

# Algorithmique du texte

Alignment de séquences : dot plot et programmation dynamique

- Introduction
- Dotplot
- Alignement : définitions et scores
- Algorithme d'alignement (programmation dynamique)
- Autres types d'alignements
- La suite ...

# Pourquoi aligner des séquences ?

---

Alignment = comparaison

- Recherche d'homologie
- Prédiction de gènes
- Recherche de fonction commune / transfert d'annotation

On dispose de beaucoup de séquences ⇒ ressource à exploiter

# Exemples d'alignement : insuline

Score = 271, Expect = 2e-24

B FVNQHLCGSHLVEALYLVCGERGFYTPKAGIVEQCCASTCSLYQLENYCN

**E FVNQHLCGSHLVEALYLVCGERGFFYTPKTGIVEQCCTGVCSLYQLENYCN**

Score = 256, Expect = 1e-22

B NQHLCGSHLVEALYLVCGERGFYTPKAGIVEQCCASTCSLYQLENYCN

C NQHLCGSHLVEALYLVCGERGFYSPKTGIVEQCCEPDSL YQLENYCN

Score = 243, Expect = 4e-21

B FVNQHLCGSHLVEALYLVCGERGFYTPKAGIVEQCCASTCSLYQLENYCN

O FPNQHLCGSHLVEALYLVCGEKGFYYIPRMGIVEECCKGVCSMYQLENYC

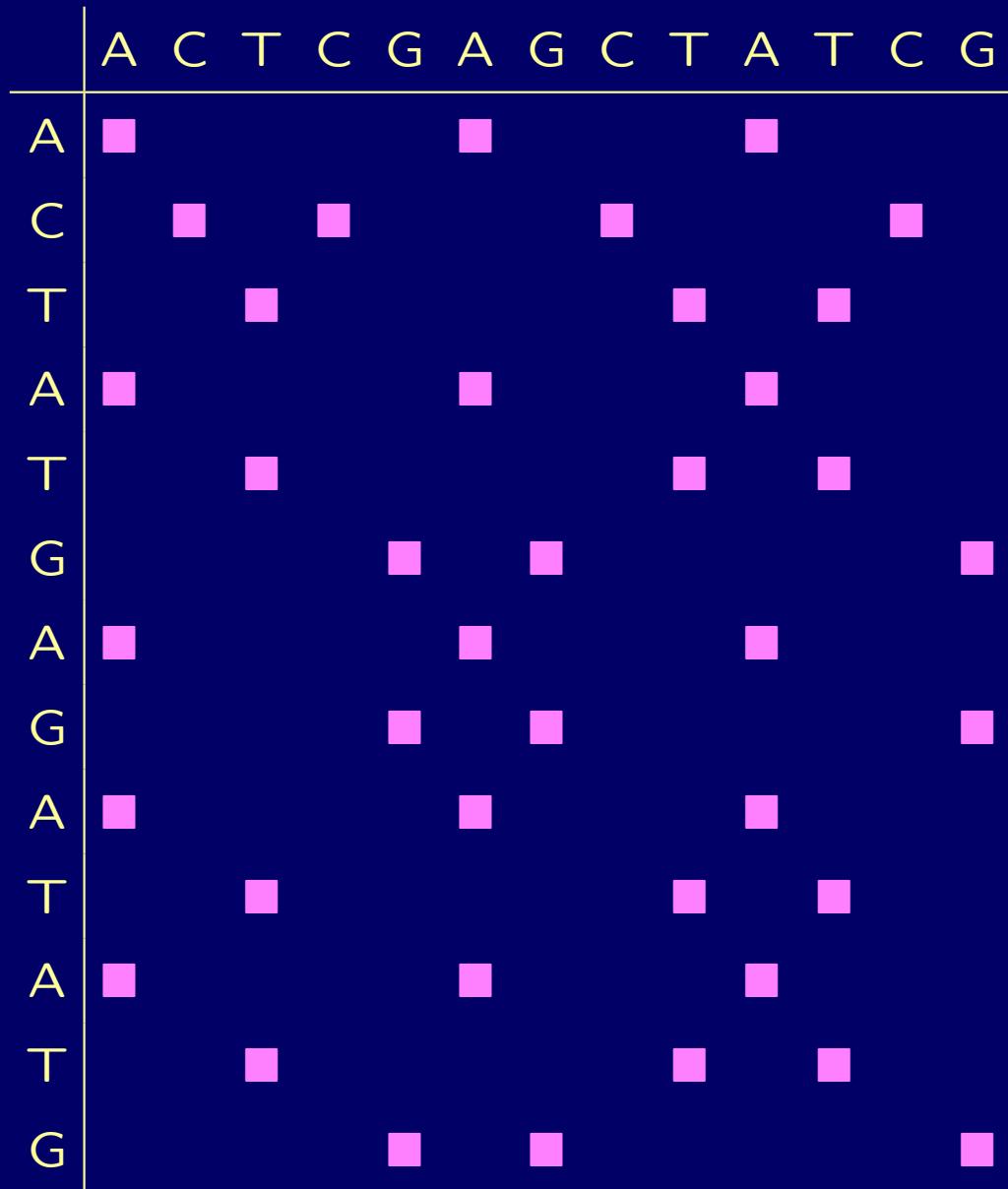
- Introduction
- Dotplot
- Alignement : définitions et scores
- Algorithme d'alignement (programmation dynamique)
- Autres types d'alignements
- La suite ...

