Sequence Alignments

Dr. Fayyaz ul Amir Afsar Minhas
PIEAS Biomedical Informatics Research Lab
Department of Computer and Information Sciences
Pakistan Institute of Engineering & Applied Sciences
PO Nilore, Islamabad, Pakistan
http://faculty.pieas.edu.pk/fayyaz/

Similarity between biological sequences

- One of the most fundamental tasks in Bioinformatics
  - Sequence Alignment
- Why?
  - Same reason we use Google
    - Search for a gene across genomes of multiple species
    - Find the importance of a gene
  - Example: Calmodulin
    - >sp|P62158|CALM_HUMAN Calmodulin OS=Homo sapiens GN=CALM1 PE=1 SV=2
      MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADG
      NGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGYISAAELRHVMTNLGEKLTDE
      EVDEMIREADIDGDGQVNYEEFVQMMTAK

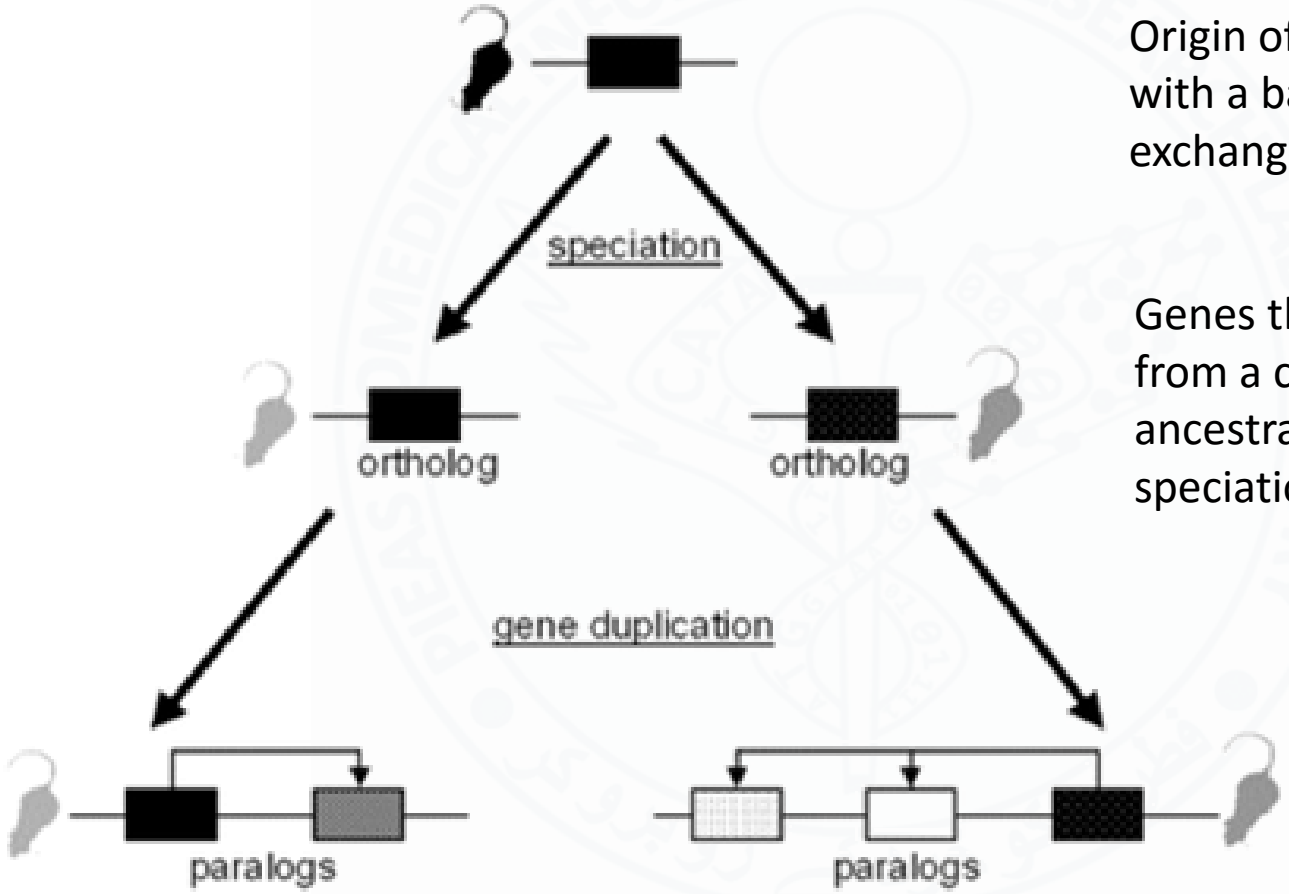
      http://www.uniprot.org/uniprot/P62158

Searching for Calmodulin

```
Human          MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADG 60
Mouse          MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADG 60
Round_Worm     MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADG 60
Drosophila     MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAELQDMINEVDADG 60
Arabidopsis    MADQLTDEQISEFKEAFSLFDKDGDGCITTKELGTVMRSLGQNPTEAELQDMINEVDADG 60
Yeast          MSSNLTEEQIAEFKEAFALFDKDNNGSISSSELATVMRSLGLSPSEAEVNDLMNEIDVDG 60
               *:.:**:***:******:*****.:* *::.**.******* .*:***::*::**:*.**

Human          NGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGYISAAELRHVMTNLGEKLTDE 120
Mouse          NGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGYISAAELRHVMTNLGEKLTDE 120
Round_Worm     NGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGFISAAELRHVMTNLGEKLTDE 120
Drosophila     NGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGFISAAELRHVMTNLGEKLTDE 120
Arabidopsis    NGTIDFPEFLNLMAKKMKDTDSEEELKEAFRVFDKDQNGFISAAELRHVMTNLGEKLTDE 120
Yeast          NHQIEFSEFLALMSRQLKSNDSEQELLEAFKVFDKNGDGLISAAELKHVLTSIGEKLTDA 120
               *  *:*.*** :*::::*..***:*: ***:****: :* ******:**:*.:******

Human          EVDEMIREADIDGDGQVNYEEFVQMMTAK 149
Mouse          EVDEMIREADIDGDGQVNYEEFVQMMTAK 149
Round_Worm     EVDEMIREADIDGDGQVNYEEFVTMMTTK 149
Drosophila     EVDEMIREADIDGDGQVNYEEFVTMMTSK 149
Arabidopsis    EVEEMIREADVDGDGQINYEEFVKIMMAK 149
Yeast          EVDDMLREVS-DGSGEINIQQFAALLSK- 147
               **::*:**.. **.*::* ::*. ::
```

Homologs, Orthologs and Paralogs

- Homologs: A gene related to a second gene by descent from a common ancestral DNA sequence. The term, homolog, may apply to the relationship between genes separated by the event of speciation (see ortholog) or to the relationship betwen genes separated by the event of genetic duplication (see paralog).
- Orthologs: Orthologs are genes in different species that evolved from a common ancestral gene by speciation. Normally, orthologs retain the same function in the course of evolution. Identification of orthologs is critical for reliable prediction of gene function in newly sequenced genomes. (See also Paralogs.).
- Speciation: Speciation is the origin of a new species capable of making a living in a new way from the species from which it arose. As part of this process it has also aquired some barreir to genetic exchage with the parent species.
- Paralogs: Paralogs are genes related by duplication within a genome. Orthologs retain the same function in the course of evolution, whereas paralogs evolve new functions, even if these are related to the original one.
- Conservation: If a sequence remains relatively unchanged across a number of species, it is said to be conserved



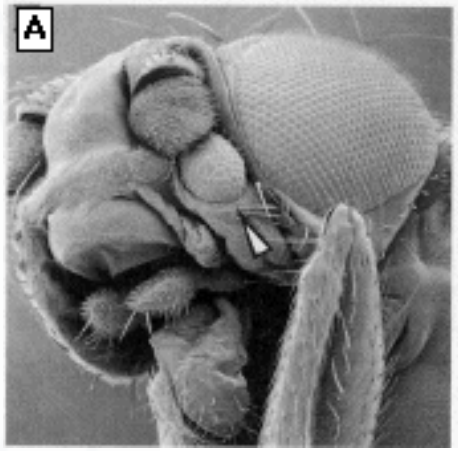
Origin of new species with a barrier to genetic exchange

Genes that evolved from a common ancestral gene by speciation

Genes related by duplication within a genome

Similarity between biological sequences

- Why…
  - Infer the function of a gene
- Example
  - ‘Eyeless’ gene
    - W. Gehring discovered that turning on the “eyeless” gene in drosophila leads to the growth of ectopic eyes
    - “eyeless” is a master control gene for eye formation (transcription factor)
  - The aniridia gene in humans has a sequence that is similar to the drosophila eyeless gene
  - **Eye morphogenesis is under similar genetic control in vertebrates and insects**



Scanning electron micrograph of an ectopic eye (arrowhead) in the head region

PAX6_HUMAN aligned against PAX6_DRO

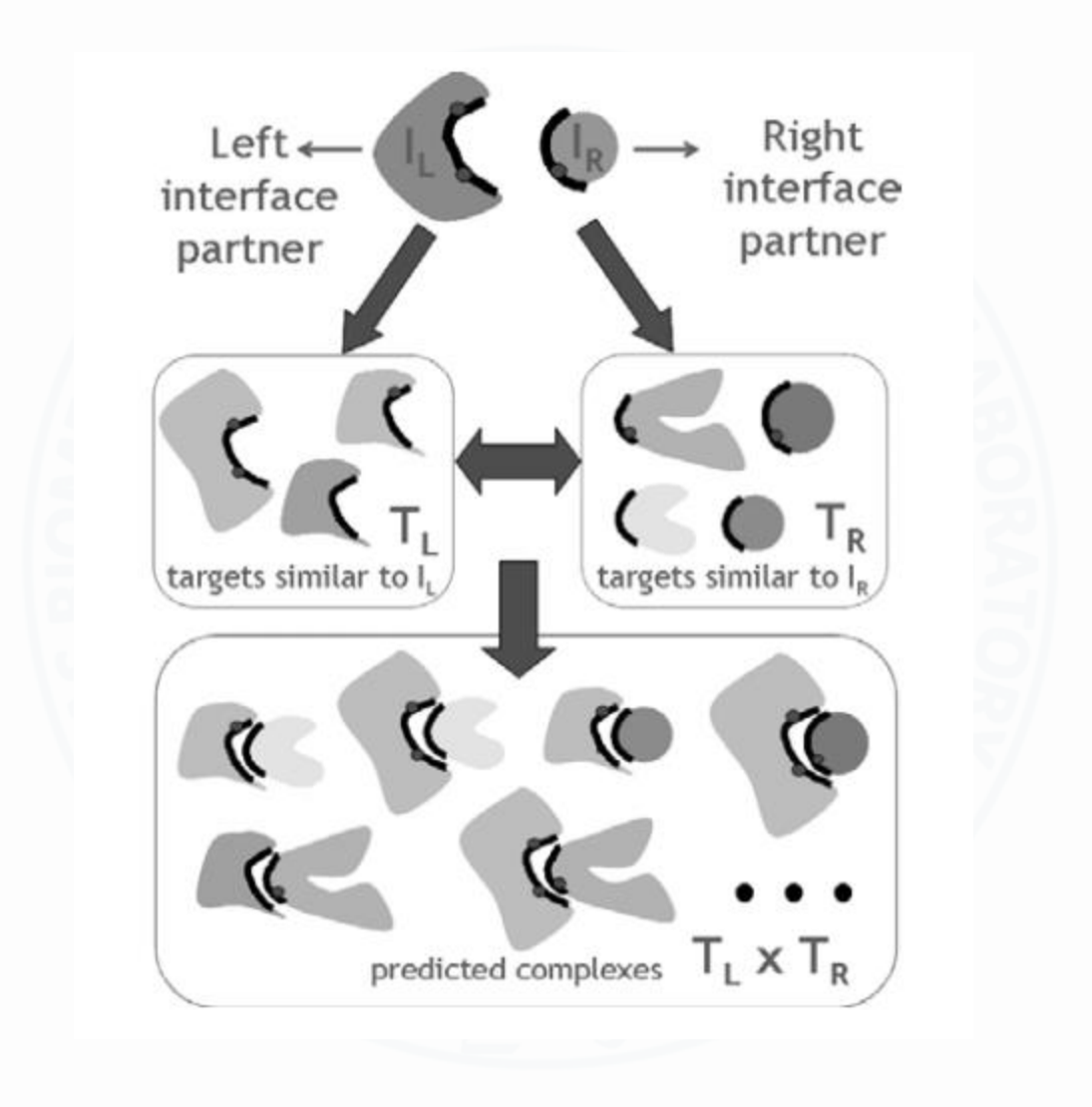
```
HSGVNQLGGVFVNGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVS 54
||||||||||||.||||||||||||||||||||||||||||||||||||
HSGVNQLGGVFVGGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVS 106

KILGRYYETGSIRPRAIGGSKPRVATPEVVSKIAQYKRECPSIFAWEIRD 104
||||||||||||||||||||||||||.|||||:||||||||||||||||
KILGRYYETGSIRPRAIGGSKPRVATAEVVSKISQYKRECPSIFAWEIRD 156

RLLSEGVCTNDNIPSVSSINRVLRNLASEKQQMGA-------------- 139
|||.|.|||||||||||||||||||::|:|...
RLLQENVCTNDNIPSVSSINRVLRNLAAQKEQQSTGSGSSSTSAGNSISA 206

----------SWGTR---PGWYPGTSVPGQPTQ---------------- 174
          ||..|   ..||| ||:...|..
NHQALQQHQQQSWPPRHYSGSWYP-TSLSEIPISSAPNIASVTAYASGPS 355

-----------------------------------DGCQQQE---GGGE 185
                                  ||.|..|   |.||
LAHSLSPPNDIESLASIGHQRNCPVATEDIHLKKELDGHQSDETGSGEGE 405

NTNSISSNGEDSDEAQMRLQLKRKLQRNRTSFTQEQIEALEKEFERTHYP 235
|:|..:||..::::.|.||.|||||||||||||.:||::||||||||||
NSNGGASNIGNTEDDQARLILKRKLQRNRTSFTNDQIDSLEKEFERTHYP 455
```

Sequence Alignments

- Understanding disease mechanisms
- Example:
  - Oncogene ν-sis is a gene found in Simian Sarcoma Virus
    - It causes uncontrolled cell growth and leads to cancer in monkeys infected by this virus
  - In 1983 Russell Doolittle and colleagues found similarities between a cancer-causing gene from the Simian Sarcoma virus and a normal growth factor gene (PDGF)

R. F. Doolittle, M. W. Hunkapiller, L. E. Hood, S. G. Devare, K. C. Robbins, S. A. Aaronson, and H. N. Antoniades. Simian sarcoma virus oncogene, ν -sis, is derived from the gene (or genes) encoding a platelet-derived growth factor. Science, 221:275–277, 1983.

Interface prediction



Prediction of protein–protein interactions by combining structure and sequence conservation in protein interfaces
by A. Selim Aytuna, Attila Gursoy and Ozlem Keskin in Bioinformatics, Vol. 21 no. 12 2005, pages 2850–2855, 2005.

Similarity between biological sequences

- Why…
  - Genome Assembly

Similarity between biological sequences

- Why …
  - Find if two genes are related
  - Find if two species are related
  - Construct the phylogenetic tree
- Example

WNV NS3 vs. DENV4 NS3

```
DENV-WNV
alignment done in 48.73 seconds
sequence 1: Dengue4_Aedes : 618
SGALWDVPSPAATQKAALSEGVYRIMQRGLFGKTQVGVGIHMEGVFHTMWHVTRGSVICHETGRLEPSWADVRNDMISYGGGWRLGDKWDKEEDVQVLAIEPGKNPKH
VQTKPGLFKTLTGEIGAVTLDFKPGTSGSPIINRKGKVIGLYGNGVVTKSGDYVSAITQAERIGEP---DYEVDEDIFRKKRLTIMDLHPGAGKTKRILPSIVREALK
RRLRTLILAPTRVVAAEMEEALRGLPIRYQTPAVKSEHTGREIVDLMCHATFTTRLLSSTRVPNYNLIVMDEAHFTDPSSVAARGYISTRVEMGEAAAIFMTATPPGA
TDPFPQSNSPIEDIEREIPERSWNTGFDWITDYQGKTVWFVPSIKAGNDIANCLRKSGKKVIQLSRKTFDTEYPKTKLTDWDFVVTTDISEMGANFRAGRVIDPRRCL
KPVILTDGPERVILAGPIPVTPASAAQRRGRIGRNPAQEDDQYVFSGDPLKNDEDHAHWTEAKMLLDNIYTPEGIIPTLFGPEREKTQAIDGEFRLRGEQRKTFVELM
RRGDLPVWLSYKVASAGISYKDREWCFTGERNNQILEENMEVEIWTREGEKKKLRPKWLDARVYADPMALKDFKEFASGRK
sequence 2: West_Nile_Culex : 619
GGVLWDTPSPKEYKKGDTTTGVYRIMTRGLLGSYQAGAGVMVEGVFHTLWHTTKGAALMSGEGRLDPYWGSVKEDRLCYGGPWKLQHKWNGHDEVQMIVVEPGKNVKN
VQTKPGVFKTPEGEIGAVTLDYPTGTSGSPIVDKNGDVIGLYGNGVIMPNGSYISAIVQGERMEEPAPAGFE--PEMLRKKQITVLDLHPGAGKTRKILPQIIKEAIN
KRLRTAVLAPTRVVAAEMSEALRGLPIRYQTSAVHREHSGNEIVDVMCHATLTHRLMSPHRVPNYNLFIMDEAHFTDPASIAARGYIATKVELGEAAAIFMTATPPGT
SDPFPESNAPISDMQTEIPDRAWNTGYEWITEYVGKTVWFVPSVKMGNEIALCLQRAGKKVIQLNRKSYETEYPKCKNDDWDFVITTDISEMGANFKASRVIDSRKSV
KPTIIEEGDGRVILGEPSAITAASAAQRRGRIGRNPSQVGDEYCYGGHTNEDDSNFAHWTEARIMLDNINMPNGLVAQLYQPEREKVYTMDGEYRLRGEERKNFLEFL
RTADLPVWLAYKVAAAGISYHDRKWCFDGPRTNTILEDNNEVEVITKLGERKILRPRWADARVYSDHQALKSFKDFASGKR
alignment score: 2180.7
```

