www.facebook.com/DomaineSNV/

Domaine SNV : Biologie,Agronomie,Science Alimentaire,Ecologie

Alignements par paires

Jacques van Helden

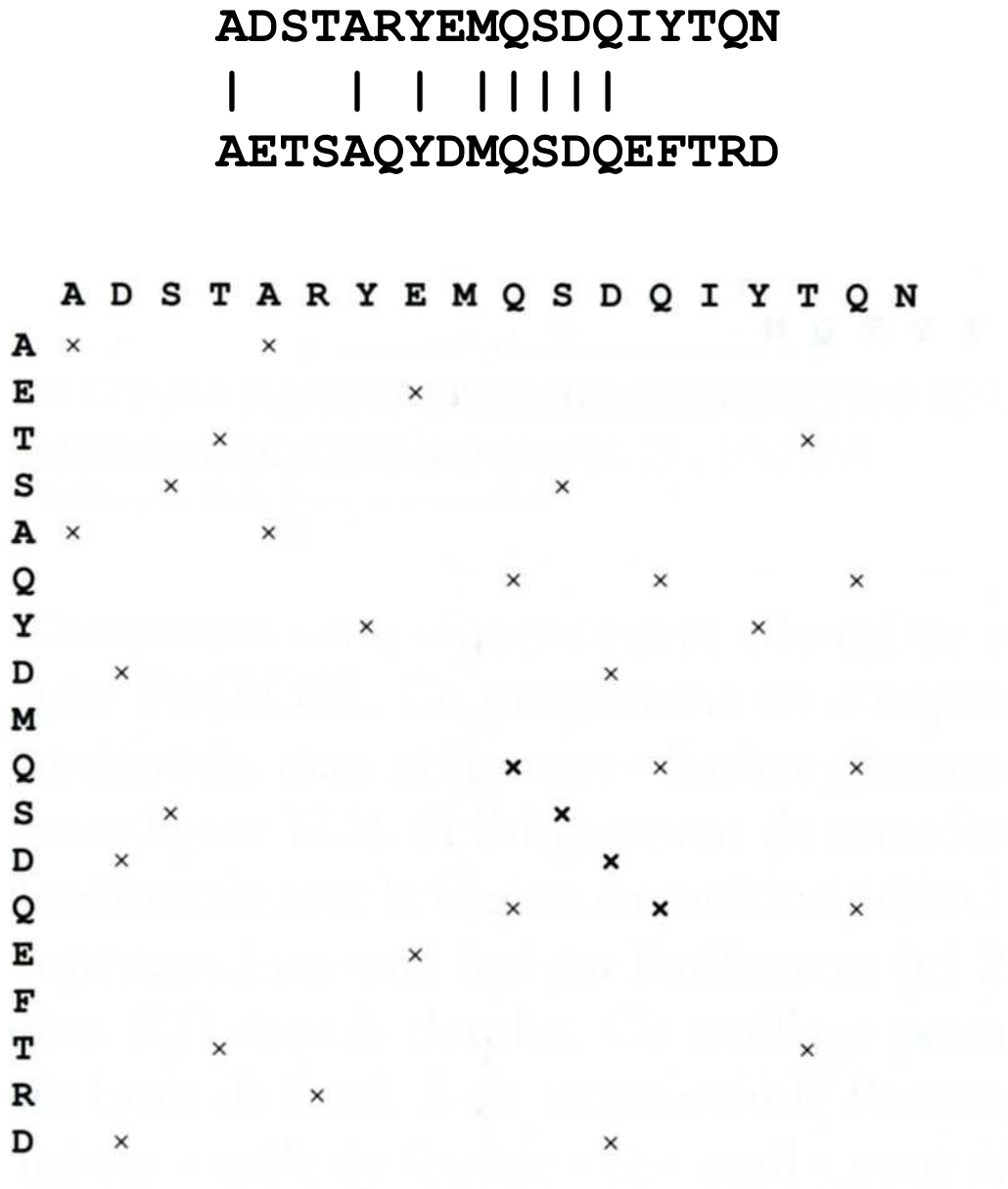
Jacques.van-Helden@univ-amu.fr
Université d'Aix-Marseille, France
Lab. Technological Advances for Genomics and Clinics
(TAGC, INSERM Unit U1090)
http://tagc.univ-mrs.fr/

FORMER ADDRESS (1999-2011)
Université Libre de Bruxelles, Belgique
Bioinformatique des Génomes et des Réseaux (BiGRe lab)
http://www.bigre.ulb.ac.be/

Dot plots : visualisation intuitive des segments “alignables” entre deux séquences

Matrice de points (Matrice de pixels; dot-plot)

- Le dot plot est une représentation graphique simple des résidus identiques entre les deux séquences.
  - Les deux séquences sont représentées sur les deux axes
  - Un point (dot) est tracé pour chaque correspondance entre deux résidus de séquences.
  - Les lignes diagonales révèlent les régions alignables entre les deux séquences.



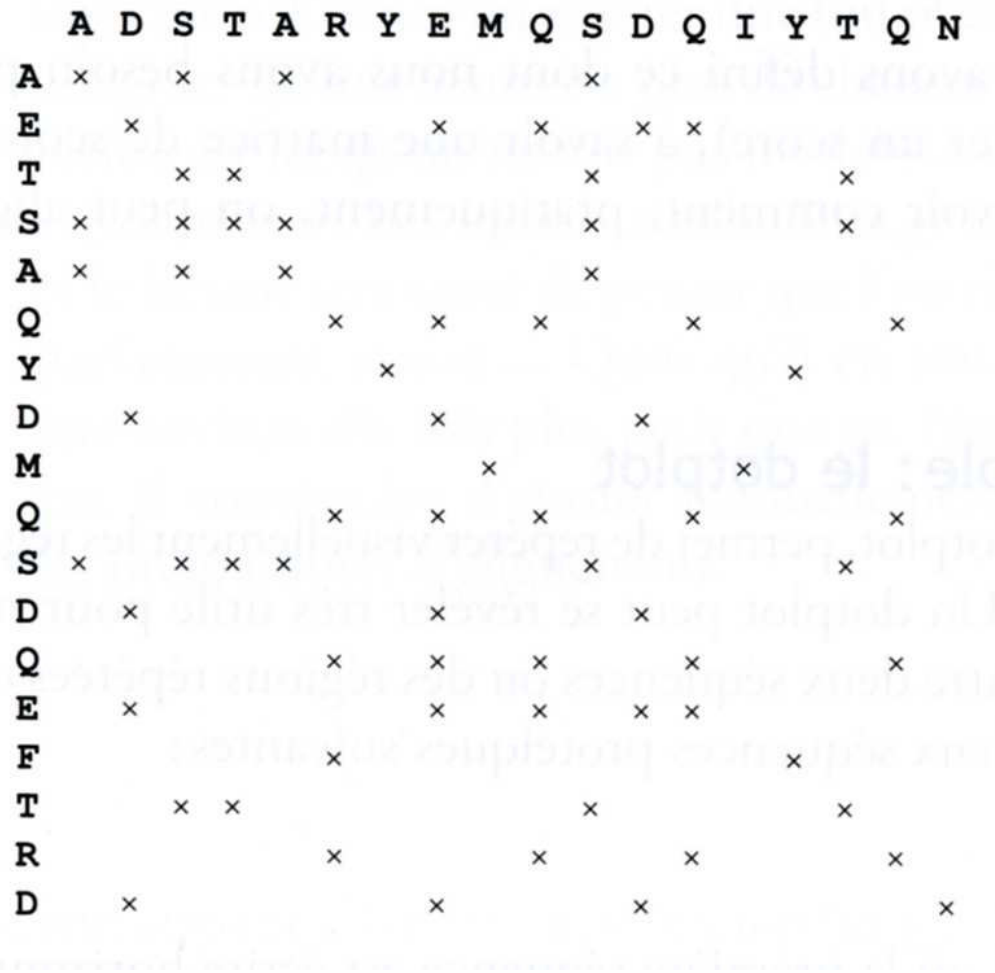
- Diapo: Emese Meglecz

Matrice de points (Matrice de pixels; dot-plot)

- Outre les identités, on peut marquer les similarités entre acides aminés (substitutions conservatives, d'après une matrice de substitution donnée)
  - Exemple: marquage des paires de résidus ayant un score BLOSUM62 > 1.

		Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
Ala	A	4																			
Arg	R	-1	5																		
Asn	N	-2	0	6																	
Asp	D	-2	-2	1	6																
Cys	C	0	-3	-3	-3	9															
Gln	Q	-1	1	0	0	-3	5														
Glu	E	-1	0	0	2	-4	2	5													
Gly	G	0	-2	0	-1	-3	-2	-2	6												
His	H	-2	0	1	-1	-3	0	0	-2	8											
Ile	I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

```
ADSTARYEMQSDQIYTQN
|:::|:|:|||||  :|::
AETSAQYDMQSDQEFTRD
```

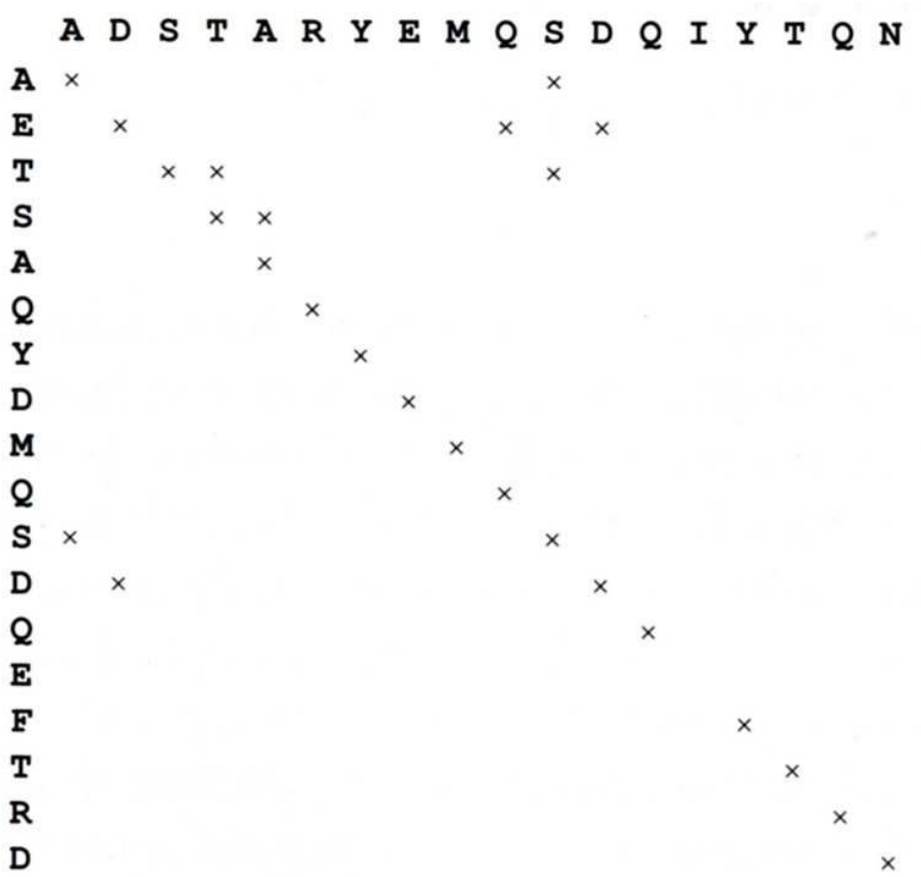


- Diapo: Emese Meglecz

Matrice de points (Matrice de pixels; dot-plot)

- On peut appliquer à la matrice de points un filtrage par fenêtre glissante.
  - On n'affiche que les diagonale comportant au moins 2 points marqués successifs.
- À chaque position de la matrice on extrait la paire de mots de taille w qui commence aux positions correspondantes des deux séquences.
- Le score de la paire de mots est calculé en additionnant les scores des paires des résidus (d'après la matrice de substitution).
- Si le score dépasse un seuil défini par l'utilisateur, une diagonale noire s'affiche à la position correspondante.

		Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
		A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
Ala	A	4																			
Arg	R	-1	5																		
Asn	N	-2	0	6																	
Asp	D	-2	-2	1	6																
Cys	C	0	-3	-3	-3	9															
Gln	Q	-1	1	0	0	-3	5														
Glu	E	-1	0	0	2	-4	2	5													
Gly	G	0	-2	0	-1	-3	-2	-2	6												
His	H	-2	0	1	-1	-3	0	0	-2	8											
Ile	I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

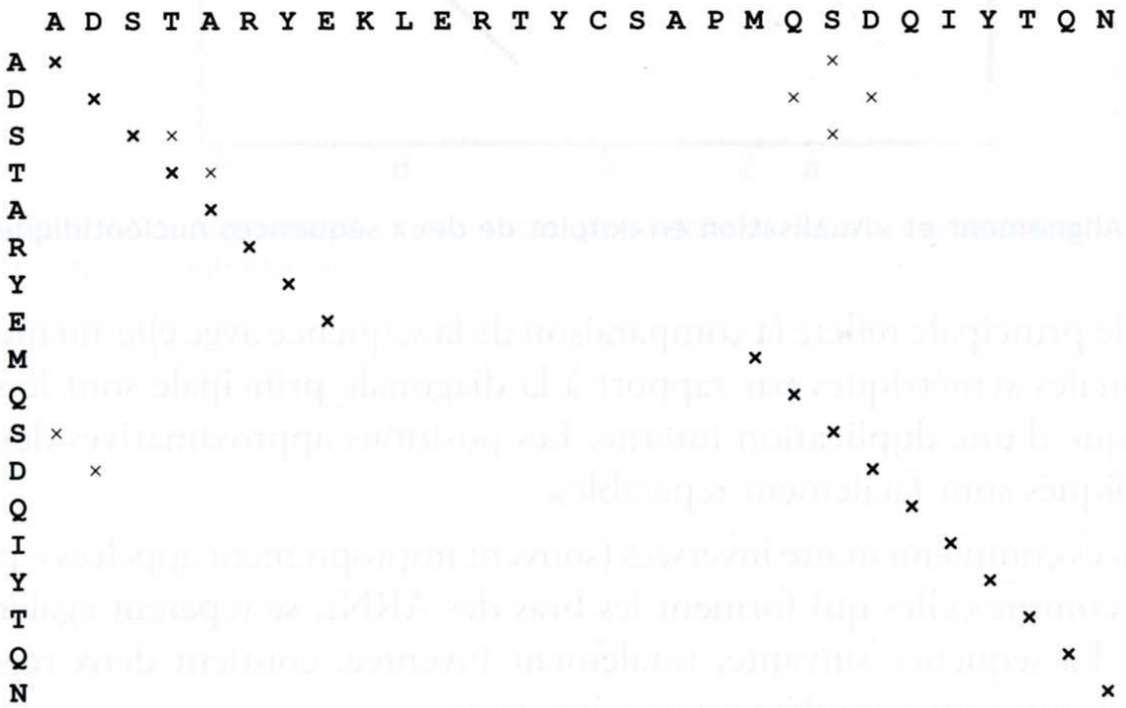


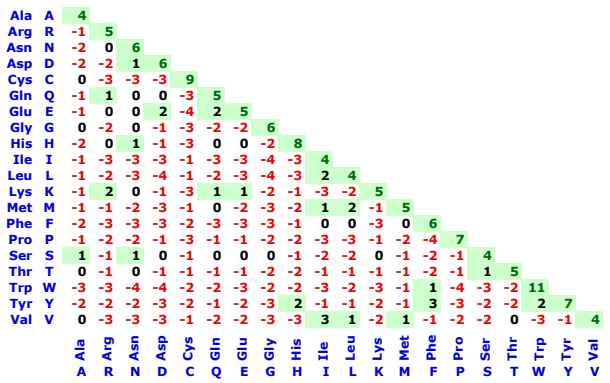
Matrice de points - Détections des INDELs

- Un décalage (gap) entre deux diagonales suggère soit une délétion (sur la première séquence dans l'exemple ci-contre), soit une insertion (dans la seconde séquence).
- Le simple alignement entre deux séquences ne permet pas de décider si l'événement évolutif était une délétion ou une insertion.
- On désigne par « indel » l'événement évolutif supposé.

