Comparaisons locales et matrices de score

Équipe Bonsai

http://www.lifl.fr/bonsai

année 2011

Recherche dans une base de données

- on se donne :
  - une séquence requête q
  - une banque de séquences $T = \{t_1, ..., t_n\}$
- on veut :

  trouver des alignements significatifs entre q et les t_i
- les algorithmes classiques ne fonctionnent pas : prennent trop de temps, il faut trouver des parades

FASTA

Pearson et Lipman, 1988

- alignement global avec gaps
- traite les séquences de la banque les unes après les autres
- fonctionnement :
  1. trouve tous les mots identiques de longueur $\geq l$ communs à q et t_i
  2. sélectionne ceux de score suffisament élevé (score PAM par exemple)
  3. sélectionne une diagonale d (du dotplot) contenant le maximum de mots identiques de longueur $\geq l$
  4. procède à un alignement global "classique" dans une bande de largeur $2k$ autour de la diagonale d
- deux paramètres : k et l, l généralement de longueur 6 pour l'ADN et 2 pour les protéines

Blast

- naît en 1990 : trouve des matchs significatifs sans gaps
- évolution vers une version 2, avec gaps
  - NCBI-Blast
  - WU-Blast : très similaire à NCBI-Blast (mix entre Blast1 et FASTA pour la dernière étape)
- évolution vers des versions avec raffinement des résultats

Blast 1

1. Trouve les mots **similaires** de taille fixe entre q et t_i.

   (taille par defaut : ADN $\rightarrow$ 11 , Protéines $\rightarrow$ 3)
2. Ne considère que les couples de mots ...

(Protéines) **similaires** $\rightarrow$ score des mots alignés $\geq$ seuil T

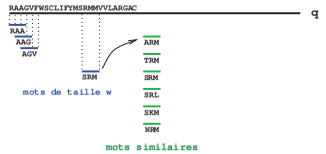
(defaut : $T = 11$ sur BLOSUM-62)

(ADN) **identiques** $\rightarrow$ pas de seuil T donc moins sensible.

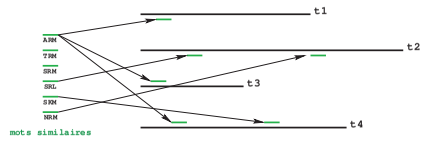
3. Chaque couple de mots entre q et un t_i forme un *hit*
4. Chaque hit est étendu à gauche et à droite : l'extension est stoppée lorsque le score du hit décroît de plus de X (X-*drop*)

Blast 1 - Schématiquement

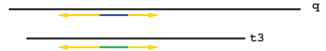
1 -



2 -



3 -

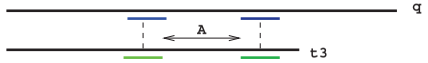


Blast 1

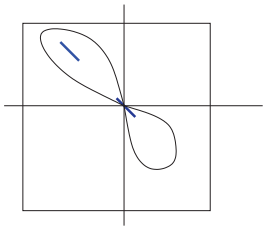
- un hit est un mot "commun" de taille fixée w (et de score supérieur à un seuil T dans le cas de BLAST-P) sur les deux séquences q et t_i.
- chaque hit **étendu** forme un HSP : *High scoring Segment Pair*
- ne conserve que les HSP de score supérieur à un score seuil donné : les LMSP, *Locally Maximal Segment Pair*
- pour deux séquences données : le LMSP de meilleur score est le MSP : *Maximal scoring Segment Pair*

NCBI - Blast 2 (Gapped-blast)

- idée : incorporer des gaps dans le critère de *hit*
- mise en oeuvre : se baser sur 2 hits distants au maximum de A



- étendre les hits comme dans Blast 1 (avec limitation de score) mais en autorisant les gaps



MegaBLAST

pour l'ADN

- idée : un Blast plus rapide lorsqu'on recherche une grande similarité
- mise en oeuvre : utiliser des mots de taille plus grande (28 contre 11)
- à réserver à des requêtes du style : trouver la séquence dans la banque
- **évolution** : Discontiguous MegaBLAST
  - principe : utiliser une *graine espacée* plutôt qu'un *mot exact* (graine contiguë)
  - **exemple** : graine espacée 100101100101100101101 plutot que graine contiguë 11111111111
  - peut se réveler meilleur que BLAST (en particulier avec *graines espacées multiples*).