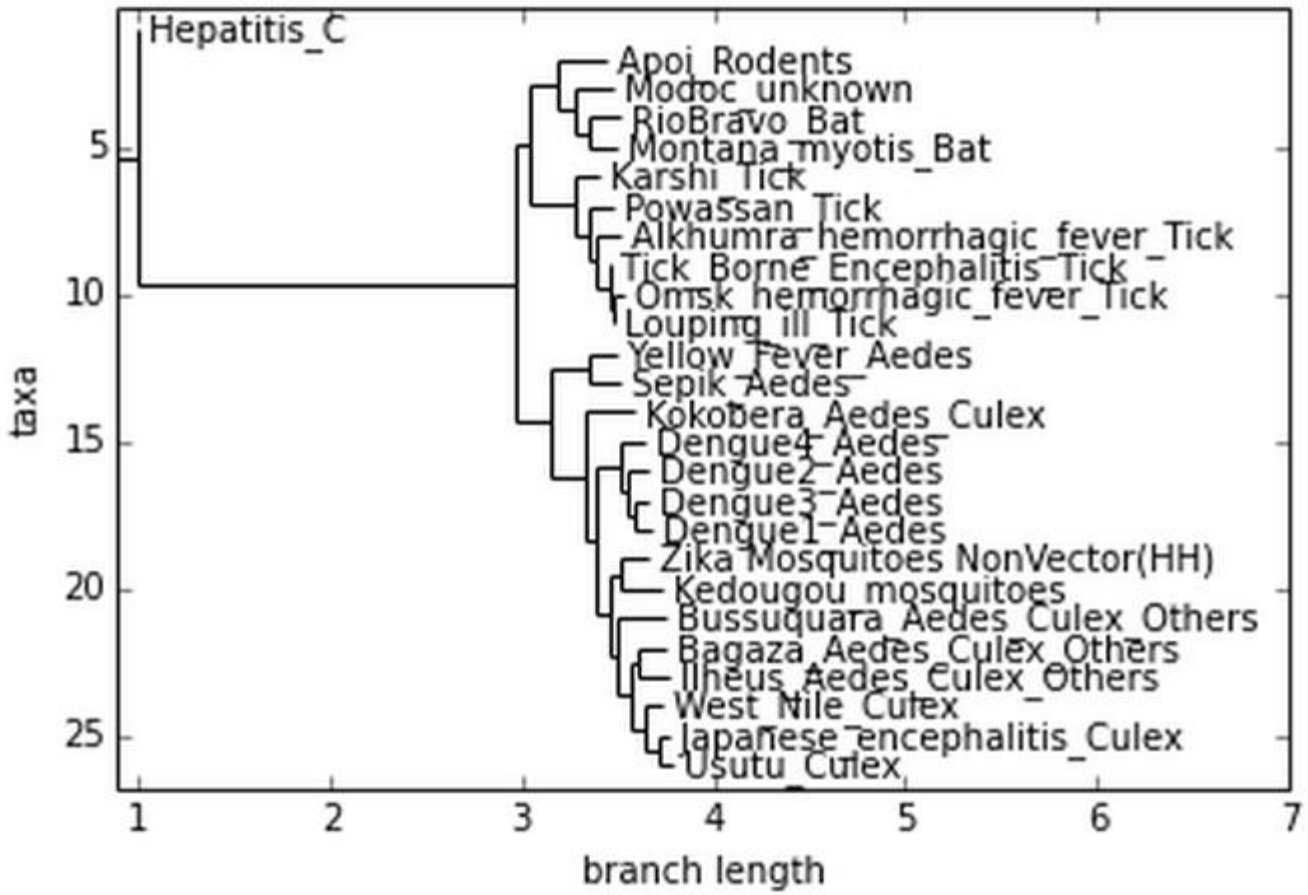
HEPC NS3 vs. DENV4 NS3

```
DENV-HEPC
alignment done in 55.22 seconds
sequence 1: Dengue4_Aedes : 618
SGALWDVPSPAATQKAALSEGVYRIMQRGLFG----------KTQVGVGIHM-------------EGVFHTMWHVTRGSVICHETGRLEPSWADVRNDMISYGGGW--
------------------------------RLGDKWDKEEDVQVLAIEPGK----NPKHVQTKPGLFKTLTGEIGAVTLDFKPGTSGSPIINRKGKVIGLYGNGVVT-
---KSGDYVS--------------------AITQAERIGEPDYEVDEDIFRKKRLTIMDLHP--GAGKTKRILPSIVREALKRRLRTLILAPTRVVAAEM--------
----EEALR--------GLPIRYQTPAVKSEHTGREIVDLMCHATFTTRLLSSTRVPNYNLIVMDEAHFTDPSSVAARGYISTRVEM-GEAAAIFMTATPPGA-TDPF
PQSNSPIEDIER----EIPERSWNTGFDWITDYQGKTVWFVP--SIKAGNDIANCLRKSGKKVIQLSRK--------------------------TFDTEYPKTKLT
-DWDFVV--------TTDISEMGANFRAGRVIDPRRCLKPVILTDGPERVILAGPIPVTPASAAQRRGRIGRNPAQEDDQYVFSGDP---------------------
------------------------------LKNDEDHAH-WTEAKMLLDNIYTPEGIIPTLFGPEREKTQAIDGEF----RLRGEQRKTFVELMRRGDLPVWLSYKV-
----ASAGISYKDREW-CFT---------------GERNNQI-------------LEENMEVEIWTREGEKKKLRPKWLDARVYADPMALKDFKEFASGRK
sequence 2: Hepatitis_C : 631
-----------------APITAYAQQTRGLLGCIITSLTGRDKNQVEGEVQIVSTAAQTFLATCINGVCWTVYHGAGTRTIASSKGPVIQMYTNVDQDLV----GWPA
PQGARSLTPCTCGSSDLYLVTRHADVIPVRRRGD--------------GRGSLLSPRPIS-------------------YLKGSSGGPLLCPAGHAVGIFRAAVCTR
GVAKAVDFIPVEGLETTMRSPVFSDNSSPPAVPQSYQVAH-------------------LHAPTGSGKSTKVPAAYAAQGYK----VLVLNPS--VAATLGFGAYMSK
AHGIDPNIRTGVRTITTGSPITYST-------YGKFLADGGCSGGA------------YDIIICDECHSTDATSILGIGTVLDQAETAGARLTVLATATPPGSVTVPH
PN----IEEVALSTTGEIP-------------FYGKAI---PLEAIKGGRHLIFCHSK--KKCDELAAKLVALGVNAVAYYRGLDVSVIPASGDVVVVATDALMTGFT
GDFDSVIDCNTCVTQTVDFS-----------LDPTFTIETTTL-------------PQDAVSRTQRRGRTGR------------GKPGIYRFVTPGERPSGMFDSSVL
CECYDAGCAWYELTPAETTVRLRAYMNTPGLPVCQDHLEFW-------------EGVFTGL--------THIDAHFLSQTKQSGE------------NLPYLVAYQAT
VCARAQAPPPSWDQMWKCLIRLKPTLHGPTPLLYRLGAVQNEITLTHPITKYIMTCMSADLEVVT-----------------------------------
alignment score: 284.8
```

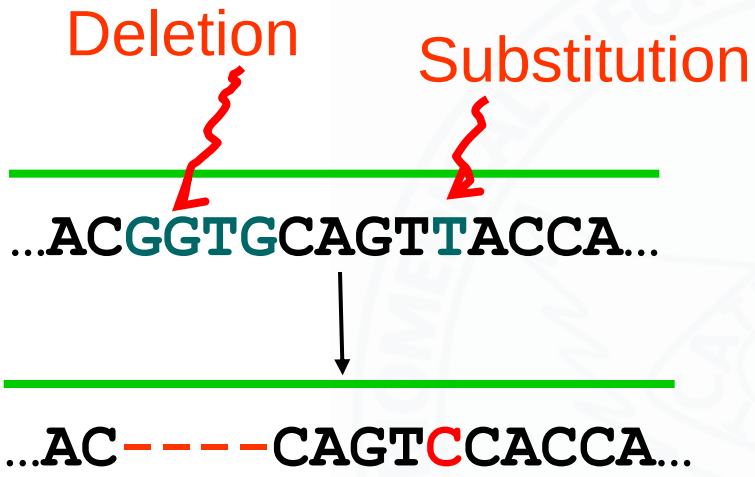
Constructing Phylogenetic Tree using Flaviviridae NS3

```
-GGVLWDTPSPKEYK-KGDTTTGVYRIMTRGLLG-SYQAGAGVM...--- West_Nile_Culex
--GVFWDTPSPKPCS-KGDTTTGVYRIMARGILG-TYQAGVGVM...R-- Japanese_encephalitis_Culex
-SGAIWDVPAPKEKK-RAELSTGVFRIMARGILG-KYQAGVGVM...--- Bagaza_Aedes_Culex_Others
SNTSFSLITDILESDDHESPNEGIYRIFASGVLG-KKQVGVGIW...--- Apoi_Rodents
-SGALWDVPSPAATQ-KAALSEGVYRIMQRGLFG-KTQVGVGIH...--- Dengue4_Aedes
--GVMWDVPAPKQFG-KTELKPGVYRVMTMGILG-RYQSGVGVM...--- Ilheus_Aedes_Culex_Others
-SGVLWDVPSPPETQ-KAELEEGVYRIKQQGILG-KTQVGVGVQ...--- Dengue3_Aedes
APITAYAQQTRGLLGCIITSLTGRDKNQVEGEVQIVSTAAQTFL...--- Hepatitis_C
-SGVLWDTPSPPEVE-KAVLDDGIYRILQRGLLG-RSQVGVGVF...--- Dengue1_Aedes
-GDVLWDIPTPKIIEECEHLEDGIYGIFQSTFLG-ASQRGVGVA...--- Yellow_Fever_Aedes
SDLVFSGQSGSERGSQPFEVRDGVYRILSPGLLWGHRQVGVGFG...--- Omsk_hemorrhagic_fever_Tick
-SGALWDVPAPKEVK-KGETTDGVYRVMTRRLLG-STQVGVGVM...--- Zika_Mosquitoes_NonVector(HH)
-SGVLWDIPTVVPPEEVGYLEDGVYTINQKNVLG-MAQKGVGVV...--- Sepik_Aedes
-AGALWDIPSPREAK-PAKVEDGVYRIFSRKLFG-ESQIGAGVM...SA- Kokobera_Aedes_Culex
-SGALWDMPAPRPAP-PAVLGDGVYRIMSKKLLG-PSQLGVGVM...--- Kedougou_mosquitoes
SLIVYGGRSESANDETEGDLLEGVYRITAASIFG-RKQVGIGVL...--- Modoc_unknown
--SIIVFGVGEDPEGEKVSIQDGVYRIMVSSLFG-KKQVGVGIW...--- RioBravo_Bat
SDLVYSGQGGQERGDRPFEVKDGVYRIFSPGLFWGQRQVGVGYG...--- Louping_ill_Tick
...
-AGVLWDVPSPPPMG-KAELEDGAYRIKQKGILG-YSQIGAGVY...--- Dengue2_Aedes
```

Flaviviridae Family Tree

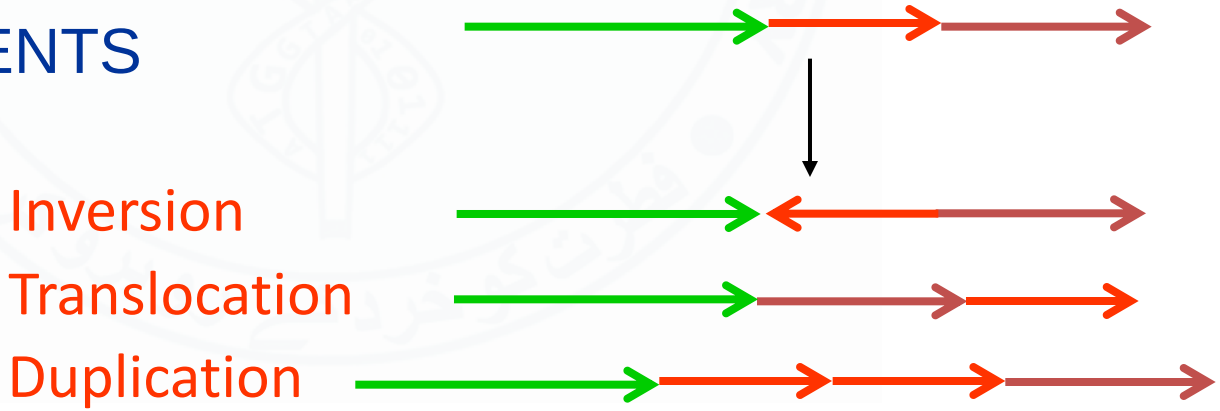


Mutations at the DNA level



SEQUENCE EDITS

REARRANGEMENTS



Sequence alignment

```
AGGCTATCACCTGACCTCCAGGCCGATGCCC
TAGCTATCACGACCGCGGTCGATTTGCCCGAC

-AGGCTATCACCTGACCTCCAGGCCGA--TGCCC---
TAG-CTATCAC--GACCGC--GGTCGATTTGCCCGAC
```

Definition

Given two strings $\mathbf{v} = v_1v_2...v_m$, $\mathbf{w} = w_1w_2...w_n$,

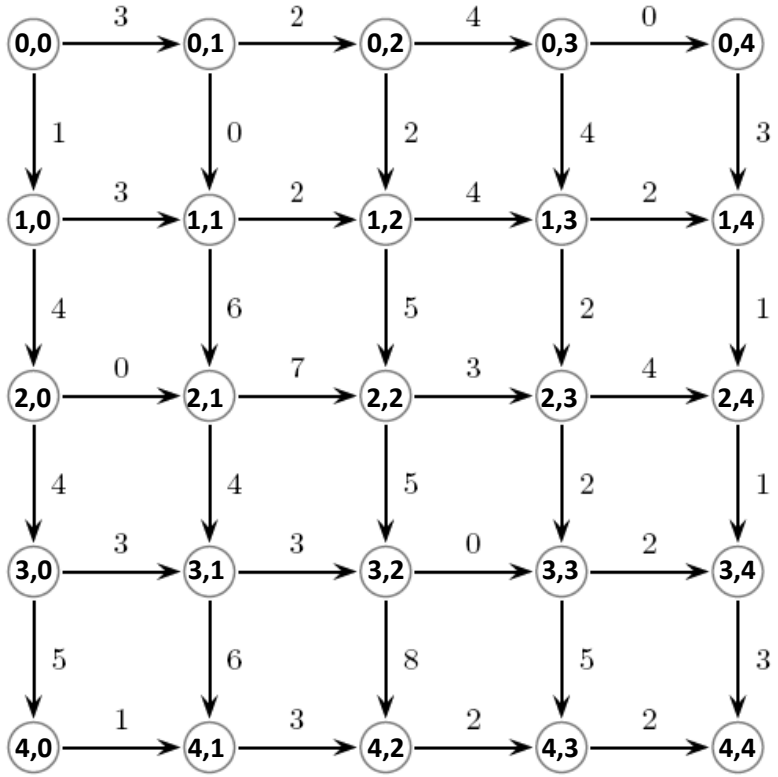
an alignment is an assignment of gaps to positions 0,…,m in **v**, and 0,…,n in **w**, so as to line up each letter in one sequence with either a letter, or a gap in the other sequence

So how do we do alignments?

- An alignment can be represented as an optimization problem with the objective of maximizing a score that represents the similarity of the two sequences after alignment

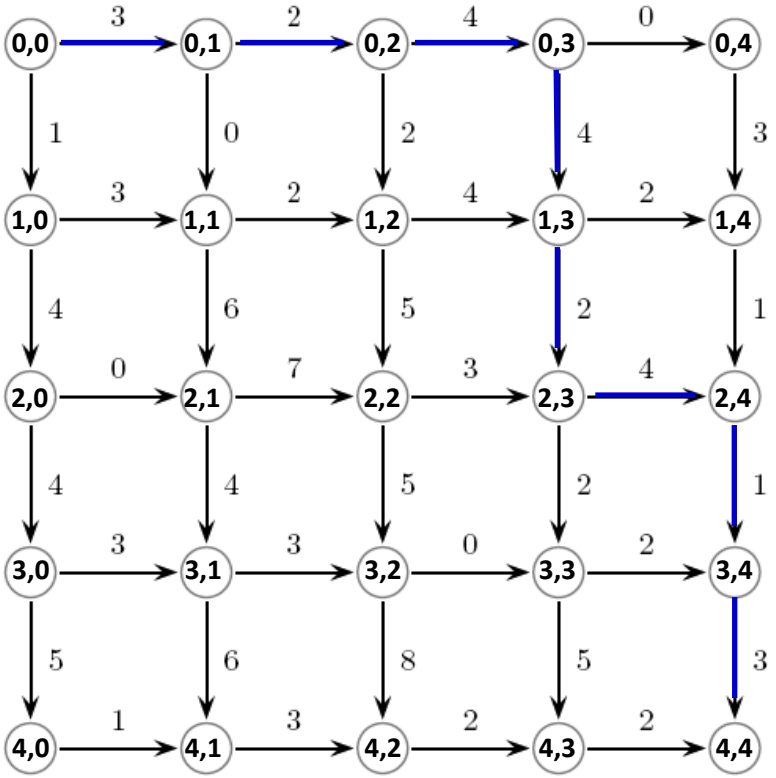
How do we solve such optimization problems?

- Analogous Example
  - Move from start (0,0) to end (4,4) while making the most money
  - On each edge, there is a certain amount of money to be made indicated by weights on that edge
  - You can only move in the directions indicated

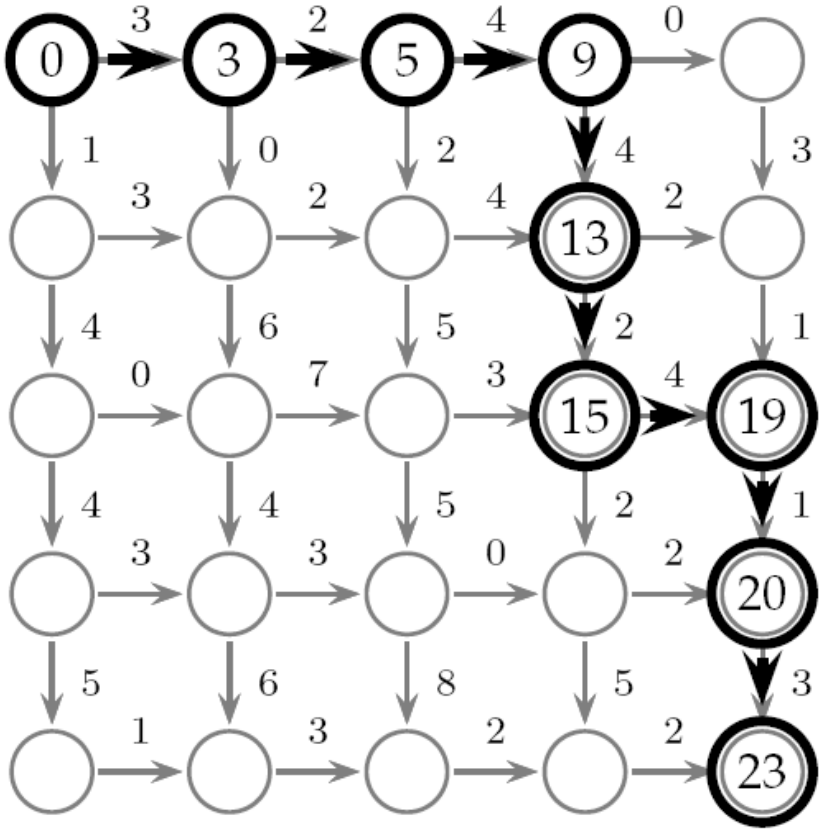


Making the most money: Greedy Solution

- Start at (0,0)
- At each node
  - Choose the edge with maximum money
- This solution gives:
  - 3+2+4+4+2+4+1+3 = 23
- Is this optimal?