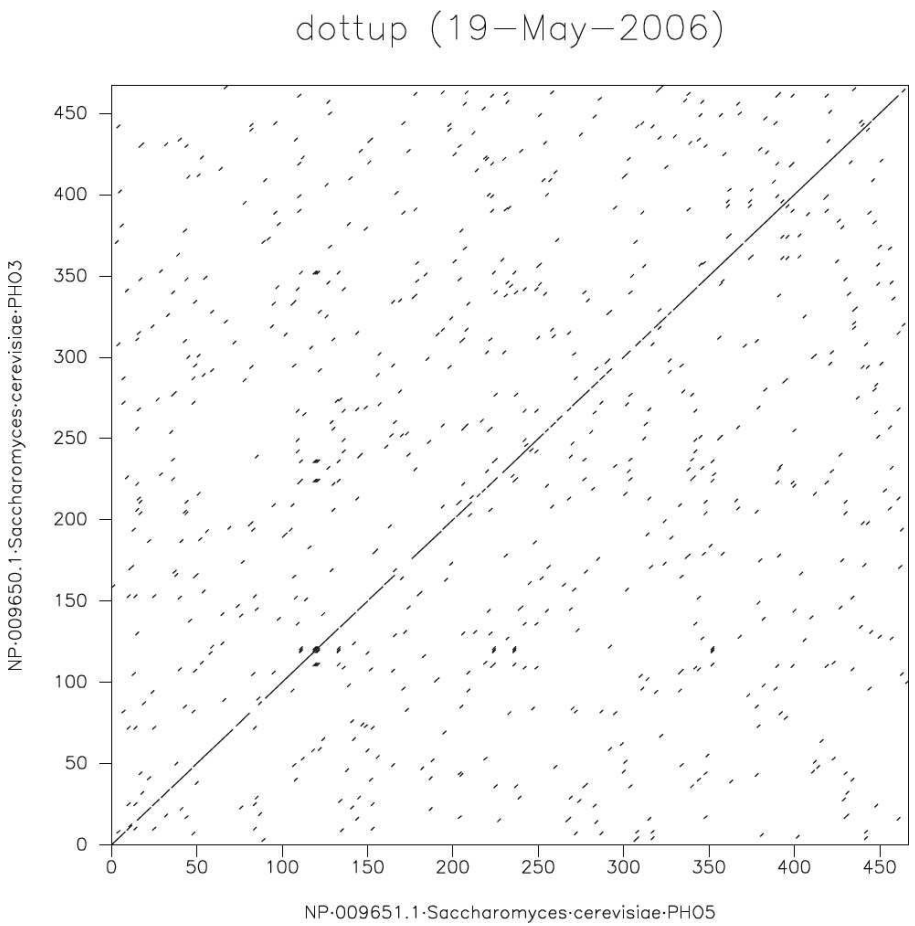
a) ADSTARYEKLERTYCSAPMQSDQIYTQN
b) ADSTARYE----------MQSDQIYTQN

donnera :





Dot plot

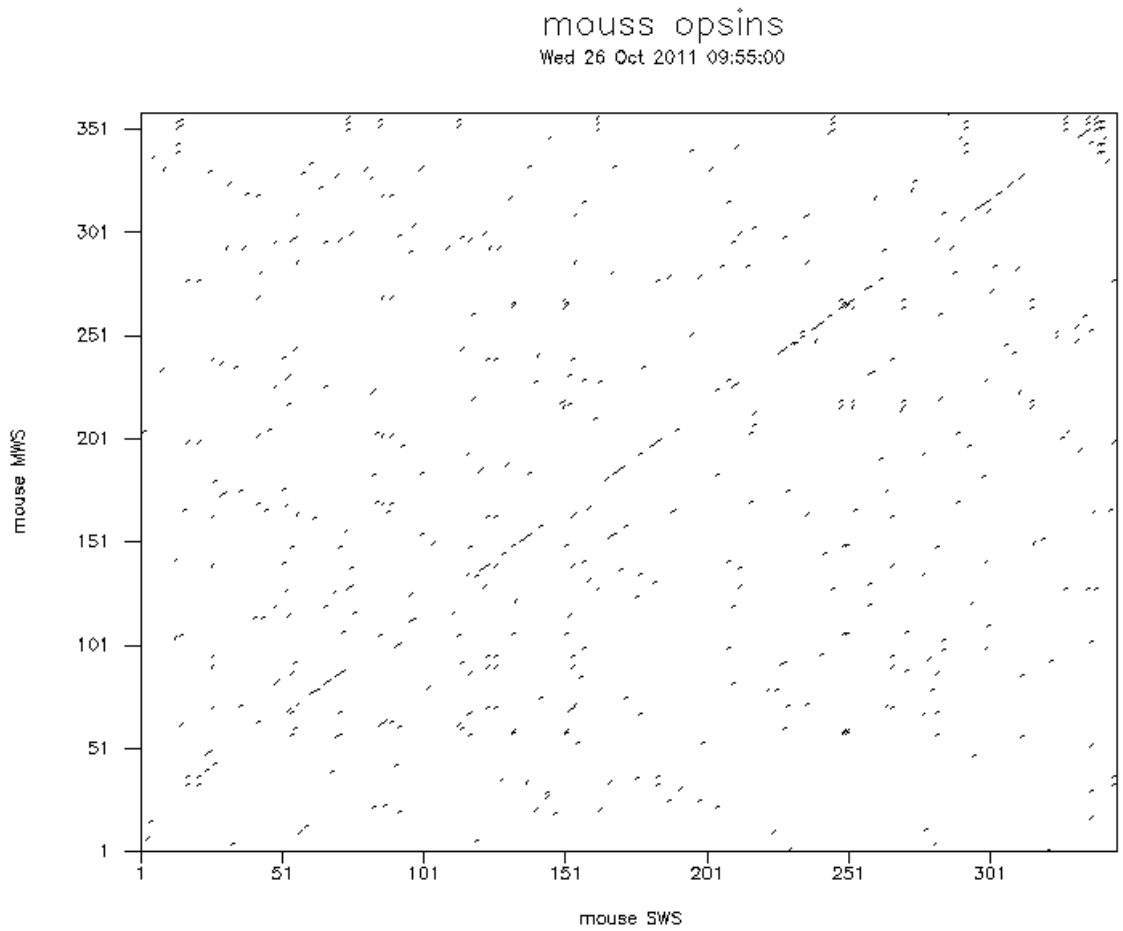


Example: protein sequences of the Pho5p and Pho3p phosphatases in the yeast *Saccharomyces cerevisiae*

- Exemple de dot plot: comparaison de deux gènes de levure codant pour des phosphatases.
  - L'axe X représente la première séquence (PHO5),
  - L'axe Y représente la seconde séquence (PHO3)
  - Un point est affiché pour chaque correspondance entre deux résidus des séquences.
  - Les lignes obliques représentent des régions d'identité entre deux séquences.
  - La diagonale est fortement marquée, car les deux séquences sont homologues et conservées sur toute leur longueur.

Séquences peptidiques – Opsines de souris

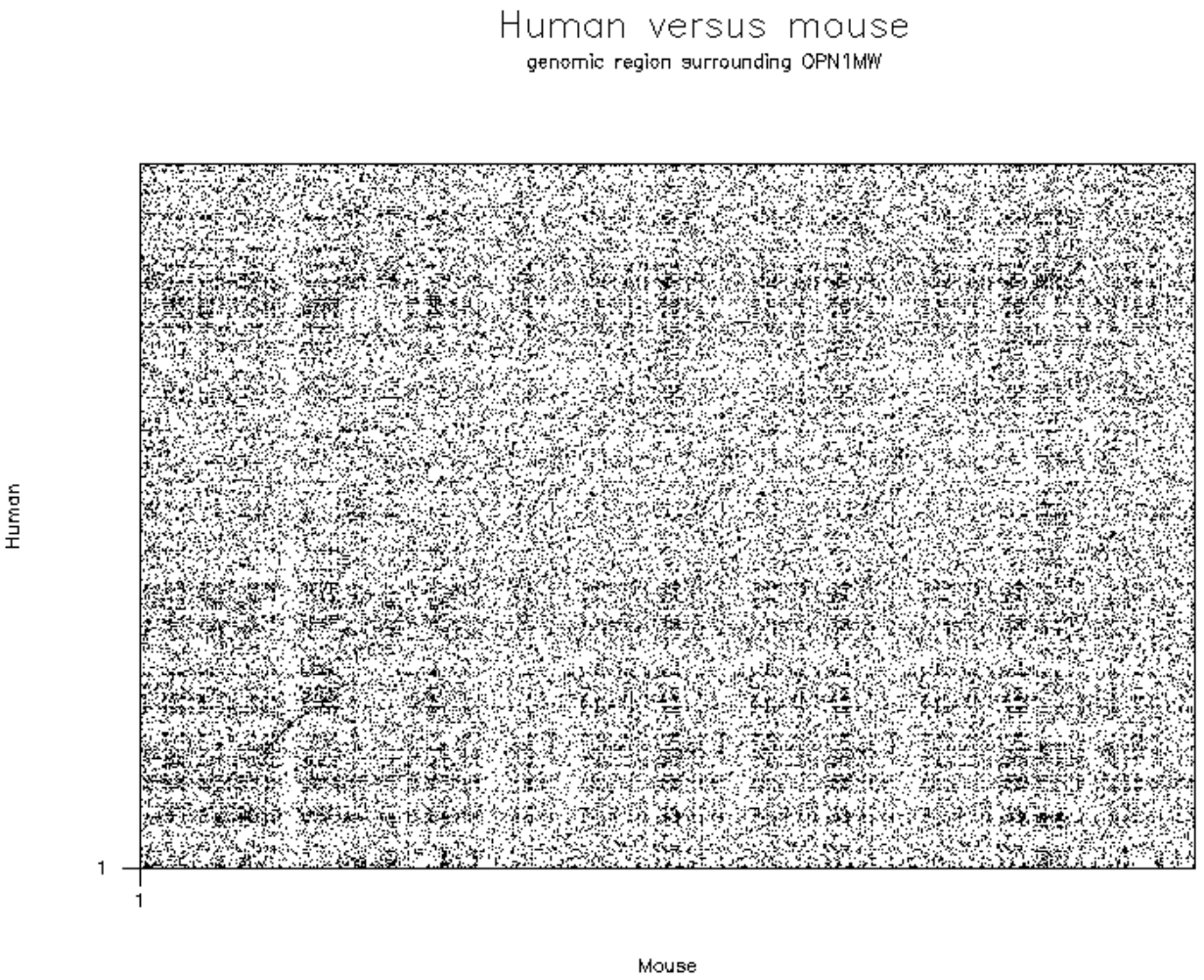
- Dot plot des séquences peptidiques de deux opsines de souris.
- Ces séquences sont homologues, mais ont divergé depuis quelques dizaines de millions d'années (avant la radiation évolutive des mammifères).
- On distingue des traits plus allongés dans la diagonale, qui indiquent une similarité des deux séquences.



- http://emboss.bioinformatics.nl/cgi-bin/emboss/dottup

Séquences peptidiques – Opsines de souris

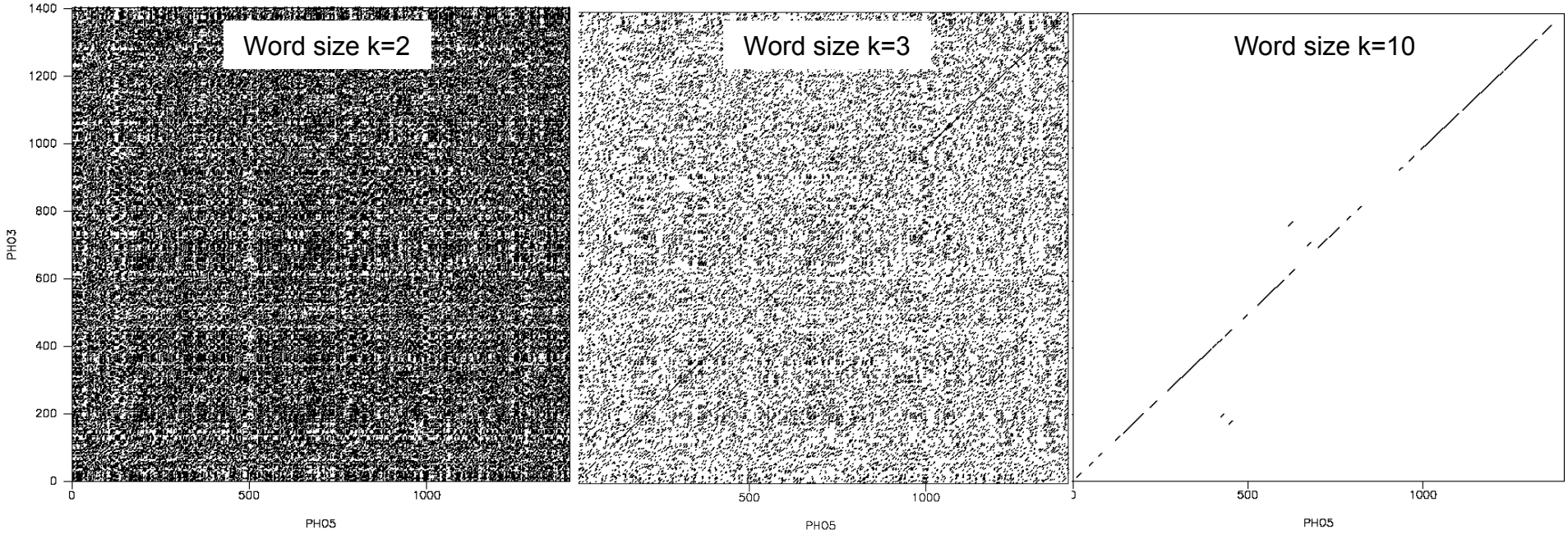
- Dot plot des séquences nucléiques de deux opsines de souris.
- Le plot « brut » est difficilement lisible, car on trouve des identités entre nucléotides un peu partout (1 chance sur 4).
- On distingue néanmoins un renforcement de la diagonale, même sur ce graphique extrêmement bruité.