Définition des graines contiguës vs espacées

graine contiguë : 111111

```
      111111
ATCAGTGCAAATGCGCAAGA
|||||:|||||||.|||||
ATCAGCGCAAATGCTCAAGA
```

graine espacée: 11101011

```
       11101011
ATCAGTGCGAATGCGCAAGA
|||||:|||:|||||.|||||
ATCAGCGCAAATGCTCAAGA
```

- Les graines espacées peuvent être *bien choisies* pour mieux détecter les alignements (*Keich Li Ma Tromp DAM 2004*)
- Il est possible d'utiliser plusieurs graines espacées de formes différentes pour améliorer la sensibilité de la recherche

π_c = 111111

π_s = 111001011

α =

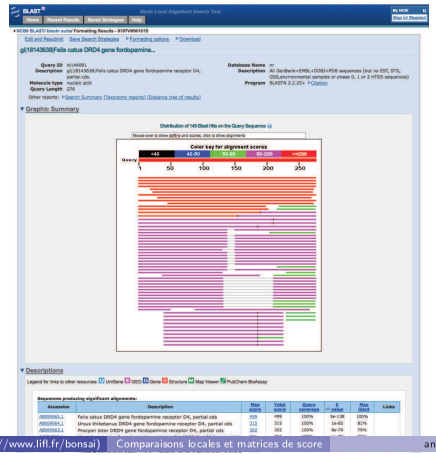
```
ATCAGTGCAAATGCTCAAGA
||||||||||||||||||||
ATCAGTGCAAATGCTCAAGA
```

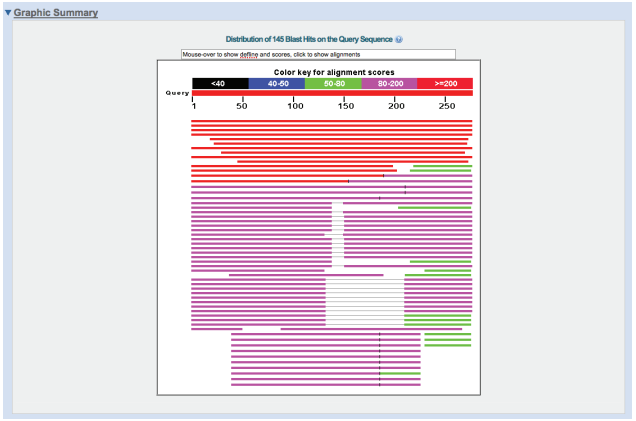


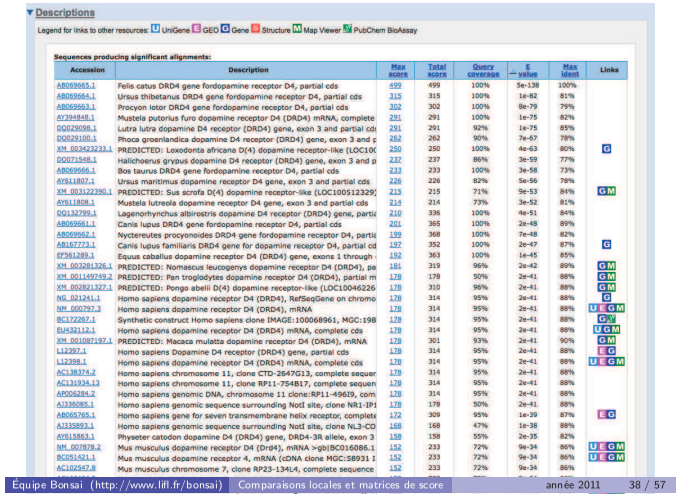
```
ATCAGTGCAAATGCTCAAGA
||||||||||||||||||||
ATCAGTGCAAATGCTCAAGA
111111
 111111
  111111
   111111
    111111
     111111
      111111
       111111
        111111
         111111
          111111
           111111
            111111
             111111
              111111
               111111
```

```
ATCAGTGCAAATGCTCAAGA
||||||||||||||||||||
ATCAGTGCAAATGCTCAAGA
111001011
 111001011
  111001011
   111001011
    111001011
     111001011
      111001011
       111001011
        111001011
         111001011
          111001011
           111001011
```

BLAST : Exemple de résultats







BLAST : Exemple de résultats

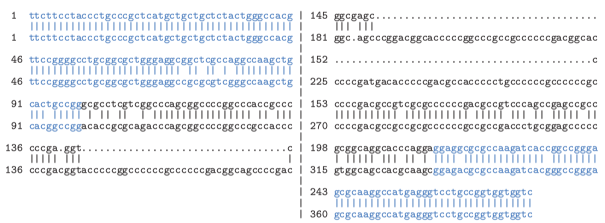
```
Query=  Felis catus DRD4 gene fordopamine receptor D4
        (276 letters)

Database: All GenBank+EMBL+DDBJ+PDB sequences
          1,174,453 sequences; 5,001,591,585 total letters


                                                    Score    E
Sequences producing significant alignments:         (bits) Value

gi|AB069665  Felis catus DRD4 gene f...              210    5e-52
gi|AB069662  Nyctereutes procyonoide...              157    7e-36
gi|AB069661  Canis lupus DRD4 gene f...              157    7e-36
gi|AB069666  Bos taurus DRD4 gene fo...              143    1e-31
gi|291947|   Homo sapiens Dopamine D4 recep...       135    2e-29
```