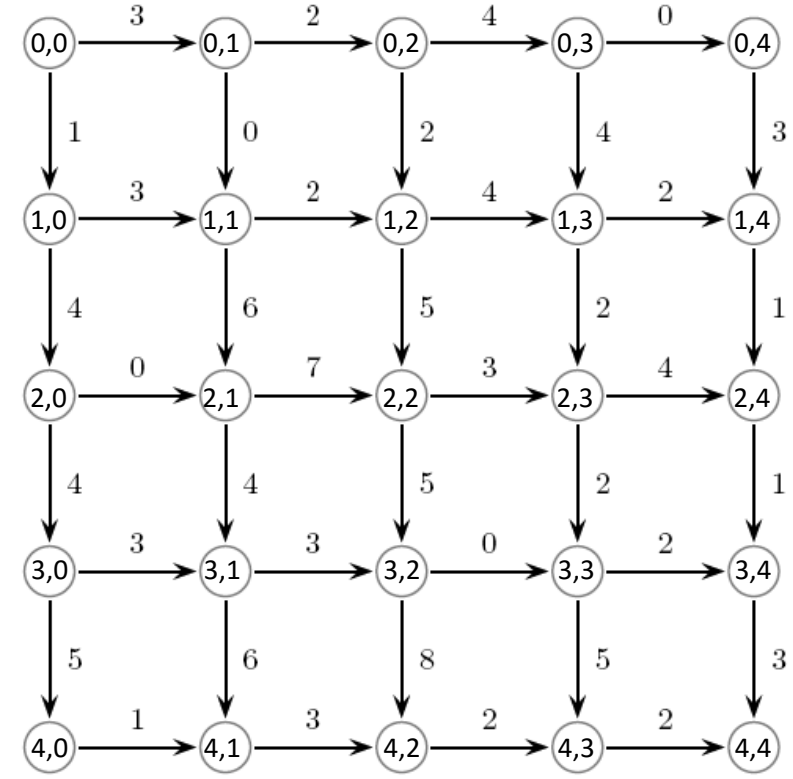


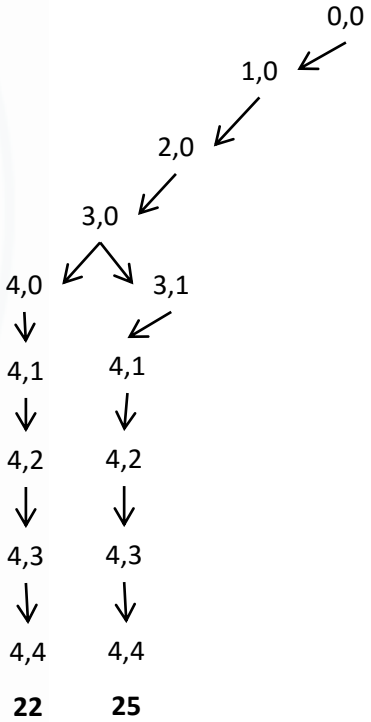
The greedy solution



Making the most money: Exhaustive Depth First

- We can also search exhaustively
  - Find the costs of all paths using a depth first traversal
  - Choose the path with least cost
- Number of paths grows exponentially with the number of nodes

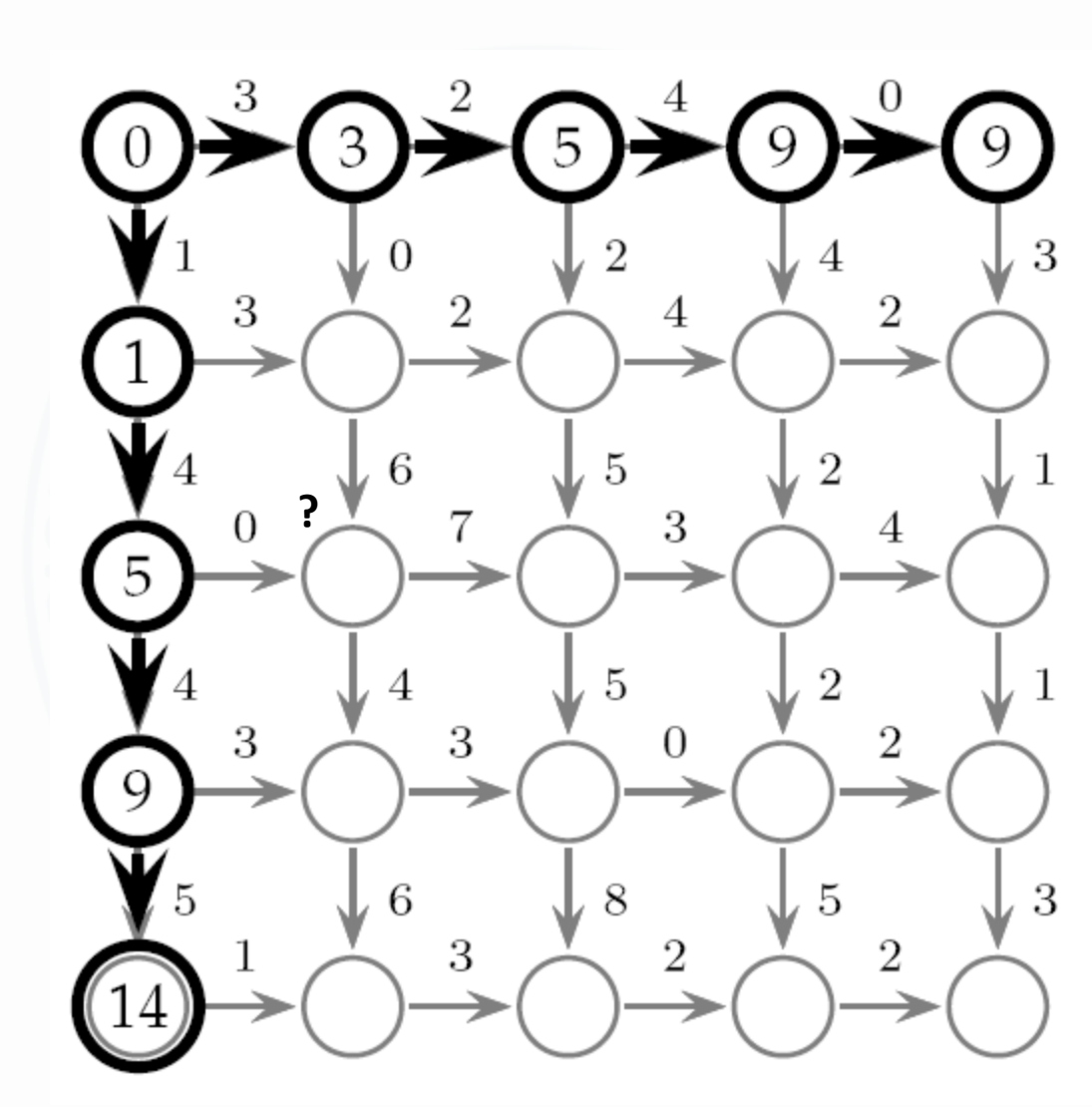




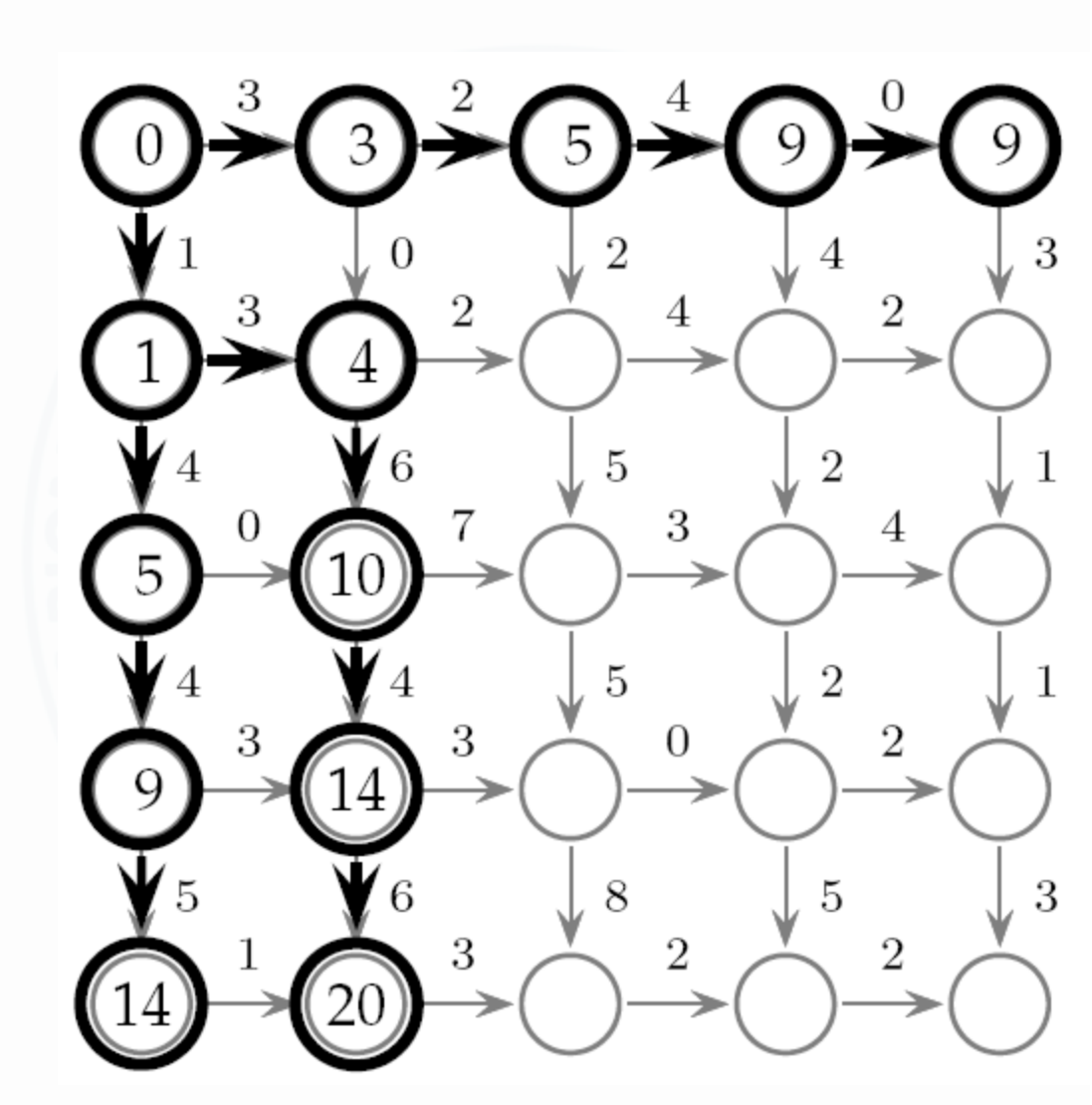
Making the most money: Dynamic Programming

- Break down the problem into sub-problems
  - Solve a general problem of finding the best path from the source to any node (not just the sink node!)
  - Instead of directly solved the source to the sink solution, we now find m x n solutions!
  - But knowing the solution to a sub-problem allows us to solve bigger problems that include that sub-problem!

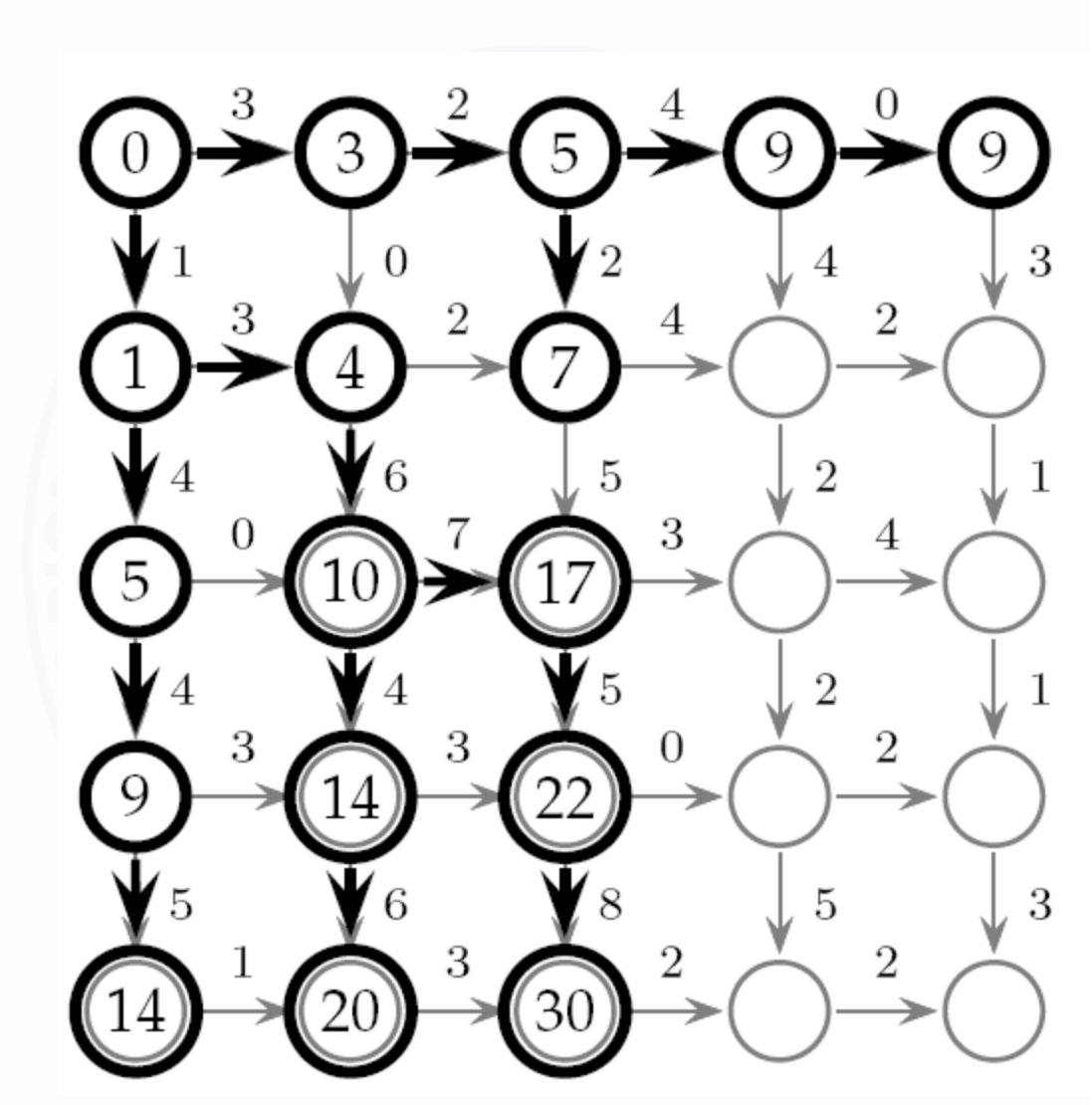
Starting off



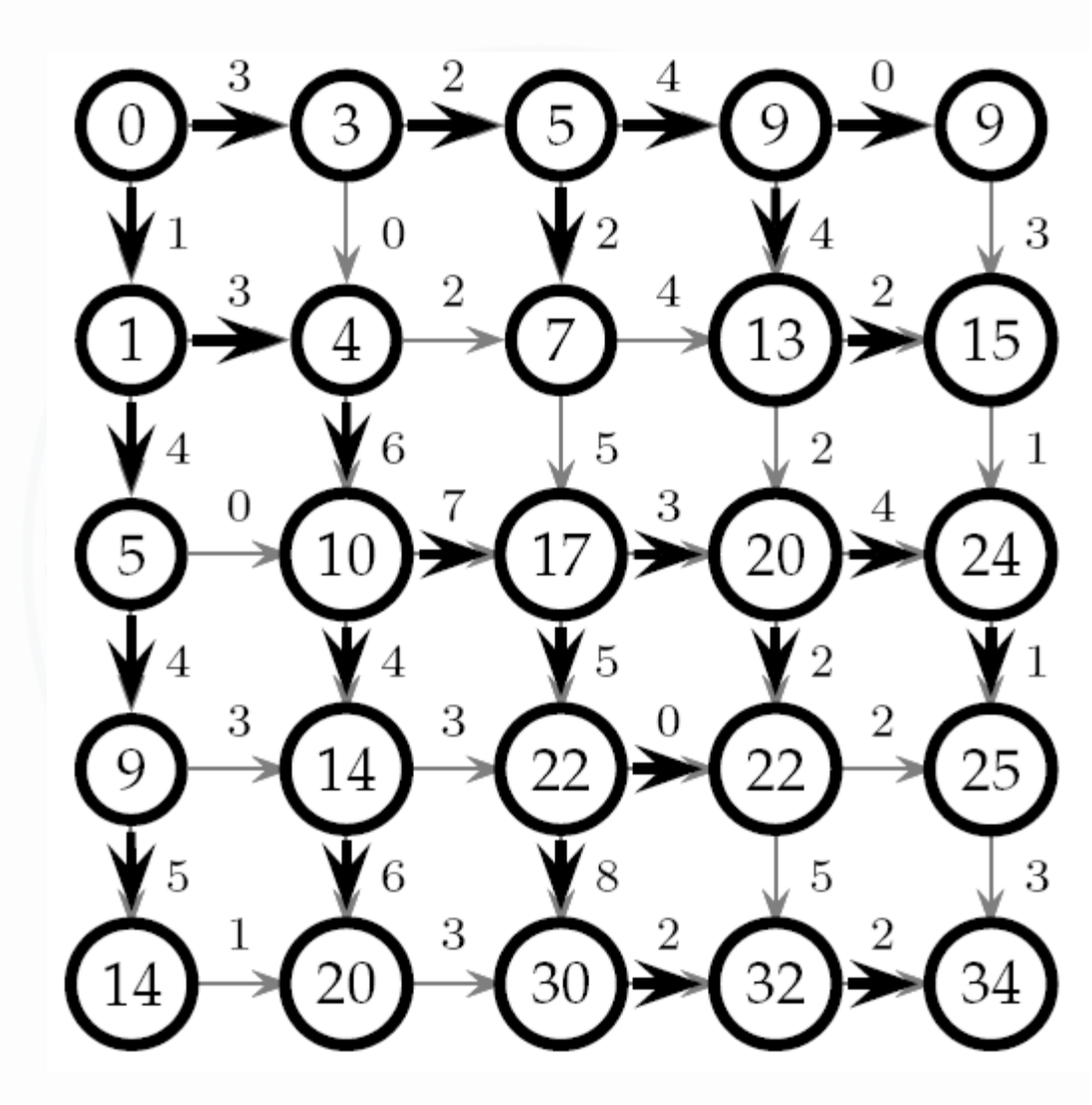
In the 2nd column



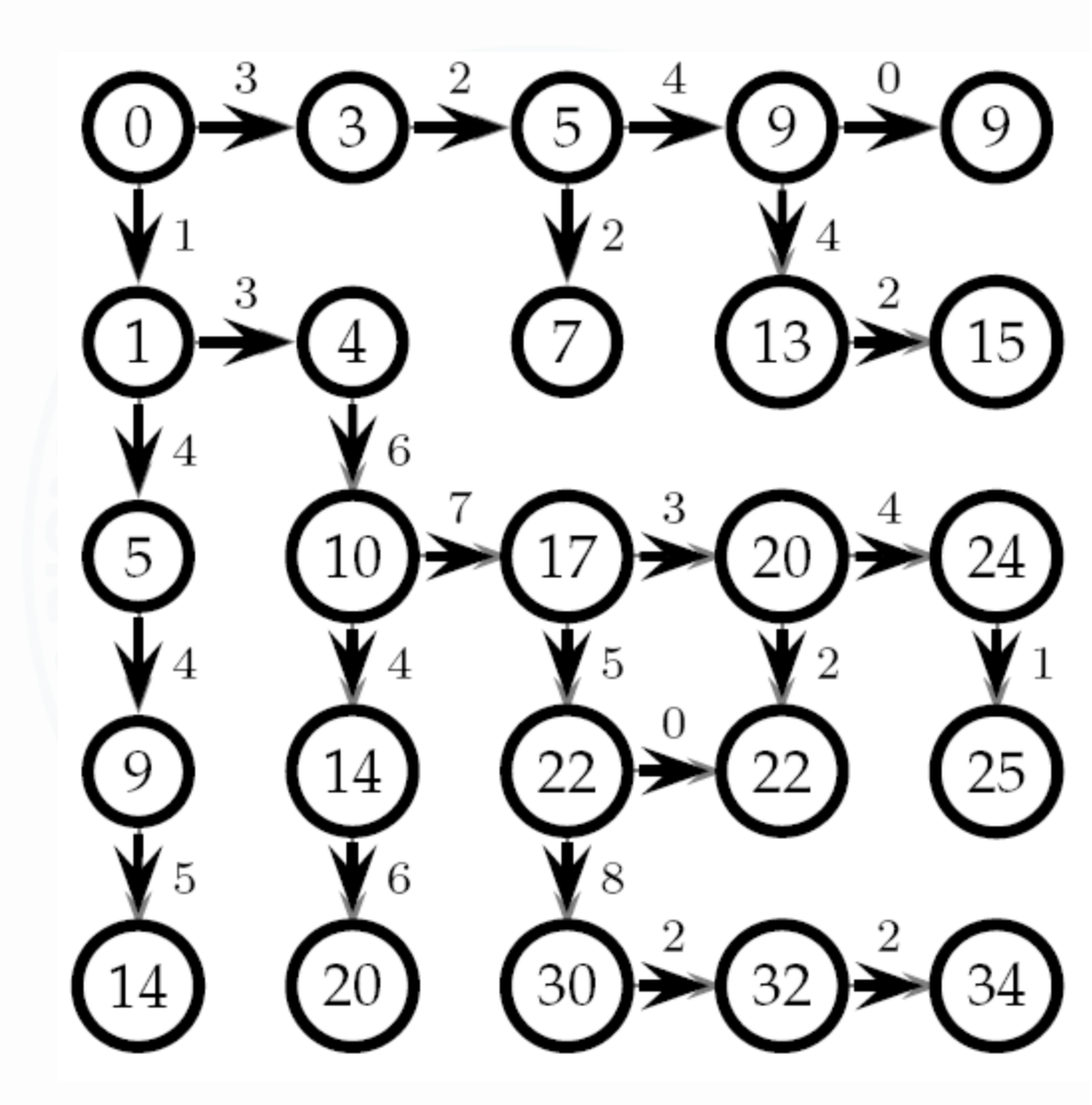
3rd column


0 3 5 9 9
3 2 4 0
1 0 2 4 3
1 4 7
3 2 4 2
4 6 5 2 1
5 10 17
0 7 3 4
4 4 5 2 1
9 14 22
3 3 0 2
5 6 8 5 3
14 20 30
1 3 2 2