- http://emboss.bioinformatics.nl/cgi-bin/emboss/dottup

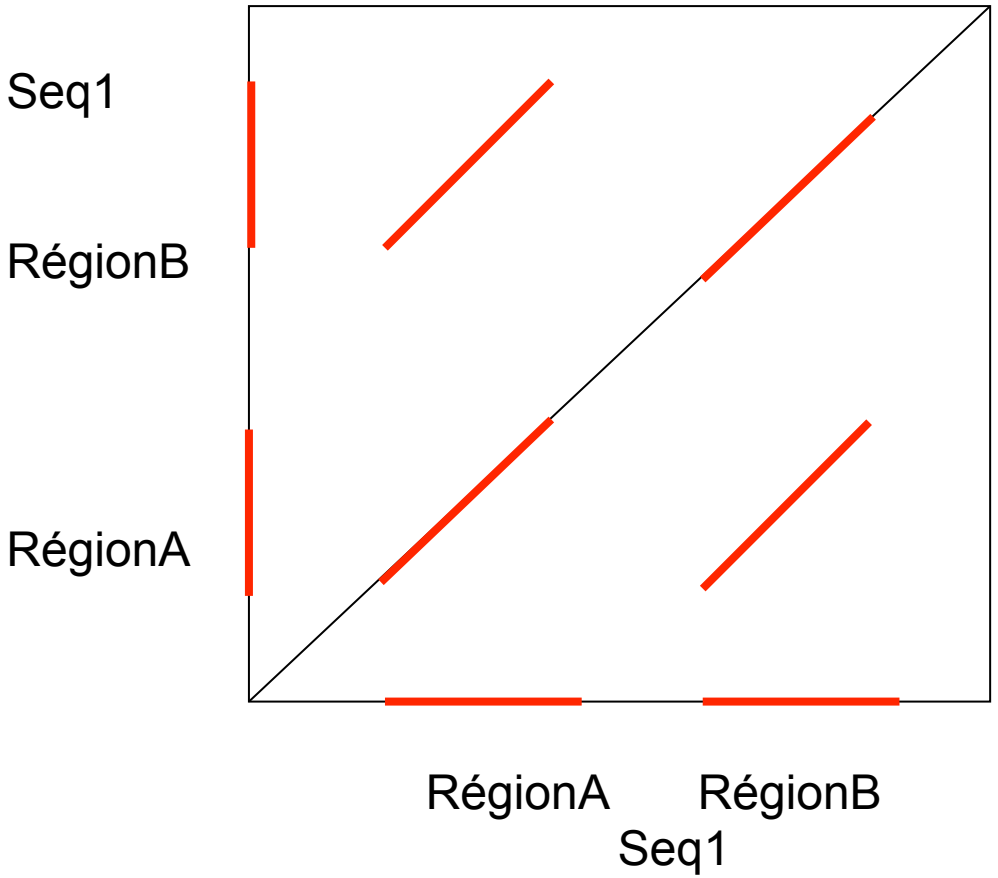
Dot plot de "mots"

- Dans les séquences nucléiques, les « lettres » (nucléotides) ont une fréquence moyenne de 0.25. On s'attend donc à trouver une identité pour 1 point sur 4, ce qui rend le graphique illisible.
- Pour distinguer les régions de similarité sur les dot plots, on peut restreindre l'affichage à des « mots » identiques (oligonucléotides) de taille *k* (ci-dessous: k=2, k=3, k=10).
- Ceci revient à remplacer par un point les lignes obliques de *k* éléments.
- Exemple: dot plot des gènes PHO5 and PHO3 de levure, avec différentes tailles de mots.



Matrice de points - Répétitions intra-séquence

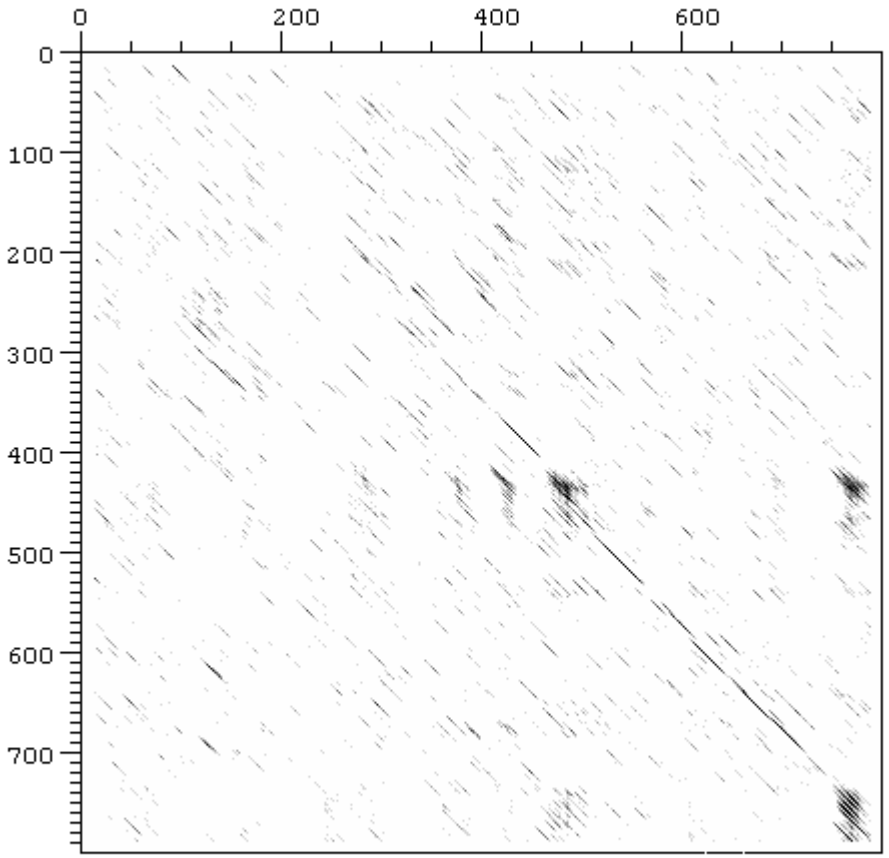
- La matrice de points (dotplot) permet notamment de repérer des *segments répétés* au sein d'une séquence.
- Pour cela, on aligne la séquence avec elle-même.
- Les segments répétées apparaissent comme des lignes obliques parallèles à la diagonale principale.



- Adaptée d'après Emese Meglecz

Echelles de gris (logiciel dottups)

- Les graphiques affichés précédemment requièrent une correspondance parfaite entre les mots.
- Une façon plus raffinée de représenter les similarités partiales est d'utiliser des fenêtres.
  - On fait glisser une « fenêtre » de taille *k* le long des deux séquences en comparant les mots correspondants.
  - Pour chaque point, le ***score de similarité par fenêtre*** est la somme de résidus correspondants.
  - Le niveau de gris est proportionnel au score de similarité de fenêtre.



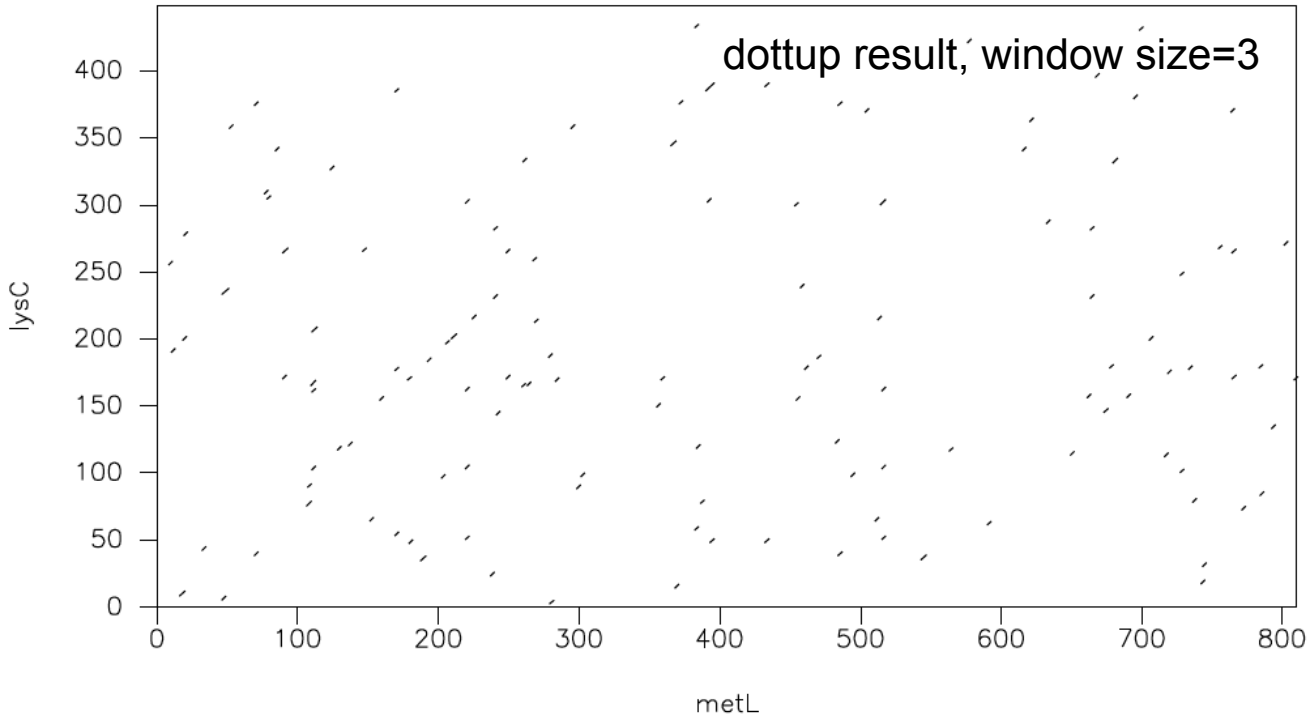
Source ?

Matrices de substitutions et dot plots

- Plutôt que de compter les identités, on peut utiliser une matrice de substitution pour calculer le score de simliarité par fenêtres d'un dot plot.
- L'utilisateur doit spécifier les paramètres suivants:
  - Taille de la fenêtre (= taille des mots, k).
  - Matrice de substitution
  - Seuil de score
- Pour chaque position du plot
  - La paire de mots de taille *k* est extraite des positions correspondantes sur la première et la seconde séquence.
  - Le score du mot est calculé en faisant la somme des scores des paires de résidus.
  - Si le score dépasse le seuil, une ligne oblique s'affiche à la position correspondante du dot plot.
- Les régions de similarité entre deux séquences apparaissent comme des lignes obliques allongées sur le dot plot.

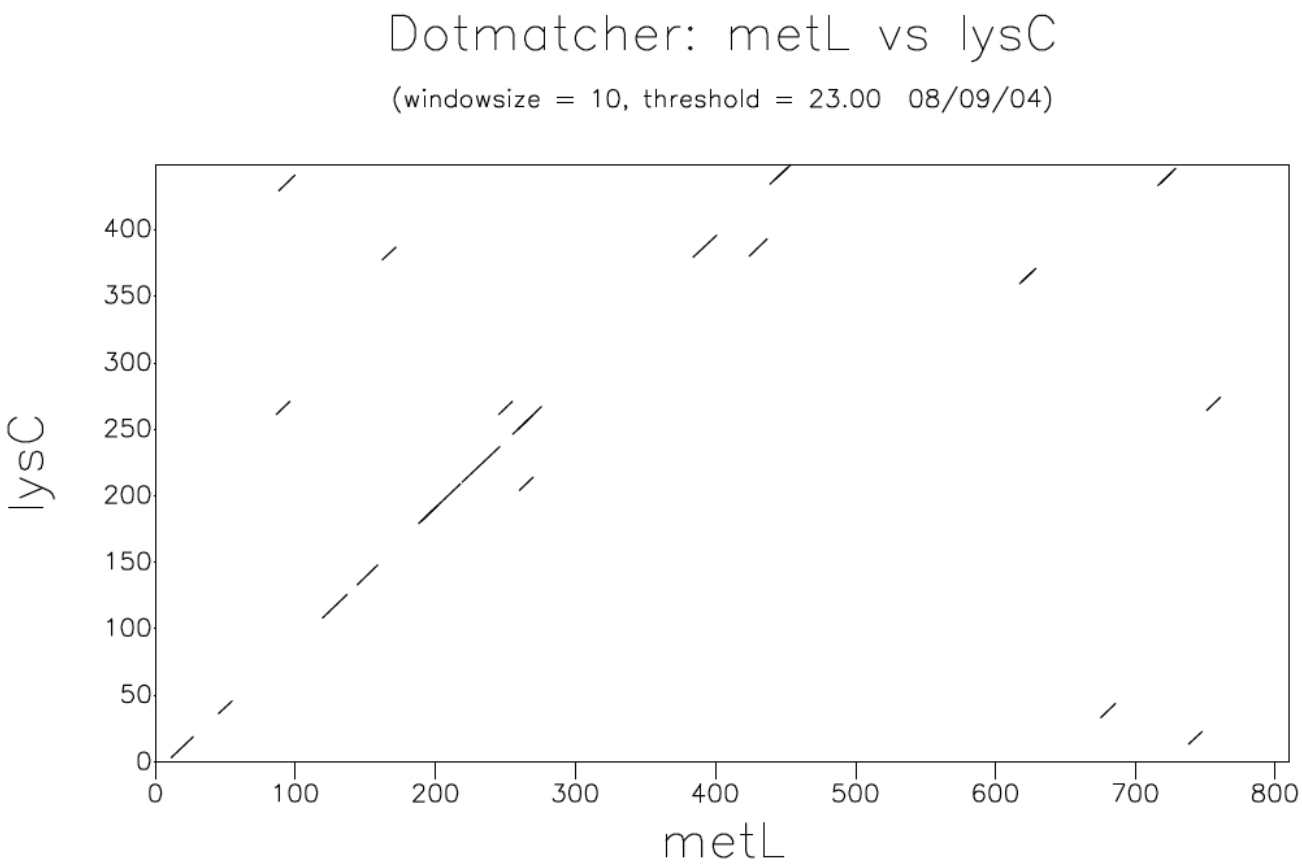
Aspartokinases: dot plot avec simple identité de mots

- Comparaison entre les séquences de deux enzymes de la bactérie *Escherichia coli K12*.
  - LysC aspartokinase impliquée dans la biosynthèse de la lysine
  - MetL enzyme bifonctionnelle qui combine deux domaines:
    - L'aspartokinase catalyse la première étape de la biosynthèse de la méthionine
    - L'homoserine déshydrogenase catalyse la troisième étape de cette même voie métabolique.
- Sur le dot plot, on distingue à peine la région de similarité entre les deux domaines.



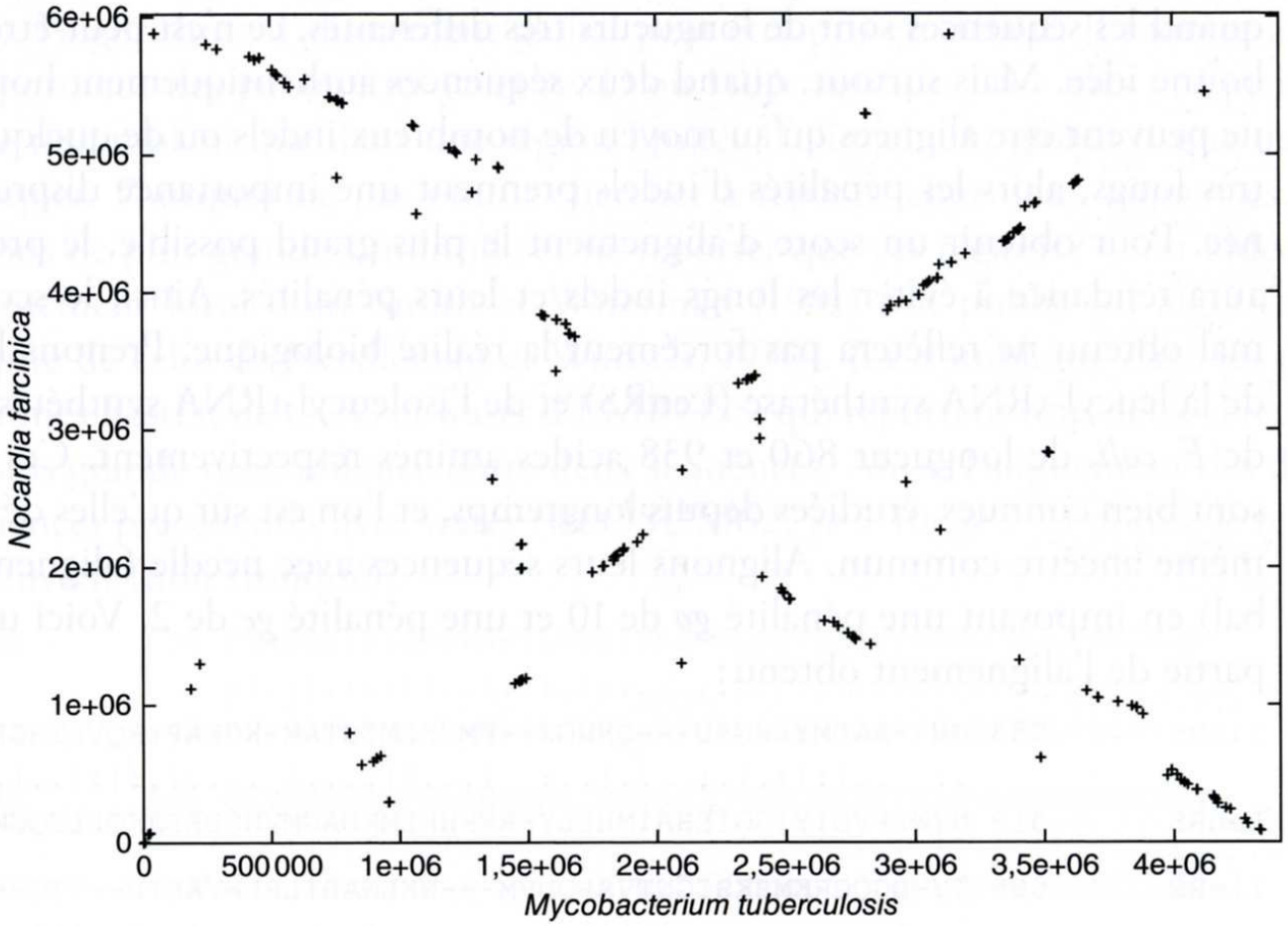
Aspartokinases: dot plot avec matrice de substitution (BLOSUM62)

- Avec le logiciel *dotmatcher*, une matrice de substitution est utilisée pour assigner un score à chaque paire de résidus.
- Ceci révèle la similarité entre les domaines aspartokinase de LysC (l'ensemble de la séquence) et de MetL (positions 1 à ~450).
- Notes
  - Cette région de similarité ne recouvre que la partie N-terminale de MetL, car il s'agit d'une enzyme bi-fonctionnelle. La région C-terminale contient un domaine homosérine déshydrogenase.
  - Le choix des paramètres est délicat, il varie selon les séquences alignées.