BLAST : Exemple de résultats

```
>gi|18143632|dbj|AB069662.1|AB069662 Nyctereutes procyonoides
 DRD4 gene  fordopamine receptor D4. Length = 393

 Score =  157 bits (79), Expect = 7e-36
 Identities = 94/99 (94%)
 Strand = Plus / Plus

Query 1 ttcttcctaccctgcccgctcatgctgctgctctactgggccacgttcc 48
        ||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct 1 ttcttcctaccctgcccgctcatgctgctgctctactgggccacgttcc 48

Query 49 gggcctgcggcgctgggaggccggctcgccaggccaagctgcactgccgg 99
         ||||||||||||||||||| || || | ||||||||||| |||||
Sbjct 49 gggcctgcggcgctgggaggccgcgcgctcgggccaagctgcacggccgg 99


 Score =  107 bits (54), Expect = 5e-21
 Identities = 60/62 (96%)
 Strand = Plus / Plus

Query 215 ggaggcgcgccaagatcaccggccgggagcgcaaggccatgagggtcct 252
          |||| ||||||||||||||| ||||||||||||||||||||||||||
Sbjct 332 ggagacgcgccaagatcacgggccgggagcgcaaggccatgagggtcct 379

Query 253 tgccggtggtggtc 276
          |||||||||||||
Sbjct 380 tgccggtggtggtc 393
```

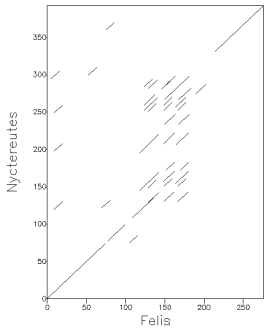
BLAST : ... vs Alignement Réel

Felis Catus/ Nyctereute



BLAST : ... vs Alignement Réel

Felis Catus/ Nyctereute



BLAST : Significativité des MSPs

- deux séquences peuvent toujours être alignées
- il existe toujours un (au moins) alignement de meilleur score S entre deux séquences (un MSP)

question : *ce score est-il suffisamment élevé pour prouver une homologie ?*

problème : *peut-on trouver un MSP de meilleur score dans deux séquences aléatoires ?*

BLAST : Mesures de significativité

- la **p-valeur (p-value)**

  mesure la *Probabilité* que 2 séquences aléatoires de même longueur et de même composition possèdent un MSP de score $\geq S$

- la **e-valeur (e-value)**

  mesure l'*Esperance* E du nombre n de MSPs de score $\geq S$ dans 2 séquences aléatoires de même longueur et de même composition

$$E = \sum_n p(n) \times n$$

Calcul de la e-value

- soient deux séquences a et b aléatoires suivant une distribution de probabilité connue
- on suppose que les MSPs sont données par les diagonales du dotplot
- plutôt que de décrire un alignement par des paires de lettres tirées aléatoirement, on peut le décrire par une suite de scores tirés aléatoirement
- on veut calculer l'esperance du nombre de MSPs de score $\geq$ à S

Calcul de la e-value

- Selon *Karlin et Altschul 1991* :

$$\textbf{e-value} = Kmne^{-\lambda S} \qquad \textbf{p-value} = 1 - e^{-\text{e-value}}$$

avec m la taille de la séquence requête, n la taille de la banque de données, S le score du hit (K et λ dépendent de la matrice de score, K peut être ajusté en fonction du coût des gaps)

Calcul du bit-score

- si S est le score d'un hit
- le bit-score (score normalisé) est :

$$S' = \frac{\lambda S - \ln K}{\ln 2}$$

- l'expression de la e-value devient :

$$E(S) = mn2^{-S'}$$

Variation de la e-value

- si la taille de la séquence query augmente : la e-value
- si la taille de la banque est divisée par deux : la e-value
- si le score augmente : la e-value
- quel bit-score pour obtenir une e-value de 0.05 pour une séquence de longueur 250 et une bd de longueur 50000000 ?
- si on passe la e-value à 0.01, quel sera le bit-score ?