Final


0 3 2 4 0
3 5 9 9
1 0 2 4 3
1 3 4 2 7 4 13 2 15
4 6 5 2 1
5 0 10 7 17 3 20 4 24
4 4 5 2 1
9 3 14 3 22 0 22 2 25
5 6 8 5 3
14 1 20 3 30 2 32 2 34

Solution


0 3 3 2 5 4 9 0 9
1 2 4
1 3 4 7 13 2 15
4 6
5 10 7 17 3 20 4 24
4 4 5 2 1
9 14 22 0 22 25
5 6 8
14 20 30 2 32 2 34

Logic

$$s_{i,j} = \max \begin{cases} s_{i-1,j} + \text{ weight of the edge between } (i-1,j) \text{ and } (i,j) \\ s_{i,j-1} + \text{ weight of the edge between } (i,j-1) \text{ and } (i,j) \end{cases}$$

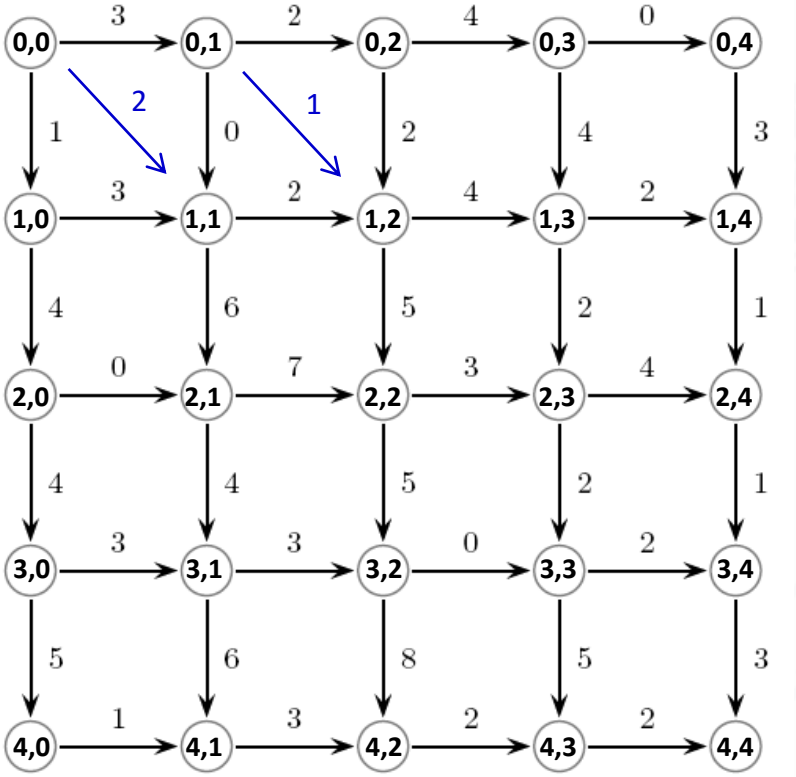
MaximumMoney $(\downarrow \mathbf{w}, \rightarrow \mathbf{w}, n, m)$

1 $s_{0,0} \leftarrow 0$
2 **for** $i \leftarrow 1$ **to** n
3 $\quad s_{i,0} \leftarrow s_{i-1,0} + \downarrow w_{i,0}$
4 **for** $j \leftarrow 1$ **to** m
5 $\quad s_{0,j} \leftarrow s_{0,j-1} + \rightarrow w_{0,j}$
6 **for** $i \leftarrow 1$ **to** n
7 $\quad$ **for** $j \leftarrow 1$ **to** m
8 $\quad\quad s_{i,j} \leftarrow \max \begin{cases} s_{i-1,j} + \downarrow w_{i,j} \\ s_{i,j-1} + \rightarrow w_{i,j} \end{cases}$
9 **return** $s_{n,m}$

Stores money made while arriving at (i,j) from West

Stores money made while arriving at (i,j) from North

What if?


0,0
0,1
0,2
0,3
0,4
1,0
1,1
1,2
1,3
1,4
2,0
2,1
2,2
2,3
2,4
3,0
3,1
3,2
3,3
3,4
4,0
4,1
4,2
4,3
4,4
3 2 4 0
1 0 2 4 3
3 2 4 2
4 6 5 2 1
0 7 3 4
4 4 5 2 1
3 3 0 2
5 6 8 5 3
1 3 2 2
2
1

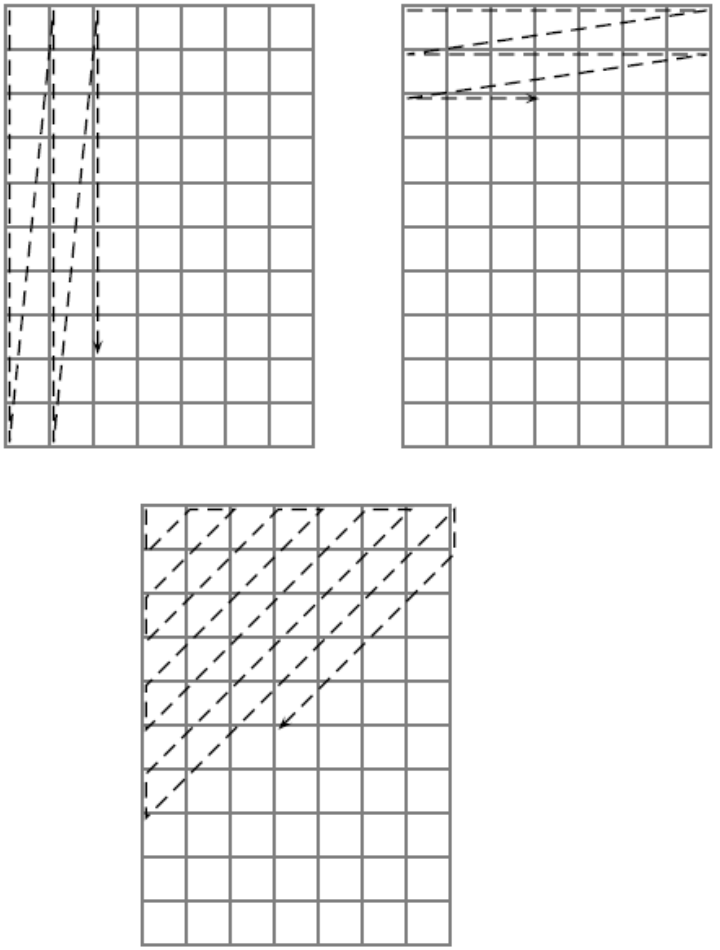
General Solution

- For finding the maximum value path in any directed acyclic graph using Dynamic Programming

$$s_v = \max_{u \in Predecessors(v)} (s_u + \text{ weight of edge from } u \text{ to } v)$$

- Cost of solution:
  - Time:
    - O(mn)
  - Memory:
    - O(mn)

General Solution

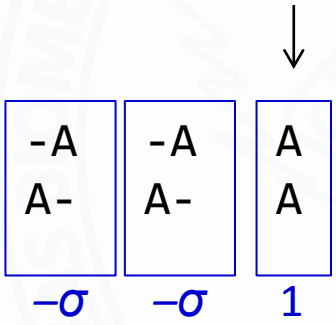


Reading

- Section 6.3 in Jones & Pevzner 2004

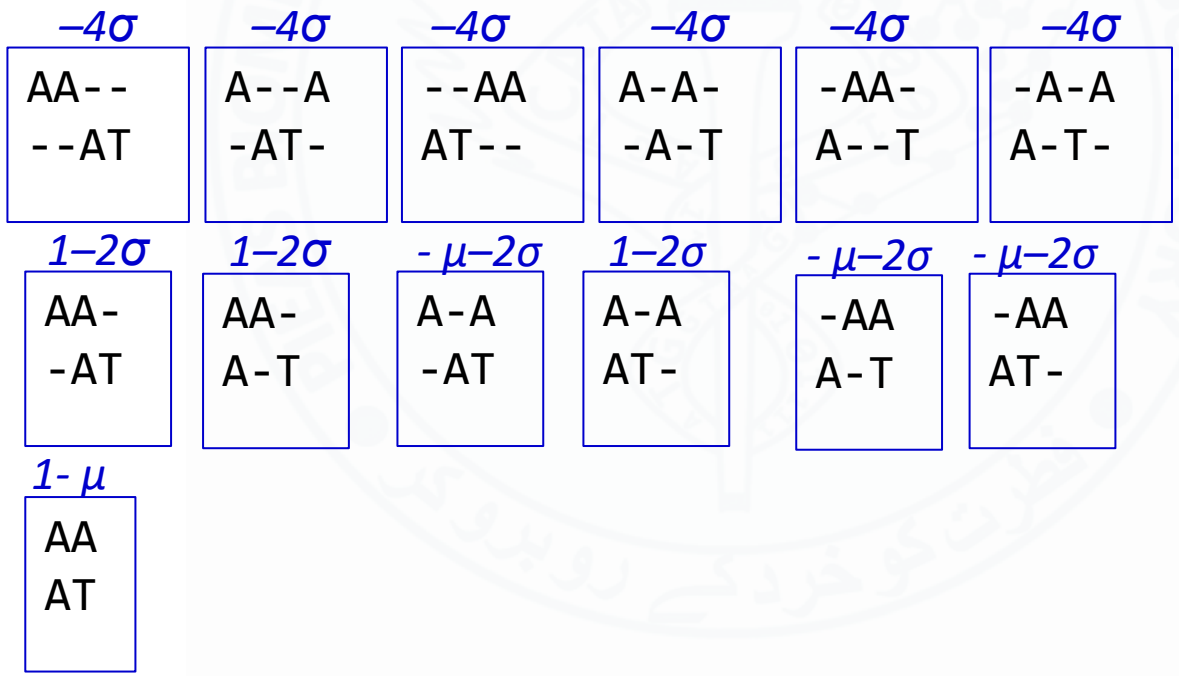
Back to alignments

- What are the possible alignments of the following strings:
  - s_1 = A
  - s_2 = A
- Answer:



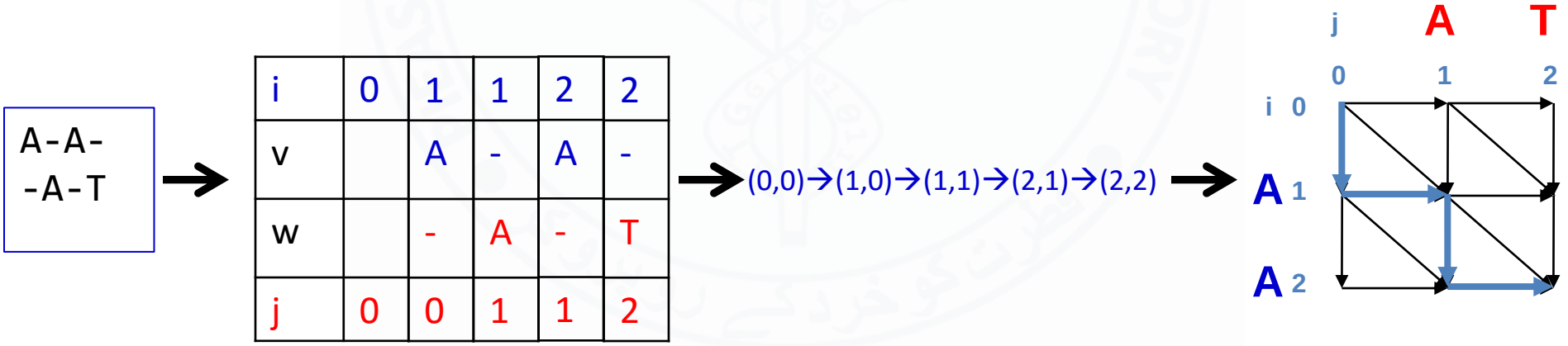
- Scoring: Let's
  - Penalize mismatches by $-\mu$
  - Penalize indels (match to gaps) by $-\sigma$,
  - Reward matches with *+1*

- What are the possible alignments of the following strings:
  - v = AA
  - w = AT



Alignment as a path in 2D grid

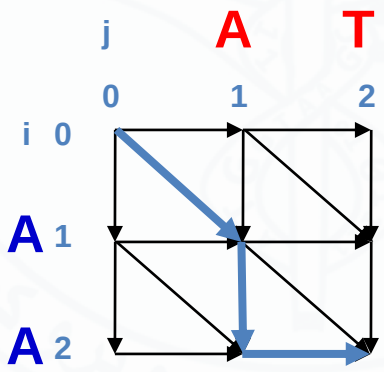
- We can represent an alignment as a path on a 2D grid as follows
  - Start at 0
  - Whenever a character from a string is used increment its coordinate otherwise keep the previous
- In case of alignments these graphs are called edit graphs



Alignment as a path in 2D grid

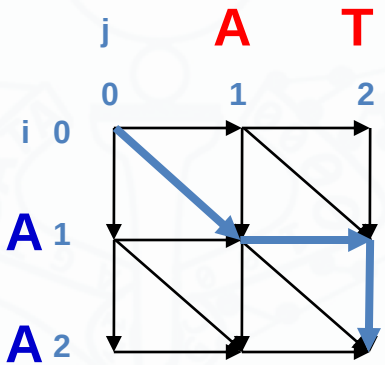
- Represent the following alignment

AA-
A-T

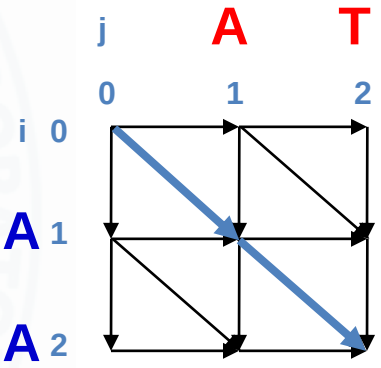


Alignments from paths

- What's the corresponding alignment for the path shown
- Note
  - ↓ and → Correspond to matches to gaps (indels)
  - Diagnoal arrows correspond to matches or mismatches



AA-
A-T

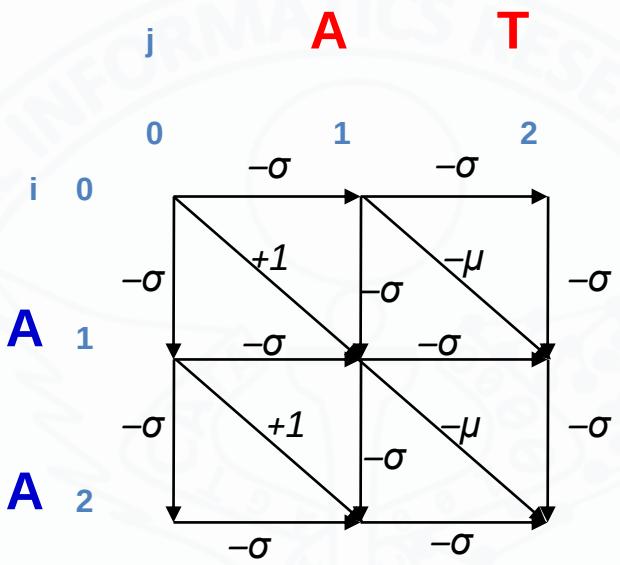


AA
AT

Assigning values to the paths

- Let's assign values to paths:
  - What should be the value of any $\longrightarrow$ or $\downarrow$?
    - *These are matches to gaps*
      - $-\sigma$
  - What should be the value of diagonals?
    - *These correspond to matches or mismatches*
      - *If match: +1*
      - *If mismatch: $-\mu$*

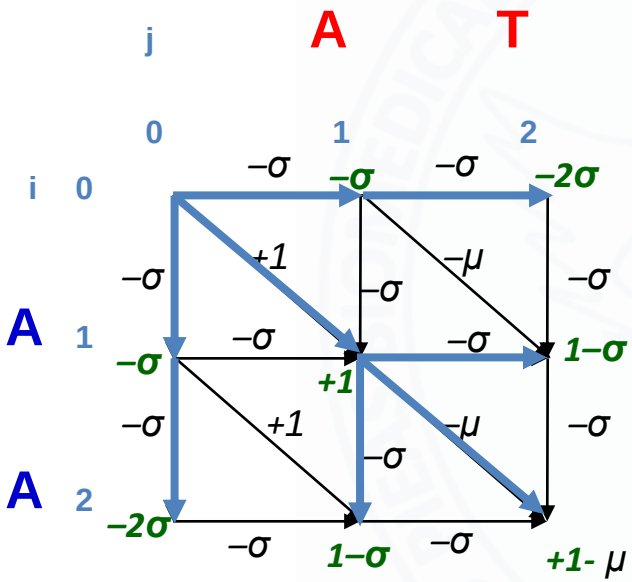
Complete DAG for alignments

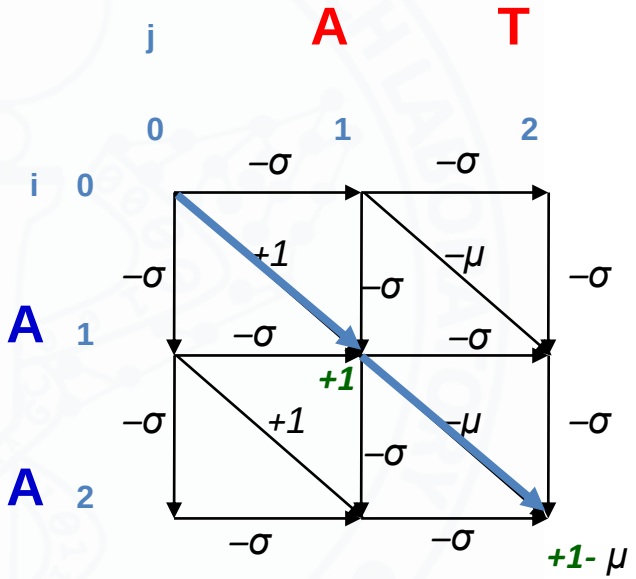


- The alignment problem is then to find the path of the maximum value in this DAG
  - Use Dynamic programming!

Let's solve it!

- Assume: $\sigma = 0.5, \mu = 0.5$





AA
AT

Needleman-Wunsch

- This algorithm is called Needleman-Wunsch Algorithm!

- Needleman, Saul B.; and Wunsch, Christian D. (1970). "A general method applicable to the search for similarities in the amino acid sequence of two proteins". Journal of Molecular Biology 48 (3): 443–53. doi:10.1016/0022-2836(70)90057-4. PMID 5420325.