# Une méthode graphique : le Dot Plot

match (identité) → ■

mismatch → □

diagonale = région similaire

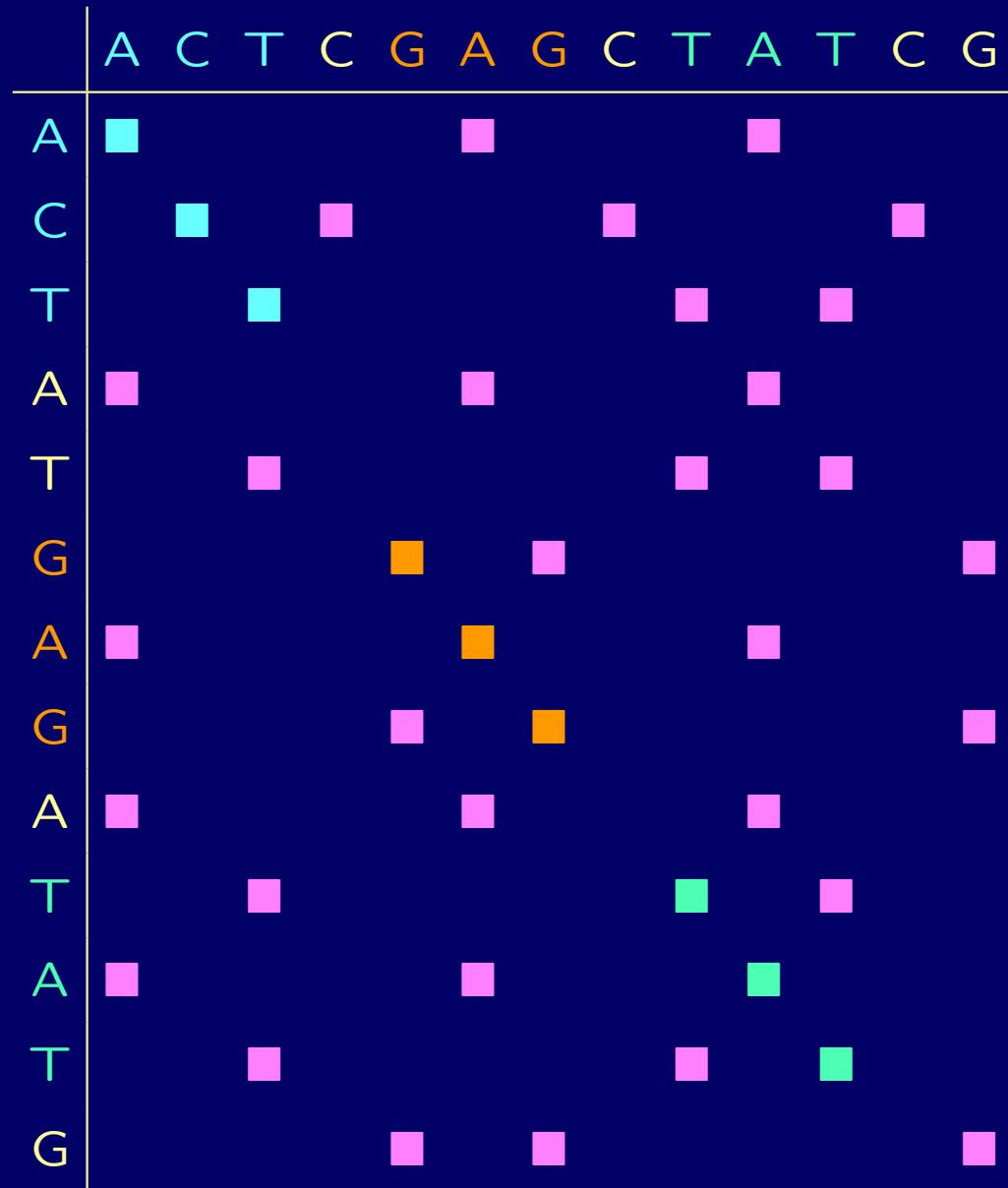


# Une méthode graphique : le Dot Plot

match (identité) → ■

mismatch → □

diagonale = région similaire

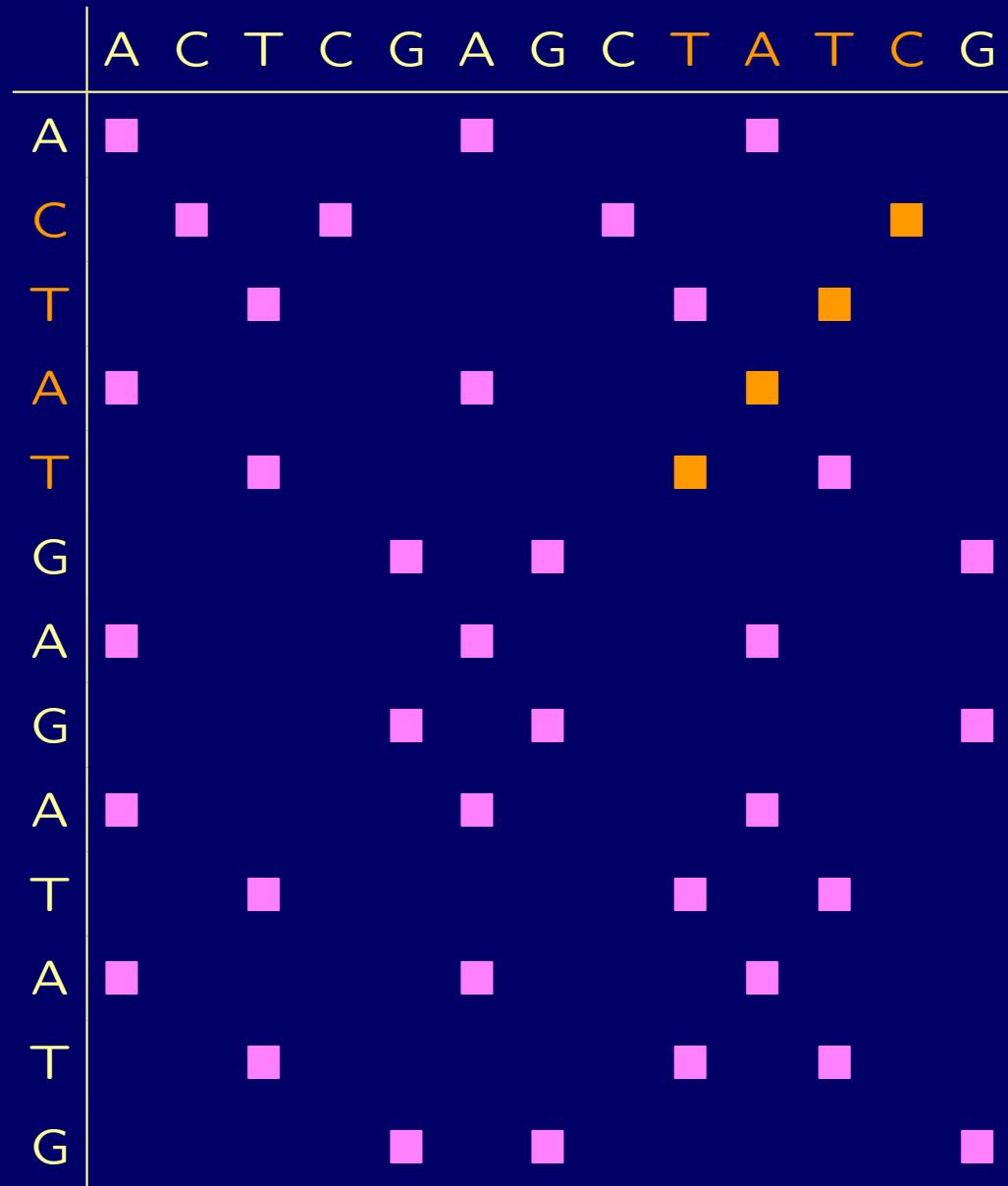


# Une méthode graphique : le Dot Plot

match (identité) → ■

mismatch → □

diagonale inversée = région miroir

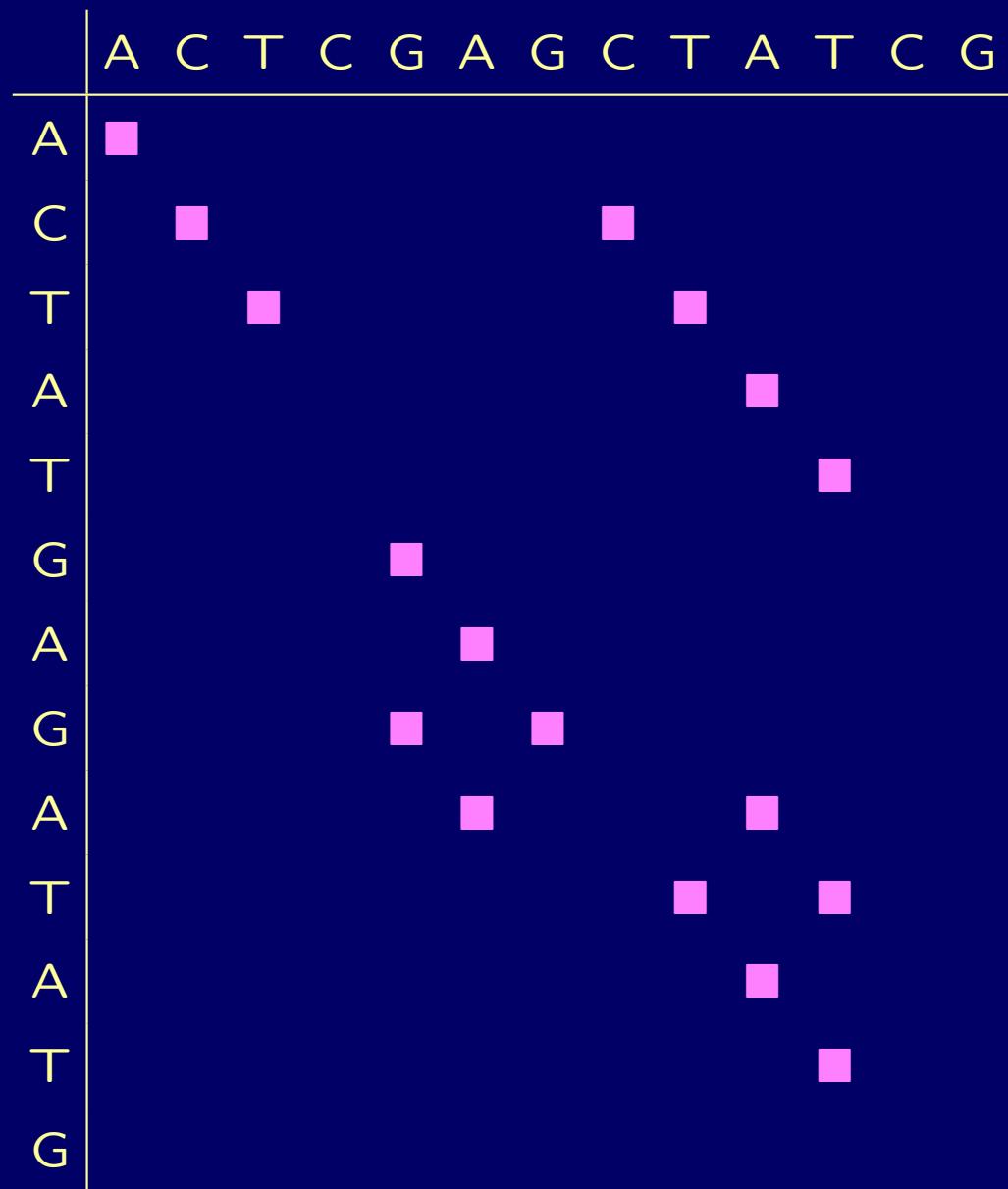


# Dot Plot : vue par fenêtre

match (identité) → ■

mismatch → □

Fenêtre de taille 2

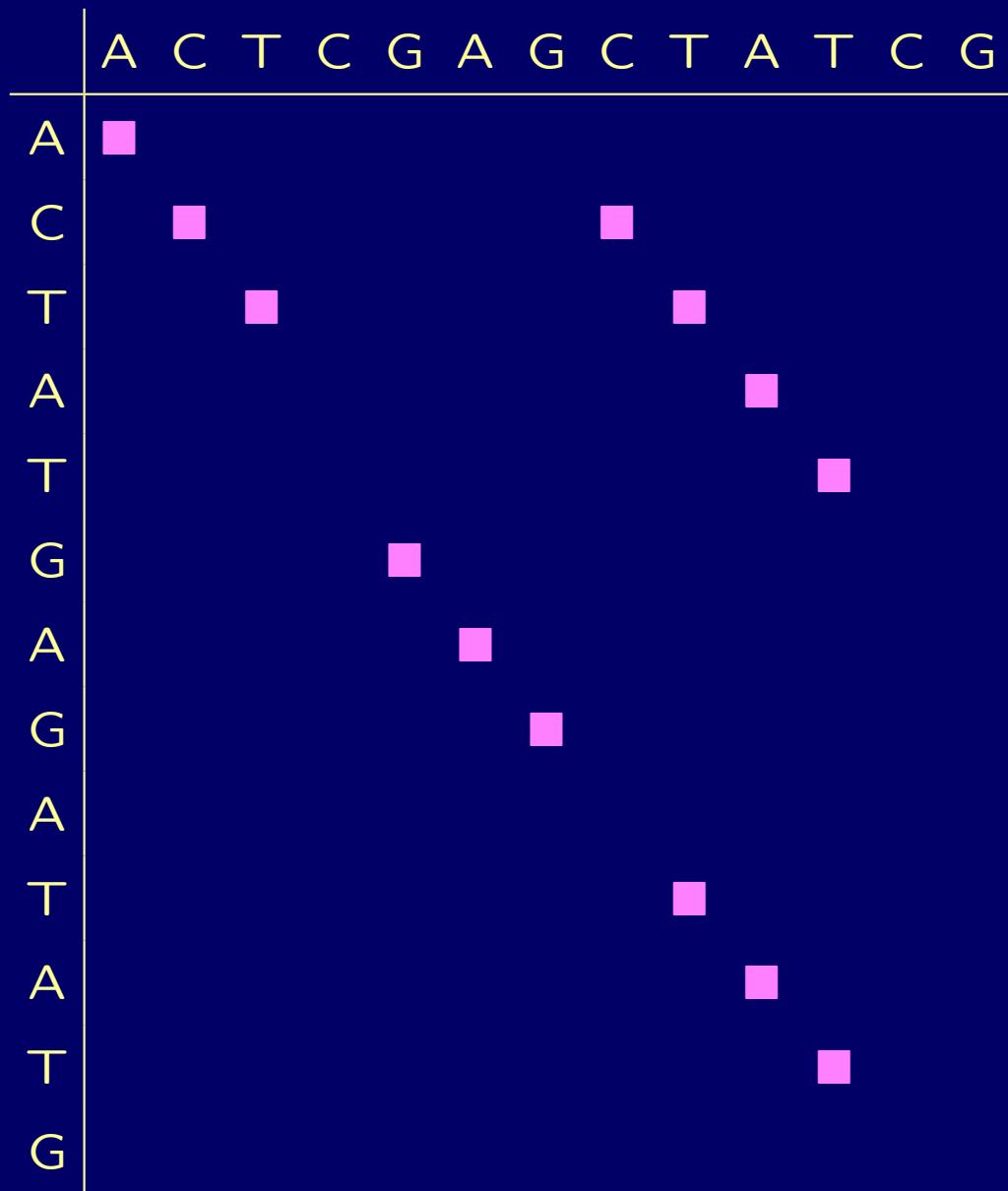


# Dot Plot : vue par fenêtre

match (identité) → ■

mismatch → □

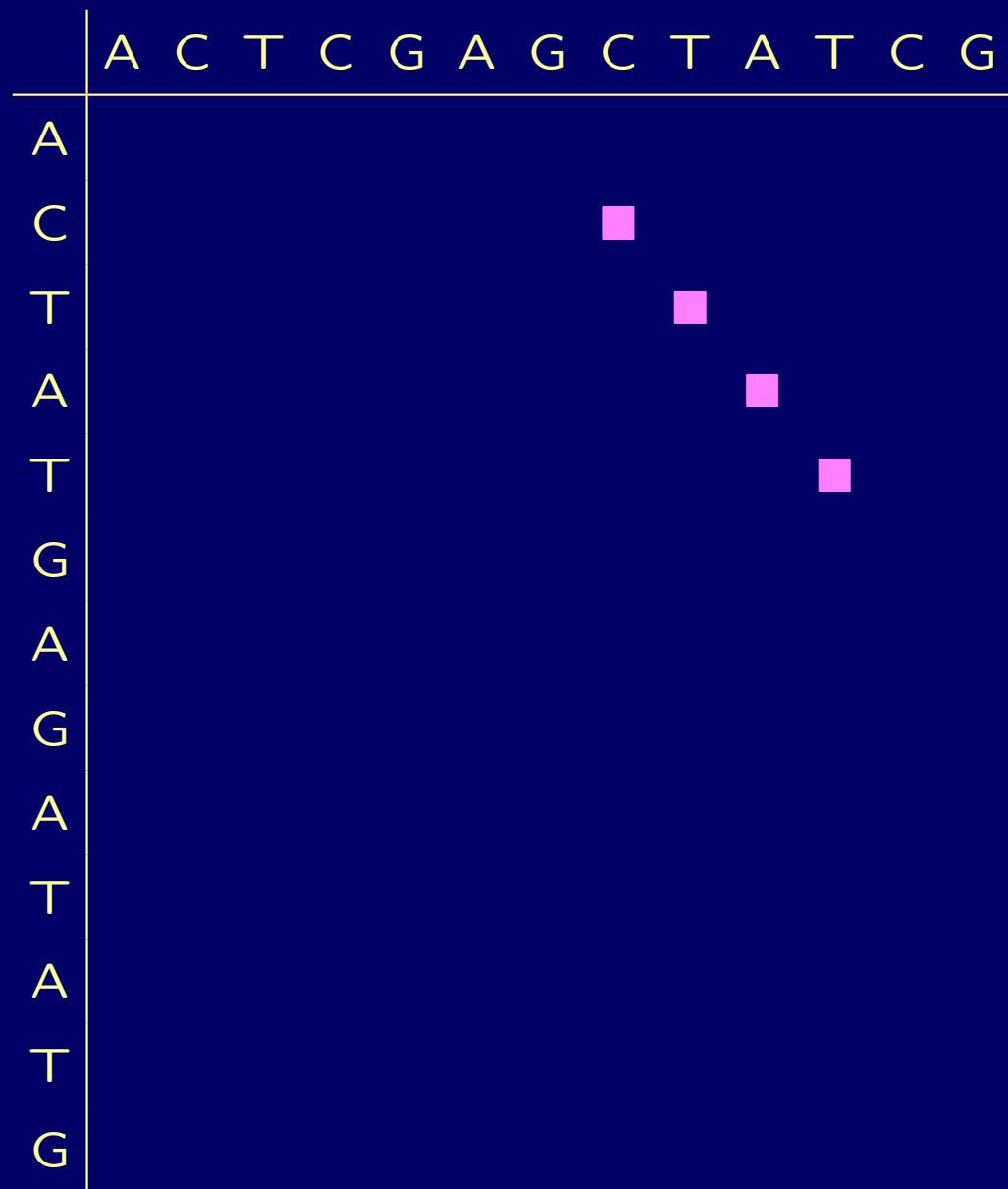
Fenêtre de taille 3



