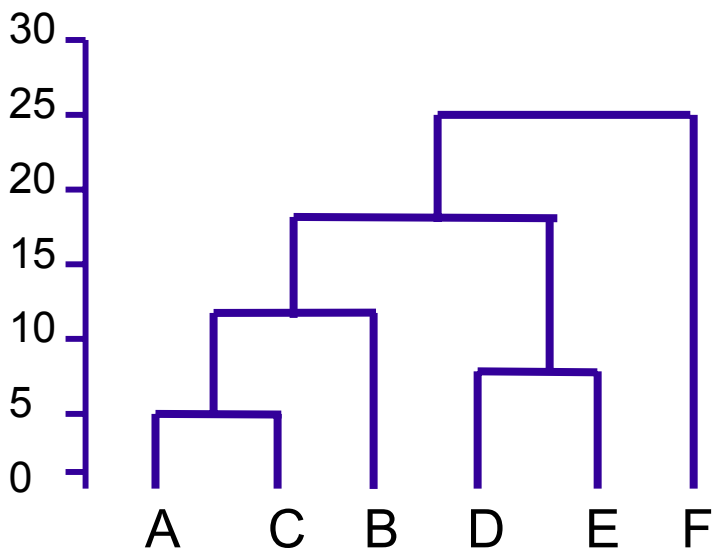
UPGMA

★ Considérons la matrice de distances :

- Regroupement des séquences les plus proches (ici AC=4)
- Recalcul de la matrice avec nouvelle entité remplaçant les deux séquences regroupées. Les nouvelles distances se calculent par moyenne (ici: moy(Ax,Cx))

	A	B	C	D	E
B	5				
C	4	7			
D	7	10	7		
E	6	9	6	5	
F	8	11	8	9	8

Arbre avec racine. Distances non additives:



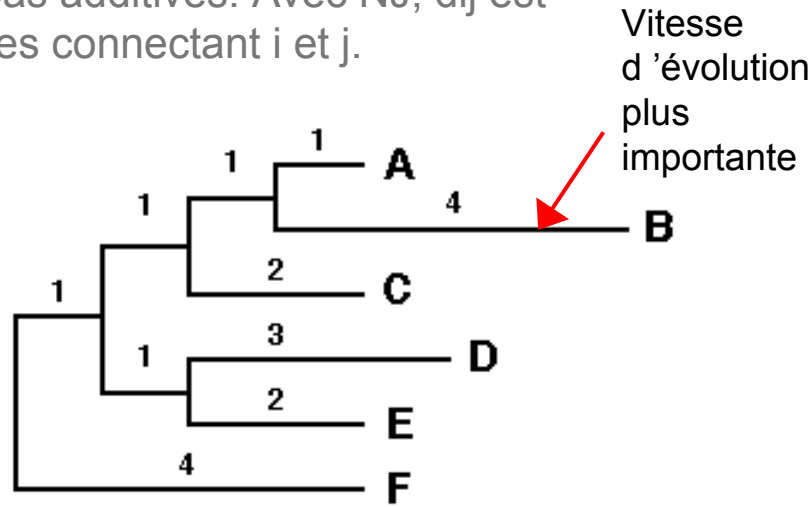
L'algorithme "Neighbor Joining" (NJ)

Saitou & Nei, 1986

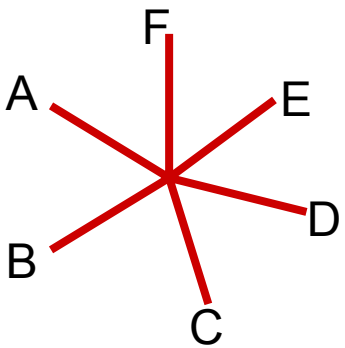
★ **Les problèmes résolus par la méthode NJ:**

- UPGMA ne fonctionne pas lorsque les vitesses d'évolution varient d'une branche à l'autre. Reprenons la matrice précédente:
- UPGMA placerait A avec C plutot qu'avec B. Or, B est plus proche de A que de toute autre séquence. Explication: comme il a évolué rapidement, B se trouve plus éloigné de A que C de A. C'est l'arbre suivant qui devrait être produit.
- NJ regroupe les espèces en fonction de leur distance avec l'ensemble des autres espèces, et non pas de leur distance entre elles. Ce faisant, NJ minimise aussi la longueur totale des branches.
- Dans l'arbre UPGMA, les distances ne sont pas additives. Avec NJ, dij est proportionnel à la longueur totale des branches connectant i et j.