Matrice de points – Extension à l'échelle génomique

- On peut transposer le concept de matrice de points à l'échelle génomique, en indiquant par un point la présence de gènes homologues entre les deux génomes.
- Cette approche permet de repérer
  - des régions génomiques où les gènes se succèdent de façon similaires (syntons);
  - des inversions chromosomiques.



Matrice d'alignement

- Une ***matrice*** d'alignement est conceptuellement liée au dot plot.
- On affiche une séquence horizontalement, l'autre verticalement.
- Un score est assigné à chaque paire de résidus.
- Dans la matrice d'alignement, les « bons » alignements apparaissent comme des lignes obliques ininterrompues de scores élevés.
- Les séparations entre les segments obliques indiquent les gaps.
  - Séparation verticale: un segment de la séquence verticale est aligné avec un gap dans la séquence horizontale.
  - Séparation horizontale: gap dans la séquence verticale.

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

```
AATCTTCAGC-----GTATTGCT
-ATCTT-AGCCGGAGGTATT---
```

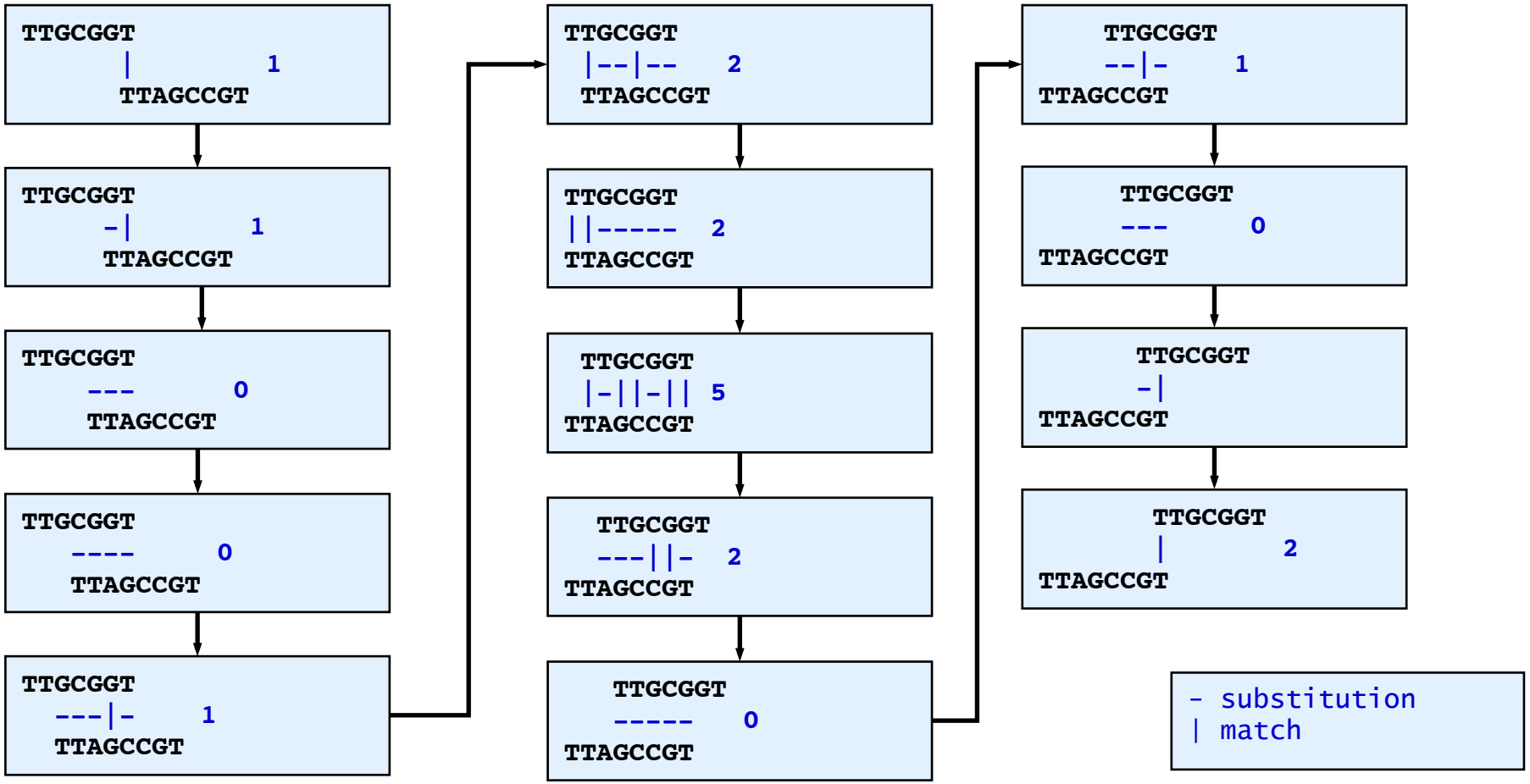
Logiciels de dot plot

- Dotlet
  - Junier T, Pagni M (2000) Dotlet: diagonal plots in a web browser. Bioinformatics. 16:178-9.
  - http://myhits.isb-sib.ch/cgi-bin/dotlet
- Dnadot
  - http://www.vivo.colostate.edu/molkit/dnadot/
  - Draw nucleic acid dot plots, convenient for DNA/RNA alignment.
- Dotter
  - Sonnhammer EL, Durbin R (1995) A dot-matrix program with dynamic threshold control suited for genomic DNA and protein sequence analysis. Gene. 167:GC1-10.
  - http://www.cgb.ki.se/cgb/groups/sonnhammer/Dotter.html

Identification des alignements optimaux entre deux séquences

Exemple d'alignement sans gaps

- La façon la plus simple d'aligner deux séquences ans gap est de faire glisser l'une sur l'autre, et de calculer le score d'alignement pour chaque décalage.



Exercice

- On dispose des deux séquences suivantes
  - **Seq1** **TTTGCGTTAAATCGTGTAGCAATTTAA**
  - **Seq2** **AAGAATGGCGTTTTTAATAGCAATAT**
- Questions
  1. En décalant progressivement les séquences, identifiez le(s) décalage(s) qui révèlent des régions de similarité.
  2. A chaque position de décalage, identifiez les segments parfaitement conservés (successions ininterrompue de résidus identiques).
  3. Au vu du résultat, pensez-vous que l'insertion d'un gap permettrait d'augmenter le score d'alignement?

Solution de l'exercice

- On dispose des deux séquences suivantes
  - **Seq1** **TTTGCGTTAAATCGTGTAGCAATTTAA**
  - **Seq2** **AAGAATGGCGTTTTTAATAGCAATAT**
- Questions
  1. En décalant progressivement les séquences, identifiez le(s) décalage(s) qui révèlent des régions de similarité.
  2. A chaque position de décalage, identifiez les segments parfaitement conservés (successions ininterrompue de résidus identiques).
  3. Au vu du résultat, pensez-vous que l'insertion d'un gap permettrait d'augmenter le score d'alignement?
- Décalage -4
  - **Position** -4 123456789
  - **Seq1** 1234**TTTGCGTTAAATCGTGTAGCAATTTAA**
  - **Seq2** **AAGAATGGCGTTTTTAATAGCAATAT**
- Décalage -1
  - **Seq1** **TTTGCGTTAAATCGTGTAGCAATTTAA**
  - **Seq2** **AAGAATGGCGTTTTTAATAGCAATAT**

Alignement avec « gaps » (brèches)

- Les alignements sans gaps sont rarement pertinents, car les divergences entre séquences incluent souvent des insertions et délétions.

- Les gaps permettent de mettre en évidence les régions de similarités multiples.

```
----TTTGCGTT--AAATCGTGTAGCAATTTAA     s=substitution; |=identité
1111s|s|||||11s||22222|||||||s|22     1=gap dans la 1ère séquence
AAGAATGGCGTTTTTAA-----TAGCAATAT--     2=gap dans la 2de séquence
```

- Gaps, insertions et délétions
  - Les "***gaps***" (***brèches***) reflètent soit une insertion dans l'une des séquences, soit une délétion dans l'autre.
  - Sur seule base de l'alignement d'une paire de séquences, on ne peut pas déterminer si un gap correspond à une délétion ou une insertion.
  - On utilise le terme ***indel*** pour désigner cet événement évolutif de nature indéterminée (insertion ou délétion) qui a donné lieu à un gap observé dans un alignement.

Alignements locaus versus globaux

Alignements globaux (Needleman-Wunsch) versus locaux (Smith-Waterman)

- Alignement global
  - Algorithme: ***Needleman-Wunsch*** (1970).
  - Outil web EMBOSS : ***needle*** (nucleic acids or proteins).
  - Approprié, par exemple, pour les protéines homologues qui sont conservées sur toute leur longueur.
  - L'alignement final inclut obligatoirement les deux séquences complètes.

```
--------LQGPSKGTGKGS-SRSWDNTGKAIVVWRS
--------|----|--|||---|--|-----------
IVLLAKSMLN-ITKSAGKGAIMRLGDA----------
```

- Alignement local
  - Algorithme: ***Smith-Waterman*** (1981).
  - Outil Web EMBOSS : ***water*** (nucleic acids or proteins).
  - Approprié, par exemple, pour les protéines qui partagent un domaine commun, restreint à un segment de chaque séquence.

```
          LQGPSSKTGKGS-SSRIWDN
                |-|||
IVLLAKSMLN-ITKKAGKGAIMRLGDA
```

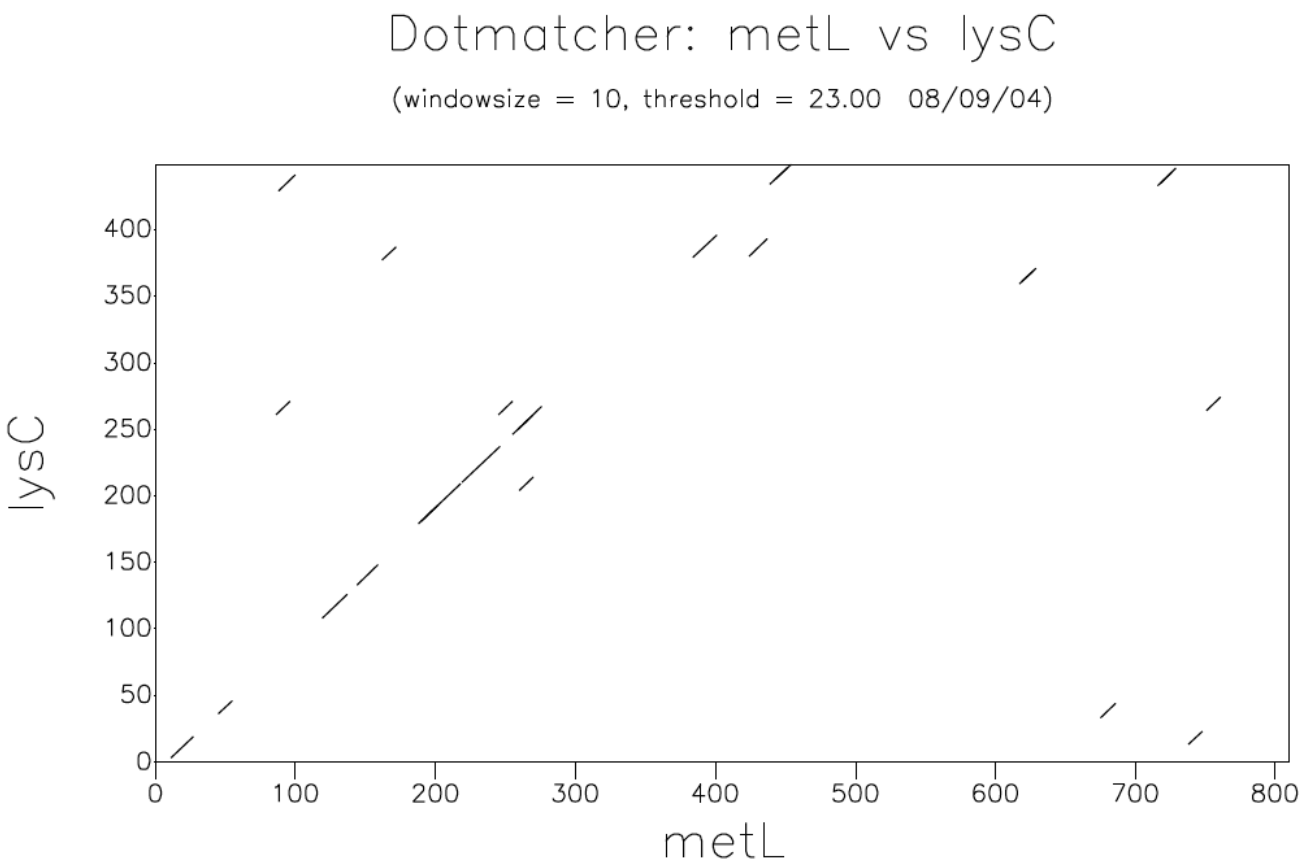
  - L'alignement final est restreint aux segments conservés.

```
KTGKG
|-|||
KAGKG
```


- Needleman, S. B. & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol 48, 443-53.
- Smith, T. F. & Waterman, M. S. (1981). Identification of common molecular subsequences. J Mol Biol 147, 195-7.

Aspartokinases: dot plot avec matrice de substitution (BLOSUM62)

- Avec le logiciel *dotmatcher*, une matrice de substitution est utilisée pour assigner un score à chaque paire de résidus.
- Ceci révèle la similarité entre les domaines aspartokinase de LysC (l'ensemble de la séquence) et de MetL (positions 1 à ~450).
- Notes
  - Cette région de similarité ne recouvre que la partie N-terminale de MetL, car il s'agit d'une enzyme bi-fonctionnelle. La région C-terminale contient un domaine homosérine déshydrogenase.
  - Le choix des paramètres est délicat, il varie selon les séquences alignées.



