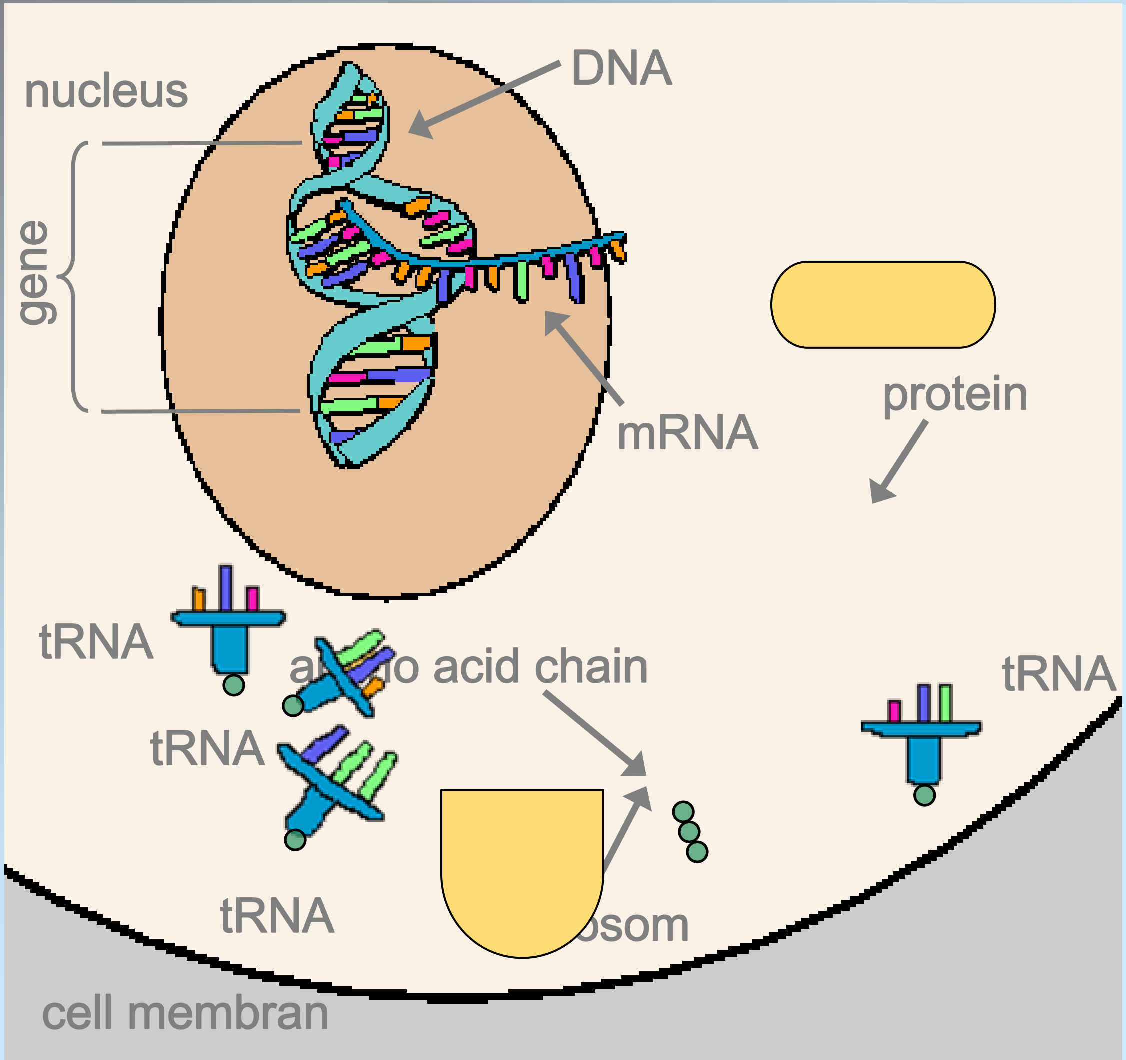
Bioinformatics 1

Biology, Sequences, Phylogenetics

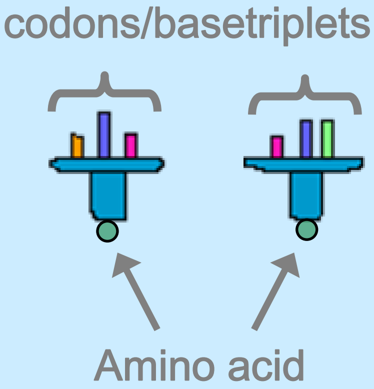
Part 2

Sepp Hochreiter

Central Dogma



1. **transcription** (mRNA)
2. transport
3. **translation** (ribosom, tRNA)
4. folding (protein)