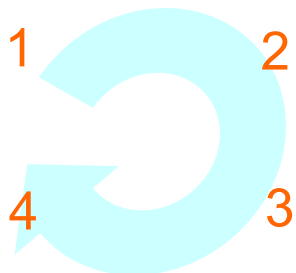
	A	B	C	D	E
B	5				
C	4	7			
D	7	10	7		
E	6	9	6	5	
F	8	11	8	9	8

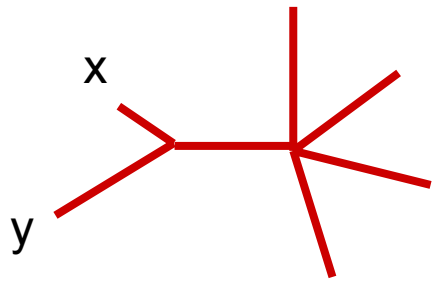


L'Algorithme NJ: principe

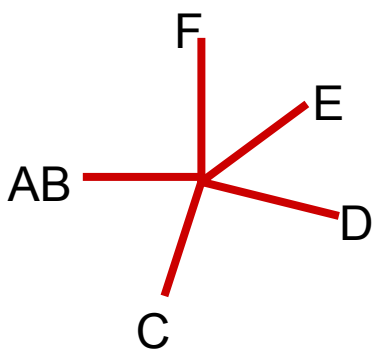


Toutes les espèces sont regroupées sur 1 noeud

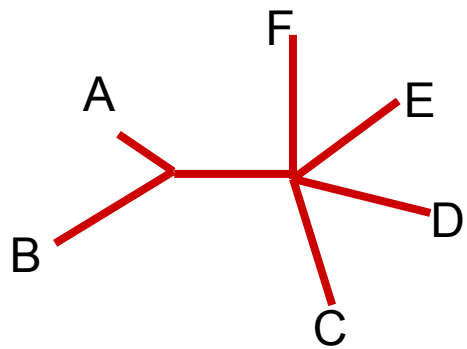




2 séquences sont prises au hasard et jointes dans un nouveau nœud. La longueur totale des branches est recalculée. On reprend l'opération avec tous les couples de séquences



Les séquences jointes sont fusionnées en une nouvelle feuille et les distances sont recalculées. On reprend à l'étape 1.

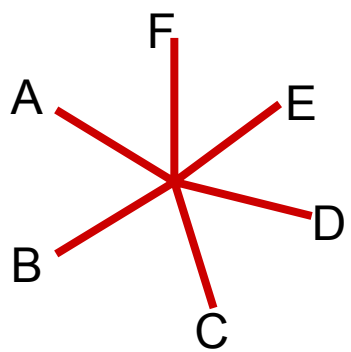


Les séquences jointes sont celles qui minimisent la longueur totale des branches

Algorithme NJ (détail:1)

- **Etape 1**: On appelle OTU (Operational Taxonomic Unit) une feuille ou un noeud de l'arbre. Au début, les OTU sont les espèces. On calcule la divergence nette r(i) de chaque OTU avec toutes les autres
  - r(A) = 5+4+7+6+8=30
  - r(B) = 42
  - r(C) = 32
  - r(D) = 38
  - r(E) = 34
  - r(F) = 44
- **Etape 2**: On calcule une nouvelle matrice de distance qui va donner pour chaque paire d'OTU la distance moyenne de cette paire avec tous les autres OTUs. Formule:
  - M(ij)=d(ij) - [r(i) + r(j)]/(N-2) soit pour la paire A,B:
  - M(AB)=d(AB) -[(r(A) + r(B)]/(N-2) = -13
- On crée maintenant un arbre en étoile:

	A	B	C	D	E
B	-13				
C	-11.5	-11.5			
D	-10	-10	-10.5		
E	-10	-10	-10.5	-13	
F	-10.5	-10.5	-11	-11.5	-11.5



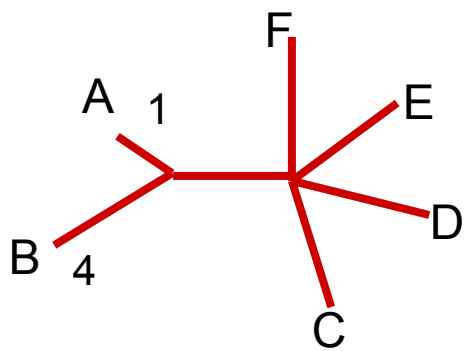
Algorithme NJ (détail: 2)

- **Etape 3**: On choisit comme voisins les 2 OTU pour lesquels Mij est le plus petit. Ce sont A et B; ou D et E. Prenons A et B et créons un nouveau noeud appelé U. On calcule ensuite les longueurs des branches entre le noeud interne U et les OTU A et B.
  - (AU) =d(AB) / 2 + [r(A)-r(B)] / 2(N-2) = 1
  - dd(BU) =d(AB) -d(AU) = 4