# Dot Plot : vue par fenêtre

match (identité) → ■  
mismatch → □

Fenêtre de taille 4

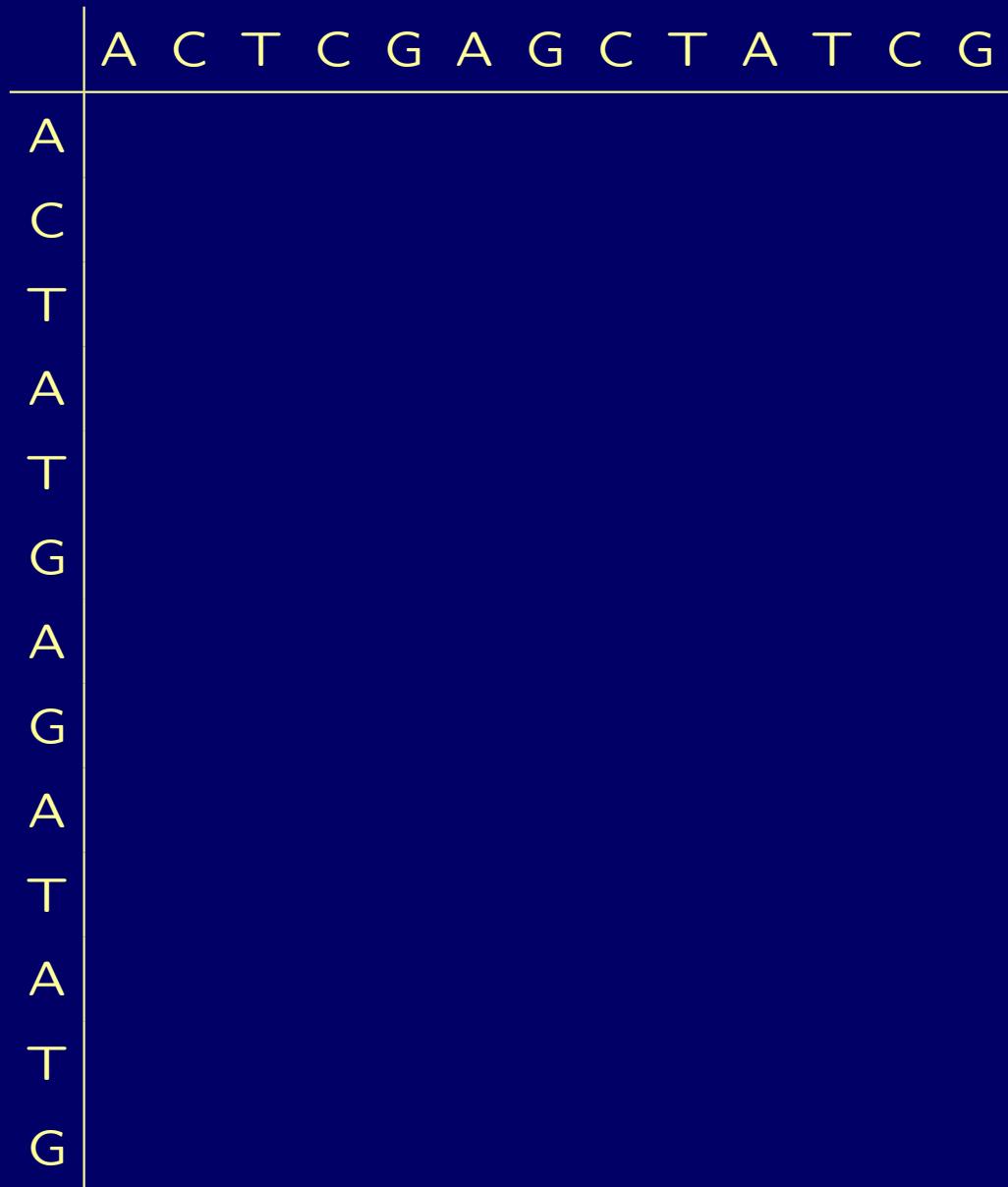


# Dot Plot : vue par fenêtre

match (identité) → ■

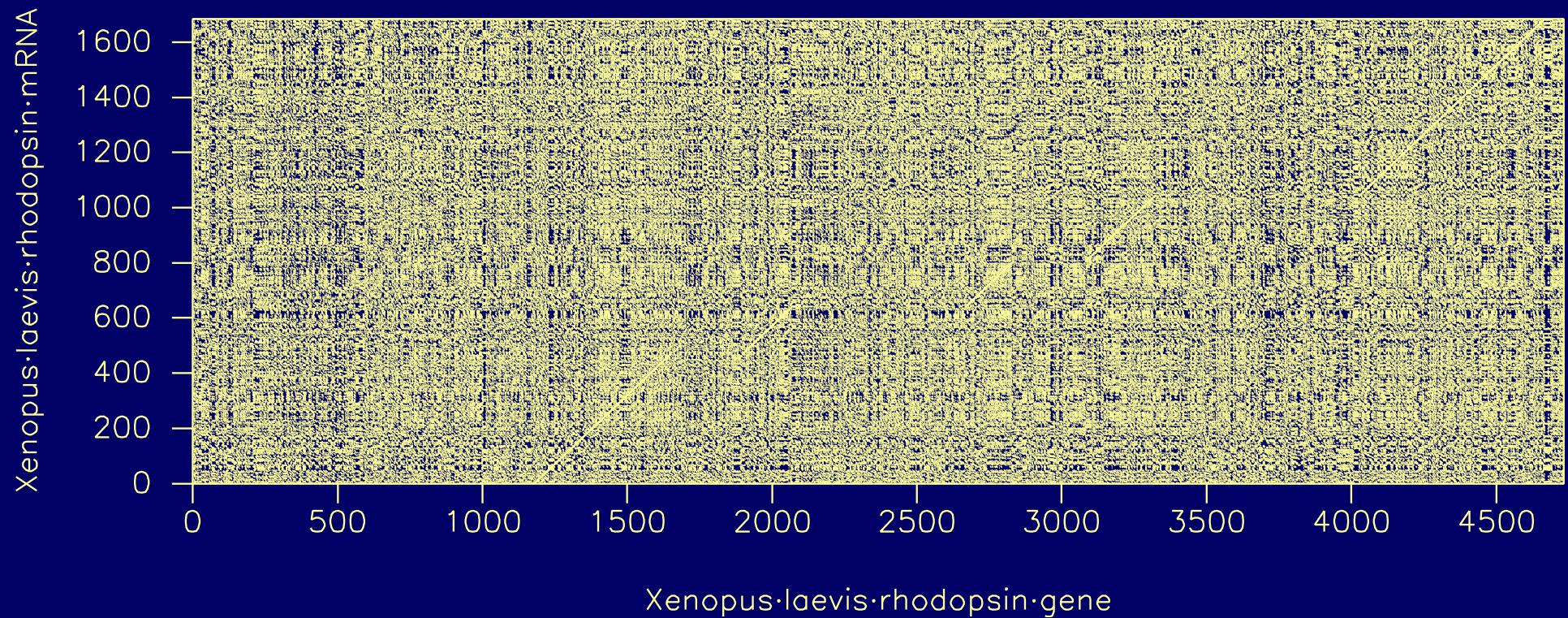
mismatch → □

Fenêtre de taille 5



# Dotplot sur des séquences réelles

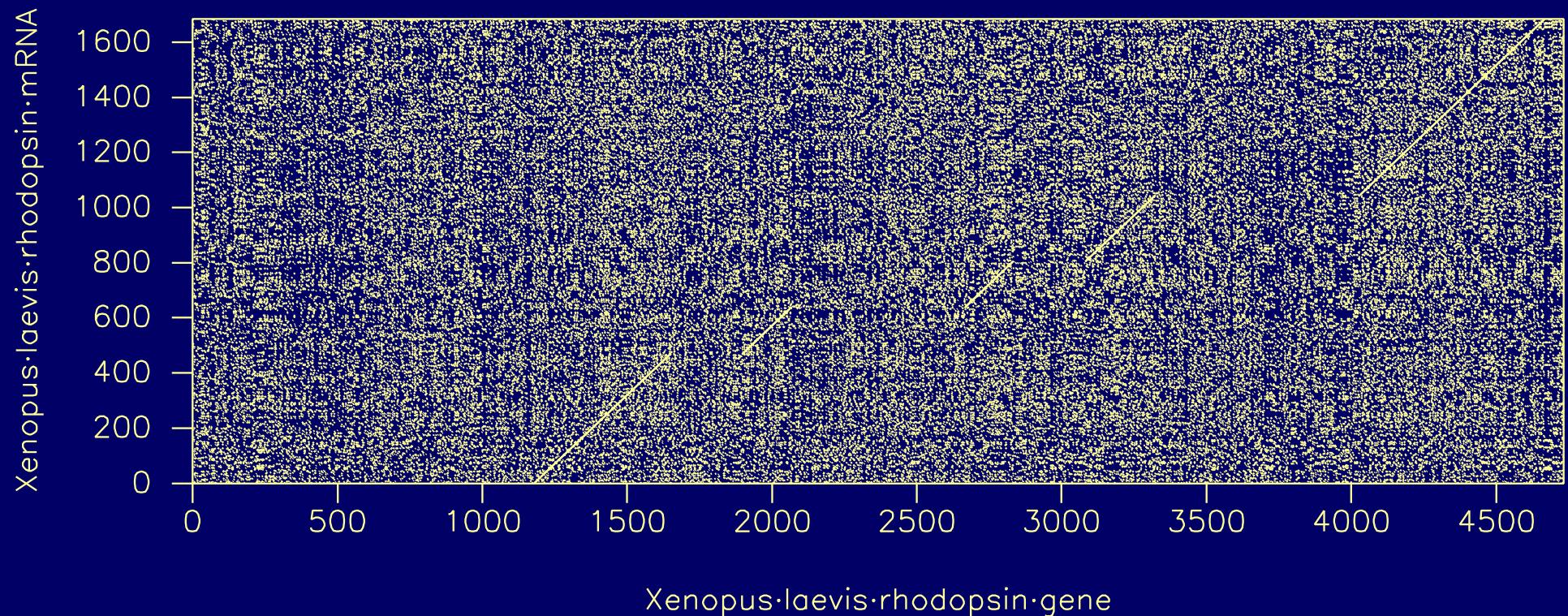
dotpath (08/06/05)



Taille fenêtre = 2

# Dotplot sur des séquences réelles

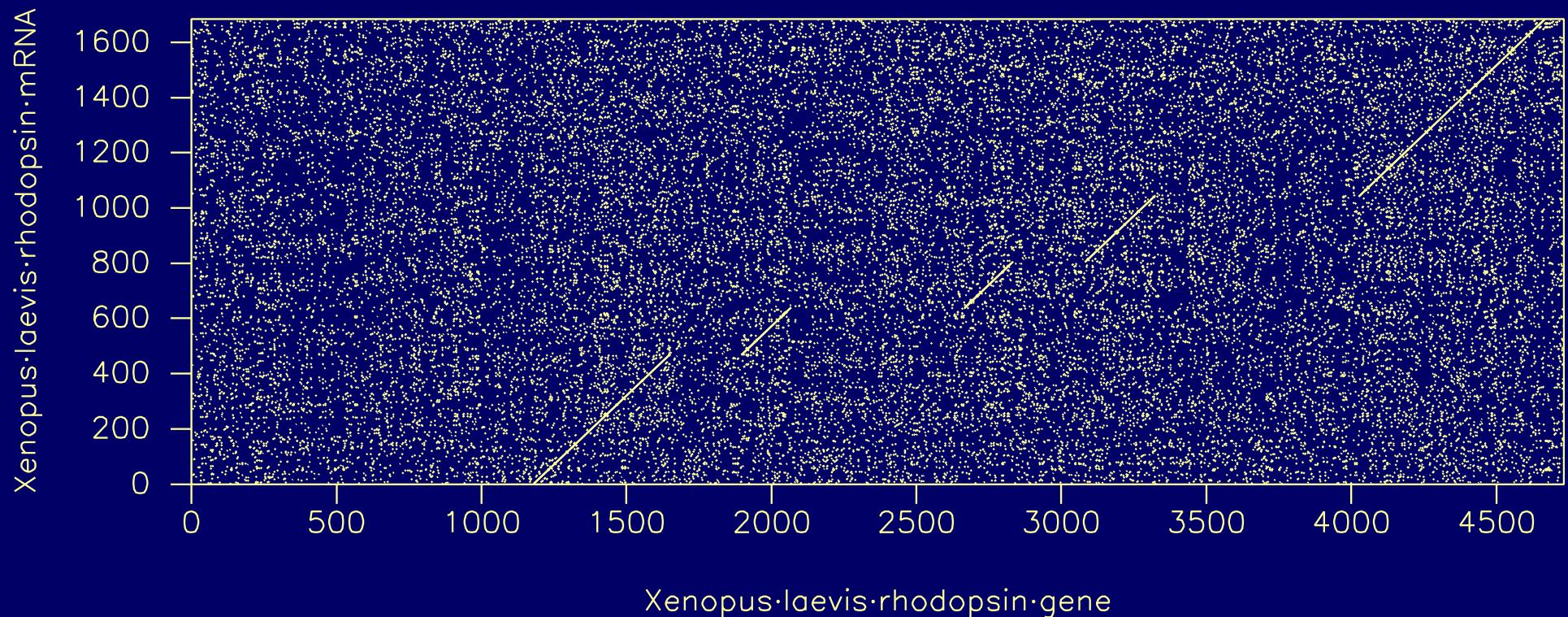
dotpath (08/06/05)



Taille fenêtre = 3

# Dotplot sur des séquences réelles

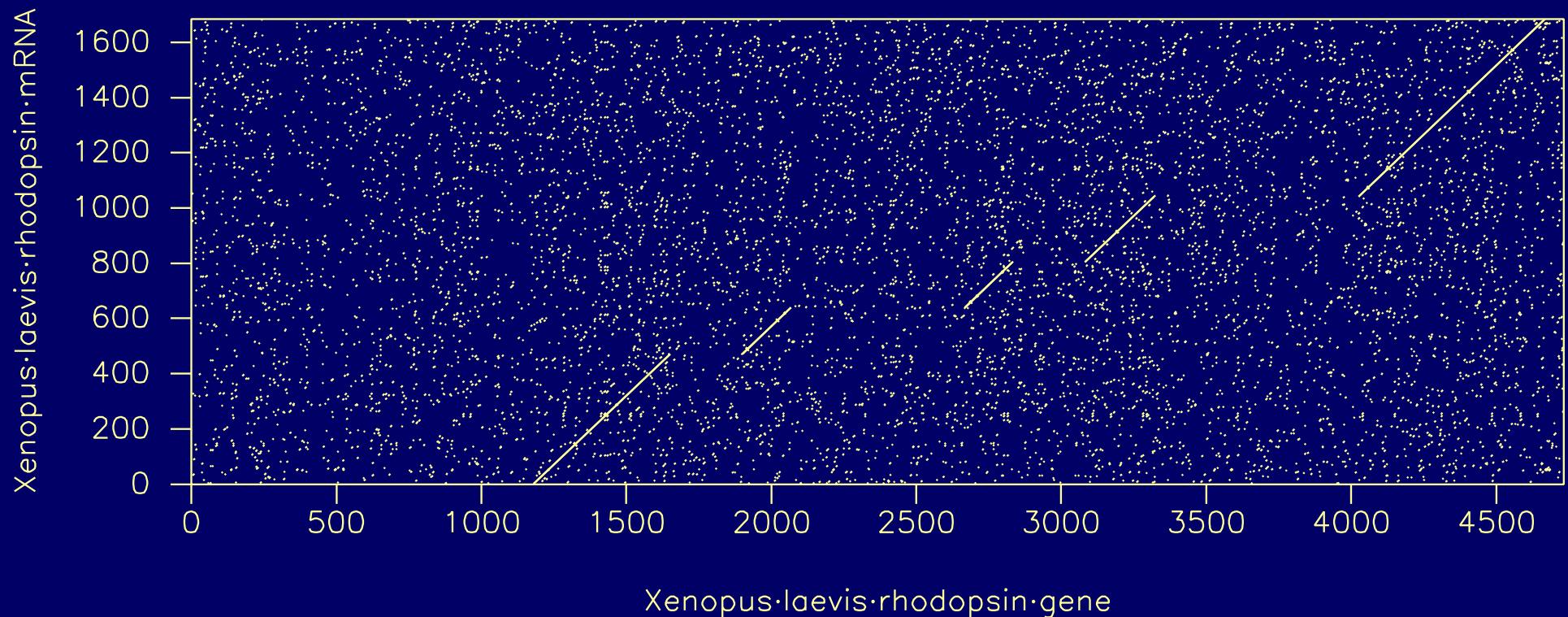
dotpath (08/06/05)



Taille fenêtre = 4

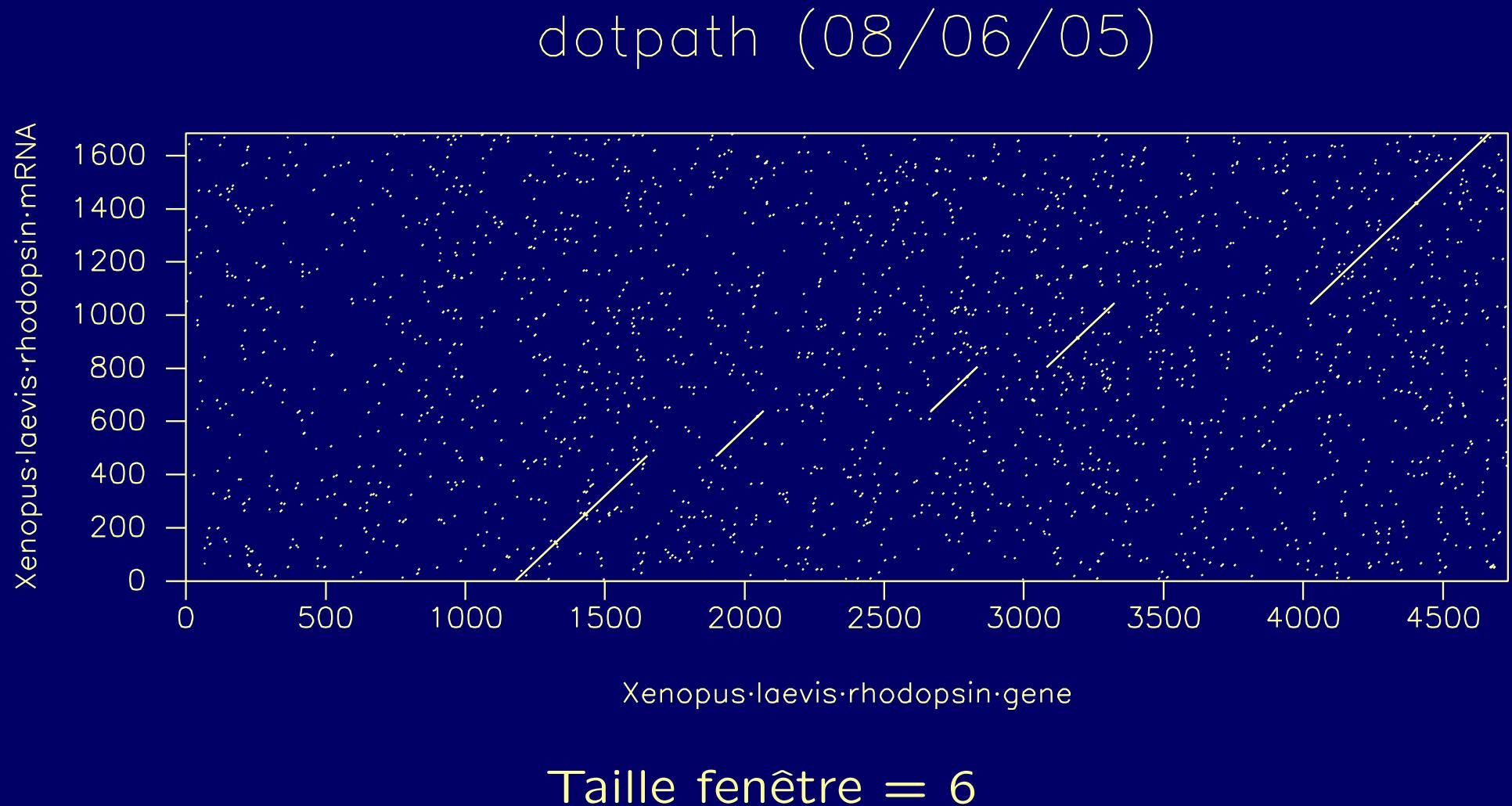
# Dotplot sur des séquences réelles

dotpath (08/06/05)