Contents

Data Bases

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles

Resources on the WWW and data bases

➥European Molecular Biology Laboratory (EMBL - http://www.embl-heidelberg.de): nucleotide data base (daily updated)

➥European Bioinformatics Institute (EBI - http://www.ebi.ac.uk/ebi_home.html): SwissProt protein sequence data base and Sequence Retrieval System (SRS - http://srs.ebi.ac.uk/)

➥ExPASy (http://www.expasy.org/): SwissProt & TrEMBL, PROSITE

➥University College London: PRINTS (protein fingerprints) database and the CATH protein structure database

Data Bases

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles

- National Center for Biotechnology Information (NCBI - http://www.ncbi.nlm.nih.gov/): GenBank (NIH), DNA sequence data base and BLAST software (with NR); ENTREZ (http://www.ncbi.nlm.nih.gov/Entrez/): biological data and articles, nucleotide sequences from GenBank, EMBL, DDBJ (DNA data base of Japan) as well as SWISS-PROT, PIR, PRF, SEQDB, PDB

- European EMBnet (http://www.embnet.org)

- Sanger Centre / Wellcome Trust (www.sanger.ac.uk/Info/)

- Martinsried Institute for Protein Sequences (MIPS - http://www.mips.biochem.mpg.de/)

Data Bases

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles

EMBL	http://www.embl-heidelberg.de/
PDB	http://www.rcsb.org/pdb/Welcome.do
SCOP	http://scop.berkeley.edu/
CATH	http://cathwww.biochem.ucl.ac.uk/latest/
PIR	http://pir.georgetown.edu/
SWISS-PROT	http://www.expasy.org/sprot/
TrEMBL	http://www.expasy.org/sprot/
Homstrad	http://www-cryst.bioc.cam.ac.uk/~homstrad/
InterPro	http://www.ebi.ac.uk/interpro/
NR	ftp://ftp.ncbi.nih.gov/blast/db
Pfam	http://www.sanger.ac.uk/Software/Pfam/
UniProt	http://www.expasy.uniprot.org/
PROSITE	http://www.expasy.org/prosite/
PRINTS	http://umber.sbs.man.ac.uk/dbbrowser/PRINTS/
BLOCKS	http://blocks.fhcrc.org/
CAMPASS	http://www-cryst.bioc.cam.ac.uk/~campass/