- Puis les distance entre U et tous les autres noeud terminaux:
  - d(CU) = d(AC) + d(BC) - d(AB) / 2 = 3
  - d(DU) = d(AD) + d(BD) - d(AB) / 2 = 6
  - d(EU) = d(AE) + d(BE) - d(AB) / 2 = 5
  - d(FU) = d(AF) + d(BF) - d(AB) / 2 = 7
- Ce qui crée la matrice ci contre

	U	C	D	E
C	3			
D	6	7		
E	5	6	5	
F	7	8	9	8

- Il en résulte l'arbre ci contre

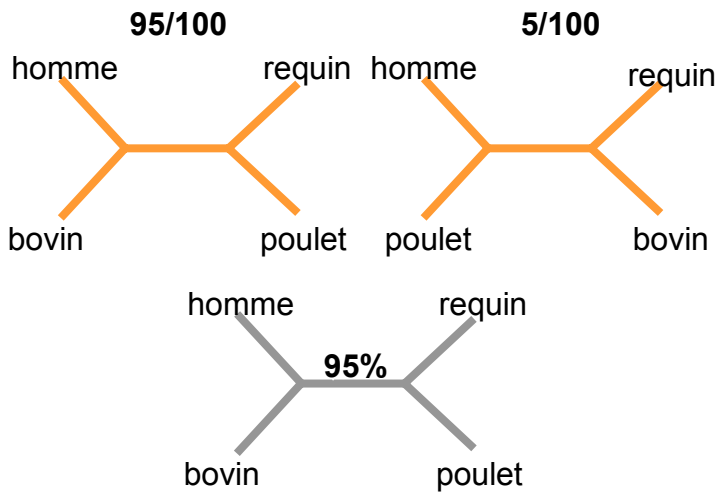


- N= N-1 = 5
  Puis on reprend à l'Etape 1

Explications tirées du site de Fred Opperdoes

Le bootstrap

- Les risques d'erreur dans la construction d'arbres sont multiples. Les distances sont évaluées sur un ensemble de sites (chaque position de l'alignement), mais ces sites peuvent se comporter différemment, avoir des origines différentes etc. Par conséquent, la matrice de distance comprend souvent des incohérences ou des situations mal résolues. Comment détecter ce genre de proiblème dans l'arbre produit?
- Le bootstrap consiste à créer un pseudo-alignement en tirant au hasard un certain nombre de sites, puis à recalculer l'arbre. On réalise ainsi un grand nombre d'abres (p. ex. 100) et on mémorise tous les arbres obtenus. Pour chaque branche de l'arbre initial, on note combien de fois cette branche est observée dans les arbres du bootstrap. Plus le nombre est élevé, plus cette branche est fiable.