Needleman-Wunsch with partial similarities

```
# Matrix: EBLOSUM62
# Gap_penalty: 10.0
# Extend_penalty: 0.5
# Length: 854
# Identity:     136/854 (15.9%)
# Similarity:   209/854 (24.5%)
# Gaps:         449/854 (52.6%)
# Score: 351.0
metL               1 MSVIAQAGAKGRQLHKFGGSSLADVKCYLRVAGIMAEYSQPDDMMVVSAA     50
                     ||.|.        :.||||:|:||.....|.|.|:...:.. .::|:||:
lysC               1 MSEIV--------VSKFGGTSVADFDAMNRSADIVLSDANV-RLVVLSAS     41

metL              51 GSTTNQLINWLK-LSQTDRLSAHQVQQTLRRYQCDLISGL----LPAEEA     95
                     ...||.|:...: |...:|.   :....:|..|..::..|    :..||.
lysC              42 AGITNLLVALAEGLEPGERF---EKLDAIRNIQFAILERLRYPNVIREEI     88

metL              96 DSLISAFVSDLERLAALLDSGINDAVYAEVVGHGEVWSARLMSAVLNQQG    145
                     :.|:.. ::.|...|||..|   .|:..|:|.|||:.|..|...:|.::.
lysC              89 ERLLEN-ITVLAEAAALATS---PALTDELVSHGELMSTLLFVEILRERD    134

metL             146 LPAAWLDAREFLRA-ERAAQPQVDEGLSYPLLQQLLVQHPGKRLVVT-GF    193
                     :.|.|.|.|:.:|. :|..:.:.|......|....|....:.||:| ||
lysC             135 VQAQWFDVRKVMRTNDRFGRAEPDIAALAELAALQLLPRLNEGLVITQGF    184

metL             194 ISRNNAGETVLLGRNGSDYSATQIGALAGVSRVTIWSDVAGVYSADPRKV    243
                     |...|.|.|..|||.||||:|..:.....|||.||:||.|:|:|:.|||.|
lysC             185 IGSENKGRTTTLGRGGSDYTAALLAEALHASRVDIWTDVPGIYTTDPRVV    234

metL             244 KDACLLPLLRLDEASELARLAAPVLHARTLQPVSGSEIDLQLRCSYTPDQ    293
                     ..|..:..:...||:|:|...|.|||..||.|...|:|.:.:..|..|..
lysC             235 SAAKRIDEIAFAEAAEMATFGAKVLHPATLLPAVRSDIPVFVGSSKDPRA    284

metL             294 GSTRI---------ERVLASGTGARIVTSHDDVCLIEFQVPASQDFKLAH    334
                     |.|.:         .|.||......::|.|      ...:..|:.| ||
lysC             285 GGTLVCNKTENPPLFRALALRRNQTLLTLH------SLNMLHSRGF-LA-    326

metL             335 KEIDQILKRAQVRPLAVGVHNDRQLLQFCYTSEVA----------D    370
                      |:..||.|          ||..  :....||||:             |
lysC             327 -EVFGILAR----------HNIS--VDLITTSEVSVALTLDTTGSTSTGD    363
```

- Alignment of *E.coli* lysC and metL proteins with Needleman-Wunsch algorithm.
- metL contains two domains: aspartokinase and homoserine dehydrogenase.
- LysC only contains the aspartokinase domains.
- With Smith-Waterman, the %similarity is calculated over the whole length of the alignment (854aa), which gives 24.5%.
- Actually, most of the alignment length is in the terminal gap (the homoserine dehydrogenase domain of metL).
- This percentage is lower than the usual threshold for considering two proteins as homolog.

Smith-Waterman with partial similarities

```
# Matrix: EBLOSUM62
# Gap_penalty: 10.0
# Extend_penalty: 0.5
# Length: 482
# Identity:     133/482 (27.6%)
# Similarity:   205/482 (42.5%)
# Gaps:          85/482 (17.6%)
# Score: 353.5
metL              16 KFGGSSLADVKCYLRVAGIMAEYSQPDDMMVVSAAGSTTNQLINWLK-LS     64
                     ||||:|:||.....|.|.|:...:.. .::|:||:...||.|:...: |.
lysC               8 KFGGTSVADFDAMNRSADIVLSDANV-RLVVLSASAGITNLLVALAEGLE     56

metL              65 QTDRLSAHQVQQTLRRYQCDLISGL----LPAEEADSLISAFVSDLERLA    110
                     ..:|.   :....:|..|..::..|    :..||.:.|:.. ::.|...|
lysC              57 PGERF---EKLDAIRNIQFAILERLRYPNVIREEIERLLEN-ITVLAEAA    102

metL             111 ALLDSGINDAVYAEVVGHGEVWSARLMSAVLNQQGLPAAWLDAREFLRA-    159
                     ||..|   .|:..|:|.|||:.|..|...:|.:::.:.|.|.|.|:.:|.
lysC             103 ALATS---PALTDELVSHGELMSTLLFVEILRERDVQAQWFDVRKVMRTN    149

metL             160 ERAAQPQVDEGLSYPLLQQLLVQHPGKRLVVT-GFISRNNAGETVLLGRN    208
                     :|..:.:.|......|....|:...:.||:| |||...|.|.|..|||.
lysC             150 DRFGRAEPDIAALAELAALQLLPRLNEGLVITQGFIGSENKGRTTTLGRG    199

metL             209 GSDYSATQIGALAGVSRVTIWSDVAGVYSADPRKVKDACLLPLLRLDEAS    258
                     ||||:|..:.....|||.||:||.|:|:.|||.|..|..:..:...||:
lysC             200 GSDYTAALLAEALHASRVDIWTDVPGIYTTDPRVVSAAKRIDEIAFAEAA    249

metL             259 ELARLAAPVLHARTLQPVSGSEIDLQLRCSYTPDQGSTRI---------E    299
                     |:|...|.|||..||.|...|:|.:.:..|..|..|.|.:         .
lysC             250 EMATFGAKVLHPATLLPAVRSDIPVFVGSSKDPRAGGTLVCNKTENPPLF    299

metL             300 RVLASGTGARIVTSHDDVCLIEFQVPASQDFKLAHKEIDQILKRAQVRPL    349
                     |.||......::|.|      ...:..|:.| ||  |:..||.|
lysC             300 RALALRRNQTLLTLH------SLNMLHSRGF-LA--EVFGILAR------    334

metL             350 AVGVHNDRQLLQFCYTSEVA--------------DSAL--KILDEAGLPG    383
                         ||..  :....||||:              |:.|  .:|.|.....
lysC             335 ----HNIS--VDLITTSEVSVALTLDTTGSTSTGDTLLTQSLLMELSALC    378
```

- Alignment of *E.coli* lysC and metL proteins with Smith-Waterman algorithm.
- The alignment is almost identical to the one reported by Needleman-Wunsch, but the score is now considered on the aligned segments only (482 aa).
- On this region, there is 42.5% of similarity.

Aspects algorithmiques

Recherche de l'alignement optimal par programmation dynamique

- Les diapos suivantes ne font pas partie de la matière d'examen pour le cours BI4U2 (bioinformatique appliquée de L2 sciences de la vie).

Nombre d'alignements possibles sans gaps

- Si l'on ne prend en compte que les substitutions (ni délétion, ni insertion), l'alignement de séquence ***sans gaps*** est très simple.

- Temps requis
  - Définissons que
    - L1 longueur de seq1
    - L2 longueur de seq2
    - L2 <= L1 (on désigne par seq2 la séquence la plus courte)
  - Il existe *L1+L2-1* décalages possibles entre les séquences sequence 1 and 2.
  - Pour chaque décalage, on calcule le score sur la longueur alignée (max=L2, quand seq2 est complètement couverte par seq1).
  - ***T ~ (L1 + L2 - 1)L2 – L2(L2 - 1)***

Nombre d'alignements possibles sans gaps

- Les alignements sans gaps sont rarement pertinents, car les divergences entre séquences incluent souvent des insertions et délétions.
- Les gaps permettent de mettre en évidence les régions de similarités multiples.

```
----TTTGCGTT--AAATCGTGTAGCAATTTAA     s=substitution; |=match
1111s|s|||||11s||22222|||||||s|22     1=gap dans la 1ère séquence
AAGAATGGCGTTTTTAA-----TAGCAATAT--     2=gap dans la 2de séquence
```

- Le fait d'autoriser les gaps augmente considérablement la complexité de l'alignement. A chaque position, on peut trouver l'une des trois possibilités suivantes:
  - gap dans la première séquence
  - gap dans la seconde séquence
  - Superposition des résidus des deux séquences (identité ou substitution)
- Au total, la complexité du problème $N \sim 3^L$, où L est la taille de la séquence la plus courte. Le nombre de possibilités augmente donc ***exponentiellement*** avec la longueur des séquences. Ceci devient rapidement infaisable.
  - Pour deux séquences de taille 1000, il existe ~3^{1000} (~10^{477}) alignements possibles.
- On voudrait identifier l'alignement optimal, càd celui qui obtient le score le plus élevé. Il est cependant impossible de tester chaque alignement et de calculer son score, car ceci prendrait un temps virtuellement infini.

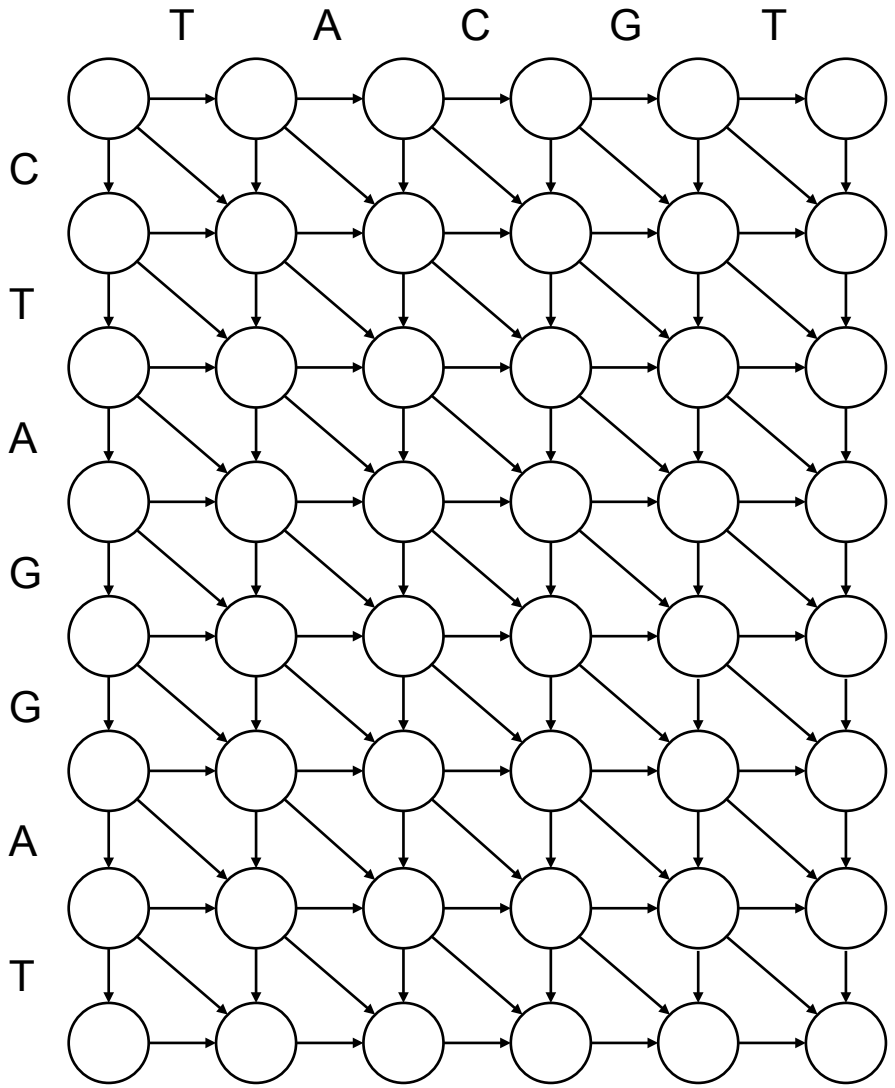
Dynamical programming - global alignment

- **Needleman-Wunsch** proposed an algorithm called dynamical programming
  - Performs a ***global alignment*** (the sequences are aligned on the whole length)
  - The time of processing is proportional to the product of sequence lengths. It si thus increasing ***quadratically*** with the sequence length, instead of exponentially.
  - Guarantees to return ***the highest scoring alignment*** between two sequences.

Reference: Needleman, S. and Wunsch, C. (1970). A general method applicable to the search for similarities in the amino acid sequences of two proteins. *J. Mol. Biol.*, 48:444–453.

Dynamical programming - global alignment

- For each position in the alignment, one can have
  - a gap in sequence 1
  - a gap in sequence 2
  - aligned residues (match or substitution)

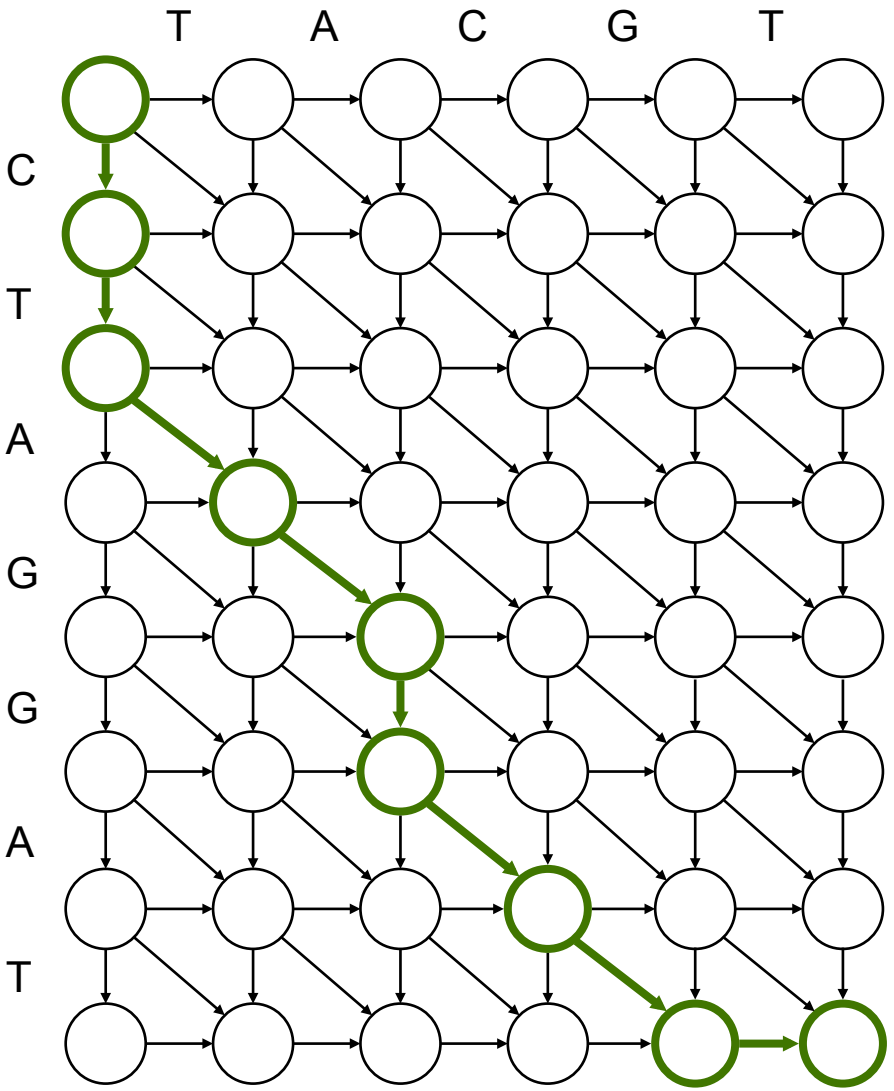


Dynamical programming - paths

- Any possible alignment between the two sequences can be represented as a path from the left top corner to the right bottom corner.
- For example, the path highlighted in green corresponds to the alignment below.