Variations de la e-value

```
Mus musculus chromosome 5, clone RP23-301L9, complete sequence
          Length = 212246

 Score = 32.2 bits (16), Expect = 2.1
 Identities = 19/20 (95%)
 Strand = Plus / Plus

Query: 2      ttcattatgaagcacgagga 21
               ||||||||| |||||||||
Sbjct: 136843 ttcattatgatgcacgagga 136862


Mus musculus BAC clone RP23-13L19 from chromosome 9, complete sequence
          Length = 224108

 Score = 30.2 bits (15), Expect = 8.1
 Identities = 15/15 (100%)
 Strand = Plus / Plus

Query: 6      ttatgaagcacgagg 20
               |||||||||||||||
Sbjct: 93798 ttatgaagcacgagg 93812


Lambda     K      H
    1.37    0.711     1.31
Number of Hits to DB: 99,084,306
Number of Sequences: 2130505
Number of extensions: 2852
Number of successful extensions: 19
Number of sequences better than 10.0: 0
Number of HSP's better than 10.0 without gapping: 0
Number of HSP's successfully gapped in prelim test: 16
length of query: 21
length of database: 10,249,863,584
```


Variations de la e-value

```
Mus musculus chromosome 5, clone RP23-301L9, complete sequence
          Length = 212246

 Score = 32.2 bits (16), Expect = 2.1
 Identities = 19/20 (95%)
 Strand = Plus / Plus

Query: 2      ttcattatgaagcacgagga 21
               ||||||||| |||||||||
Sbjct: 136843 ttcattatgatgcacgagga 136862

------------------------------------------------------------

Mus musculus, clone RP23-277C24, complete sequence
          Length = 199946

 Score = 32.2 bits (16), Expect = 2.1
 Identities = 16/16 (100%)
 Strand = Plus / Minus


Query: 1      attcattatgaagcac 16
               ||||||||||||||||
Sbjct: 69080 attcattatgaagcac 69065
```


Variations de la e-value

```
Query length : 21
Mus musculus, clone RP23-277C24, complete sequence
          Length = 199946

 Score = 32.2 bits (16), Expect = 7.6
 Identities = 16/16 (100%)
 Strand = Plus / Minus


Query: 1      attcattatgaagcac 16
               ||||||||||||||||
Sbjct: 69080 attcattatgaagcac 69065

------------------------------------------------------------

Query length : 20
Mus musculus, clone RP23-277C24, complete sequence
          Length = 199946

 Score = 32.2 bits (16), Expect = 5.1
 Identities = 16/16 (100%)
 Strand = Plus / Minus


Query: 1      attcattatgaagcac 16
               ||||||||||||||||
Sbjct: 69080 attcattatgaagcac 69065
```


Les différents programmes BLAST

Query \ Database	nucléique	protéique	nucléique traduit
nucléique	blastn	x	x
protéique	x	blastp	tblastn
nucléique traduit	x	blastx	tblastx

Le bon programme pour la bonne requête

extrait de "BLAST Program Selection Guide"

- MEGABLAST is the tool of choice to identify a nucleotide sequence
- Discontiguous MEGABLAST is better at finding nucleotide sequences similar, but not identical, to your nucleotide query
- les pages "Search for short nearly exact matches"
  - nucleotide: useful for primer or short nucleotide searches
  - proteins: optimized to find matches to a short peptide
  - principales différences :
    - taille de mots plus petite
    - suppression des filtres
    - relâchement de la E-value
    - matrice de score PAM30 (au lieu de BLOSUM62) pour les protéines

Evolutions de Blast : PSI-Blast

PSI-BLAST is designed for more sensitive protein-protein similarity searches

Position Specific Iterated BLAST

1. recherche initiale avec BLASTp
2. construction d'un alignement multiple, puis d'un profil
   - à partir d'un alignement multiple des meilleurs hits
   - construit une matrice position-spécifique :
     - chaque colonne représente un AA
     - chaque ligne une position dans l'alignement
3. nouvelle recherche avec le profil et modification

réitère le processus un certain nombre de fois ou juqu'à convergence

Profil - exemple

```
                    # Pure Frequency Matrix
Alignement multiple:# Columns are amino acid counts A->Z
                    # Rows are alignment positions 1->n
                    Simple
T-VAAPSVFIFPPSDEQ   Name            mymatrix
A-DAAPTVSIFPPSSEQ   Length          17
A-NAAPTVSIFPPSTZZ   Maximum score   60
D-PVAPTVLIFPPAADQ   Thresh          75
DPPIAPTVLLFPPSADQ   Consensus       APPAAPTVLIFPPSADQ
                    20020000000000000010000000
                    00000000000000100000000000
                    00010000000010200000100000
                    30000000100000000000100000
                    50000000000000000000000000
                    00000000000000500000000000
                    00000000000000000140000000
                    00000000000000000000500000
                    00000100000200000200000000
                    00000000400100000000000000
                    00005000000000000000000000
                    00000000000000500000000000
                    00000000000000500000000000
                    10000000000000000400000000
                    20010000000000000110000000
                    00022000000000000000000010
                    00000000000000004000000010
```


Evolution de Blast : PHI-Blast

PHI-BLAST can do a restricted protein pattern search

Pattern Hit Initiated BLAST

- pour les séquences protéiques
- entrée : une séquence et un motif (expression régulière à la Prosite)
- restriction de la banque aux séquences pour lesquelles le motif est retrouvé
- puis application de BLAST
- couplage possible avec PSI-Blast