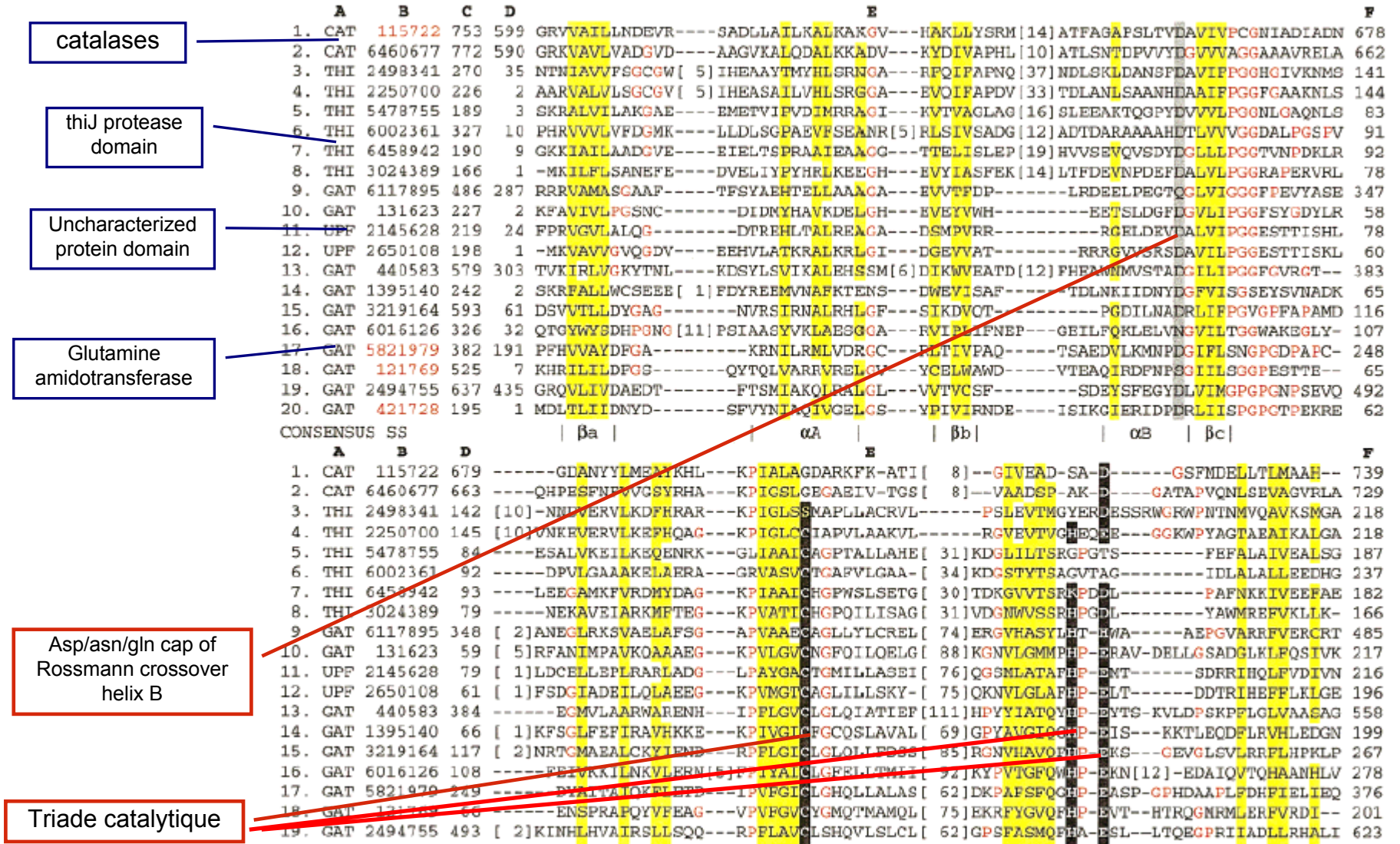


Un protocole pour l'analyse fonctionnelle complète d'une protéine

- Déterminer avec fiabilité la fonction d'une protéine d'après sa séquence demande l'application d'un protocole bien précis. On peut lire à ce sujet le papier de Bork et Koonin (Nature genetics, 1998, 18, 313). Les principaux points sont les suivants:
- **1. Eléments structuraux**
- L'identification des éléments suivants permet de s'assurer que les recherches d'homologies se déroulent sans artefact.
  - Régions de basse complexité (pour masquage)
  - Régions transmembranaires
  - Répétitions internes (ex: D1-D2-D1)
- **2. Homologies**
- C'est par homologie que l'on identifie les domaines/motifs fonctionnels de la protéine...
  - Identification de domaines connus (Prosite-Prodom)
  - Recherches itératives pour l'extension de la famille d'homologues
    - Manuellement (avec BLAST) ou automatiquement (avec PSI Blast)
    - PAM250 au lieu de BLOSUM62
    - Si protéine modulaire: masquer les domaines identifiés et lancer la recherche sur la partie restante.
    - Précautions essentielles: ne pas considérer que les meilleurs scores Blast, ne pas croire systématiquement les annotations, Tenir compte du contexte (par ex: pas de séquence signal dans un doigt de zinc)
- **3. Synthèse**
  - Alignement multiple pour visualisation finale des domaines conservés/absents.
  - Vérifier la présence de paralogues (par ex: substrats différents). La meilleure approche est de réaliser l'arbre phylogénétique.