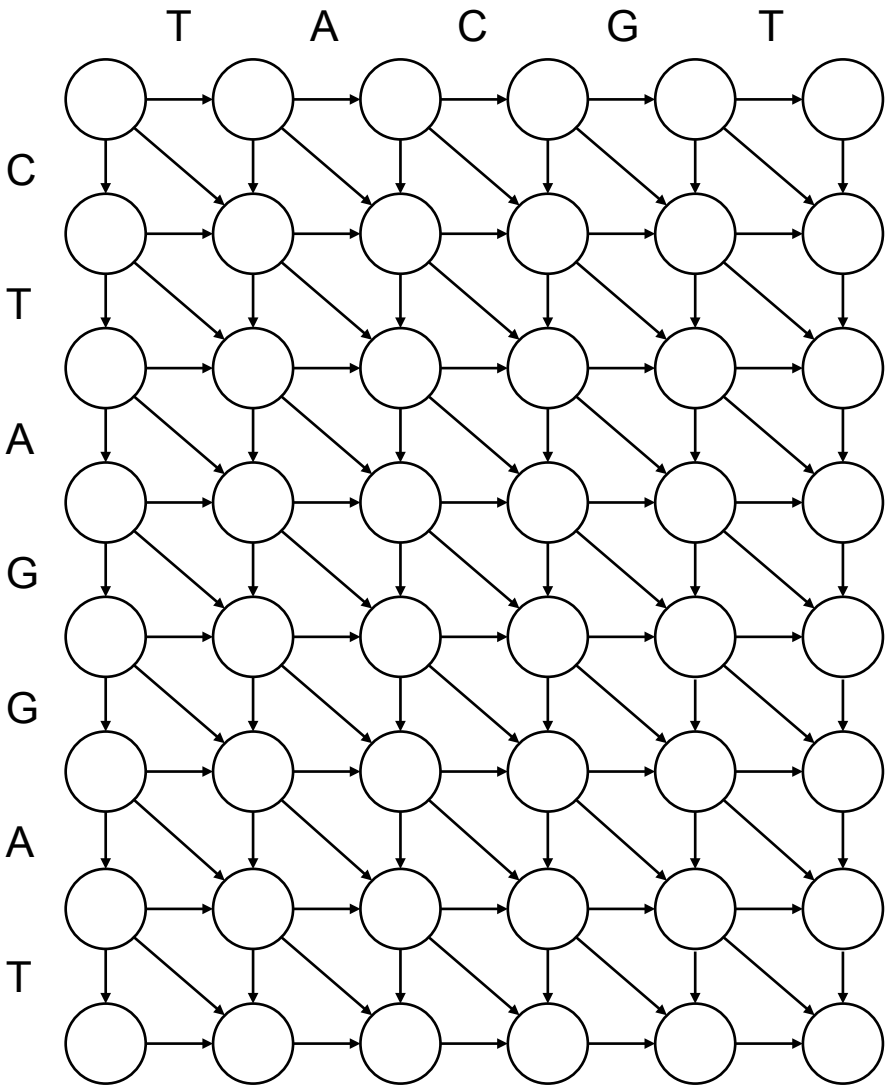
```
- - T A - C G T
g g s s g s s g
C T A G G A T –
```

- Obviously, this alignment is not optimal : it does not contain a single match !



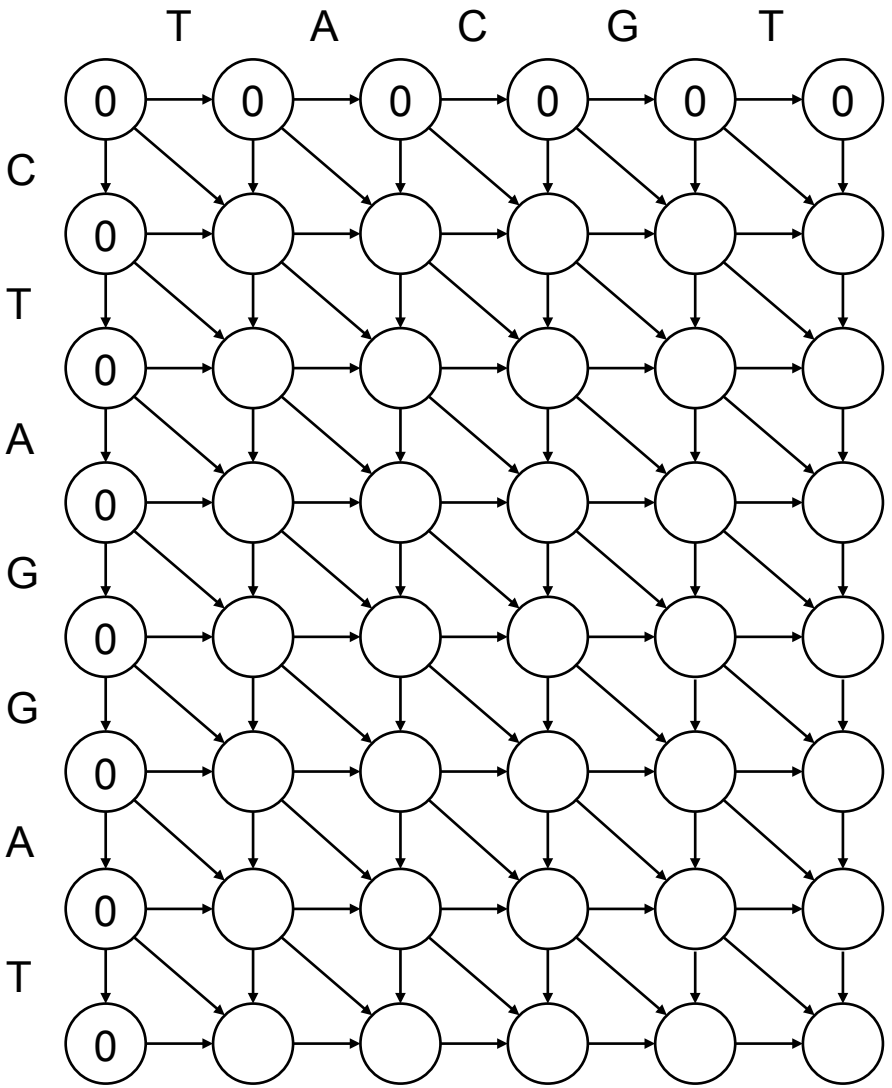
Dynamical programming - global alignment

- Our goal is to find the best possible alignment, which corresponds to the path giving the optimal score (we still need to define how this score will be computed).
- As discussed before, the number of possible paths increases exponentially with the sequence sizes.
- Dynamical programming however allows us to find this optimal score in a quadratic time (L1*L2), by building the solution progressively.
- This progressive path finding allows us to avoid evaluating a large number of paths which would anyway return a sub-optimal score.



Dynamical programming - initialization

- The top row and left column are initialized .
- We will now progressively fill the other cells of the matrix by calculating, for each cell, its optimal score as a function of the path used to reach this cell.
- Two possible ways to initialize
  - If you consider that there is no cost for an terminal gaps (start, end), initialize first row and column with 0.
  - If you want to penalize terminal gaps, initialize first row and column with gap penalties.



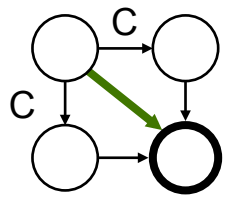
Dynamical programming – the 3 way to leave a cell

- From each starting cell, we can take 3 possible moves:
  - Diagonal
    - Align the two residues (match or substitution)
  - Rightward
    - Align one residue of the horizontal sequence with a gap in the vertical sequence.
  - Downward
    - Align one residue of the vertical sequence with a gap in the horizontal sequence.

Diagonal move : align the two residues

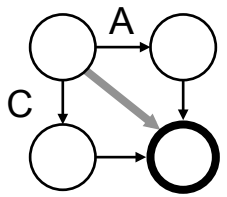
Alignment

Match



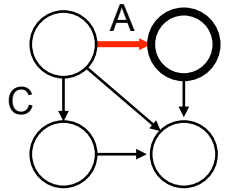
C
C

Substitution



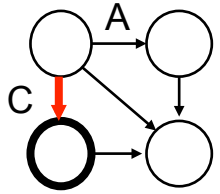
A
C

Rightward move: insert gap in vertical sequence



A
-

Downward move: insert gap in horizontal sequence



-
C

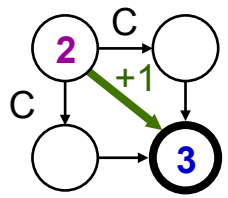
Dynamical programming – the 3 way to leave a cell

- From each starting cell, we can take 3 possible moves:
  - Diagonal
    - Align the two residues (match or substitution)
  - Rightward
    - Align one residue of the horizontal sequence with a gap in the vertical sequence.
  - Downward
    - Align one residue of the vertical sequence with a gap in the horizontal sequence.
- We can define a scoring scheme, and define the score of each possible direction.
  - match +1
  - substitution -1
  - gap -2

Diagonal move : align the two residues

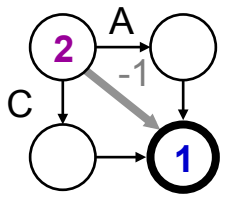
Alignment

Match



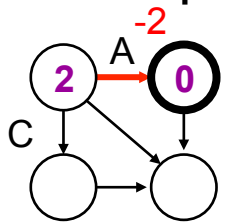
C
C

Substitution



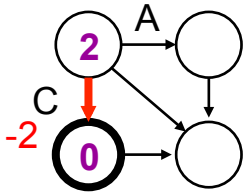
A
C

Rightward move: insert gap in vertical sequence



A
-

Downward move: insert gap in horizontal sequence

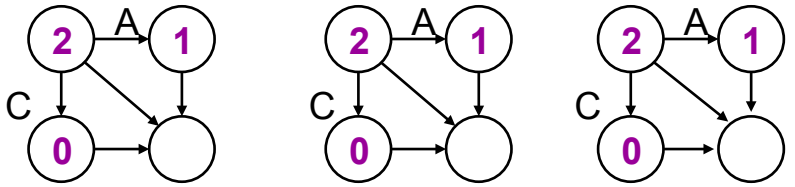


-
C

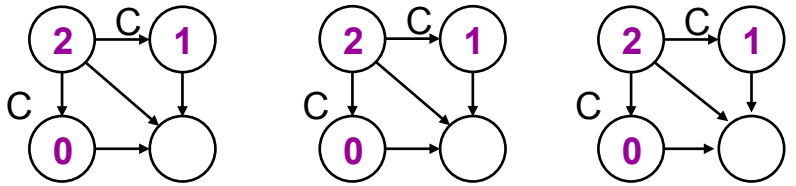
Exercise – what is the best way to reach a cell

- Let us assume the following scoring scheme
  - match +1
  - substitution -1
  - gap -2
- At a given destination cell, 3 scores are calculated depending on the 3 possible starting positions:
  - upper neighbour + gap cost
  - left neighbour + gap cost
  - upper-left neighbour +
    - match score if the residue match
    - substitution cost if residues do not match
- For each of the cases besides
  - Compute the 3 possible scores.
  - Indicate the best score in the destination cell.
  - Mark the arrow giving this best score.