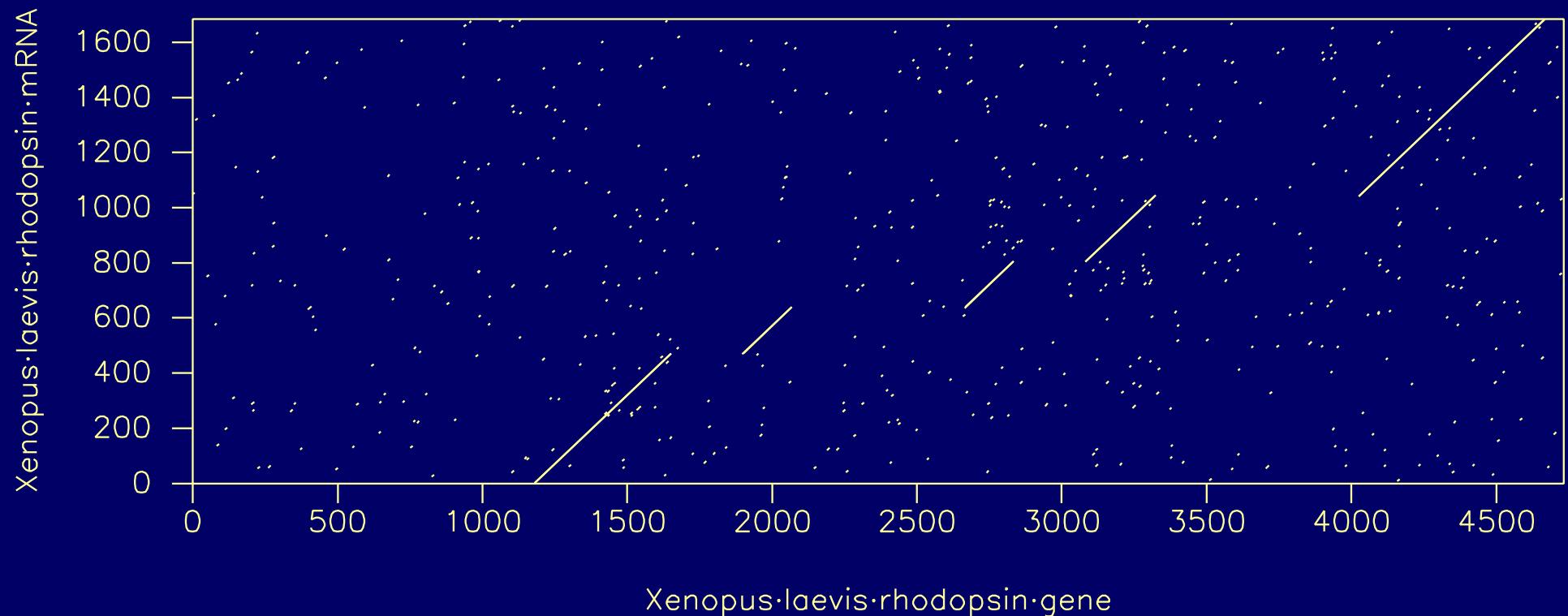
Taille fenêtre = 5

# Dotplot sur des séquences réelles



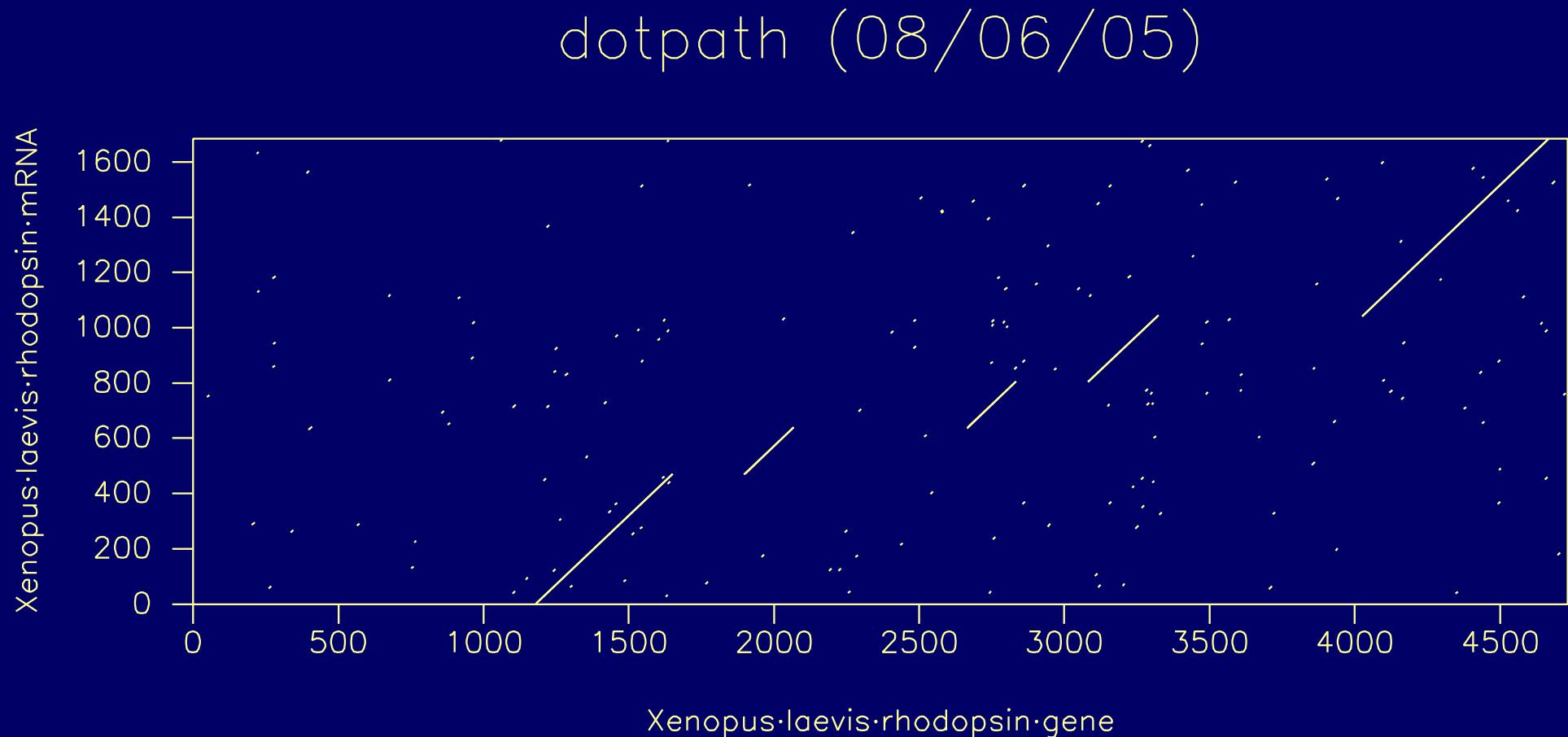
# Dotplot sur des séquences réelles

dotpath (08/06/05)