Pseudo code

```
NWAlign(v,w)
    m ← length(v)
    n ← length(m)
    for i ← 0 to m
        s_{i,0} ← 0
    for j ← 0 to N
        s_{0,j} ← 0
    for i ← 1 to m
        for j ← 1 to n
```

$$s_{i,j} \leftarrow max \begin{cases} s_{i-1,j} - \sigma \\ s_{i,j-1} - \sigma \\ s_{i-1,j-1} + 1 \quad if v_i = w_j \\ s_{i-1,j-1} - \mu \quad if\ v_i \neq w_j \end{cases}$$

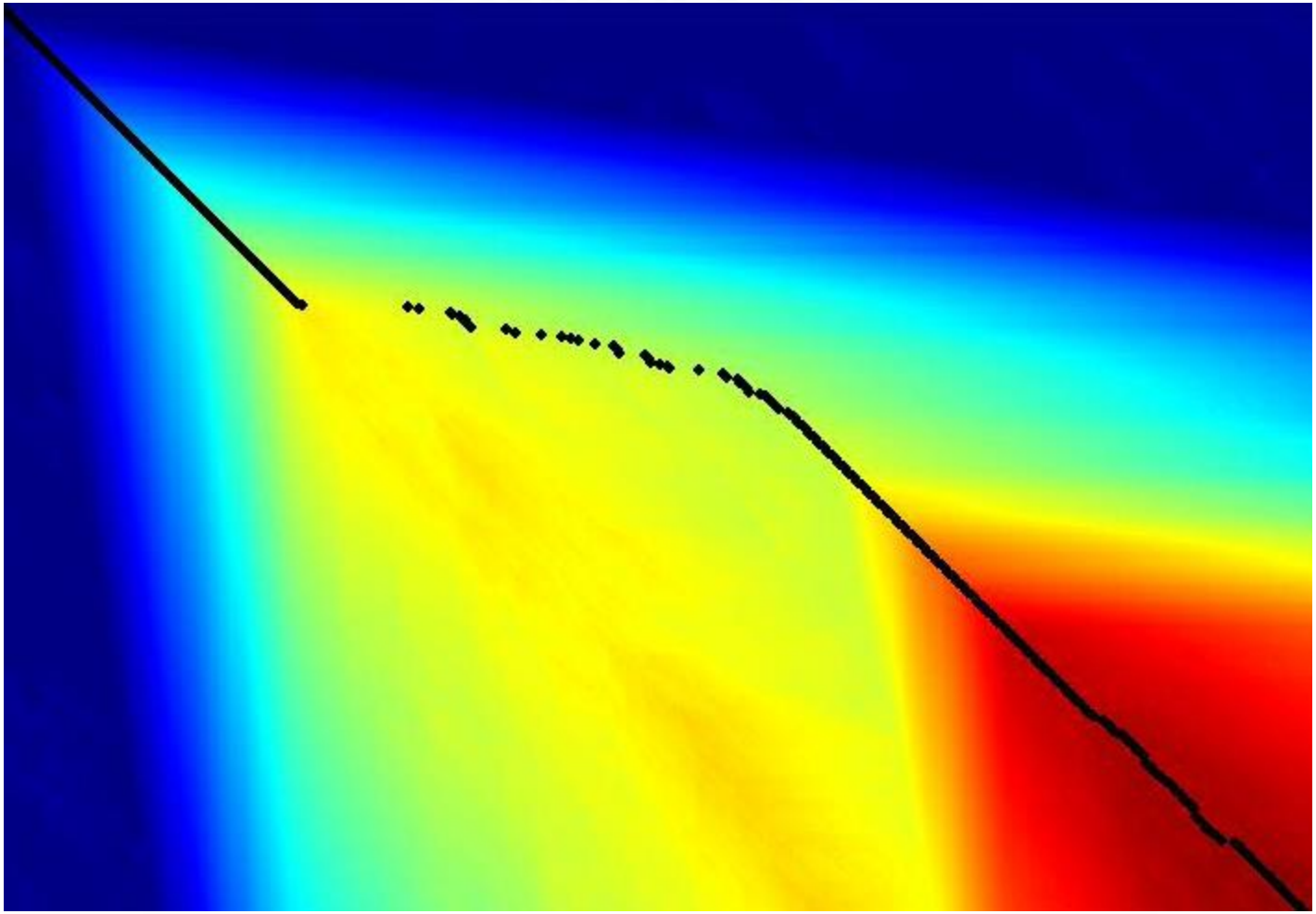
$$b_{i,j} \leftarrow \begin{cases} \downarrow & if s_{i,j} = s_{i-1,j} - \sigma \\ \rightarrow & if s_{i,j} = s_{i,j-1} - \sigma \\ \searrow & if s_{i,j} = s_{i-1,j-1} + 1 \\ \searrow & if s_{i,j} = s_{i-1,j-1} - \mu \end{cases}$$

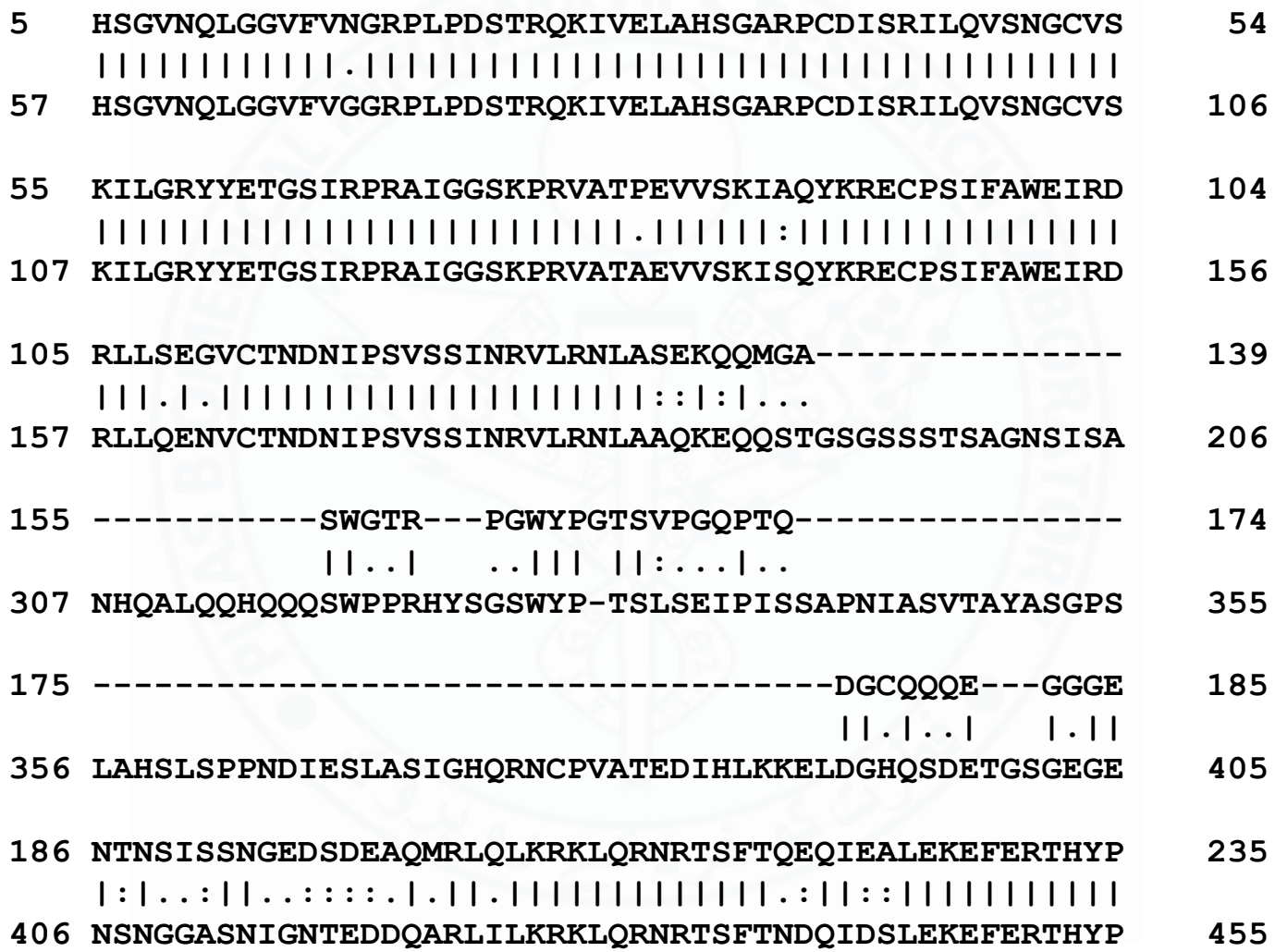
```
    Return s_{m,n}, b
```

```
PrintAlign(b,v,w)
    A_v ← ""
    A_w ← ""
    i ← length(A)
    j ← length(B)
    while (i > 0 or j > 0)
        if (i > 0 and j > 0 and b_{i,j} == ↘
            A_v ← v_i + A_v
            A_w ← w_j + A_w
            i ← i - 1
            j ← j - 1
        else if i > 0 and b_{i,j} == ↓
            A_v ← v_i + A_v
            A_w ← "-" + A_w
            i ← i - 1
        else if j > 0 and b_{i,j} == →
            A_v ← "-" + A_v
            A_w ← w_j + A_w
            j ← j – 1
    Return A_v , A_w
```

$s_{m,n}$ stores the alignment score, O (mn) time, O (mn) space

Scores Matrix




5 HSGVNQLGGVFVNGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVS 54
57 HSGVNQLGGVFVGGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVS 106
55 KILGRYYETGSIRPRAIGGSKPRVATPEVVSKIAQYKRECPSIFAWEIRD 104
107 KILGRYYETGSIRPRAIGGSKPRVATAEVVSKISQYKRECPSIFAWEIRD 156
105 RLLSEGVCTNDNIPSVSSINRVLRNLASEKQQMGA 139
157 RLLQENVCTNDNIPSVSSINRVLRNLAAQKEQQSTGSGSSSTSAGNSISA 206
155 SWGTR---PGWYPGTSVPGQPTQ 174
307 NHQALQQHQQQSWPPRHYSGSWYP-TSLSEIPISSAPNIASVTAYASGPS 355
175 DGCQQQE---GGGE 185
356 LAHSLSPPNDIESLASIGHQRNCPVATEDIHLKKELDGHQSDETGSGEGE 405
186 NTNSISSNGEDSDEAQMRLQLKRKLQRNRTSFTQEQIEALEKEFERTHYP 235
406 NSNGGASNIGNTEDDQARLILKRKLQRNRTSFTNDQIDSLEKEFERTHYP 455

Alignment scores

- This algorithm scores alignments as:
  - #matches − μ · #mismatches − σ · #indels
- For an alphabet of size k
  - We can use a unified square matrix $\boldsymbol{\delta}(x, y)$ of size (k+1)x(k+1) called the substitution matrix
  - Example: The previous scoring scheme can be written as follows
  - Protein matrices: More on how they are obtained later

	A	C	G	T	-
A	1	-μ	-μ	-μ	$-\sigma$
C		1	-μ	-μ	$-\sigma$
G			1	-μ	$-\sigma$
T				1	$-\sigma$
-					∞

$$s_{i,j} \leftarrow max \begin{cases} s_{i-1,j} + \boldsymbol{\delta}(v_i, -) \\ s_{i,j-1} + \boldsymbol{\delta}(-, w_j) \\ s_{i-1,j-1} + \boldsymbol{\delta}(v_i, w_j) \end{cases}$$

Scoring protein alignments: Blosum62

	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
Ala	4																			
Arg	-1	5																		
Asn	-2	0	6																	
Asp	-2	-2	1	6																
Cys	0	-3	-3	-3	9															
Gln	-1	1	0	0	-3	5														
Glu	-1	0	0	2	-4	2	5													
Gly	0	-2	0	-1	-3	-2	-2	6												
His	-2	0	1	-1	-3	0	0	-2	8											
Ile	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

Scoring protein alignments: PAM

Table 1 – The log odds matrix for 250 PAMs (multiplied by 10)

	A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y
A	2	-2	0	0	-4	1	-1	-1	-1	-2	-1	0	1	0	-2	1	1	0	-6	-3
C		12	-5	-5	-4	-3	-3	-2	-5	-6	-5	-4	-3	-5	-4	0	-2	-2	-8	0
D			4	3	-6	1	1	-2	0	-4	-3	2	-1	2	-1	0	0	-2	-7	-4
E				4	-5	0	1	-2	0	-3	-2	1	-1	2	-1	0	0	-2	-7	-4
F					9	-5	-2	1	-5	2	0	-4	-5	-5	-4	-3	-3	-1	0	7
G						5	-2	-3	-2	-4	-3	0	-1	-1	-3	1	0	-1	-7	-5
H							6	-2	0	-2	-2	2	0	3	2	-1	-1	-2	-3	0
I								5	-2	2	2	-2	-2	-2	-2	-1	0	4	-5	-1
K									5	-3	0	1	-1	1	3	0	0	-2	-3	-4
L										6	4	-3	-3	-2	-3	-3	-2	2	-2	-1
M											6	-2	-2	-1	0	-2	-1	2	-4	-2
N												2	-1	1	0	1	0	-2	-4	-2
P													6	0	0	1	0	-1	-6	-5
Q														4	1	-1	-1	-2	-5	-4
R															6	0	-1	-2	2	-4
S																2	1	-1	-2	-3
T																	3	0	-5	-3
V																		4	-6	-2
W																			17	0
Y																				10

Global & Local Alignments

- Needleman-Wunsch performs Global alignments
  - It tries to align the entire sequences
- What if we have two sequences such that we expect only a part of one sequence to occur in another?
  - Local alignments

Local vs. Global Alignment (cont'd)

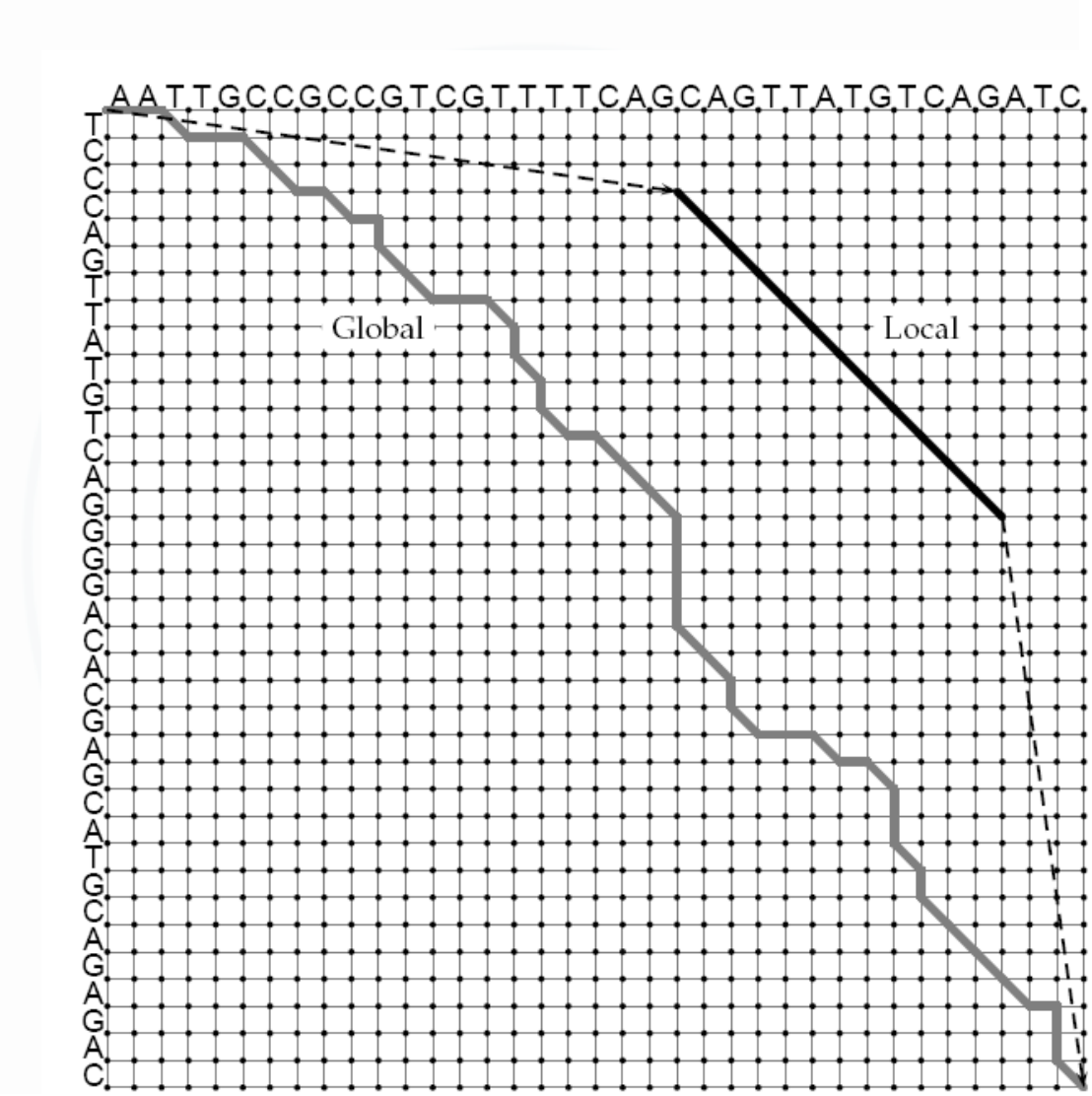
Global Alignment

```
--T--CC-C-AGT--TATGT-CAGGGGACACG-A-GCATGCAGA-GAC
  |  || |  ||  | | | |||    || | | |  | ||||   |
AATTGCCGCC-GTCGT-T-TTCAG----CA-GTTATG-T-CAGAT--C
```

Local Alignment—better for finding a conserved segment

```
gccgccgtcgttttcagCAGTTATGTCAGatc
                |||||||||||||
                     tccCAGTTATGTCAGgggacacgagcatgcagagac
```

Alignment paths for local and global alignments

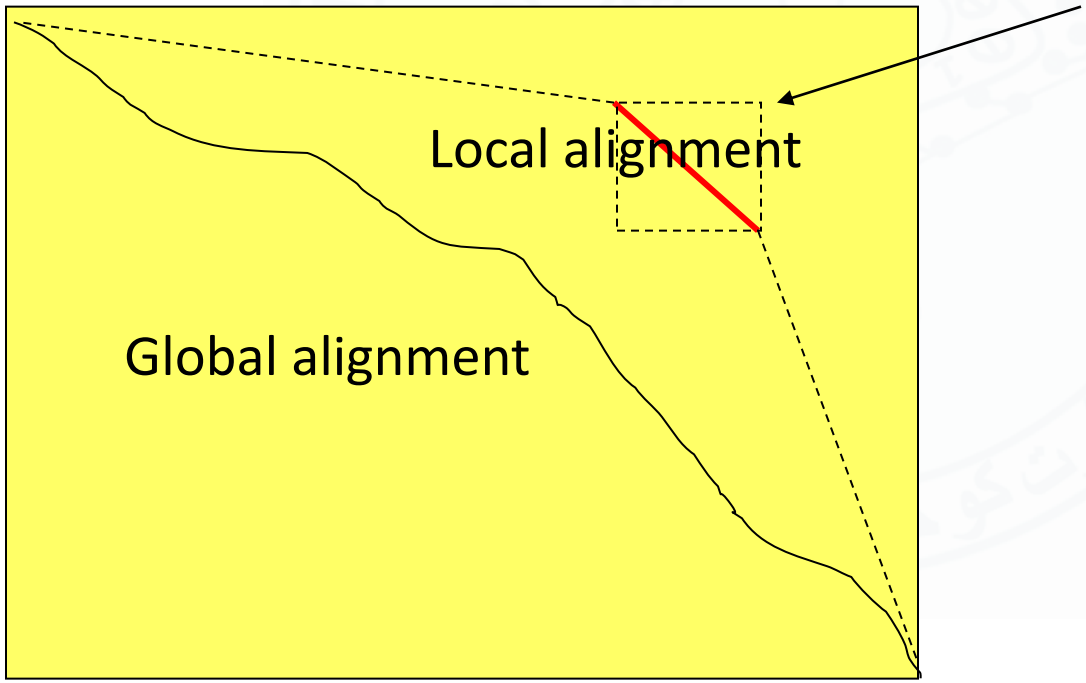


Alignments as paths in the edit graph

- The Global Alignment Problem tries to find the longest path between vertices *(0,0)* and (*n,m*) in the edit graph.
- The Local Alignment Problem tries to find the longest path among paths between arbitrary vertices (*i,j*) and (*i', j'*) in the edit graph.