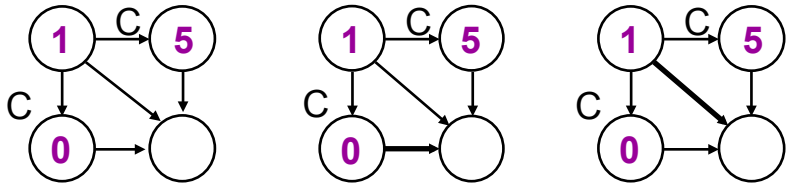
Example 1 **Alignment?**



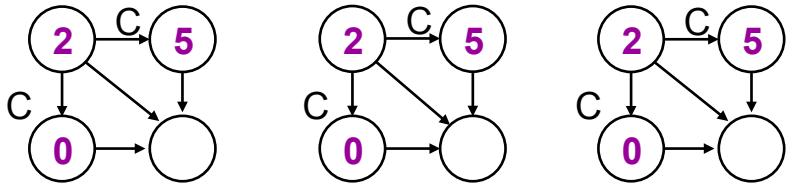
Example 2



Example 3

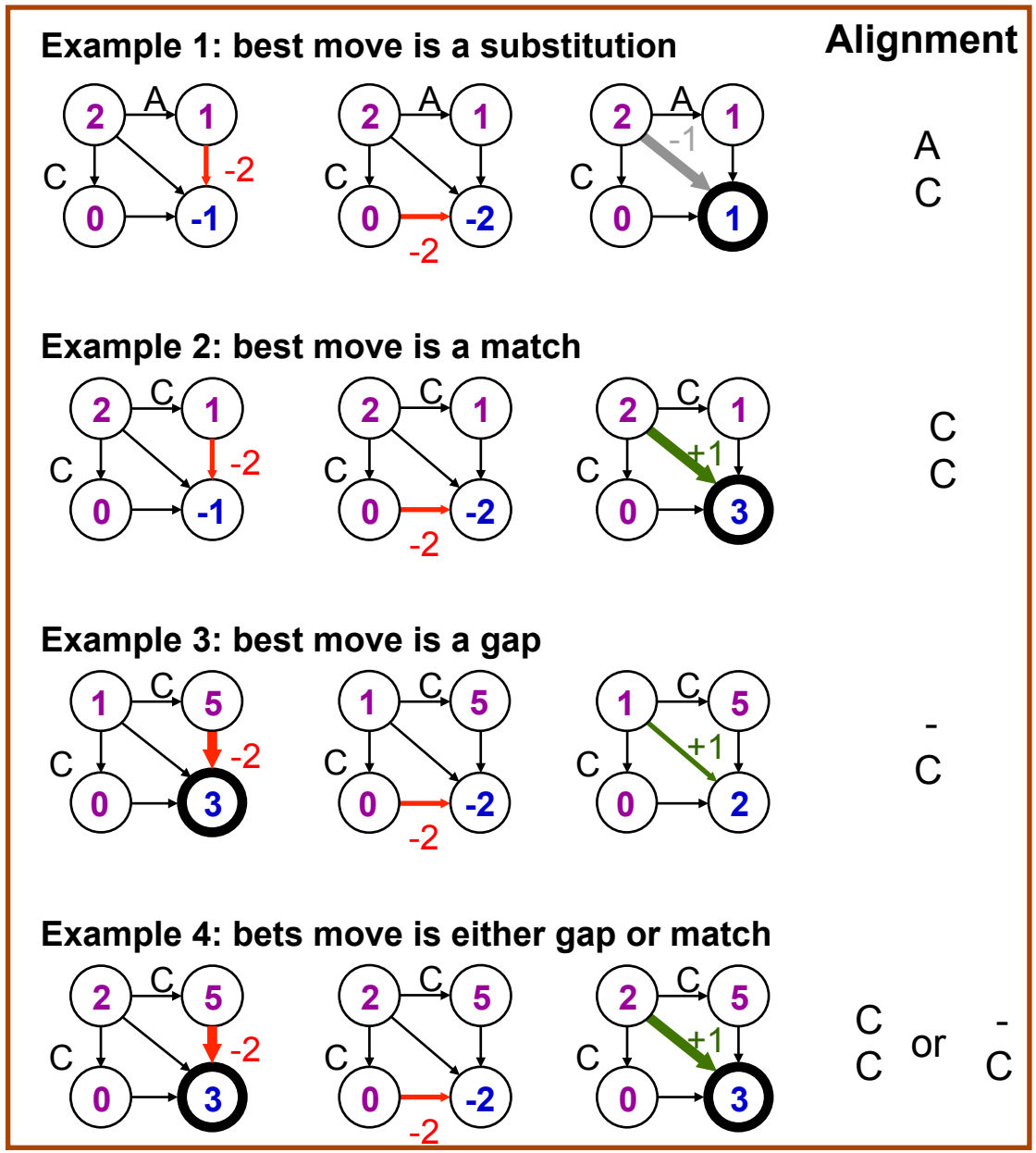


Example 4



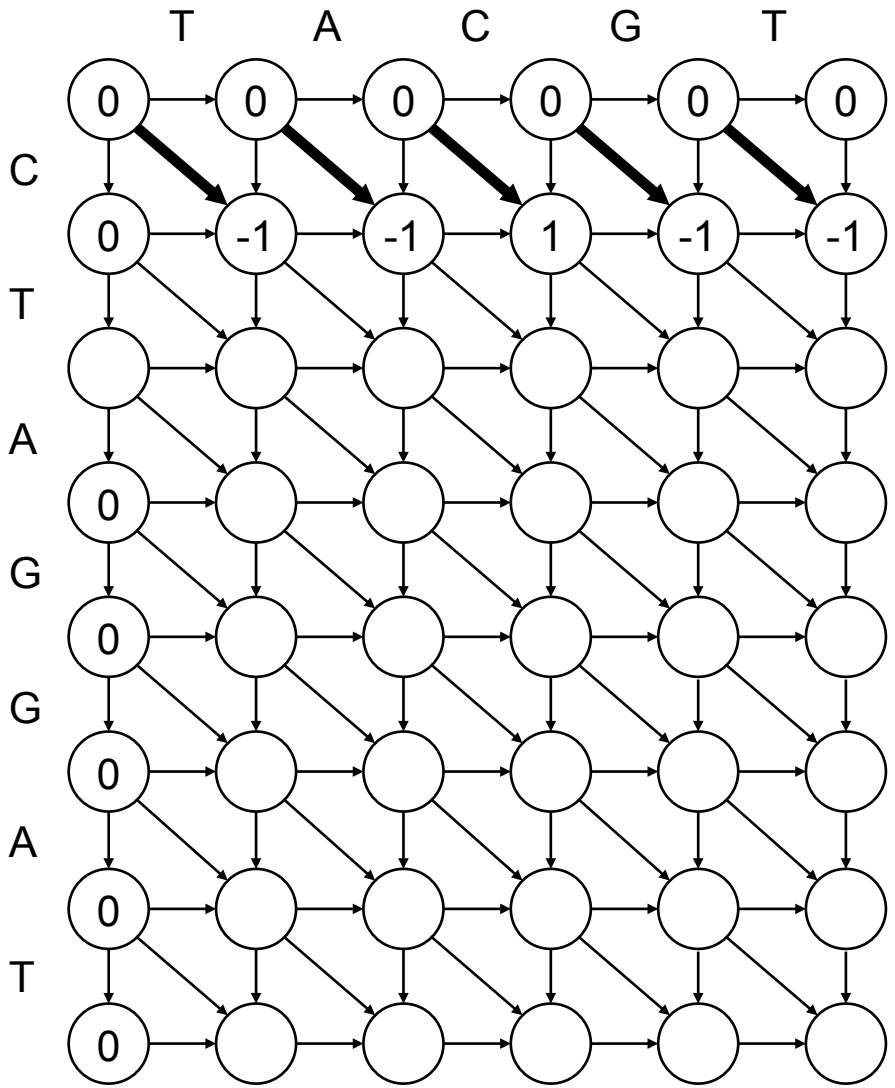
Dynamical programming – the best way to reach a cell

- Let us assume the following scoring scheme
  - match +1
  - substitution -1
  - gap -2
- At a given destination cell, 3 scores are calculated depending on the 3 possible starting positions:
  - upper neighbour + gap cost
  - left neighbour + gap cost
  - upper-left neighbour +
    - match score if the residue match
    - substitution cost if residues do not match
- The **highest score** is retained and the arrow is labelled
- In some cases (example 4), there are **several** equivalent highest scores



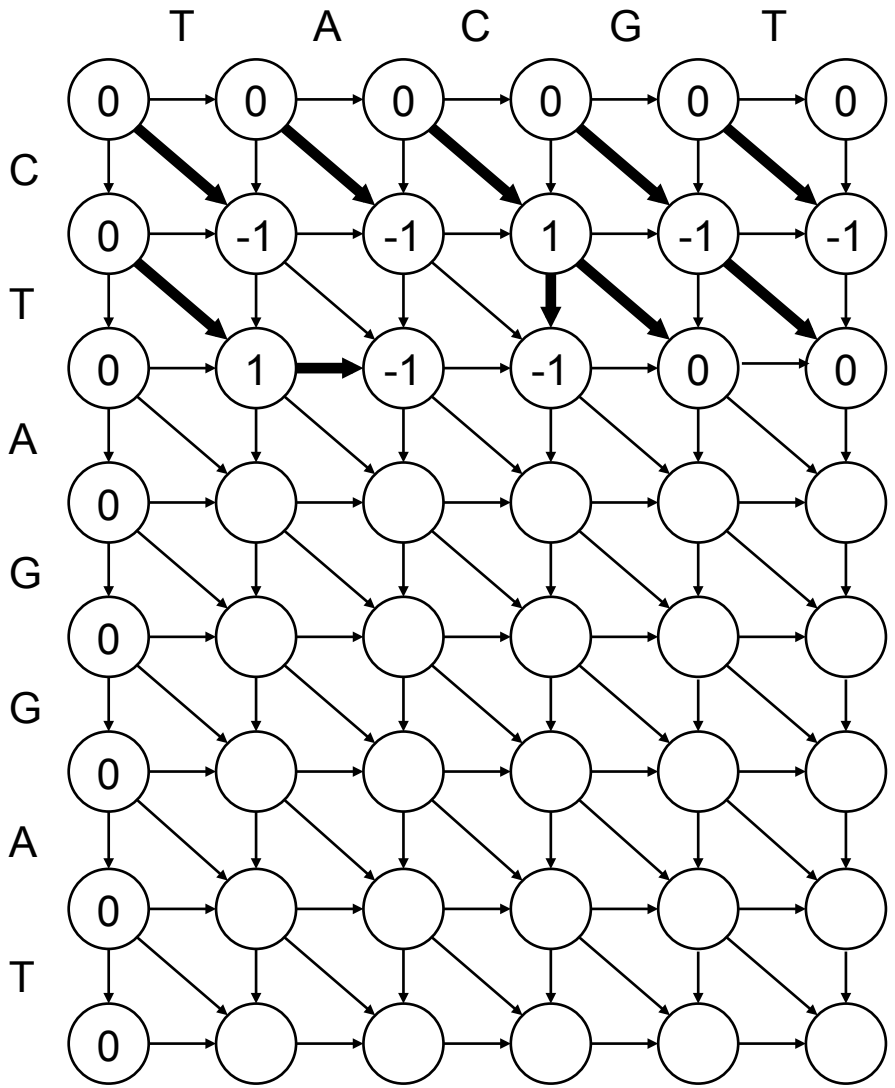
Dynamical programming - recursive computation

- The first row is processed from left to right



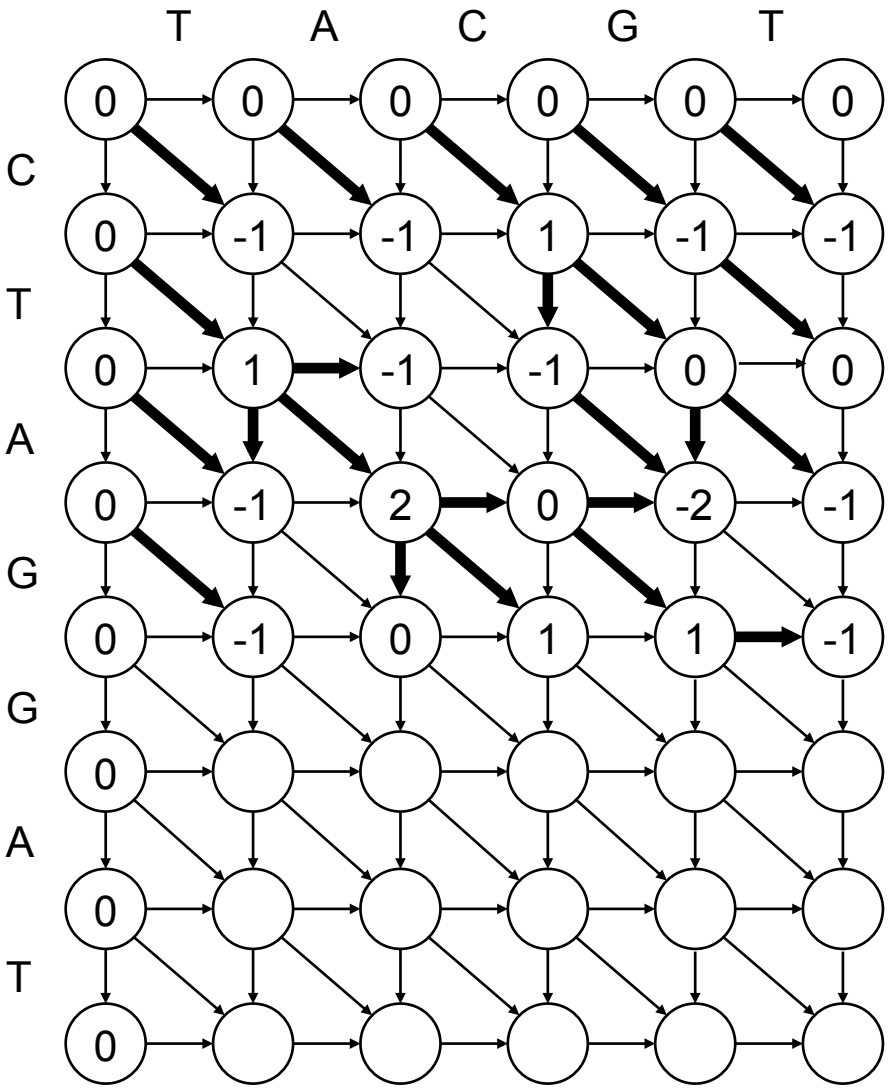
Dynamical programming - recursive computation

- The second row is then processed



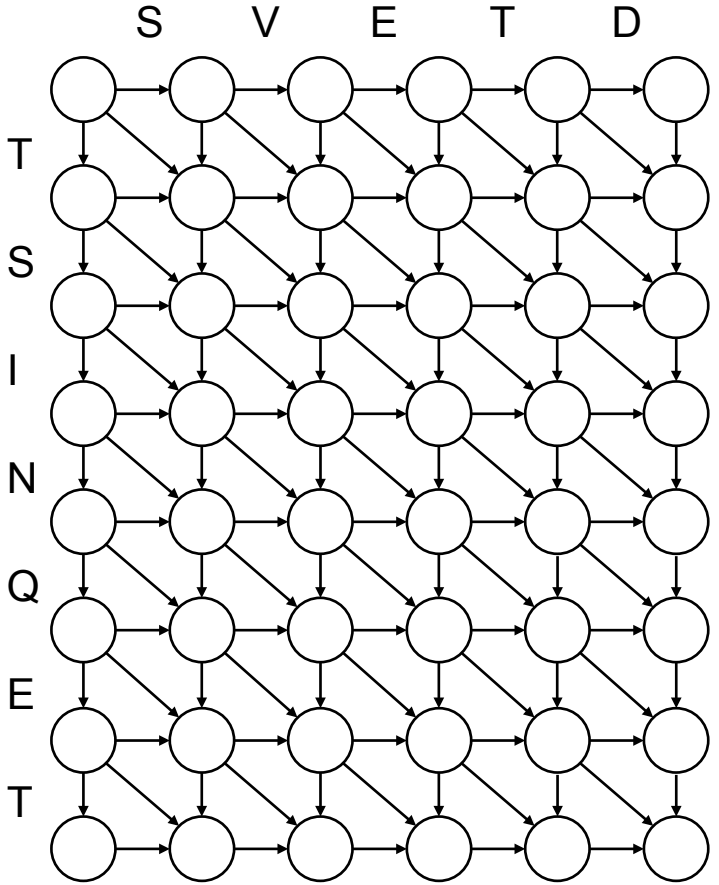
Dynamical programming - recursive computation

- The process is iterated until the right corner is reached.
- Exercise: calculate the remaining scores.



Needleman-Wunsch : exercise

- Using the Needleman-Wunsch algorithm, fill the scores of the alignment matrix with the following parameters
  - Substitution matrix: BLOSUM62
  - Gap opening penalty: -5
  - Gap extension penalty: -1
  - Initial and terminal gaps: 0

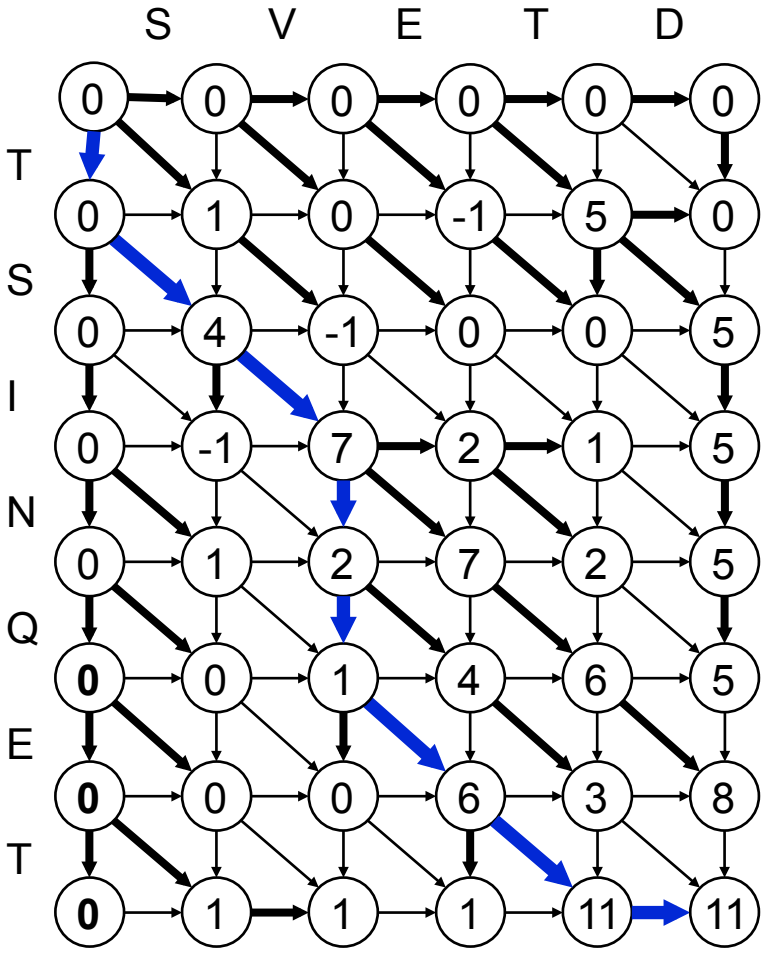


		Ala A	Arg R	Asn N	Asp D	Cys C	Gln Q	Glu E	Gly G	His H	Ile I	Leu L	Lys K	Met M	Phe F	Pro P	Ser S	Thr T	Trp W	Tyr Y	Val V
Ala	A	4																			
Arg	R	-1	5																		
Asn	N	-2	0	6																	
Asp	D	-2	-2	1	6																
Cys	C	0	-3	-3	-3	9															
Gln	Q	-1	1	0	0	-3	5														
Glu	E	-1	0	0	2	-4	2	5													
Gly	G	0	-2	0	-1	-3	-2	-2	6												
His	H	-2	0	1	-1	-3	0	0	-2	8											
Ile	I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

Needleman-Wunsch : solution of the exercise

- A substitution matrix can be used to assign specific scores to each pair of residues.
- Exercise: fill the scores of the alignment matrix using the BLOSUM62 substitution matrix.
  - Gap opening penalty: -5
  - Gap extension penalty: -1
  - Initial and terminal gaps: 0

		Ala A	Arg R	Asn N	Asp D	Cys C	Gln Q	Glu E	Gly G	His H	Ile I	Leu L	Lys K	Met M	Phe F	Pro P	Ser S	Thr T	Trp W	Tyr Y	Val V
Ala	A	4																			
Arg	R	-1	5																		
Asn	N	-2	0	6																	
Asp	D	-2	-2	1	6																
Cys	C	0	-3	-3	-3	9															
Gln	Q	-1	1	0	0	-3	5														
Glu	E	-1	0	0	2	-4	2	5													
Gly	G	0	-2	0	-1	-3	-2	-2	6												
His	H	-2	0	1	-1	-3	0	0	-2	8											
Ile	I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4




```
- S V - - E T D
- | : - - | | -
T S I N Q E T -
```