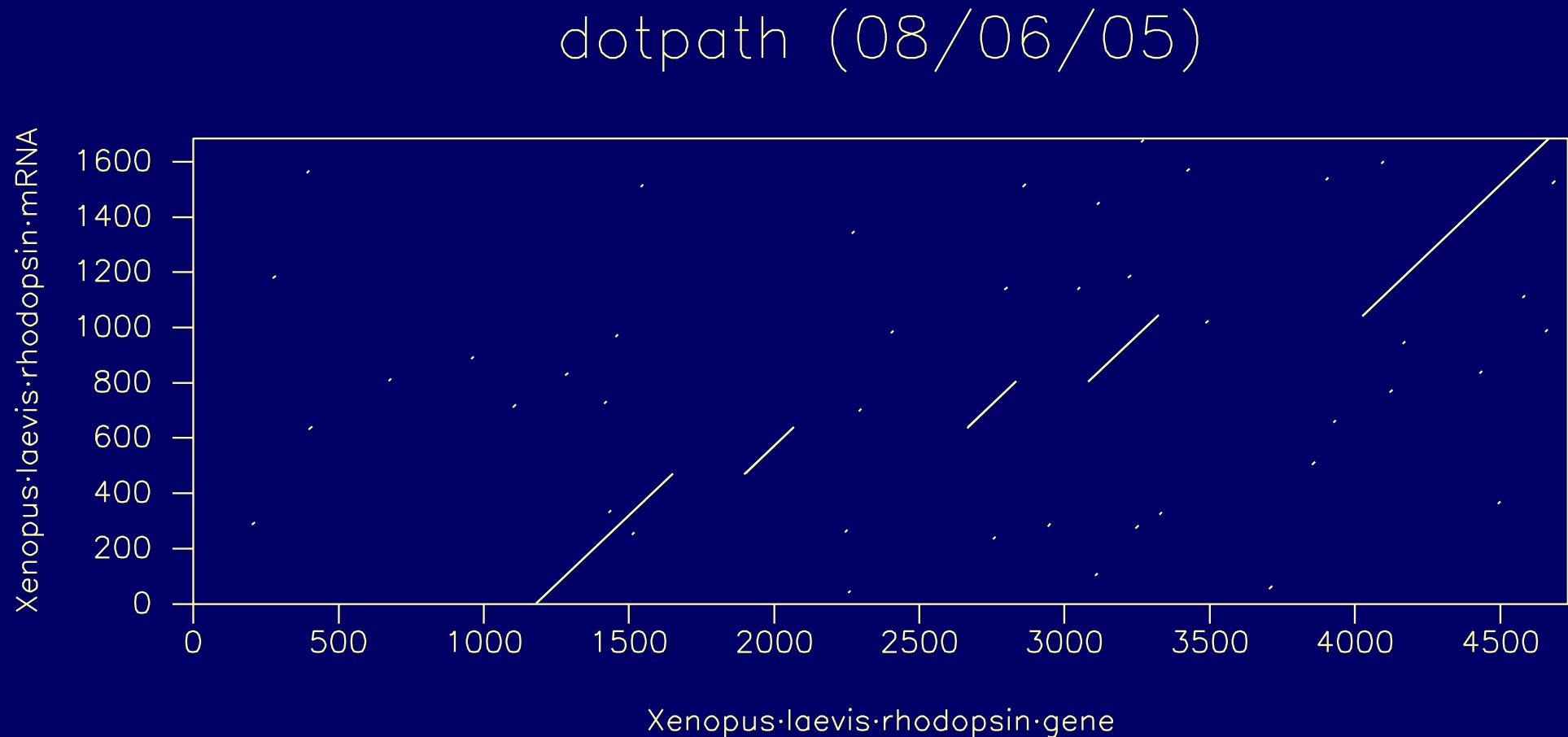
Taille fenêtre = 7

# Dotplot sur des séquences réelles



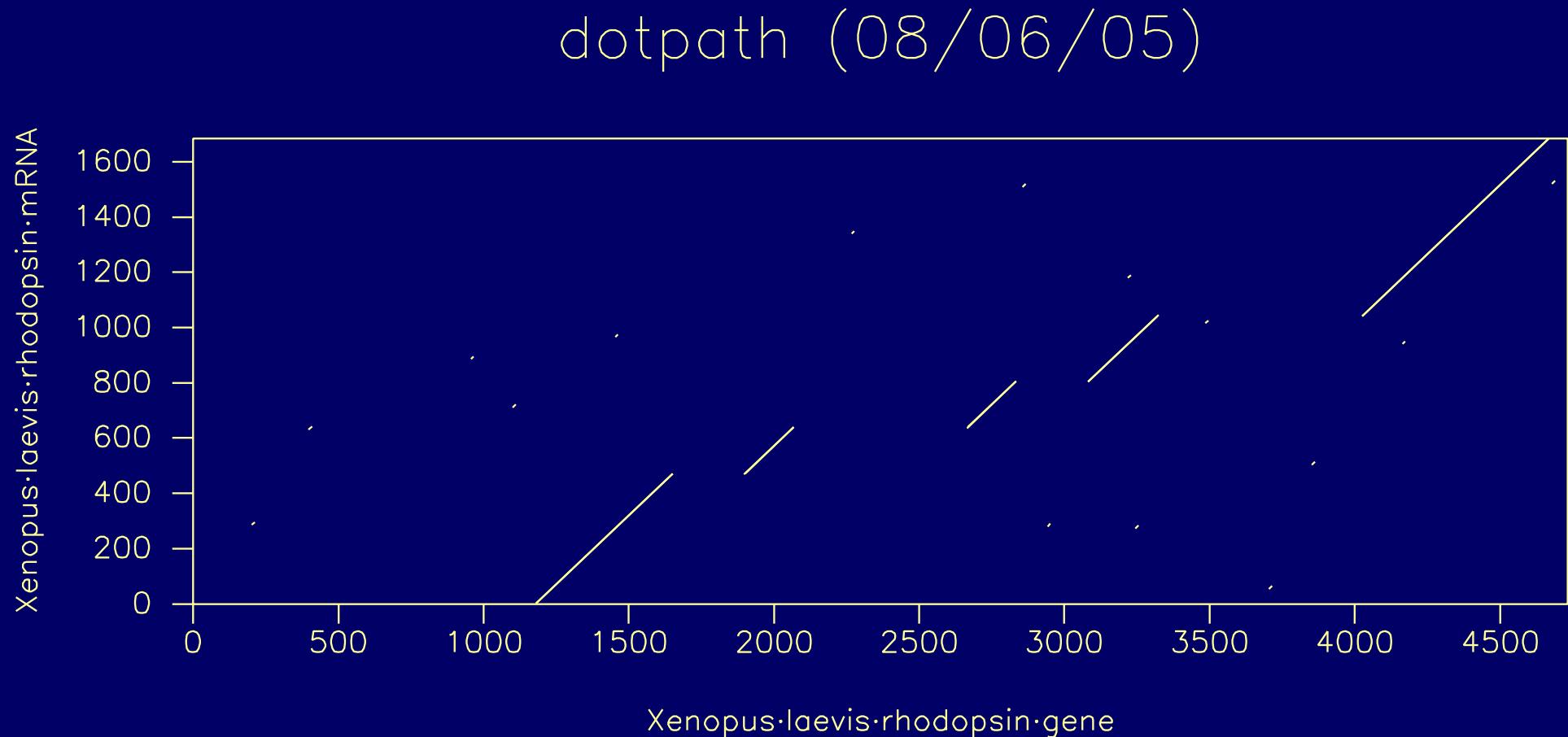
Taille fenêtre = 8

# Dotplot sur des séquences réelles



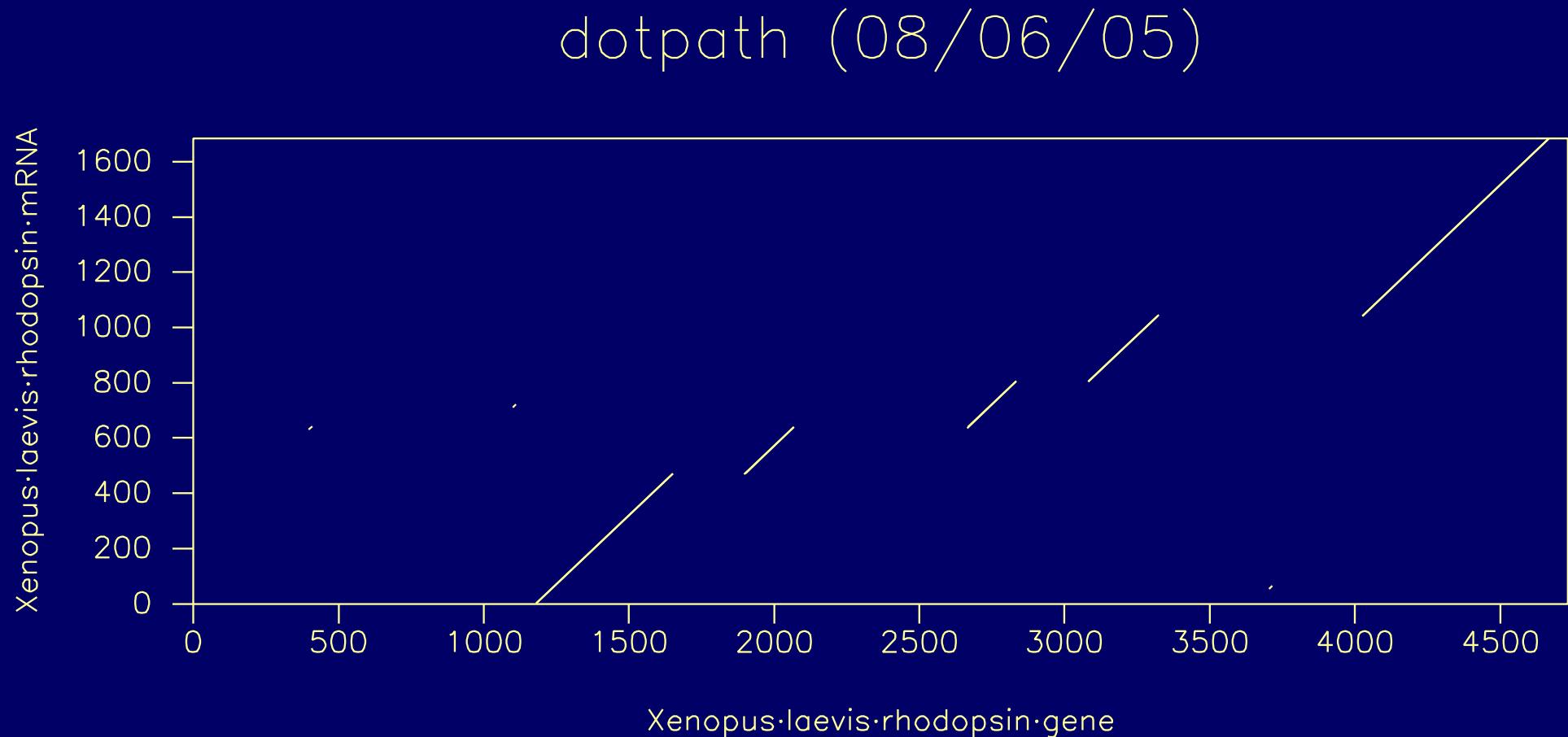
Taille fenêtre = 9

# Dotplot sur des séquences réelles



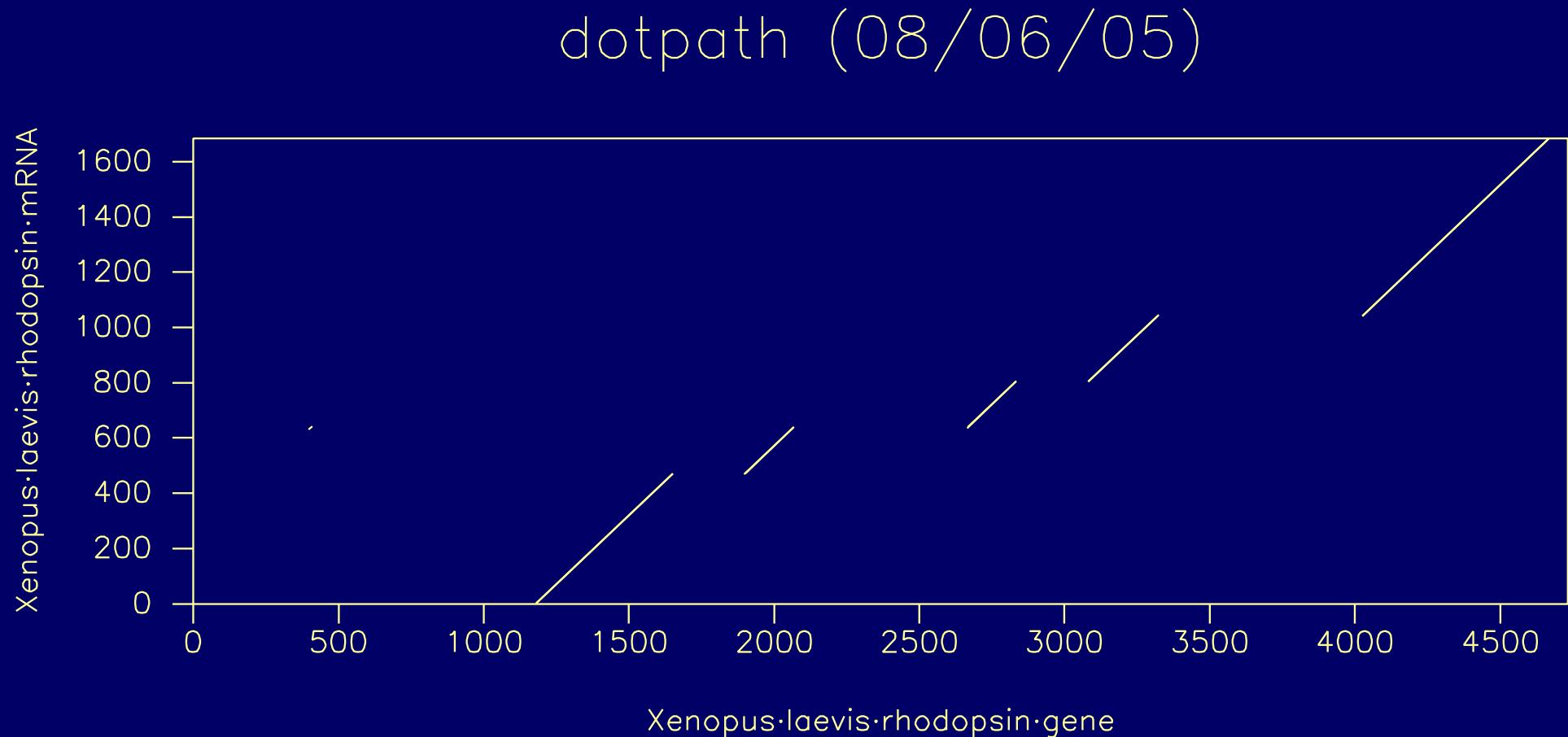
Taille fenêtre = 10

# Dotplot sur des séquences réelles



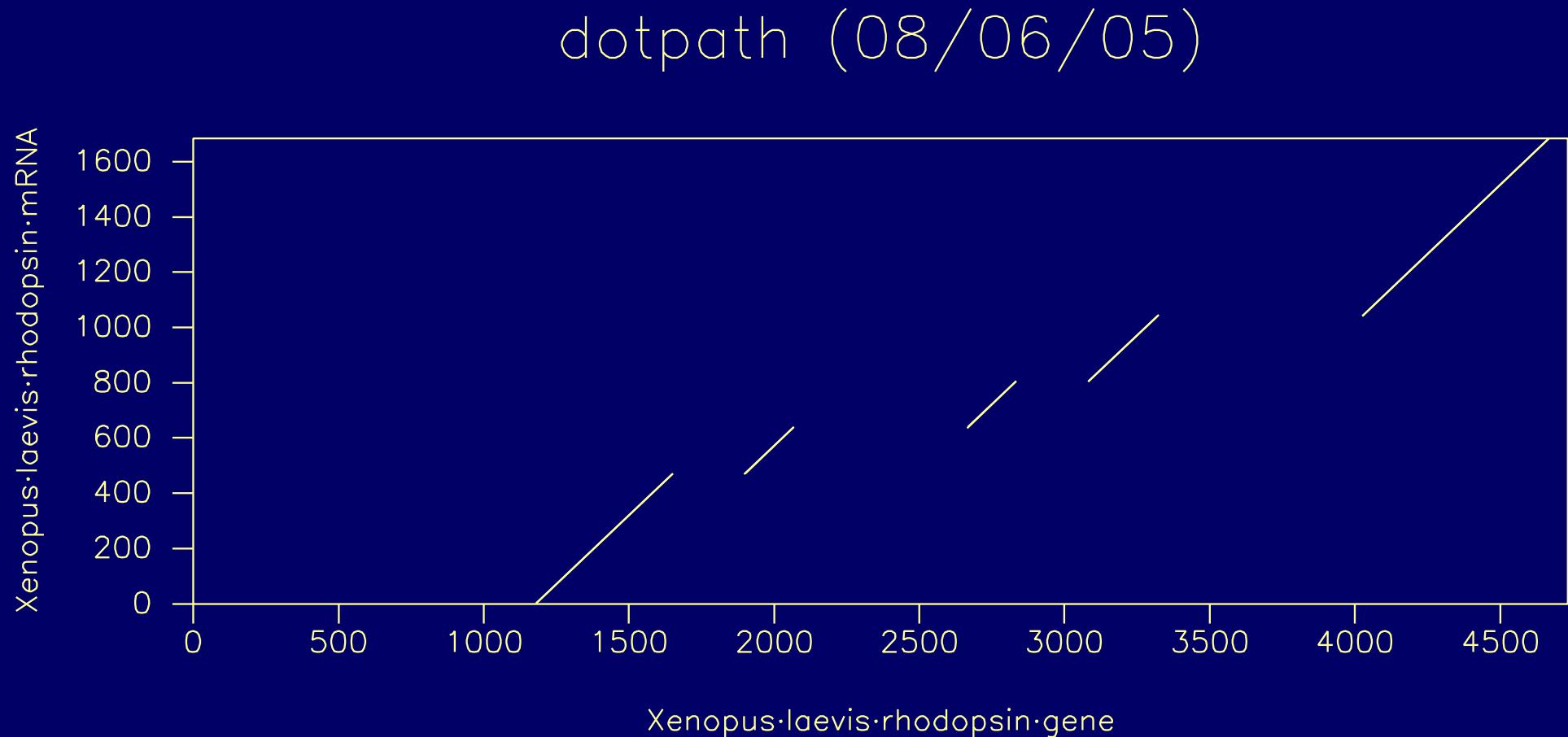
Taille fenêtre = 11

# Dotplot sur des séquences réelles



Taille fenêtre = 12

# Dotplot sur des séquences réelles

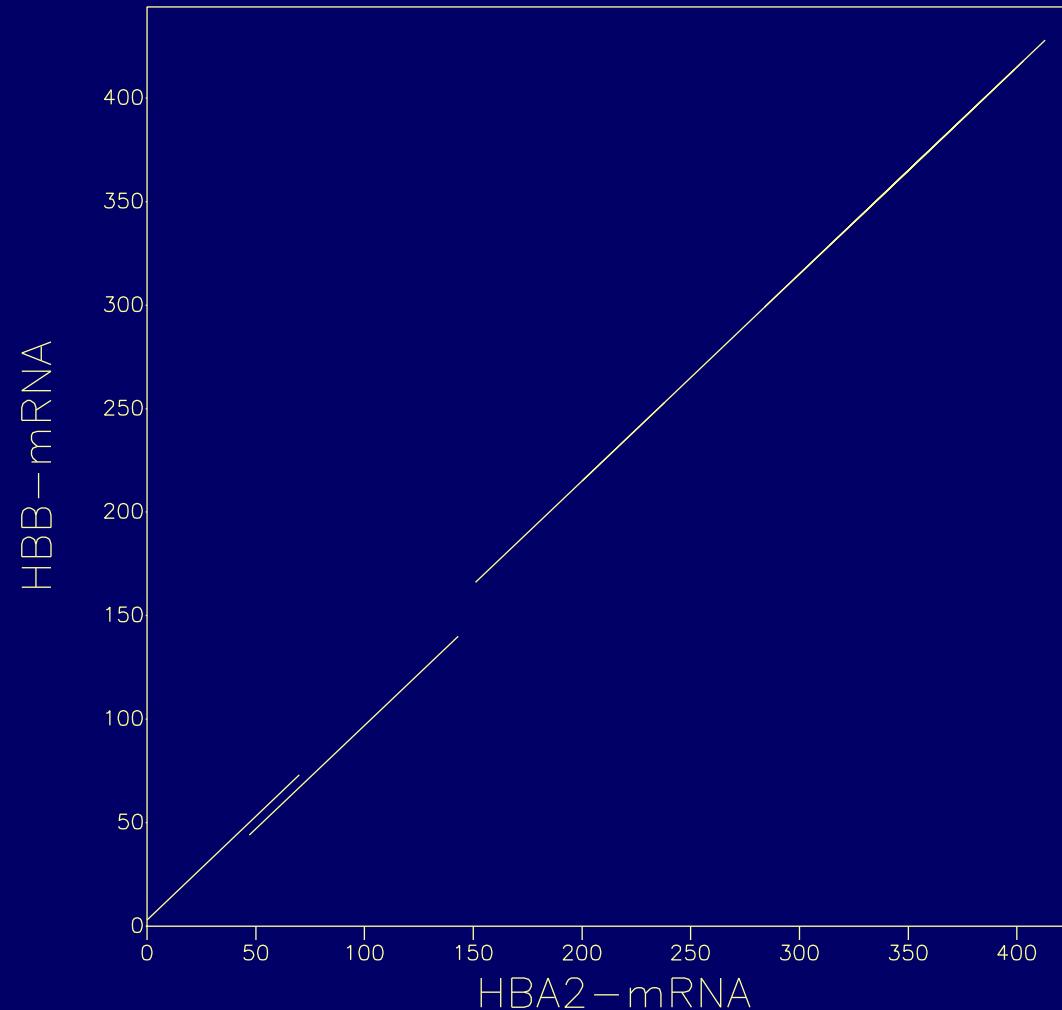


Taille fenêtre = 20

# Repérer des similarités globales

Dotmatcher: HBA2-mRNA vs HBB-mRNA

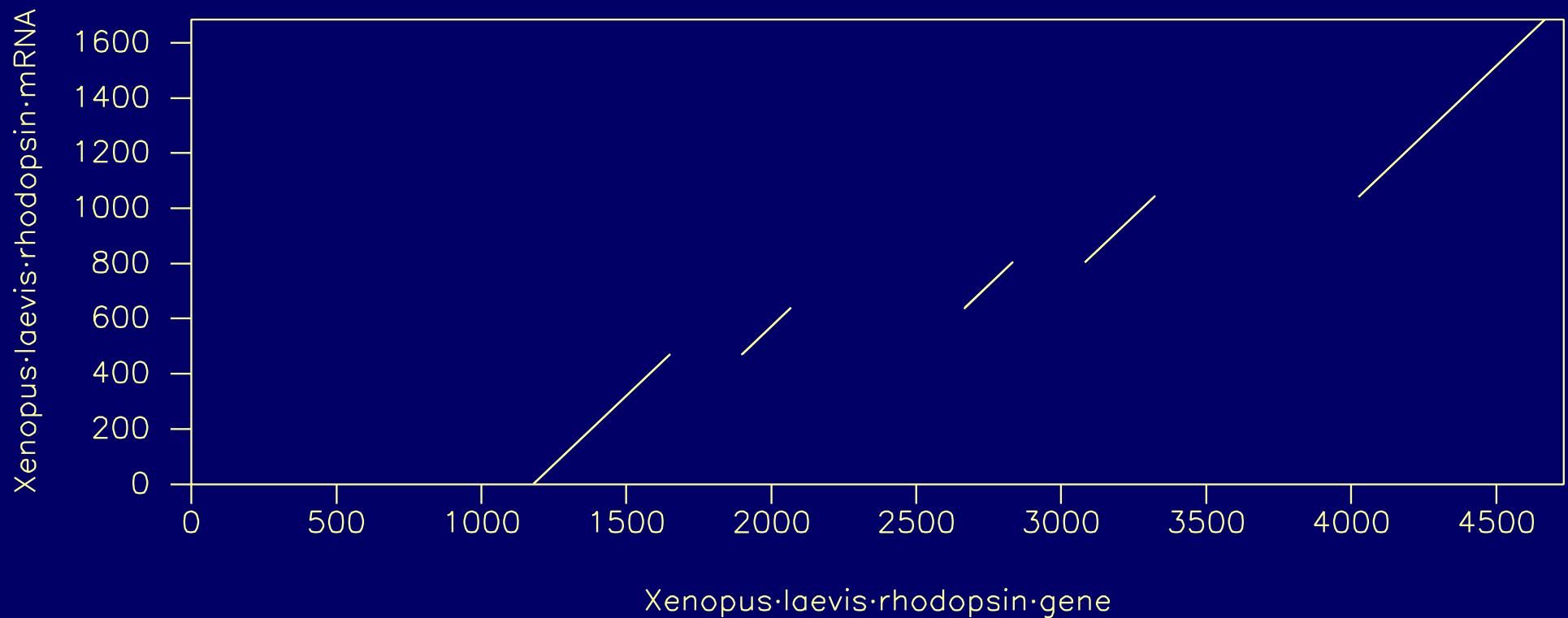
(windowsize = 60, threshold = 70.00 09/06/05)