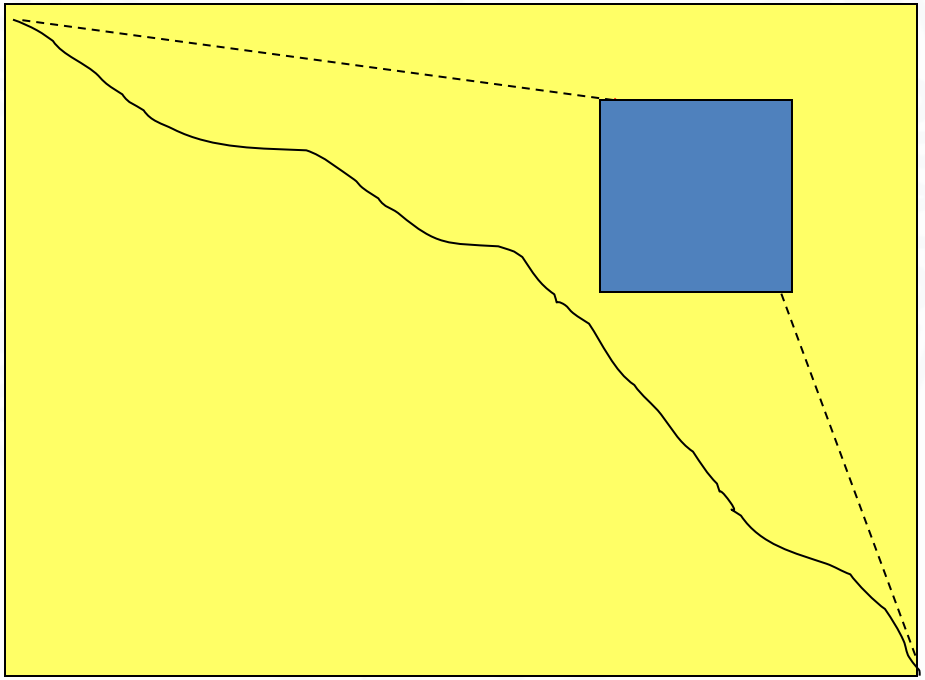
Local alignments using a global aligner

- Choose a starting and end position
- Do a global alignment between the two
- This will take $O(m^2n^2)$ time

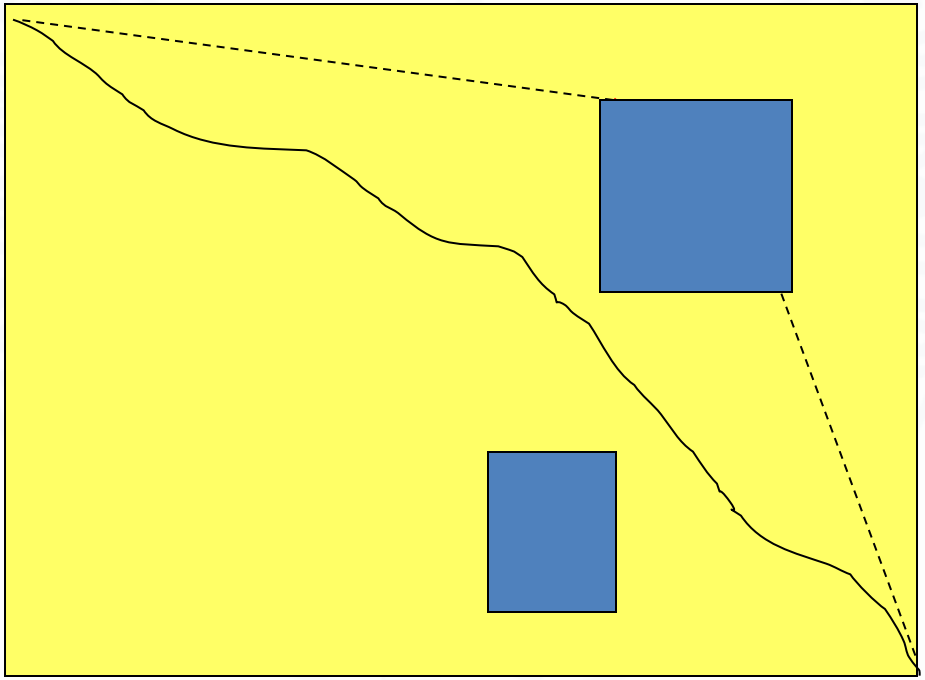


Compute a "mini" global alignment to get a local alignment

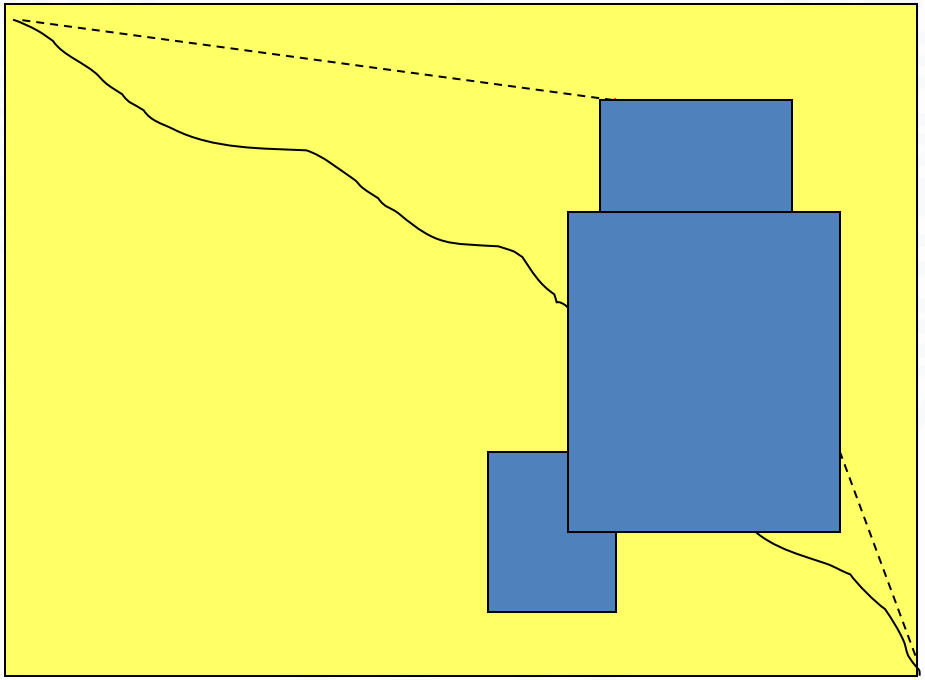
Local Alignment: Example



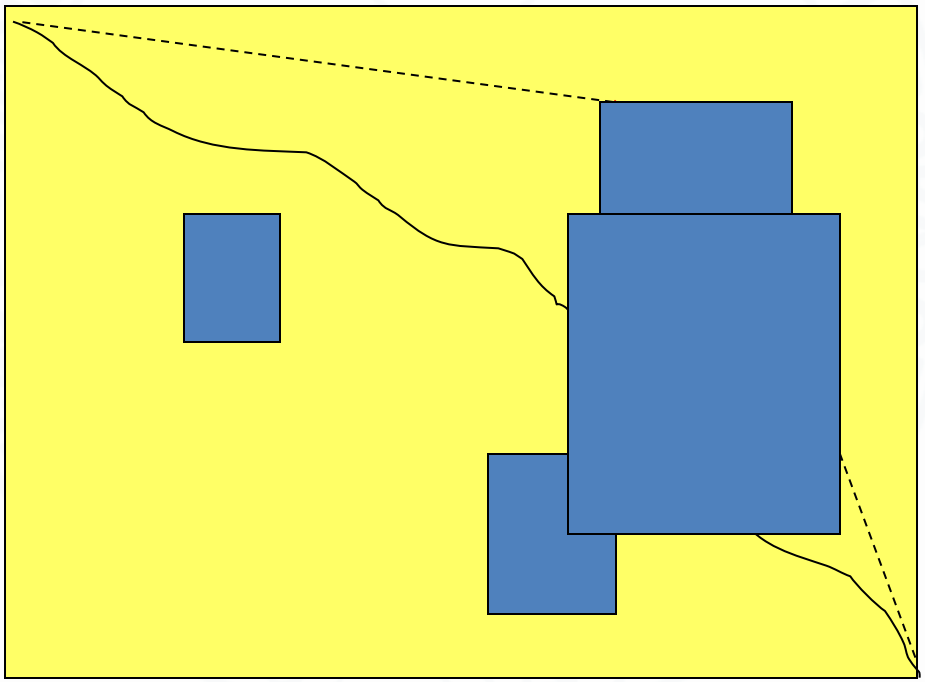
Local Alignment: Example



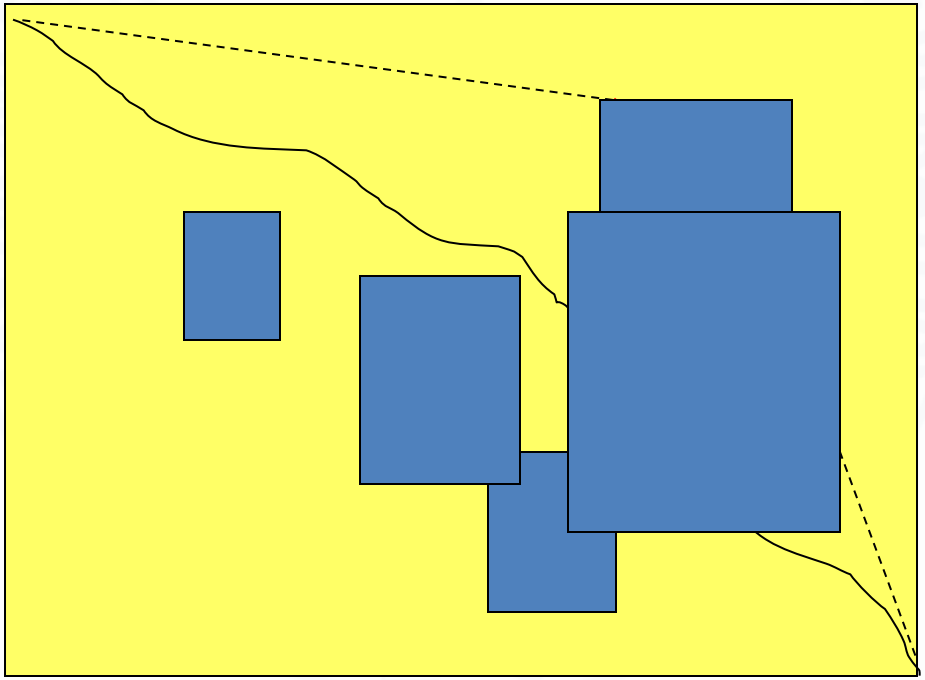
Local Alignment: Example



Local Alignment: Example

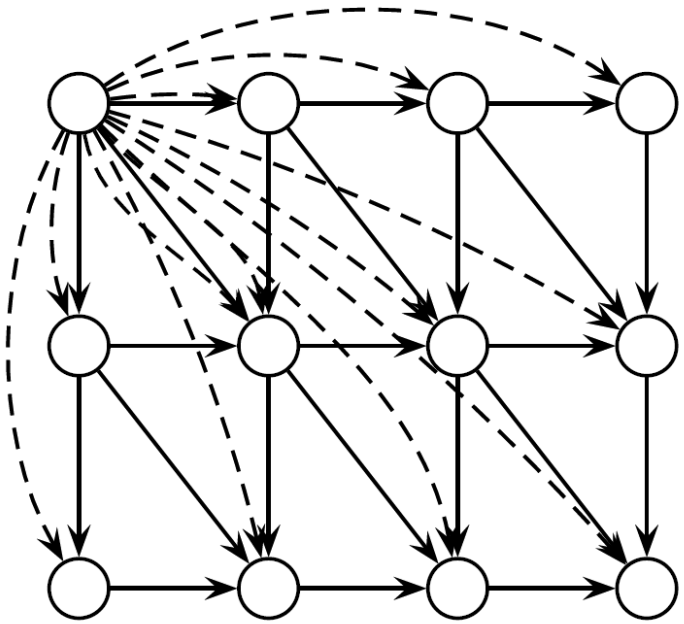


Local Alignment: Example



Free rides

The dashed edges represent the free rides from (0,0) to every other node.



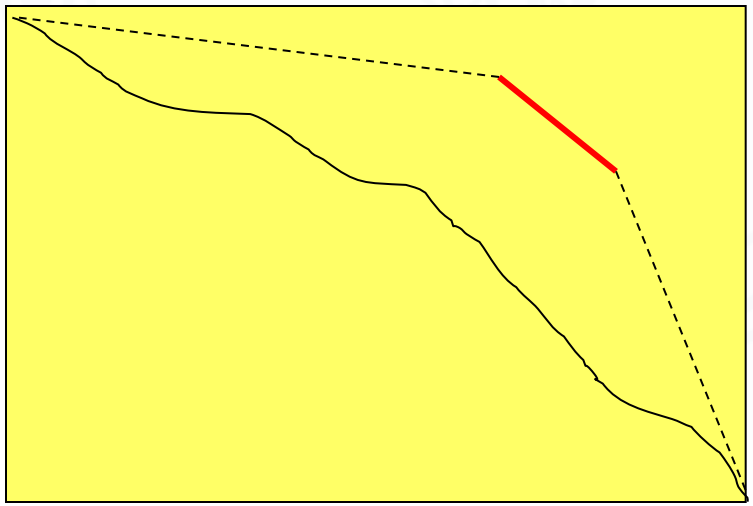
Implementation of free rides

$$s_{i,j} \leftarrow max \begin{cases} 0 \\ s_{i-1,j} + \boldsymbol{\delta}(v_i, -) \\ s_{i,j-1} + \boldsymbol{\delta}(-, w_j) \\ s_{i-1,j-1} + \boldsymbol{\delta}(v_i, w_j) \end{cases}$$

Power of ZERO: this is the only change from the original recurrence of a global alignment, representing the "free ride" edge

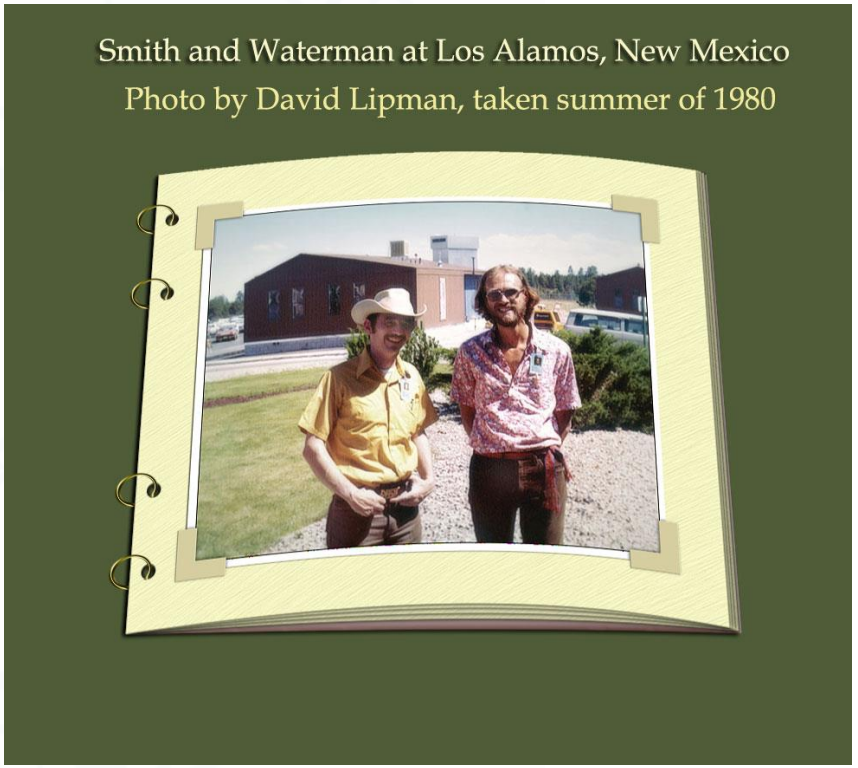
Local Alignment: Backtrace

- Score of best local alignment is the maximum entry of s_{ij}
- The alignment is found by a backtrace from the maximum node, to a node for which the score is 0.



Local Alignment: SW

- This local alignment algorithm is known as the “Smith-Waterman” algorithm[1].
- T.F. Smith and M.W. Waterman. Identification of common molecular subsequences. J. Mol. Biol. 147:195-197, 1981.



[1]The Smith-Waterman algorithm considers a more sophisticated gap penalty scheme

Example

$$H = \begin{pmatrix} & - & A & C & A & C & A & C & T & A \\ - & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A & 0 & 2 & 1 & 2 & 1 & 2 & 1 & 0 & 2 \\ G & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ C & 0 & 0 & 3 & 2 & 3 & 2 & 3 & 2 & 1 \\ A & 0 & 2 & 2 & 5 & 4 & 5 & 4 & 3 & 4 \\ C & 0 & 1 & 4 & 4 & 7 & 6 & 7 & 6 & 5 \\ A & 0 & 3 & 3 & 6 & 6 & 9 & 8 & 7 & 8 \\ C & 0 & 2 & 4 & 5 & 8 & 8 & 11 & 10 & 9 \\ A & 0 & 4 & 3 & 6 & 7 & 10 & 10 & 10 & 12 \end{pmatrix}$$

http://en.wikipedia.org/wiki/Smith%E2%80%93Waterman_algorithm

Questions

- What is the complexity of the ‘traceback’ step?

- What is the worst-case complexity of the traceback?

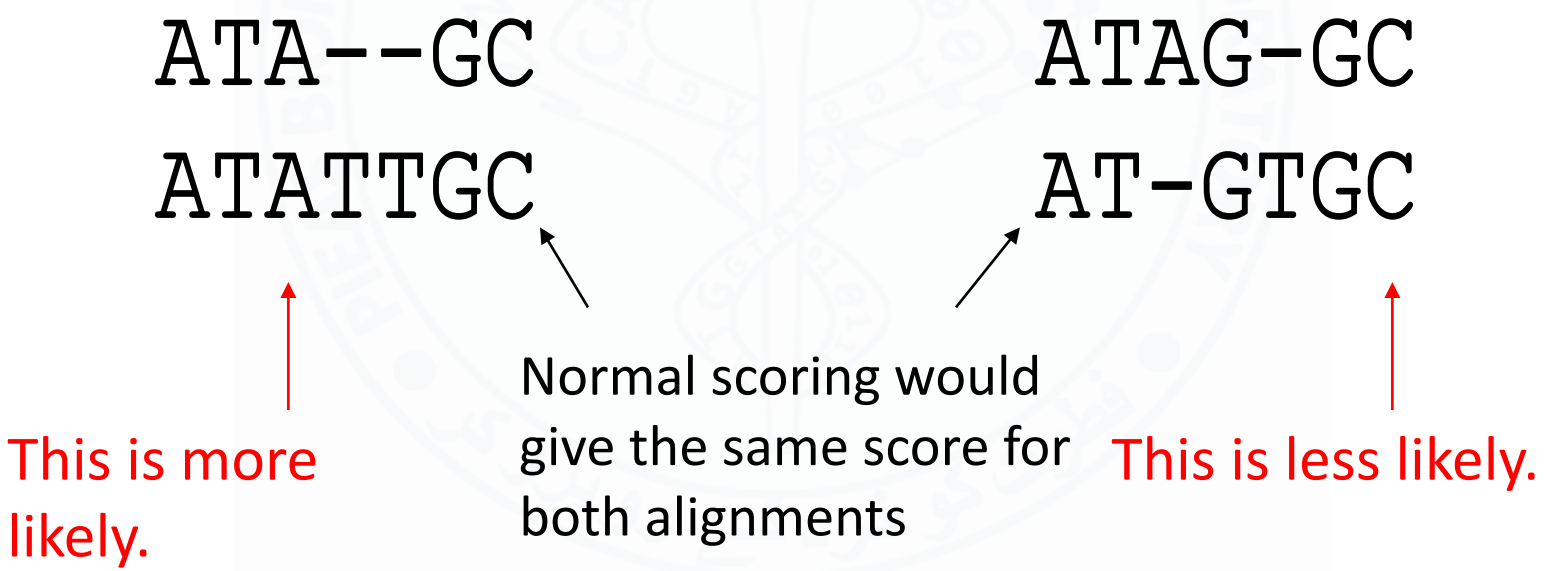
Scoring Indels: Naive Approach

- A fixed penalty σ is given to every indel:
  - $-\sigma$ for 1 indel,
  - -2σ for 2 consecutive indels
  - -3σ for 3 consecutive indels, etc.