Needleman-Wunsch example

- Alignment of *E.coli* metL and thrA proteins with Needleman-Wunsch algorithm.
- The vertical bars indicate identity between two amino-acids.
- The columns and dots indicate similarities, i.e. pairs of residues having a positive scores in the chosen substitution matrix (BLOSUM62).

```
# Matrix: EBLOSUM62
# Gap_penalty: 10.0
# Extend_penalty: 0.5
# Length: 867
# Identity:     254/867 (29.3%)
# Similarity:   423/867 (48.8%)
# Gaps:         104/867 (12.0%)
# Score: 929.0

metL              1 MSVIAQAGAKGRQLHKFGGSSLADVKCYLRVAGIMAEYSQPDDM-MVVSA     49
                    .::.||||:|:|:.:.:||||.|:...::...: .|:||
thrA              1           MRVLKFGGTSVANAERFLRVADILESNARQGQVATVLSA     39

metL             50 AGSTTNQLINWLK-----------LSQTDRLSAHQVQQTLRRYQCDLISG     88
                    ....||.|:..::           :|..:|:.|            :|::|
thrA             40 PAKITNHLVAMIEKTISGQDALPNISDAERIFA------------ELLTG     77

metL             89 LLPAEEADSL--ISAFV-SDLERLAALLDSGIN------DAVYAEVVGHG    129
                    |..|:....|  :..|| .:..::..:| .||:      |::.|.::..|
thrA             78 LAAAQPGFPLAQLKTFVDQEFAQIKHVL-HGISLLGQCPDSINAALICRG    126

metL            130 EVWSARLMSAVLNQQGLPAAWLD-AREFLRAERAAQPQVD--EGLSYPLL    176
                    |..|..:|:.||..:|.....:| ..:.|......:..||  |.......
thrA            127 EKMSIAIMAGVLEARGHNVTVIDPVEKLLAVGHYLESTVDIAESTRRIAA    176

metL            177 QQLLVQHPGKRLVVTGFISRNNAGETVLLGRNGSDYSATQIGALAGVSRV    226
                    .::...|   .:::.||.:.|..||.|:||||||||||..:.|.......
thrA            177 SRIPADH---MVLMAGFTAGNEKGELVVLGRNGSDYSAAVLAACLRADCC    223

metL            227 TIWSDVAGVYSADPRKVKDACLLPLLRLDEASELARLAAPVLHARTLQPV    276
                    .||:||.|||:.|||:|.||.||..:...||.||:...|.|||.||:.|:
thrA            224 EIWTDVDGVYTCDPRQVPDARLLKSMSYQEAMELSYFGAKVLHPRTITPI    273

metL            277 SGSEIDLQLRCSYTPDQ-----GSTRIERVLASGTGARIVTSHDDVCLIE    321
                    :..:|...::.:..|..     |::|.|..|.    .:.:::.:::.:..
thrA            274 AQFQIPCLIKNTGNPQAPGTLIGASRDEDELP----VKGISNLNNMAMFS    319

metL            322 FQVPASQDFKLAHKEIDQILKRAQVRPLAVGVHNDRQLLQFCYTSEVADS    371
```

Dynamical programming - local alignment

- The Needleman-Wunsch algorithm performs a global alignemnt (it finds the best path between the two corners of the alignment matrix).
- This is appropriate when the sequences are similar over their whole length, but local similarities could be missed.
- In 1981, Smith and Waterman published an adaptation of the Needleman-Wunsch algorithm, which allows to detect local similarities.

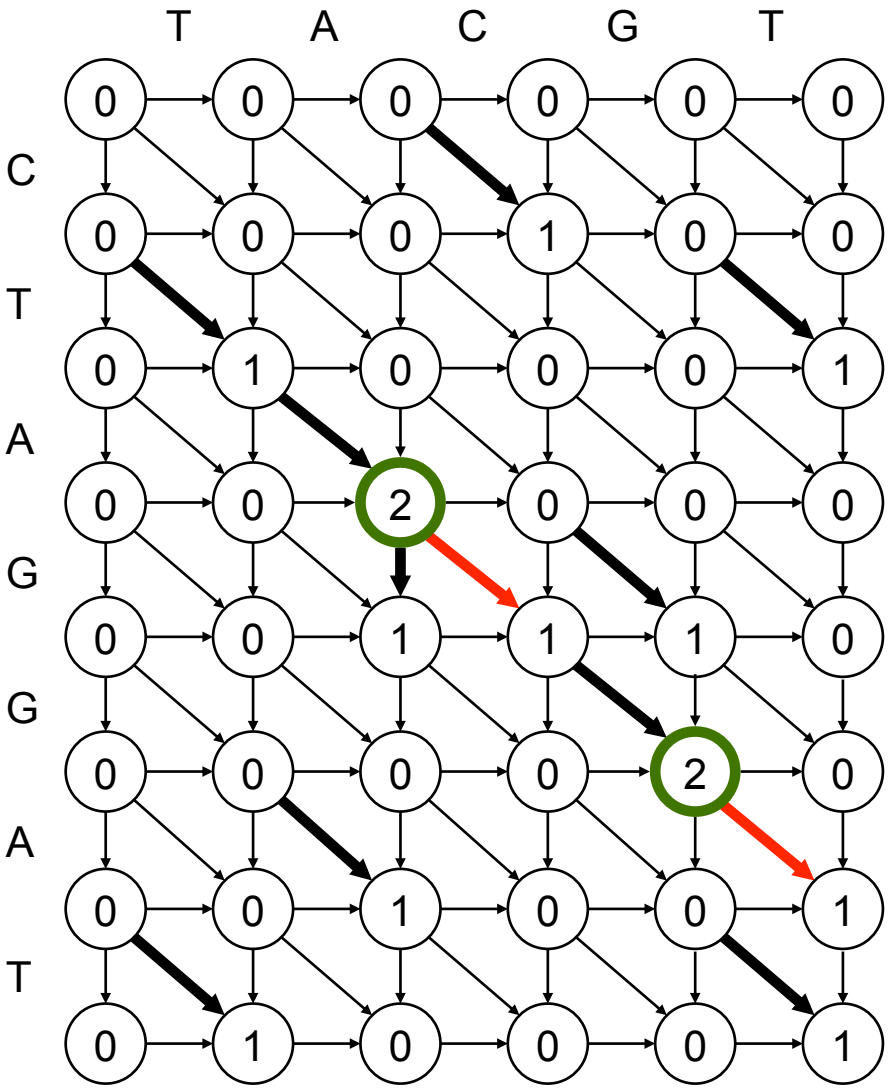
- Reference: Smith, T. F. and Waterman, M. S. (1981). Identification of common molecular subsequences. J. Mol. Biol., 147:195–197.

Dynamical programming - local alignment

- Smith-Waterman algorithm
- The algorithm is similar to Needleman-Wunsch, but negative scores are replaced by 0
- Alignments stop after **local maxima**
- In this case we have two overlapping alignments, each having a score of 2 :

  TA TACG

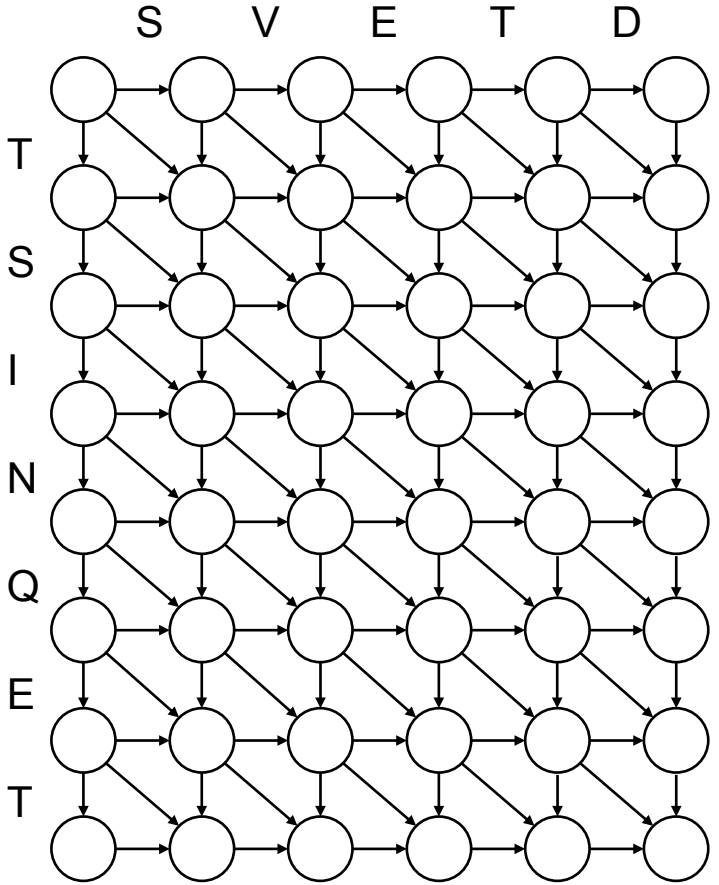
  TA TAGG
- Basically, the cost of the C/G substitution is compensate by the benefit of the G/G match.



- Reference: Smith, T. F. and Waterman, M. S. (1981). Identification of common molecular subsequences. J. Mol. Biol., 147:195–197.

Smith-Waterman: exercise

- Using the Smith-Waterman algorithm, fill the scores of the alignment matrix with the following parameters
  - Substitution matrix: BLOSUM62
  - Gap opening penalty: -5
  - Gap extension penalty: -1
  - Initial and terminal gaps: 0

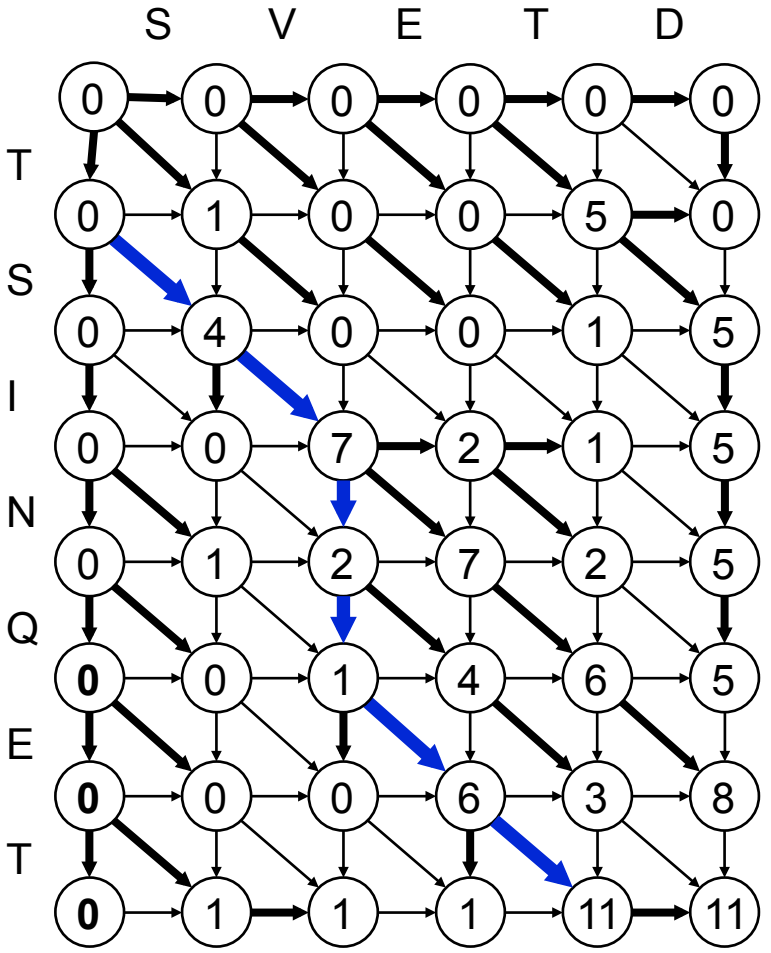


		Ala A	Arg R	Asn N	Asp D	Cys C	Gln Q	Glu E	Gly G	His H	Ile I	Leu L	Lys K	Met M	Phe F	Pro P	Ser S	Thr T	Trp W	Tyr Y	Val V
Ala	A	4																			
Arg	R	-1	5																		
Asn	N	-2	0	6																	
Asp	D	-2	-2	1	6																
Cys	C	0	-3	-3	-3	9															
Gln	Q	-1	1	0	0	-3	5														
Glu	E	-1	0	0	2	-4	2	5													
Gly	G	0	-2	0	-1	-3	-2	-2	6												
His	H	-2	0	1	-1	-3	0	0	-2	8											
Ile	I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

Smith-Waterman : solution of the exercise

- A substitution matrix can be used to assign specific scores to each pair of residues.
- Exercise: fill the scores of the alignment matrix using the BLOSUM62 substitution matrix.
  - Gap opening penalty: -5
  - Gap extension penalty: -1
  - Initial and terminal gaps: 0

		Ala A	Arg R	Asn N	Asp D	Cys C	Gln Q	Glu E	Gly G	His H	Ile I	Leu L	Lys K	Met M	Phe F	Pro P	Ser S	Thr T	Trp W	Tyr Y	Val V
Ala	A	4																			
Arg	R	-1	5																		
Asn	N	-2	0	6																	
Asp	D	-2	-2	1	6																
Cys	C	0	-3	-3	-3	9															
Gln	Q	-1	1	0	0	-3	5														
Glu	E	-1	0	0	2	-4	2	5													
Gly	G	0	-2	0	-1	-3	-2	-2	6												
His	H	-2	0	1	-1	-3	0	0	-2	8											
Ile	I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4




```
S V - - E T
|   :   -   -   |   |
S I N Q E T
```

Case study
Pairwise similarities between opsins

Needleman-Wunsch – pairwise comparisons between opsins

Needle percent s	OPS_FUGU	OPSB_BOVIN	OPSB_HUMAN	OPSB_MOUSE	OPSB_POUL	OPSB_RAT	OPSG_HUMAN	OPSR_CANIS	OPSR_HUMAN	OPSR_POUL	OPSRH2_FLY	OPSRH3_FLY	OPSV_FUGU	OPSV_HUMA	Moyenne
OPS5_MOUS	17														17
OPSB_HUMAN		92	100												96
OPSB_MOUSE		93	93												93
OPSB_POUL				65											65
OPSB_RAT			92	97											95
OPSG_HUMAN			61												61
OPSG_MOUSE			61	62											62
OPSG_RAT						62									62
OPSMELA_HUMA					36										36
OPSR_BOVIN		59													59
OPSR_CANFA			61												61
OPSR_CANIS								100							100
OPSR_HUMAN			61				98		100						86
OPSR_POUL									90						90
OPSV_HUMA										90					90
OPSV_MOUS											40	39		92	57
OPSV_POUL													30	59	44
OPSV-HUMA	25														25
Moyenne	21	82	76	75	36	62	98	100	95	90	40	39	30	76	69

Dynamical programming - summary

- Guarantees to find the optimal score.
- Is able to return all the alignments which raise the optimal score.
- Processing time is proportional to product of sequence lengths.
- Can be adapted for global (Needleman-Wunsch) or local (Smith-Waterman) alignment.
- Global alignments
  - are appropriate for aligning sequences conserved over their whole length (recent divergence).
- Local alignments
  - Are able to detect domains which cover only a fraction of the sequences
  - Are also able to return a complete alignment, when the conservation covers the whole sequence

References

- Dynamical programming applied to pairwise sequence alignment
  - Needleman-Wunsch (pairwise, global)
    - Needleman, S. B. & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol 48, 443-53.
  - Smith-Waterman (pairwise, local)
    - Smith, T. F. & Waterman, M. S. (1981). Identification of common molecular subsequences. J Mol Biol 147, 195-7.
- Software tools
  - The EMBOSS suite provides efficient implementations of the two “classical” dynamical programming algorithms
    - needle Needleman-Wunsch
    - water Smith-Waterman
  - **Pairwise Sequence Alignment tools at the EBI**
    - A web interface to EMBOSS tools (needle, water)
    - http://www.ebi.ac.uk/Tools/psa/
  - They can be installed locally or accessed via the SRS Web server
    - http://srs.ebi.ac.uk/

Exercice

T A C G T

Q

P

M

L

S

C

Q