Hémoglobine humaine chaînes  $\alpha_2$  et  $\beta$

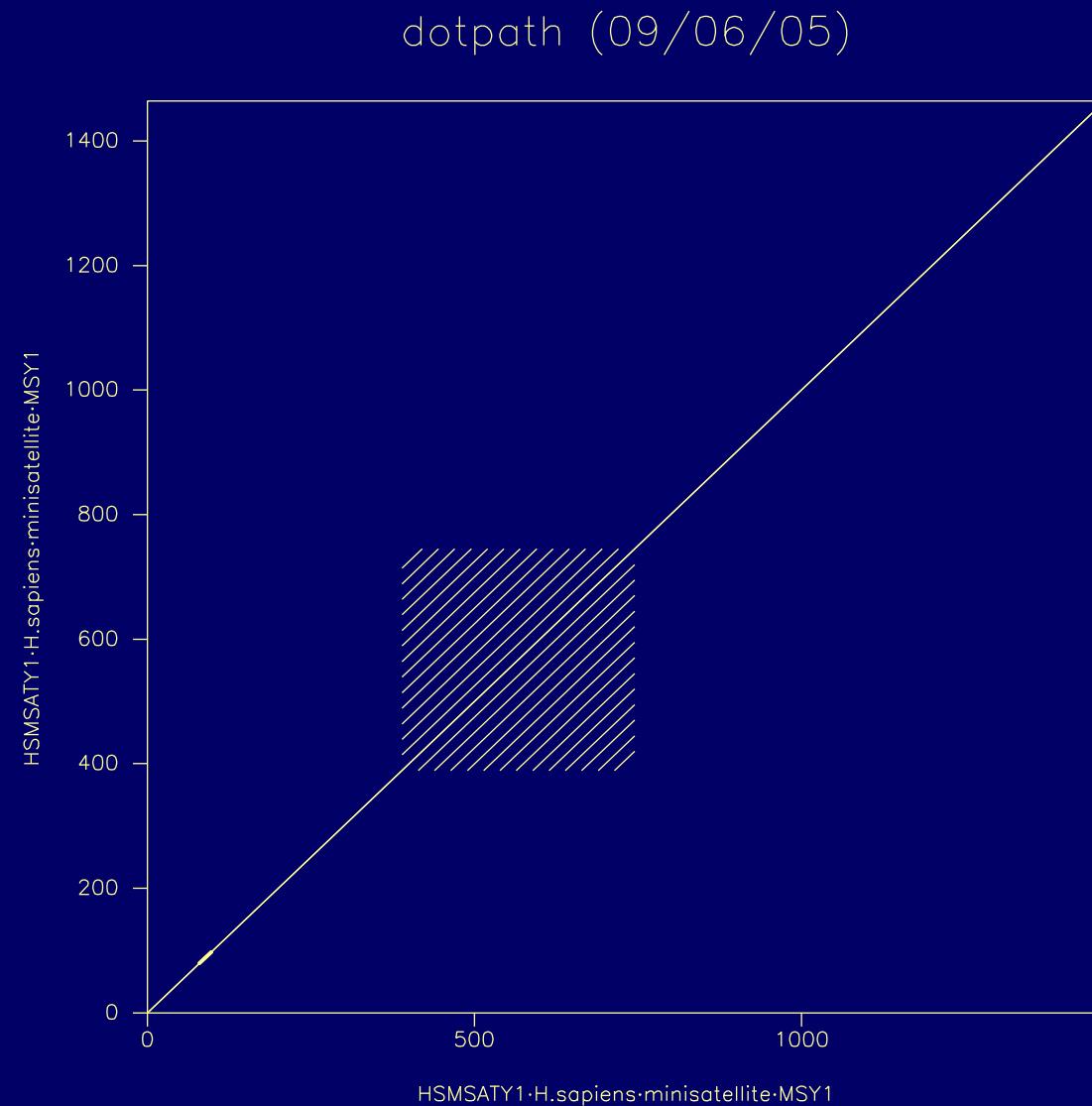
# Repérer des similarités locales

dotpath (08/06/05)



# Repérer des répétitions

10



Minisatellite humain MSY1, taille de la fenêtre = 20

## Avantages :

- simple, visuel
- très informatif

## Inconvénients :

- identification ⇒ pas de méthode de détection automatique
  - interprétation ⇒ pas de mesure objective
- ⇒ Besoin d'une mesure quantitative de similarité

- Introduction
- Dotplot
- Alignement : définitions et scores
- Algorithme d'alignement (programmation dynamique)
- Autres types d'alignements
- La suite ...

- Mise en correspondance de 2 séquences (ADN ou protéine)

<i>R</i>	<i>D</i>	<i>I</i>	<i>S</i>	<i>L</i>	<i>V</i>	-	-	-	<i>K</i>	<i>N</i>	<i>A</i>	<i>G</i>	<i>I</i>
<i>R</i>	<i>N</i>	<i>I</i>	-	<i>L</i>	<i>V</i>	<i>S</i>	<i>D</i>	<i>A</i>	<i>K</i>	<i>N</i>	<i>V</i>	<i>G</i>	<i>I</i>

- Mise en correspondance de 2 séquences (ADN ou protéine)

<i>R</i>	<i>D</i>	<i>I</i>	<i>S</i>	<i>L</i>	<i>V</i>	-	-	-	<i>K</i>	<i>N</i>	<i>A</i>	<i>G</i>	<i>I</i>
<i>R</i>	<i>N</i>	<i>I</i>	-	<i>L</i>	<i>V</i>	<i>S</i>	<i>D</i>	<i>A</i>	<i>K</i>	<i>N</i>	<i>V</i>	<i>G</i>	<i>I</i>

- 3 événements mutationnels élémentaires = 3 opérations

- ▷ substitution
  - ▷ insertion
  - ▷ délétion
- $\left. \begin{array}{l} \text{insertion} \\ \text{délétion} \end{array} \right\} indels$

- Mise en correspondance de 2 séquences (ADN ou protéine)

R	D	I	S	L	V	-	-	-	K	N	A	G	I
R	N	I	-	L	V	S	D	A	K	N	V	G	I

- 3 événements mutationnels élémentaires = 3 opérations
  - ▷ substitution
  - ▷ insertion
  - ▷ délétion

*indels*
- Donnée : une paire de séquences + une méthode de score/dist.
- But : Quantifier et localiser la similarité : score/dist. + alignement

- Mise en correspondance de 2 séquences (ADN ou protéine)

<i>R</i>	<i>D</i>	<i>I</i>	<i>S</i>	<i>L</i>	<i>V</i>	-	-	-	<i>K</i>	<i>N</i>	<i>A</i>	<i>G</i>	<i>I</i>
<i>R</i>	<i>N</i>	<i>I</i>	-	<i>L</i>	<i>V</i>	<i>S</i>	<i>D</i>	<i>A</i>	<i>K</i>	<i>N</i>	<i>V</i>	<i>G</i>	<i>I</i>

- 3 événements mutationnels élémentaires = 3 opérations
  - ▷ substitution
  - ▷ insertion
  - ▷ délétion

*indels*
- Donnée : une paire de séquences + une méthode de score/dist.
- But : Quantifier et localiser la similarité : score/dist. + alignement

⇒ Trouver la meilleure mise en correspondance = meilleur score/dist.

Soient  $s$  et  $t$  deux séquences respectivement de longueur  $n$  et  $m$ .  
On note  $s_i$  le symbole à la position  $i$  de  $s$ .

Soient  $s$  et  $t$  deux séquences respectivement de longueur  $n$  et  $m$ .  
On note  $s_i$  le symbole à la position  $i$  de  $s$ .

Définition : un Alignement global de  $s$  et  $t$  :

Soient  $s$  et  $t$  deux séquences respectivement de longueur  $n$  et  $m$ .  
On note  $s_i$  le symbole à la position  $i$  de  $s$ .

Définition : un Alignement global de  $s$  et  $t$  :

→ une matrice à 2 lignes, et entre  $\max(m, n)$  et  $n + m$  colonnes

Soient  $s$  et  $t$  deux séquences respectivement de longueur  $n$  et  $m$ .  
On note  $s_i$  le symbole à la position  $i$  de  $s$ .

Définition : un Alignement global de  $s$  et  $t$  :

- une matrice à 2 lignes, et entre  $\max(m, n)$  et  $n + m$  colonnes
- où chaque colonne met en correspondance deux symboles :  
soit  $(\begin{smallmatrix} s_i \\ t_j \end{smallmatrix})$  ou  $(\begin{smallmatrix} s_i \\ - \end{smallmatrix})$  ou  $(\begin{smallmatrix} - \\ t_j \end{smallmatrix})$

Soient  $s$  et  $t$  deux séquences respectivement de longueur  $n$  et  $m$ .  
On note  $s_i$  le symbole à la position  $i$  de  $s$ .

Définition : un Alignement global de  $s$  et  $t$  :

- une matrice à 2 lignes, et entre  $\max(m, n)$  et  $n + m$  colonnes
- où chaque colonne met en correspondance deux symboles :  
soit  $(\begin{smallmatrix} s_i \\ t_j \end{smallmatrix})$  ou  $(\begin{smallmatrix} s_i \\ - \end{smallmatrix})$  ou  $(\begin{smallmatrix} - \\ t_j \end{smallmatrix})$
- où les colonnes doivent respecter l'ordre des séquences  
après  $(\begin{smallmatrix} s_i \\ t_j \end{smallmatrix})$  on trouve  $(\begin{smallmatrix} s_{i+1} \\ - \end{smallmatrix})$ ,  $(\begin{smallmatrix} s_{i+1} \\ t_{j+1} \end{smallmatrix})$  ou  $(\begin{smallmatrix} - \\ t_{j+1} \end{smallmatrix})$

Soient  $s$  et  $t$  deux séquences respectivement de longueur  $n$  et  $m$ .  
On note  $s_i$  le symbole à la position  $i$  de  $s$ .

Définition : un Alignement global de  $s$  et  $t$  :

- une matrice à 2 lignes, et entre  $\max(m, n)$  et  $n + m$  colonnes
- où chaque colonne met en correspondance deux symboles :  
soit  $(\begin{smallmatrix} s_i \\ t_j \end{smallmatrix})$  ou  $(\begin{smallmatrix} s_i \\ - \end{smallmatrix})$  ou  $(\begin{smallmatrix} - \\ t_j \end{smallmatrix})$
- où les colonnes doivent respecter l'ordre des séquences  
après  $(\begin{smallmatrix} s_i \\ t_j \end{smallmatrix})$  on trouve  $(\begin{smallmatrix} s_{i+1} \\ - \end{smallmatrix})$ ,  $(\begin{smallmatrix} s_{i+1} \\ t_{j+1} \end{smallmatrix})$  ou  $(\begin{smallmatrix} - \\ t_{j+1} \end{smallmatrix})$

Remarque : suivant le nombre de symboles  $-$ , le nombre de colonnes d'un alignement varie.