Can be too severe penalty for a series of 100 consecutive indels

Affine Gap Penalties

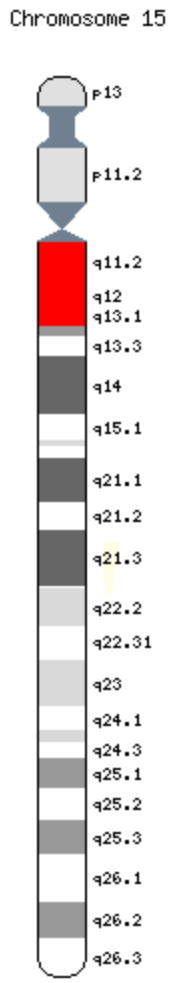
- In nature, a series of *k* indels often comes as a single event rather than a series of *k* single nucleotide events:



Missing DNA

- Many forms of Autism are cause by missing chunks of DNA
  - In one study, it was found that 5 Mbp were missing in a child with autism
  - Aspergers
- Angelman syndrome (happy puppet syndrome)
  - Deletion of the gene ATP10a on chromosome 15

http://www.ted.com/talks/james_watson_on_how_he_discovered_dna?language=en



Affine gap penalty

- Score for a gap of length x is: $-(\rho + \sigma x)$

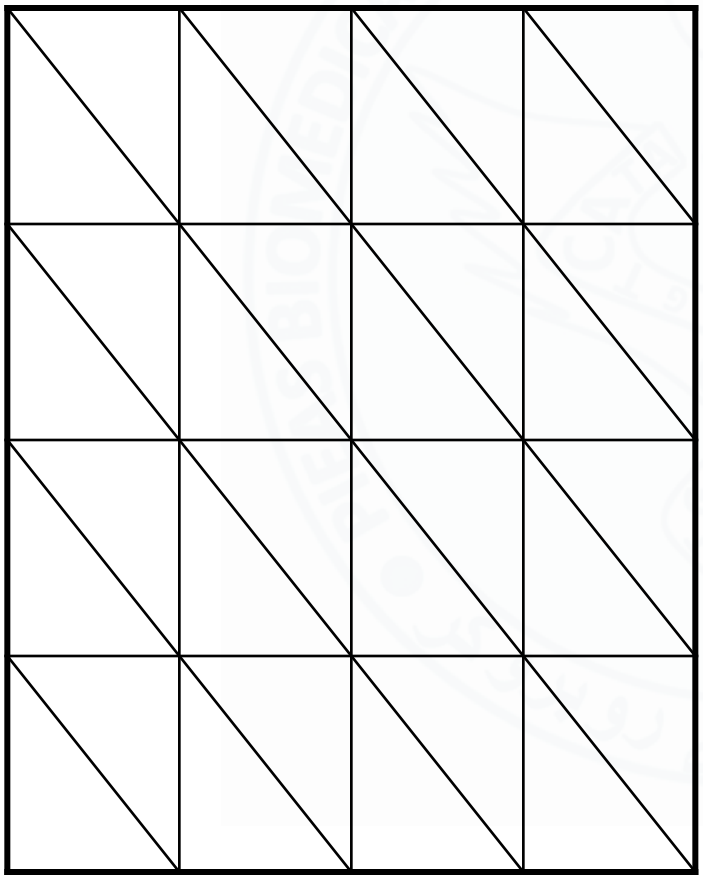
where:

$\rho > 0$ is the gap opening penalty

$\sigma > 0$ is the gap extension penalty

- ρ is large relative to σ because you do not want to add too much of a penalty for extending the gap.

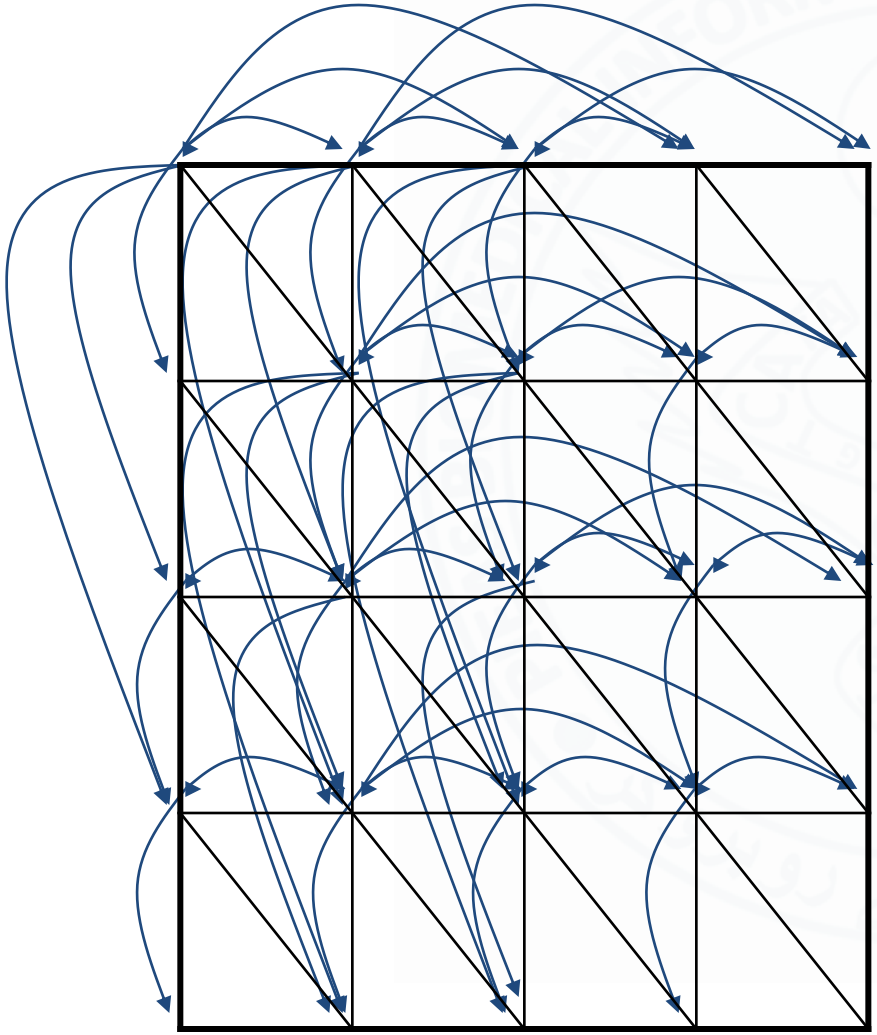
Affine Gap Penalties and Edit Graph



To reflect affine gap penalties we have to add "long" horizontal and vertical edges to the edit graph. Each such edge of length *x* should have weight

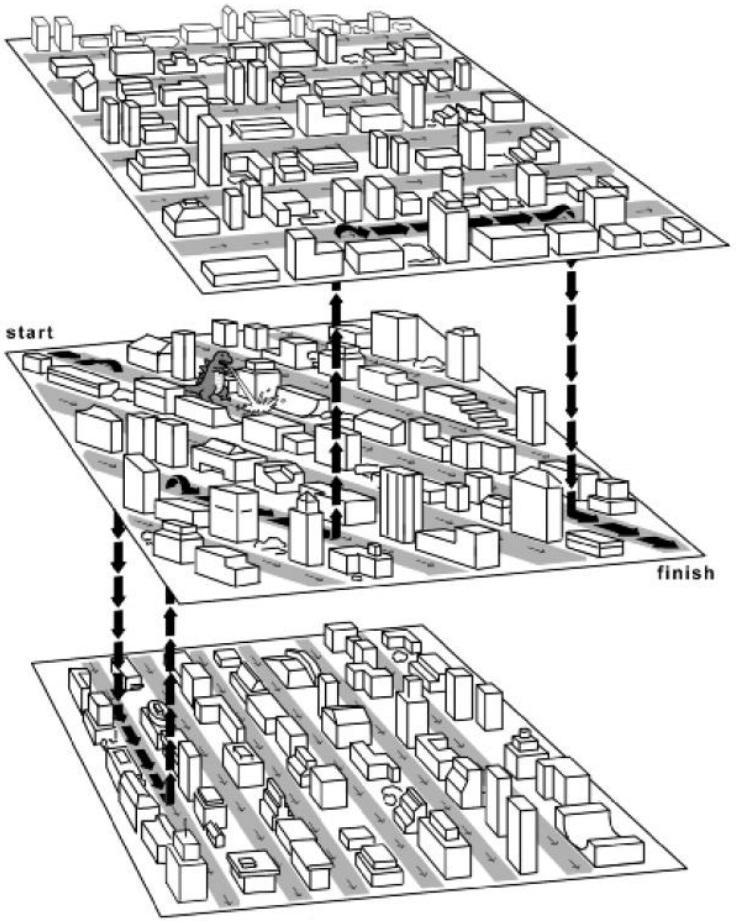
$$-\rho - x\sigma$$

Adding "Affine Penalty" Edges to the Edit Graph



- There are many such edges!
  - n (or m) incoming edges are added
- Adding them to the graph increases the running time of the alignment algorithm by a factor of *n.*
- The complexity increases from $O(n^2)$ to $O(n^3)$

The 3-leveled Manhattan Grid

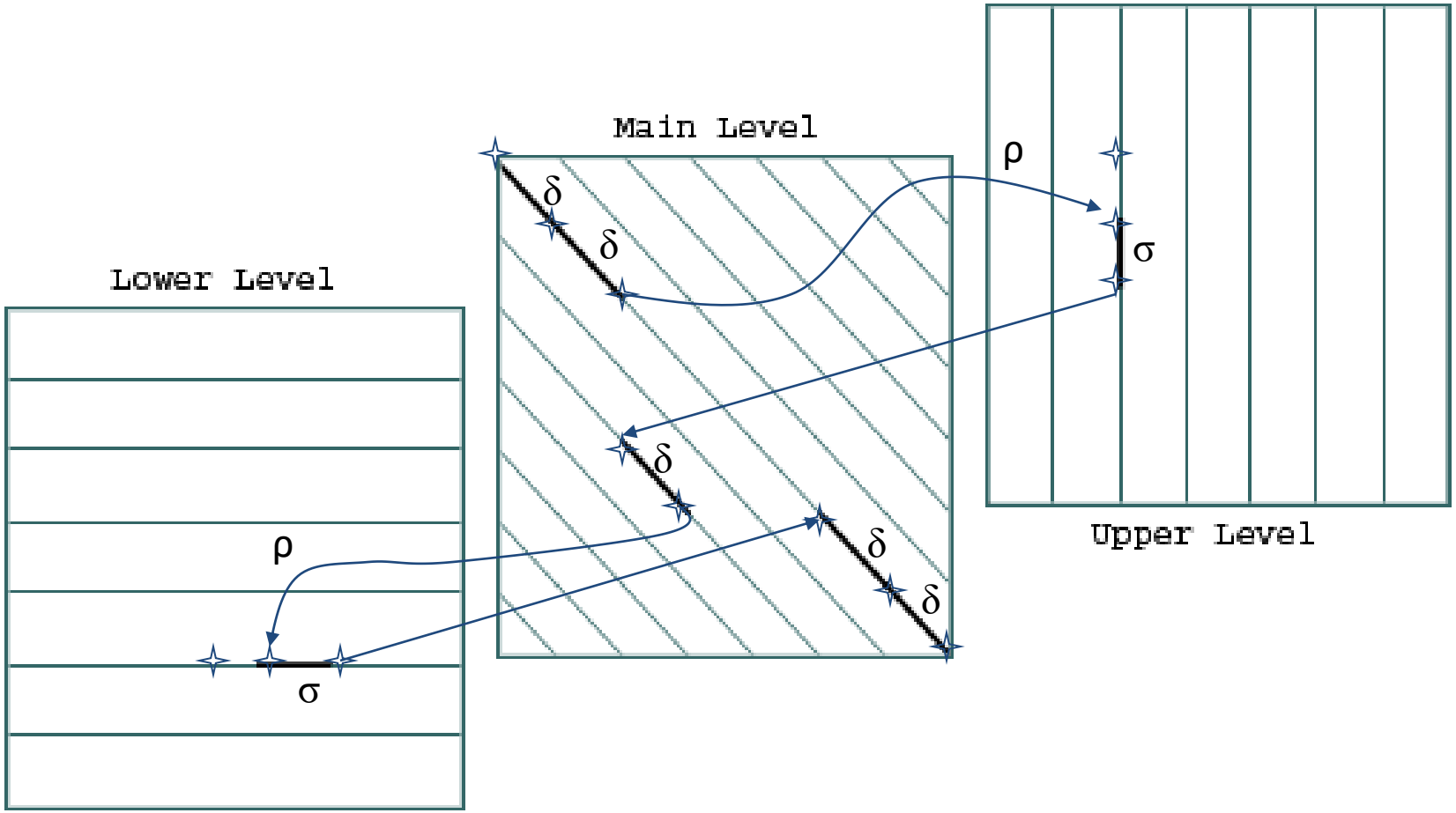


gaps in ***w***

matches/mismatches

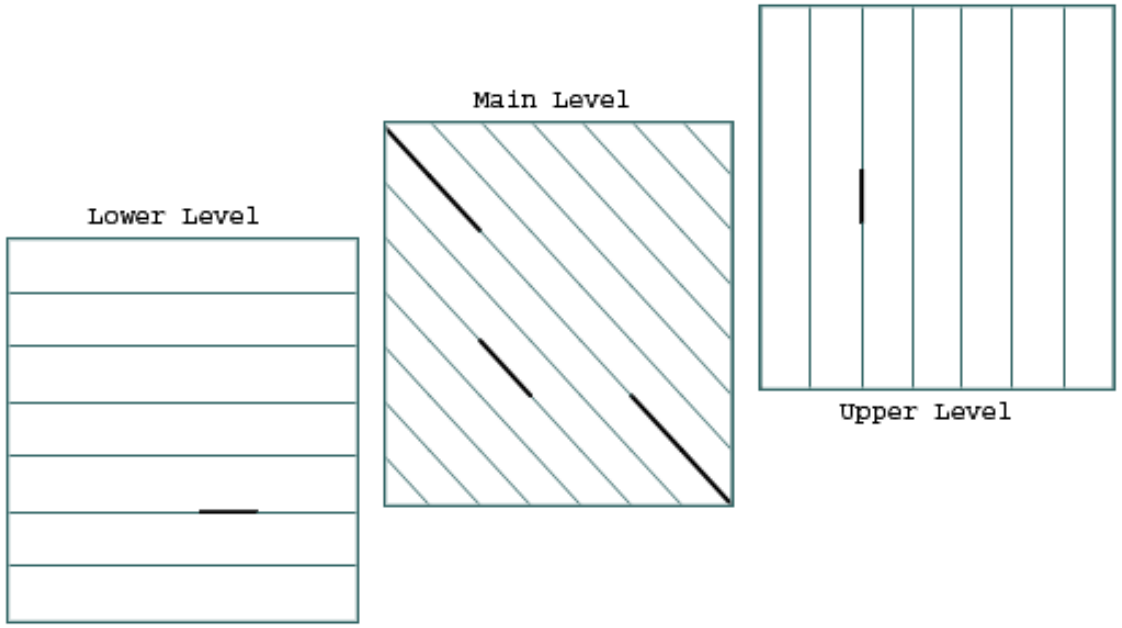
gaps in ***v***

Manhattan in 3 Layers



Affine Gap Penalties and 3 Layer Manhattan Grid

- We'll have three recurrences in a 3-layered graph.
  - The top level creates/extends gaps in the sequence *w.*
  - The bottom level creates/extends gaps in sequence *v.*
  - The middle level extends matches and mismatches.



Switching Between the Layers

- Levels:
  - The **main level** is for diagonal edges
  - The **lower level** is for horizontal edges
  - The **upper level** is for vertical edges
- A jumping penalty is assigned to moving from the main level to either the upper level or the lower level ($-\rho-\sigma$)
- There is a gap extension penalty for each continuation on a level other than the main level ($-\sigma$)

Affine Gap Penalty Recurrences

$$\downarrow s_{i,j} = \max \begin{cases} \downarrow s_{i-1,j} - \sigma \\ s_{i-1,j} - (\rho+\sigma) \end{cases}$$

Continue gap in *w*

Start gap in *w* : from middle

$$\rightarrow s_{i,j} = \max \begin{cases} \rightarrow s_{i,j-1} - \sigma \\ s_{i,j-1} - (\rho+\sigma) \end{cases}$$

Continue gap in *v*

Start gap in *v* : from middle

$$s_{i,j} = \max \begin{cases} s_{i-1,j-1} + \delta(v_i, w_j) \\ \downarrow s_{i,j} \\ \rightarrow s_{i,j} \end{cases}$$

Match or Mismatch

End gap: from top

End gap: from bottom

Should we compare DNA or protein sequences?

- DNA sequence is less conserved than protein sequence
- The protein sequence contains more information than the DNA sequence

⇒ **Less effective to compare coding regions at the nucleotide level**

When to use local/global?

Alignments

SUBSTITUTION MATRICES

Making a Scoring Matrix

- Scoring matrices are created based on the intuition that some mutations have a smaller effect on the function of a protein

  $\Rightarrow$ Such mismatch penalties should be less harsh than others.

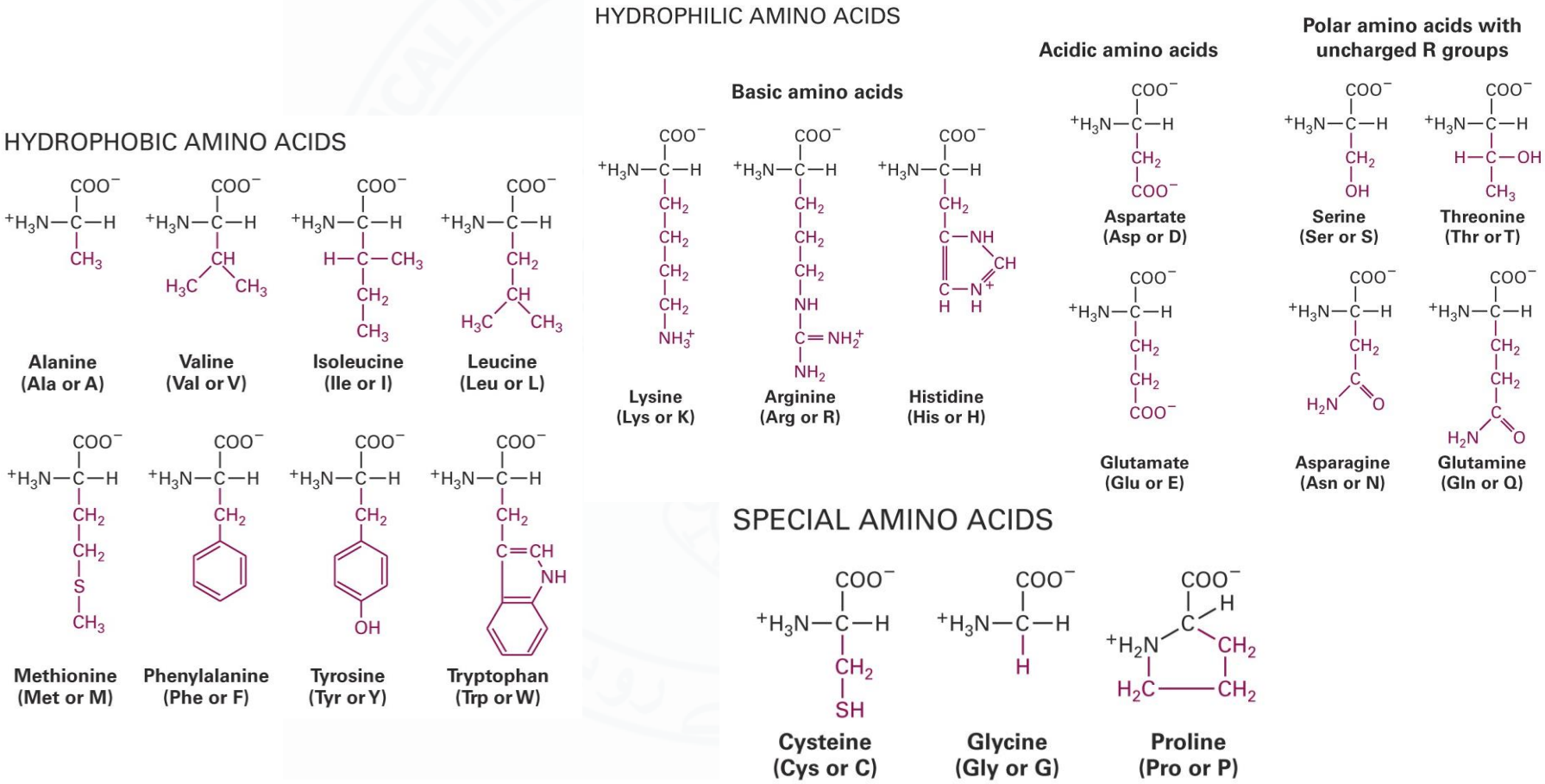
Scoring Matrix: Example

	A	R	N	K
A	5	-2	-1	-1
R	-	7	-1	3
N	-	-	7	0
K	-	-	-	6

- Although R and K are different amino acids, they have a positive score.
- Why? They are both positively charged amino acids, therefore substitution will not greatly change the function of the protein.