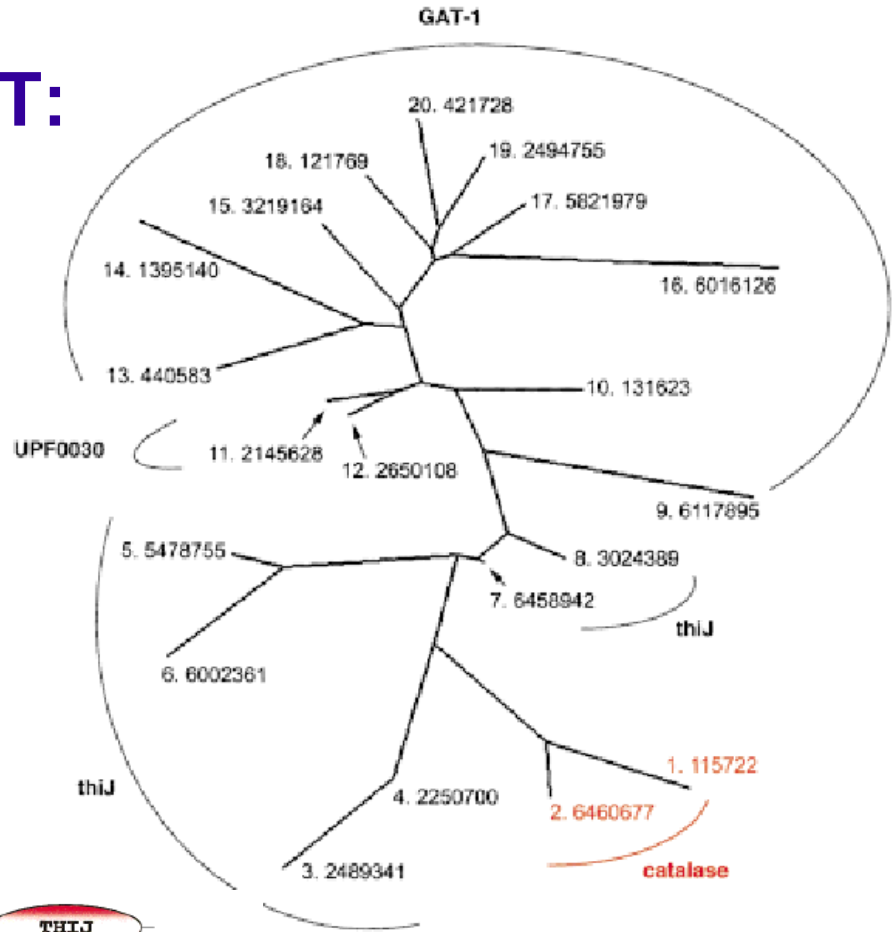
Exemple d'analyse bioinformatique: les Glutamine Aminotransferases (GAT)

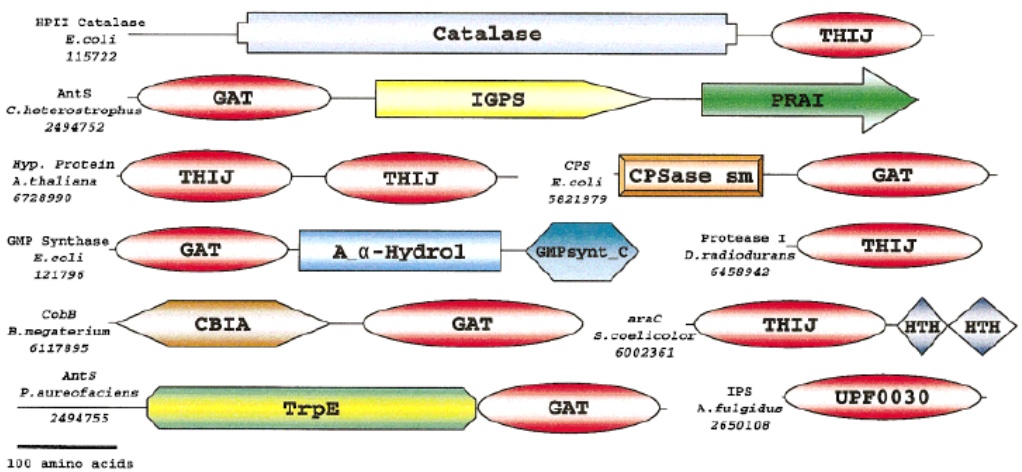
Horvath & Grishin, Proteins, 2001



1, Escherichia coli HPII catalase; 2, D. radiodurans HPII catalase; 3, Danio rerioES1; 4, Homo sapiensKNP-I a; 5, Rattus norvegicusSP22; 6, S. coelicolorAraC; 7, D. radioduransprotease I; 8, P. furiosusPfpI protease I; 9, S. coelicolor CobB; 10, Bacilis subtilisFGAM synthase I; 11, Mycobacterium leprae amidotransferase hisH; 12, A. fulgidus imidazole glycerol-phosphate synthase subunit H; 13, Saccharomyces cerevisiaeCTP synthetase; 14, Acinetobacter sp. aniline dioxygenase; 15, Arabidopsis thaliana glutamine amidotransferase; 16, A. thaliana g-glutamyl hydrolase precursor; 17, E. coli carbamoyl phosphate synthetase; 18, E. coliGMP synthase, P. aureofaciens AntS, S. sulfataricus AntS.

Arbre des GAT:





Organisation des domaines (de l'importance de séparer les domaines pour l'analyse)