Séquences  $s := I$ ,  $t := L$

Alignment 1	Alignment 2	Alignment 3
$\begin{pmatrix} I & - \\ - & L \end{pmatrix}$	$\begin{pmatrix} - & I \\ L & - \end{pmatrix}$	$\begin{pmatrix} I \\ L \end{pmatrix}$

Séquences  $s := I$ ,  $t := L$

Alignment 1	Alignment 2	Alignment 3
$\begin{pmatrix} I & - \\ - & L \end{pmatrix}$	$\begin{pmatrix} - & I \\ L & - \end{pmatrix}$	$\begin{pmatrix} I \\ L \end{pmatrix}$

Séquences  $s := IL$  et  $t := LV \Rightarrow 13$  alignements possibles

Séquences  $s := I$ ,  $t := L$

Alignement 1	Alignement 2	Alignement 3
$\begin{pmatrix} I & - \\ - & L \end{pmatrix}$	$\begin{pmatrix} - & I \\ L & - \end{pmatrix}$	$\begin{pmatrix} I \\ L \end{pmatrix}$

Séquences  $s := IL$  et  $t := LV \Rightarrow 13$  alignements possibles

$\begin{pmatrix} I & L \\ L & V \end{pmatrix}$	$\begin{pmatrix} I & L & - \\ L & - & V \end{pmatrix}$	$\begin{pmatrix} I & L & - & - \\ - & - & L & V \end{pmatrix}$	$\begin{pmatrix} - & - & I & L \\ L & V & - & - \end{pmatrix}$
$\begin{pmatrix} I & - & L \\ - & L & V \end{pmatrix}$	$\begin{pmatrix} I & - & L \\ L & V & - \end{pmatrix}$	$\begin{pmatrix} I & - & L & - \\ - & L & - & V \end{pmatrix}$	$\begin{pmatrix} - & I & - & L \\ L & - & V & - \end{pmatrix}$
$\begin{pmatrix} I & L & - \\ - & L & V \end{pmatrix}$	$\begin{pmatrix} - & I & L \\ L & - & V \end{pmatrix}$	$\begin{pmatrix} I & - & - & L \\ - & L & V & - \end{pmatrix}$	$\begin{pmatrix} - & I & L & - \\ L & - & - & V \end{pmatrix}$
	$\begin{pmatrix} - & I & L \\ L & V & - \end{pmatrix}$		

- Identité/substitution : Scores positifs/négatifs
  - matrice  $s$  de score de similarité (PAM, BLOSUM, etc.)
  - $s(a, b) =$  score d'alignement des résidus  $a$  et  $b$

- Identité/substitution : Scores positifs/négatifs
  - matrice  $s$  de score de similarité (PAM, BLOSUM, etc.)
  - $s(a, b) =$  score d'alignement des résidus  $a$  et  $b$
- Insertion/délétion (indels) : Scores négatifs
  - fonction de pénalité  $\begin{cases} \text{élémentaire : unitaire pour un indel} \\ \text{complexe : affine, logarithmique} \end{cases}$

- Identité/substitution : Scores positifs/négatifs
  - matrice  $s$  de score de similarité (PAM, BLOSUM, etc.)
  - $s(a, b) =$  score d'alignement des résidus  $a$  et  $b$
- Insertion/délétion (indels) : Scores négatifs
  - fonction de pénalité  $\begin{cases} \text{élémentaire} : \text{unitaire pour un indel} \\ \text{complexe} : \text{affine, logarithmique} \end{cases}$
- Score de l'alignement : somme des scores des événements élémentaires qui le composent

- Identité/substitution : Scores positifs/négatifs
  - matrice  $s$  de score de similarité (PAM, BLOSUM, etc.)
  - $s(a, b) =$  score d'alignement des résidus  $a$  et  $b$
- Insertion/délétion (indels) : Scores négatifs
  - fonction de pénalité  $\begin{cases} \text{élémentaire} : \text{unitaire pour un indel} \\ \text{complexe} : \text{affine, logarithmique} \end{cases}$
- Score de l'alignement : somme des scores des événements élémentaires qui le composent

⇒ Maximiser le score

Distance d'édition

$$\text{Lev} \begin{pmatrix} a \\ b \end{pmatrix} = 1$$

$$\text{Lev} \begin{pmatrix} a \\ - \end{pmatrix} = 1$$

$$\text{Lev} \begin{pmatrix} - \\ b \end{pmatrix} = 1$$

Distance d'édition généralisée

$$\text{Edit} \begin{pmatrix} a \\ b \end{pmatrix} = \text{Sub}(a, b) \geq 0$$

$$\text{Edit} \begin{pmatrix} a \\ - \end{pmatrix} = \text{Del}(a) \geq 0$$

$$\text{Edit} \begin{pmatrix} - \\ b \end{pmatrix} = \text{Ins}(b) \geq 0$$

Seule l'identité = 0

Distance d'édition

$$\text{Lev} \begin{pmatrix} a \\ b \end{pmatrix} = 1$$

$$\text{Lev} \begin{pmatrix} a \\ - \end{pmatrix} = 1$$

$$\text{Lev} \begin{pmatrix} - \\ b \end{pmatrix} = 1$$

Distance d'édition généralisée

$$\text{Edit} \begin{pmatrix} a \\ b \end{pmatrix} = \text{Sub}(a, b) \geq 0$$

$$\text{Edit} \begin{pmatrix} a \\ - \end{pmatrix} = \text{Del}(a) \geq 0$$

$$\text{Edit} \begin{pmatrix} - \\ b \end{pmatrix} = \text{Ins}(b) \geq 0$$

Seule l'identité = 0

- Coût de l'alignement : somme des coûts des événements élémentaires qui le composent

Distance d'édition

$$Lev \begin{pmatrix} a \\ b \end{pmatrix} = 1$$

$$Lev \begin{pmatrix} a \\ - \end{pmatrix} = 1$$

$$Lev \begin{pmatrix} - \\ b \end{pmatrix} = 1$$

Distance d'édition généralisée

$$Edit \begin{pmatrix} a \\ b \end{pmatrix} = Sub(a, b) \geq 0$$

$$Edit \begin{pmatrix} a \\ - \end{pmatrix} = Del(a) \geq 0$$

$$Edit \begin{pmatrix} - \\ b \end{pmatrix} = Ins(b) \geq 0$$

Seule l'identité = 0

- Coût de l'alignement : somme des coûts des événements élémentaires qui le composent

⇒ Minimiser la distance

- Introduction
- Dotplot
- Alignement : définitions et scores
- Algorithme d'alignement (programmation dynamique)
- Autres types d'alignements
- La suite ...

- Aligner les séquences : *ACGCTATC* et *ACTGTAATG*
- Scores : indel = -1, mismatch -2, match = 2

- Aligner les séquences : *ACGCTATC* et *ACTGTAATG*
- Scores : indel = -1, mismatch -2, match = 2
- Alignement optimal complet, dernière opération : 3 possibilités

- Aligner les séquences : *ACGCTATC* et *ACTGTAATG*
- Scores : indel = -1, mismatch -2, match = 2
- Alignment optimal complet, dernière opération : 3 possibilités

<u>substitution</u>	<u>délétion</u>	<u>insertion</u>
ACGGCTATC ? ? ?   C ACTGTAATG	ACGGCTATC   C ? ? ?   ACTGTAATG   -	ACGGCTATC   - ? ? ?   ACTGTAATG   G

- Aligner les séquences : *ACGCTATC* et *ACTGTAATG*
- Scores : indel = -1, mismatch -2, match = 2
- Alignment optimal complet, dernière opération : 3 possibilités

<u>substitution</u>	<u>délétion</u>	<u>insertion</u>
ACGGCTATC ? ? ?   C ACTGTAATG	ACGGCTATC   C ? ? ?   ACTGTAATG   -	ACGGCTATC   - ? ? ?   ACTGTAATG   G

Problème sur des séquences plus courtes

- Aligner les séquences : *ACGCTATC* et *ACTGTAATG*
- Scores : indel = -1, mismatch -2, match = 2
- Alignment optimal complet, dernière opération : 3 possibilités



Problème sur des séquences plus courtes

⇒ Récurrence, programmation dynamique

- La programmation dynamique résout les problèmes en combinant les solutions de sous-problèmes
  - résout chaque sous-problème 1 seule fois
  - mémorise sa solution dans une matrice (de prog. dyn.)  
(épargnant ainsi le recalcul de la sol. chaque fois que le sous-pb est rencontré)

- La programmation dynamique résout les problèmes en combinant les solutions de sous-problèmes
  - résout chaque sous-problème 1 seule fois
  - mémorise sa solution dans une matrice (de prog. dyn.)  
(épargnant ainsi le recalcul de la sol. chaque fois que le sous-pb est rencontré)

Ici :

calculs intermédiaires

=

scores d'alignements entre préfixes

# Algorithme de Needleman-Wunsch

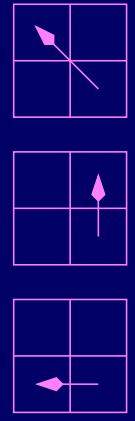
21

	0	1	2	3	4	5	6	7	8	9
0										
1	A									
2	C									
3	T									
4	G									
5	T									
6	A									
7	A									
8	T									
9	G									

Création d'une table indexée par les deux séquences

Case  $(i, j)$  : score alignment entre les  $i$  premières bases de  $ACTGTAATG$  et les  $j$  premières bases de  $ACGGCTATC$

- $s$  : une matrice de score ;  $g$  : pénalité associée à un indel
- Initialisation :  $M(0, 0) = 0$ ,  $M(0, j) = g \times j$ ,  $M(i, 0) = g \times i$
- Remplissage

$$M(i, j) = \max \begin{cases} M(i - 1, j - 1) + s(x_i, y_j) & \text{match/mismatch} \\ M(i - 1, j) + g & \text{délétion} \\ M(i, j - 1) + g & \text{insertion} \end{cases}$$


The diagram consists of three separate 2x2 grids, each with arrows indicating movement from one cell to another. The top grid shows a diagonal arrow from top-left to bottom-right, labeled 'match/mismatch'. The middle grid shows a vertical arrow pointing upwards from bottom-left to top-right, labeled 'délétion'. The bottom grid shows a horizontal arrow pointing leftwards from top-right to bottom-left, labeled 'insertion'.

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

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

		A	C	G	G	C	T	A	T	C
	0									
A										
C										
T										
G										
T										
A										
A										
T										
G										

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

		A	C	G	G	C	T	A	T	C
	0	-1								
A										
C										
T										
G										
T										
A										
A										
T										
G										

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

		A	C	G	G	C	T	A	T	C
	0	-1	-2							
A										
C										
T										
G										
T										
A										
A										
T										
G										

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

		A	C	G	G	C	T	A	T	C
	0	-1	-2	-3						
A										
C										
T										
G										
T										
A										
A										
T										
G										

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

		A	C	G	G	C	T	A	T	C
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
A										
C										
T										
G										
T										
A										
A										
T										
G										

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1									
<i>C</i>										
<i>T</i>										
<i>G</i>										
<i>T</i>										
<i>A</i>										
<i>A</i>										
<i>T</i>										
<i>G</i>										

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1									
<i>C</i>	-2									
<i>T</i>										
<i>G</i>										
<i>T</i>										
<i>A</i>										
<i>A</i>										
<i>T</i>										
<i>G</i>										

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1									
<i>C</i>	-2									
<i>T</i>	-3									
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		$A$	$C$	$G$	$G$	$C$	$T$	$A$	$T$	$C$
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
$A$	-1	2								
$C$	-2									
$T$	-3									
$G$	-4									
$T$	-5									
$A$	-6									
$A$	-7									
$T$	-8									
$G$	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1							
<i>C</i>	-2									
<i>T</i>	-3									
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0						
<i>C</i>	-2									
<i>T</i>	-3									
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		$A$	$C$	$G$	$G$	$C$	$T$	$A$	$T$	$C$
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
$A$	-1	2	1	0	-1					
$C$	-2									
$T$	-3									
$G$	-4									
$T$	-5									
$A$	-6									
$A$	-7									
$T$	-8									
$G$	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2									
<i>T</i>	-3									
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1								
<i>T</i>	-3									
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4							
<i>T</i>	-3									
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3						
<i>T</i>	-3									
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3									
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0								
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3							
<i>G</i>	-4									
<i>T</i>	-5									
<i>A</i>	-6									
<i>A</i>	-7									
<i>T</i>	-8									
<i>G</i>	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		$A$	$C$	$G$	$G$	$C$	$T$	$A$	$T$	$C$
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
$A$	-1	2	1	0	-1	-2	-3	-4	-5	-6
$C$	-2	1	4	3	2	1	0	-1	-2	-3
$T$	-3	0	3	3	2	1	3	2	1	0
$G$	-4									
$T$	-5									
$A$	-6									
$A$	-7									
$T$	-8									
$G$	-9									

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		$A$	$C$	$G$	$G$	$C$	$T$	$A$	$T$	$C$	
		0	-1	-2	-3	-4	-5	-6	-7	-8	-9
$A$	-1	2	1	0	-1	-2	-3	-4	-5	-6	
$C$	-2	1	4	3	2	1	0	-1	-2	-3	
$T$	-3	0	3	3	2	1	3	2	1	0	
$G$	-4	-1	2	5	5	4	3	2	1	0	
$T$	-5	-2	1	4	4	4	6	5	4	3	
$A$	-6	-3	0	3	3	3	5	8	7	6	
$A$	-7	-4	-1	2	2	2	4	7	7	6	
$T$	-8	-5	-2	1	1	1	4	6	9	8	
$G$	-9	-6	-3	0	3	2	3	5	8	8	

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

# Un exemple pas à pas

---

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	3	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	5	4	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	6	5	4	3
<i>A</i>	-6	-3	0	3	3	3	5	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	7	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	9	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	8

Coûts :  $s(a, b) = -1$  avec  $a \neq b$  et 2 sinon ;  $g = -1$

Case (9,9) : score de l'alignement global entre *ACGGCTATC* et *ACTGTAATG*.

```
CALCUL-GÉNÉRIQUE( $x, m, y, n$ )
```

```
1    $T[-1, -1] \leftarrow 0$ 
2   pour  $i \leftarrow 0$  à  $m - 1$  faire
3        $T[i, -1] \leftarrow T[i - 1, -1] + Dél(x[i])$ 
4   pour  $j \leftarrow 0$  à  $n - 1$  faire
5        $T[-1, j] \leftarrow T[-1, j - 1] + Ins(y[j])$ 
6   pour  $i \leftarrow 0$  à  $m - 1$  faire
7        $T[i, j] \leftarrow \min \begin{cases} T[i - 1, j - 1] + Sub(x[i], y[j]) \\ T[i - 1, j] + Dél(x[i]) \\ T[i, j - 1] + Ins(y[j]) \end{cases}$ 
8   retourner  $T[m - 1, n - 1]$ 
```

Complexité en temps :  $O(m \times n)$  et en espace :  $O(\min(m, n))$ .  
(extrait du livre Algorithmique du texte de Crochemore/Hancart/Lecroq)

- Procédure qui permet de trouver l'alignement en fonction de la matrice
- Fonctionnement :
  1. À partir de la cellule d'arrivée, remonter vers la(les) cellule(s) voisine(s) de score maximal et telle que son score plus la mutation correspondante donne le score de la cellule courante
  2. Itérer jusqu'à arriver à la cellule initiale.
- Si en une cellule, on peut revenir vers plusieurs cellules voisines, alors il existe plusieurs chemins optimaux.