Substitutions of Amino Acids

Mutation rates between amino acids have dramatic differences!

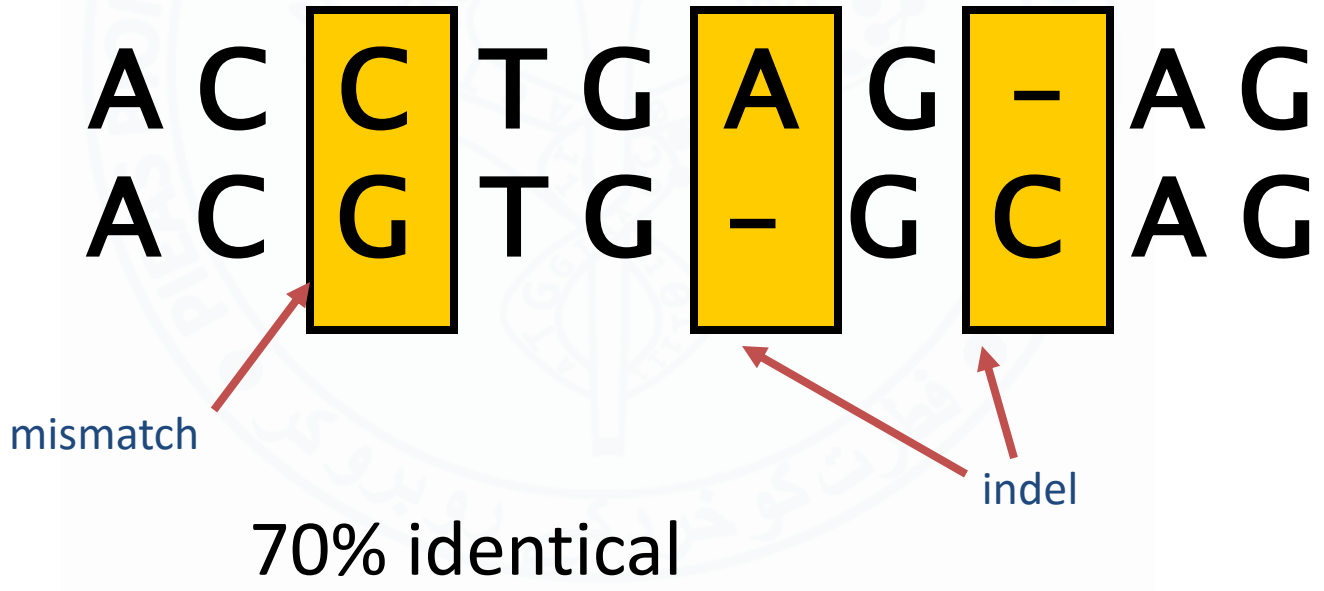


Conservation

- Amino acid changes that preserve the physico-chemical properties of the original residue should receive higher scores
  - polar to polar
    - aspartate → glutamate
  - hydrophobic to hydrophobic
    - alanine → valine

Percent Sequence Identity

- A measure of the extent to which two nucleotide or amino acid sequences are similar



BLOSUM

- **Blo**cks **Su**bstitution **M**atrix
- Scores derived from *observations* of the frequencies of substitutions in alignments of related proteins
- Matrix name indicates evolutionary distance:
  - BLOSUM62 was created using sequences sharing no more than 62% identity

Henikoff, S. and Henikoff, J. (1992)Amino acid substitution matrices from protein blocks.Proc. Natl. Acad. Sci. USA. 89(biochemistry): 10915 - 10919. 1992

An entry from the BLOCKS database

```
Block PR00851A

ID   XRODRMPGMNTB; BLOCK
AC   PR00851A; distance from previous block=(52,131)
DE   Xeroderma pigmentosum group B protein signature
BL   adapted;   width=21; seqs=8; 99.5%=985; strength=1287
XPB_HUMAN|P19447    (  74) RPLWVAPDGHIFLEAFSPVYK  54
XPB_MOUSE|P49135    (  74) RPLWVAPDGHIFLEAFSPVYK  54
P91579              (  80) RPLYLAPDGHIFLESFSPVYK  67
XPB_DROME|Q02870    (  84) RPLWVAPNGHVFLESFSPVYK  79
RA25_YEAST|Q00578   ( 131) PLWISPSDGRIILESFSPLAE 100
Q38861              (  52) RPLWACADGRIFLETFSPLYK  71
O13768              (  90) PLWINPIDGRIILEAFSPLAE 100
O00835              (  79) RPIWVCPDGHIFLETFSAIYK  86
//
```

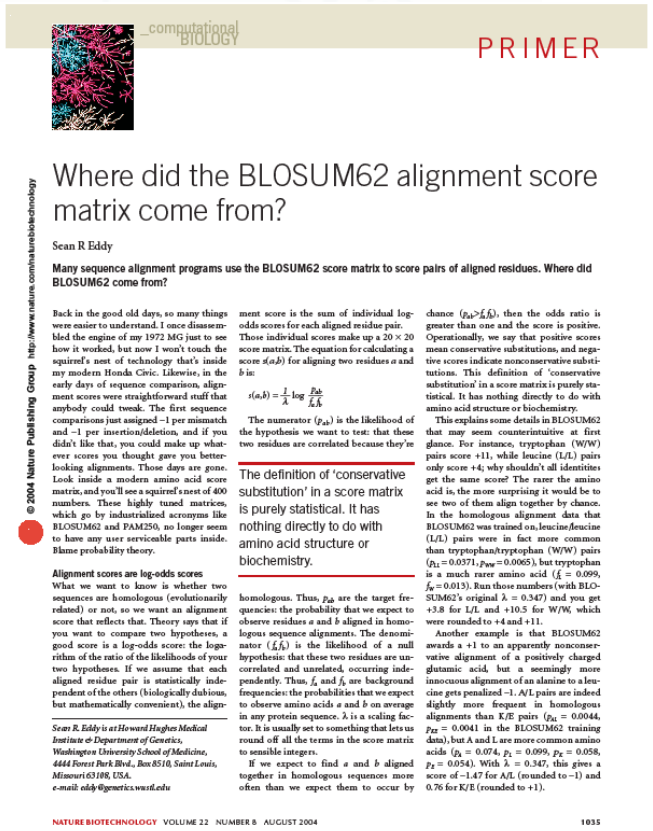
The BLOCKS database is at: http://blocks.fhcrc.org/

The Blosum50 Scoring Matrix

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	B	Z	X	*
A	5	-2	-1	-2	-1	-1	-1	0	-2	-1	-2	-1	-1	-3	-1	1	0	-3	-2	0	-2	-1	-1	-5
R	-2	7	-1	-2	-4	1	0	-3	0	-4	-3	3	-2	-3	-3	-1	-1	-3	-1	-3	-1	0	-1	-5
N	-1	-1	7	2	-2	0	0	0	1	-3	-4	0	-2	-4	-2	1	0	-4	-2	-3	4	0	-1	-5
D	-2	-2	2	8	-4	0	2	-1	-1	-4	-4	-1	-4	-5	-1	0	-1	-5	-3	-4	5	1	-1	-5
C	-1	-4	-2	-4	13	-3	-3	-3	-3	-2	-2	-3	-2	-2	-4	-1	-1	-5	-3	-1	-3	-3	-2	-5
Q	-1	1	0	0	-3	7	2	-2	1	-3	-2	2	0	-4	-1	0	-1	-1	-1	-3	0	4	-1	-5
E	-1	0	0	2	-3	2	6	-3	0	-4	-3	1	-2	-3	-1	-1	-1	-3	-2	-3	1	5	-1	-5
G	0	-3	0	-1	-3	-2	-3	8	-2	-4	-4	-2	-3	-4	-2	0	-2	-3	-3	-4	-1	-2	-2	-5
H	-2	0	1	-1	-3	1	0	-2	10	-4	-3	0	-1	-1	-2	-1	-2	-3	2	-4	0	0	-1	-5
I	-1	-4	-3	-4	-2	-3	-4	-4	-4	5	2	-3	2	0	-3	-3	-1	-3	-1	4	-4	-3	-1	-5
L	-2	-3	-4	-4	-2	-2	-3	-4	-3	2	5	-3	3	1	-4	-3	-1	-2	-1	1	-4	-3	-1	-5
K	-1	3	0	-1	-3	2	1	-2	0	-3	-3	6	-2	-4	-1	0	-1	-3	-2	-3	0	1	-1	-5
M	-1	-2	-2	-4	-2	0	-2	-3	-1	2	3	-2	7	0	-3	-2	-1	-1	0	1	-3	-1	-1	-5
F	-3	-3	-4	-5	-2	-4	-3	-4	-1	0	1	-4	0	8	-4	-3	-2	1	4	-1	-4	-4	-2	-5
P	-1	-3	-2	-1	-4	-1	-1	-2	-2	-3	-4	-1	-3	-4	10	-1	-1	-4	-3	-3	-2	-1	-2	-5
S	1	-1	1	0	-1	0	-1	0	-1	-3	-3	0	-2	-3	-1	5	2	-4	-2	-2	0	0	-1	-5
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	2	5	-3	-2	0	0	-1	0	-5
W	-3	-3	-4	-5	-5	-1	-3	-3	-3	-3	-2	-3	-1	1	-4	-4	-3	15	2	-3	-5	-2	-3	-5
Y	-2	-1	-2	-3	-3	-1	-2	-3	2	-1	-1	-2	0	4	-3	-2	-2	2	8	-1	-3	-2	-1	-5
V	0	-3	-3	-4	-1	-3	-3	-4	-4	4	1	-3	1	-1	-3	-2	0	-3	-1	5	-4	-3	-1	-5
B	-2	-1	4	5	-3	0	1	-1	0	-4	-4	0	-3	-4	-2	0	0	-5	-3	-4	5	2	-1	-5
Z	-1	0	0	1	-3	4	5	-2	0	-3	-3	1	-1	-4	-1	0	-1	-2	-2	-3	2	5	-1	-5
X	-1	-1	-1	-1	-2	-1	-1	-2	-1	-1	-1	-1	-1	-2	-2	-1	0	-3	-1	-1	-1	-1	-1	-5
*	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	1

Blosum62

- Where did the BLOSUM62 alignment score matrix come from?
  - Eddy 2004



computational BIOLOGY

PRIMER

Where did the BLOSUM62 alignment score matrix come from?

Sean R Eddy

Many sequence alignment programs use the BLOSUM62 score matrix to score pairs of aligned residues. Where did BLOSUM62 come from?

Back in the good old days, so many things were easier to understand. I once disassembled the engine of my 1972 MG just to see how it worked, but now I won't touch the squirrel's nest of technology that's inside my modern Honda Civic. Likewise, in the early days of sequence comparison, alignment scores were straightforward stuff that anybody could tweak. The first sequence comparisons just assigned −1 per mismatch and −1 per insertion/deletion, and if you didn't like that, you could make up whatever scores you thought gave you better-looking alignments. Those days are gone. Look inside a modern amino acid score matrix, and you'll see a squirrel's nest of 400 numbers. These highly tuned matrices, which go by industrialized acronyms like BLOSUM62 and PAM250, no longer seem to have any user serviceable parts inside. Blame probability theory.

Alignment scores are log-odds scores

What we want to know is whether two sequences are homologous (evolutionarily related) or not, so we want an alignment score that reflects that. Theory says that if you want to compare two hypotheses, a good score is a log-odds score: the logarithm of the ratio of the likelihoods of your two hypotheses. If we assume that each aligned residue pair is statistically independent of the others (biologically dubious, but mathematically convenient), the alignment score is the sum of individual log-odds scores for each aligned residue pair. Those individual scores make up a 20×20 score matrix. The equation for calculating a score $s(a,b)$ for aligning two residues a and b is:

$$s(a,b) = \frac{1}{\lambda} \log \frac{p_{ab}}{f_a f_b}$$

The numerator (p_{ab}) is the likelihood of the hypothesis we want to test: that these two residues are correlated because they're homologous. Thus, p_{ab} are the target frequencies: the probability that we expect to observe residues a and b aligned in homologous sequence alignments. The denominator ($f_a f_b$) is the likelihood of a null hypothesis: that these two residues are uncorrelated and unrelated, occurring independently. Thus, f_a and f_b are background frequencies: the probabilities that we expect to observe amino acids a and b on average in any protein sequence. λ is a scaling factor. It is usually set to something that lets us round off all the terms in the score matrix to sensible integers.

> The definition of 'conservative substitution' in a score matrix is purely statistical. It has nothing directly to do with amino acid structure or biochemistry.

If we expect to find a and b aligned together in homologous sequences more often than we expect them to occur by chance ($p_{ab} > f_a f_b$), then the odds ratio is greater than one and the score is positive. Operationally, we say that positive scores mean conservative substitutions, and negative scores indicate nonconservative substitutions. This definition of 'conservative substitution' in a score matrix is purely statistical. It has nothing directly to do with amino acid structure or biochemistry.

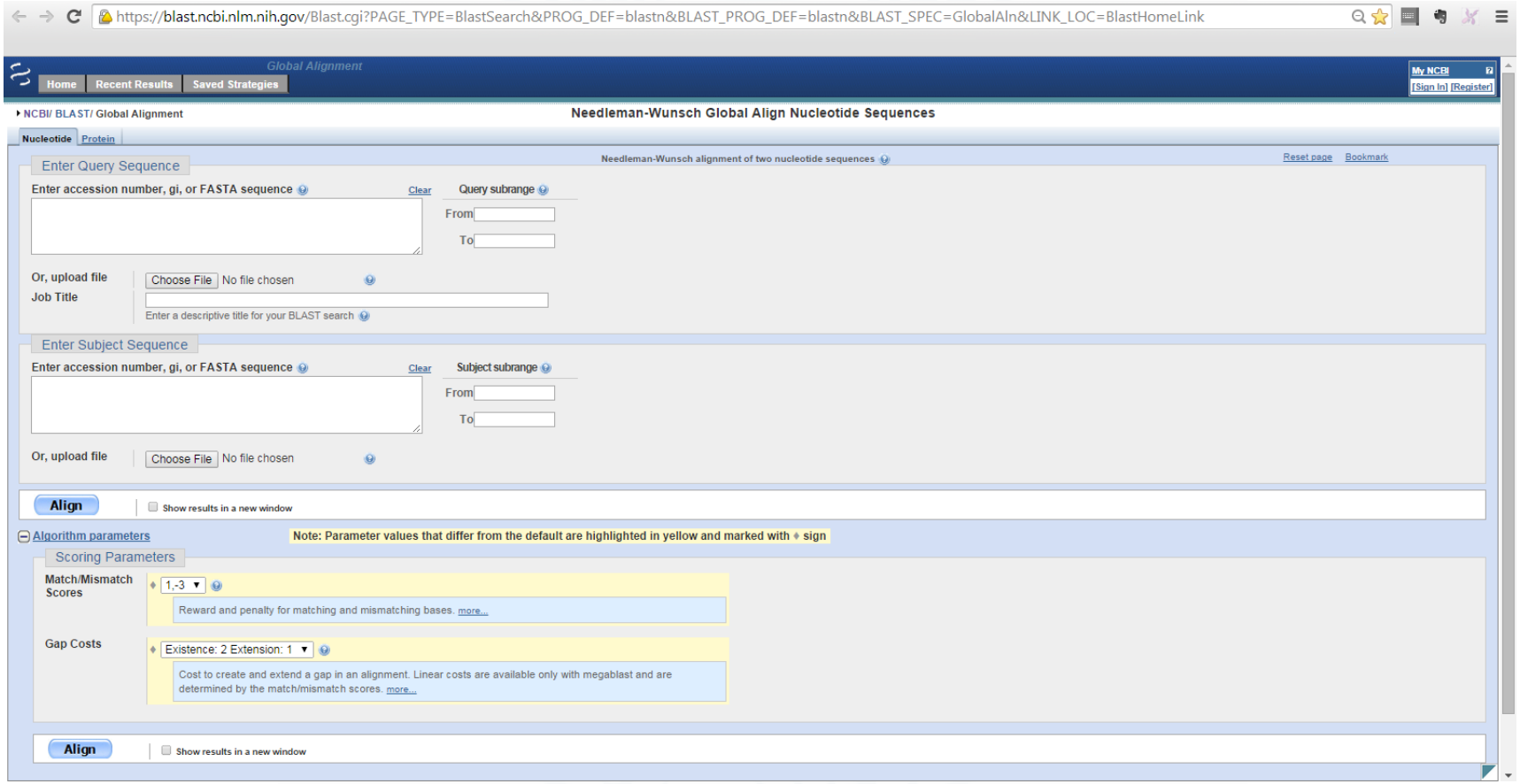
This explains some details in BLOSUM62 that may seem counterintuitive at first glance. For instance, tryptophan (W/W) pairs score +11, while leucine (L/L) pairs only score +4; why shouldn't all identities get the same score? The rarer the amino acid is, the more surprising it would be to see two of them align together by chance. In the homologous alignment data that BLOSUM62 was trained on, leucine/leucine (L/L) pairs were in fact more common than tryptophan/tryptophan (W/W) pairs ($p_L = 0.0371$, $p_{WW} = 0.0065$), but tryptophan is a much rarer amino acid ($f_L = 0.099$, $f_W = 0.013$). Run those numbers (with BLOSUM62's original $\lambda = 0.347$) and you get +3.8 for L/L and +10.5 for W/W, which were rounded to +4 and +11.

Another example is that BLOSUM62 awards a +1 to an apparently nonconservative alignment of a positively charged glutamic acid, but a seemingly more innocuous alignment of an alanine to a leucine gets penalized −1. A/L pairs are indeed slightly more frequent in homologous alignments than K/E pairs ($p_{AL} = 0.0044$, $p_{EK} = 0.0041$ in the BLOSUM62 training data), but A and L are more common amino acids ($p_A = 0.074$, $p_L = 0.099$, $p_K = 0.058$, $p_E = 0.054$). With $\lambda = 0.347$, this gives a score of −1.47 for A/L (rounded to −1) and 0.76 for K/E (rounded to +1).

Sean R. Eddy is at Howard Hughes Medical Institute & Department of Genetics, Washington University School of Medicine, 4444 Forest Park Blvd., Box 8510, Saint Louis, Missouri 63108, USA.
e-mail: eddy@genetics.wustl.edu

NATURE BIOTECHNOLOGY VOLUME 22 NUMBER 8 AUGUST 2004

Online NW



https://blast.ncbi.nlm.nih.gov/Blast.cgi?PAGE_TYPE=BlastSearch&PROG_DEF=blastn&BLAST_PROG_DEF=blastn&BLAST_SPEC=GlobalAln&LINK_LOC=BlastHomeLink

Sequence alignments in Python

- Biopython
  - http://biopython.org/DIST/docs/api/Bio.pairwise2-module.html

End of Lecture