		A	C	G	G	C	T	A	T	C
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
A	-1	2	1	0	-1	-2	-3	-4	-5	-6
C	-2	1	4	3	2	1	0	-1	-2	-3
T	-3	0	3	3	2	1	3	2	1	0
G	-4	-1	2	5	5	4	3	2	1	0
T	-5	-2	1	4	4	4	6	5	4	3
A	-6	-3	0	3	3	3	5	8	7	6
A	-7	-4	-1	2	2	2	4	7	7	6
T	-8	-5	-2	1	1	1	4	6	9	8
G	-9	-6	-3	0	3	2	3	5	8	8

		$A$	$C$	$G$	$G$	$C$	$T$	$A$	$T$	$C$
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
$A$	-1	2	1	0	-1	-2	-3	-4	-5	-6
$C$	-2	1	4	3	2	1	0	-1	-2	-3
$T$	-3	0	3	3	2	1	3	2	1	0
$G$	-4	-1	2	5	5	4	3	2	1	0
$T$	-5	-2	1	4	4	4	6	5	4	3
$A$	-6	-3	0	3	3	3	5	8	7	6
$A$	-7	-4	-1	2	2	2	4	7	7	6
$T$	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
$G$	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

 $G$  $C$

		$A$	$C$	$G$	$G$	$C$	$T$	$A$	$T$	$C$
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
$A$	-1	2	1	0	-1	-2	-3	-4	-5	-6
$C$	-2	1	4	3	2	1	0	-1	-2	-3
$T$	-3	0	3	3	2	1	3	2	1	0
$G$	-4	-1	2	5	5	4	3	2	1	0
$T$	-5	-2	1	4	4	4	6	5	4	3
$A$	-6	-3	0	3	3	3	5	8	7	6
$A$	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
$T$	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
$G$	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

$$\begin{matrix} T & G \\ | \\ T & C \end{matrix}$$

# Backtracking : exemple

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	3	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	5	4	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	6	5	4	3
<i>A</i>	-6	-3	0	3	3	3	<b>5</b>	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

<i>A</i>	<i>T</i>	<i>G</i>
<i>A</i>	<i>T</i>	<i>C</i>

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	3	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	5	4	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	<b>6</b>	5	4	3
<i>A</i>	-6	-3	0	3	3	3	<b>5</b>	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

<i>A</i>	<i>A</i>	<i>T</i>	<i>G</i>	
-	-	<i>A</i>	<i>T</i>	<i>C</i>

# Backtracking : exemple

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	3	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	5	<b>4</b>	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	<b>6</b>	5	4	3
<i>A</i>	-6	-3	0	3	3	3	<b>5</b>	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

<i>T</i>	<i>A</i>	<i>A</i>	<i>T</i>	<i>G</i>
<i>T</i>	-	<i>A</i>	<i>T</i>	<i>C</i>

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	3	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	<b>5</b>	<b>4</b>	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	<b>6</b>	5	4	3
<i>A</i>	-6	-3	0	3	3	3	<b>5</b>	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

-      *T*    *A*    *A*    *T*    *G*  
 |            |            |  
*C*    *T*    -    *A*    *T*    *C*

# Backtracking : exemple

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	4	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	<b>3</b>	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	<b>5</b>	<b>4</b>	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	<b>6</b>	5	4	3
<i>A</i>	-6	-3	0	3	3	3	<b>5</b>	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

$G \quad - \quad T \quad A \quad A \quad T \quad G$   
 |           |           |     |  
 $G \quad C \quad T \quad - \quad A \quad T \quad C$

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	2	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	<b>4</b>	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	<b>3</b>	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	<b>5</b>	<b>4</b>	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	<b>6</b>	5	4	3
<i>A</i>	-6	-3	0	3	3	3	<b>5</b>	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

<i>T</i>	<i>G</i>	-	<i>T</i>	<i>A</i>	<i>A</i>	<i>T</i>	<i>G</i>
<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	-	<i>A</i>	<i>T</i>	<i>C</i>

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	<b>2</b>	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	<b>4</b>	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	<b>3</b>	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	<b>5</b>	<b>4</b>	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	<b>6</b>	5	4	3
<i>A</i>	-6	-3	0	3	3	3	<b>5</b>	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

<i>C</i>	<i>T</i>	<i>G</i>	-	<i>T</i>	<i>A</i>	<i>A</i>	<i>T</i>	<i>G</i>
<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	-	<i>A</i>	<i>T</i>	<i>C</i>

		<i>A</i>	<i>C</i>	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>A</i>	<i>T</i>	<i>C</i>
	<b>0</b>	-1	-2	-3	-4	-5	-6	-7	-8	-9
<i>A</i>	-1	<b>2</b>	1	0	-1	-2	-3	-4	-5	-6
<i>C</i>	-2	1	<b>4</b>	3	2	1	0	-1	-2	-3
<i>T</i>	-3	0	3	<b>3</b>	2	1	3	2	1	0
<i>G</i>	-4	-1	2	5	<b>5</b>	<b>4</b>	3	2	1	0
<i>T</i>	-5	-2	1	4	4	4	<b>6</b>	5	4	3
<i>A</i>	-6	-3	0	3	3	3	<b>5</b>	8	7	6
<i>A</i>	-7	-4	-1	2	2	2	4	<b>7</b>	7	6
<i>T</i>	-8	-5	-2	1	1	1	4	6	<b>9</b>	8
<i>G</i>	-9	-6	-3	0	3	2	3	5	8	<b>8</b>

$$\begin{array}{ccccccccc}
 A & C & T & G & - & T & A & A & T & G \\
 | & | & & | & & | & & | & | \\
 A & C & G & G & C & T & - & A & T & C
 \end{array}$$

# Algorithme de calcul d'un alignement

```

UN-ALIGNEMENT( $x, m, y, n$ )
1    $z \leftarrow (\varepsilon, \varepsilon)$ 
2    $(i, j) \leftarrow (m - 1, n - 1)$ 
3   tantque  $i \neq -1$  et  $j \neq -1$  faire
4       si  $T[i, j] = T[i - 1, j - 1] + Sub(x[i], y[j])$  alors
5            $z \leftarrow (x[i], y[j]) \cdot z$ 
6            $(i, j) \leftarrow (i - 1, j - 1)$ 
7       sinon si  $T[i, j] = T[i - 1, j] + Dél(x[i])$  alors
8            $z \leftarrow (x[i], \varepsilon) \cdot z$ 
9            $i \leftarrow i - 1$ 
10      sinon  $z \leftarrow (\varepsilon, y[j]) \cdot z$ 
11       $j \leftarrow j - 1$ 
12      tantque  $i \neq -1$  faire
13           $z \leftarrow (x[i], \varepsilon) \cdot z$ 
14           $i \leftarrow i - 1$ 
15      tantque  $j \neq -1$  faire
16           $z \leftarrow (\varepsilon, y[j]) \cdot z$ 
17           $j \leftarrow j - 1$ 
18  retourner  $z$ 

```

Complexité en temps et en espace supplémentaire :  $O(m + n)$ .  
 (extrait du livre Algorithmique du texte de Crochemore/Hancart/Lecroq)

```
LES-ALIGNEMENTS( $x, m, y, n$ )
```

```
    1  LA( $m - 1, n - 1, (\varepsilon, \varepsilon)$ )
```

```
LA( $i, j, z$ )
```

```
    1  si  $i \neq -1$  et  $j \neq -1$   
        et  $T[i, j] = T[i - 1, j - 1] + Sub(x[i], y[j])$  alors
```

```
    2      LA( $i - 1, j - 1, (x[i], y[j]) \cdot z$ )
```

```
    3  si  $i \neq -1$   
        et  $T[i, j] = T[i - 1, j] + Dél(x[i])$  alors
```

```
    4      LA( $i - 1, j, (x[i], \varepsilon) \cdot z$ )
```

```
    5  si  $j \neq -1$   
        et  $T[i, j] = T[i, j - 1] + Ins(y[j])$  alors
```

```
    6      LA( $i, j - 1, (\varepsilon, y[j]) \cdot z$ )
```

```
    7  si  $i = -1$  et  $j = -1$  alors
```

```
    8      signaler que  $z$  est un alignement
```

Complexité en espace : idem et en temps :

$O(\text{somme des longueurs des alignements produits})$ .

(extrait du livre Algorithmique du texte de Crochemore/Hancart/Lecroq)

	<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>G</i>	<i>A</i>	<i>C</i>
<i>G</i>							
<i>A</i>							
<i>T</i>							
<i>C</i>							

Initialisation

$$M(0, 0) = 0$$

$$M(0, j) =$$

$$g \times j$$

$$M(i, 0) =$$

$$g \times i$$

Remplissage

$$M(i, j) =$$

$$\min \begin{cases} M(i - 1, j - 1) + s(x_i, y_j) & \text{match/mismatch} \\ M(i - 1, j) + g & \text{délétion} \\ M(i, j - 1) + g & \text{insertion} \end{cases}$$

ici  $g = 3$  et  $s(a, b) = 3$  si  $a \neq b$  et 0 si  $a = b$

- Nombre d'alignements possibles : faramineux
- Mémoire et temps utilisés  $\mathcal{O}(n \times m)$  càd proportionnels au produit de la longueur des séquences
- Amélioration : algorithme linéaire en mémoire, algorithme de  $k$ -band

- Introduction
- Dotplot
- Alignement : définitions et scores
- Algorithme d'alignement (programmation dynamique)
- Autres types d'alignements
- La suite ...

- Principe : alignement des meilleures sous-séquences [Smith-Waterman 81]



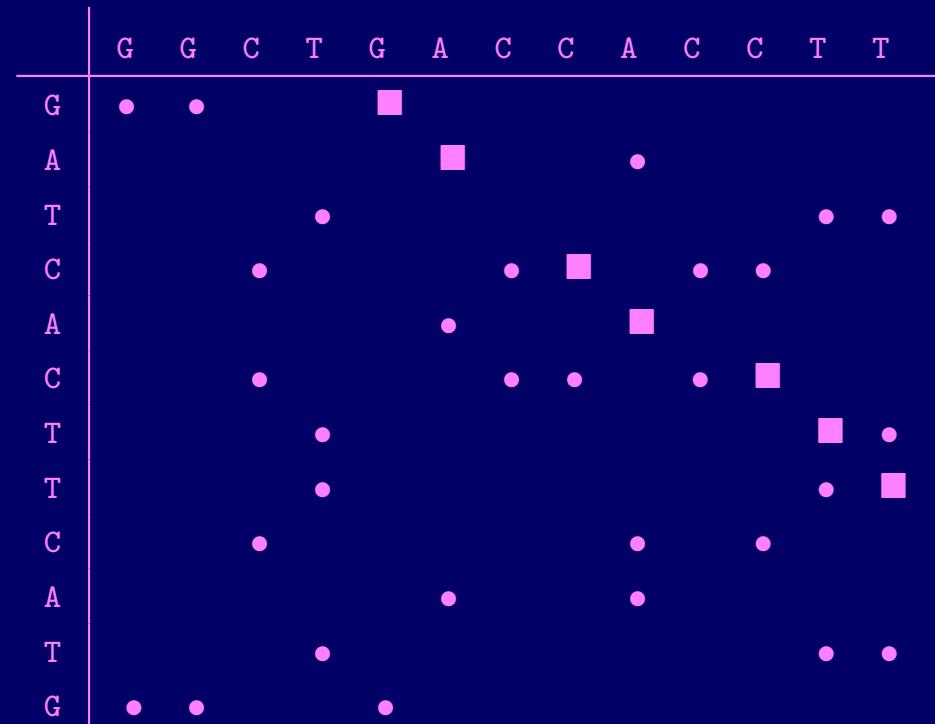
- Évaluation d'une ressemblance locale entre deux séquences ;
- Recherche de la région de plus forte similarité.

## Alignement global

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

## Alignement local

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



Les séquences présentent une similarité que l'alignement global ne révèle pas.

- $s$  : une matrice de score ;
- $g$  : pénalité associée à un indel ;

- Initialisation : 
$$\begin{cases} M(0, 0) = 0 \\ M(0, j) = 0 \quad \leftarrow \\ M(i, 0) = 0 \quad \leftarrow \end{cases}$$

- Remplissage :

$$M(i, j) = \max \begin{cases} M(i - 1, j - 1) + s(x_i, y_j) & \text{Sub. ou App. exact} \\ M(i - 1, j) + g & \text{Délétion} \\ M(i, j - 1) + g & \text{Insertion} \\ 0 & \leftarrow \end{cases}$$

		<i>G</i>	<i>G</i>	<i>C</i>	<i>T</i>	<i>G</i>	<i>A</i>	<i>C</i>
<i>G</i>								
<i>A</i>								
<i>T</i>								
<i>C</i>								

## Remplissage

## Initialisation

$$M(0, 0) = 0$$

$$M(0, j) = 0$$

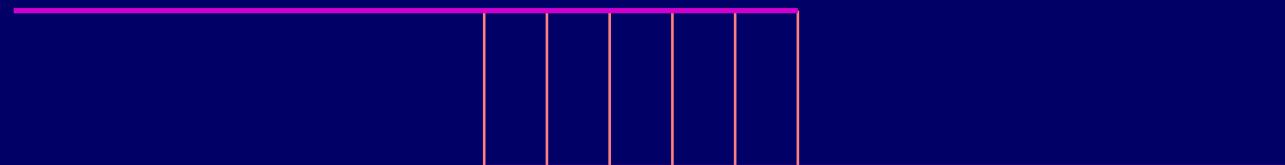
$$M(i, 0) = 0$$

$$M(i, j) = \max \begin{cases} M(i - 1, j - 1) + s(x_i, y_j) & \text{Sub./App.} \\ M(i - 1, j) + g & \text{Délétion} \\ M(i, j - 1) + g & \text{Insertion} \\ 0 & \leftarrow \end{cases}$$

ici  $g = -1$  et  $s(a, b) = -1$  si  $a \neq b$  et 2 si  $a = b$

- Ne pénalise pas les gaps aux extrémités
- Sinon, similaire à l'alignment global
- Permet de détecter des similarités de type **inclusion** et **chevauchement**

Exemple chevauchement



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

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

⇒ L'alignement global préfère le 1<sup>er</sup> alignement.

- Initialisation  $\Rightarrow$  idem à l'alignement local : 
$$\begin{cases} M(0, 0) = 0 \\ M(0, j) = 0 & \leftarrow \\ M(i, 0) = 0 & \leftarrow \end{cases}$$
- Remplissage  $\Rightarrow$  idem à l'alignement global
- Meilleur alignement  $\Rightarrow$  valeur max dans la dernière ligne et la dernière colonne.

## Pénalité associée aux gaps

- Biologie : insertion d'un segment dans un gène  
⇒ les insertions/délétions arrivent en groupe
- gap : suite de  $l$  insertions ou de  $l$  délétions  
pénalisation moindre que  $l \times g$
- Fonctions de pénalité les plus réalistes :
  - fonctions affines :  $o + e \times l$ 
    - $o$  : pénalité d'ouverture de gap
    - $e$  : pénalité d'extension de gap
  - fonctions logarithmiques
- fonctions de gaps différentes ⇒ algorithmes différents

Fonction :  $c(g) = -o - (g - 1) \times e$

Match ou mismatch :

$$M(i, j) = \max \begin{cases} M(i - 1, j - 1) + s(x_i, y_j) \\ D(i - 1, j - 1) + s(x_i, y_j) \\ I(i - 1, j - 1) + s(x_i, y_j) \end{cases}$$

Délétion :

$$D(i, j) = \max \begin{cases} M(i - 1, j) - o \\ D(i - 1, j) - e \end{cases}$$

Insertion :

$$I(i, j) = \max \begin{cases} M(i, j - 1) - o \\ I(i, j - 1) - e \end{cases}$$

- *Wraparound Dynamic Programming* (WDP) : alignement en plusieurs passes
- approche par automate reconnaissant les alignements optimaux.

- Introduction
- Dotplot
- Alignement : définitions et scores
- Algorithme d'alignement (programmation dynamique)
- Autres types d'alignements
- La suite ...

Comment peut-on savoir d'après le score d'alignement si les séquences se ressemblent ?

- Tout alignement produit un score

→ *Quel serait le score de l'alignement de 2 séquences aléatoires ?*

- Approche empirique : Tests de permutation (TP logiciel PRSS)
- Approche statistique : E-value = Nombre attendu de fois de trouver un alignement de score supérieur à  $S$  par hasard quand on aligne une séquence de longueur  $m$  avec une séquence de longueur  $n$

- Jusque ici les algorithmes considérés donnent des solutions exactes mais ils ne sont pas très rapides (complexité  $O(n \times m)$ )

$\sim 10^{25}$  résidus dans les BD

Comparer 1000pb à une BD  $\Rightarrow \sim 10^{18}$  cellules à évaluer

Calcul 10 milliards de cellules par sec.  $\Rightarrow \underline{x \text{ 100 heures}}$

$\Rightarrow$  Nécessité d'algorithmes plus rapides

- Algorithmes heuristiques, les plus connus : BLAST et FASTA