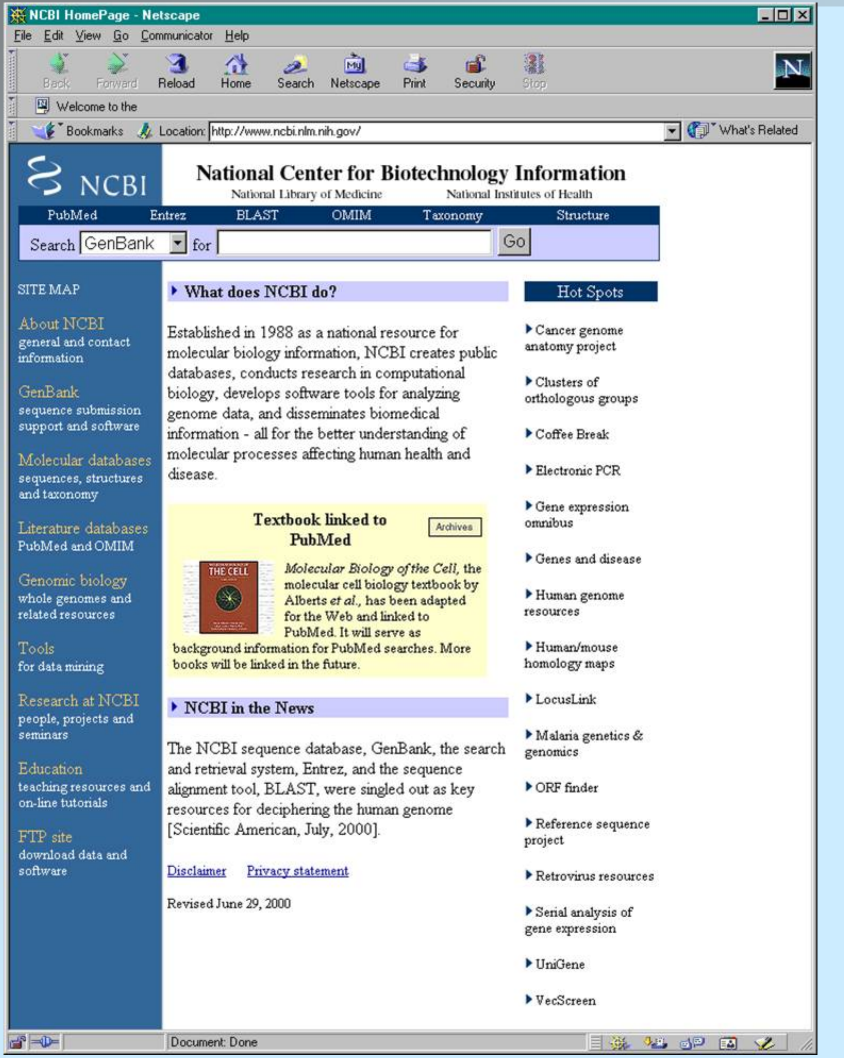
Data Bases

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles



NCBI

National Center for Biotechnology Information

National Library of Medicine National Institutes of Health

PubMed Entrez BLAST OMIM Taxonomy Structure

Search GenBank for Go

SITE MAP

About NCBI
general and contact information

GenBank
sequence submission support and software

Molecular databases
sequences, structures and taxonomy

Literature databases
PubMed and OMIM

Genomic biology
whole genomes and related resources

Tools
for data mining

Research at NCBI
people, projects and seminars

Education
teaching resources and on-line tutorials

FTP site
download data and software

What does NCBI do?

Established in 1988 as a national resource for molecular biology information, NCBI creates public databases, conducts research in computational biology, develops software tools for analyzing genome data, and disseminates biomedical information - all for the better understanding of molecular processes affecting human health and disease.

Textbook linked to PubMed

Archives

THE CELL

Molecular Biology of the Cell, the molecular cell biology textbook by Alberts *et al.*, has been adapted for the Web and linked to PubMed. It will serve as background information for PubMed searches. More books will be linked in the future.

NCBI in the News

The NCBI sequence database, GenBank, the search and retrieval system, Entrez, and the sequence alignment tool, BLAST, were singled out as key resources for deciphering the human genome [Scientific American, July, 2000].

Disclaimer Privacy statement

Revised June 29, 2000

Hot Spots

- Cancer genome anatomy project
- Clusters of orthologous groups
- Coffee Break
- Electronic PCR
- Gene expression omnibus
- Genes and disease
- Human genome resources
- Human/mouse homology maps
- LocusLink
- Malaria genetics & genomics
- ORF finder
- Reference sequence project
- Retrovirus resources
- Serial analysis of gene expression
- UniGene
- VecScreen

Software

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles

machine learning:

- Suport Vector Machines and kernel methods: www.kernel-machines.org/ under „software“, libsvm and torch are recommanded
- feature selection: „spider“ software or the PSVM software http://www.bioinf.jku.at/software/psvm

Software

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles

bioinformatics:

EMBOSS	toolbox	http://emboss.sourceforge.net
Domainatrix	tools/domains	http://emboss.sourceforge.net/embassy/domainatrix/
BLAST	homology/profiles	http://www.ncbi.nlm.nih.gov/BLAST/
PHRAP	shotgun DNA	http://www.phrap.org/
Babel	converts formats	http://openbabel.sourceforge.net/
BioPerl	toolbox perl	http://www.bioperl.org/
clustalw	multiple alig.	ftp://ftp-igbmc.u-strasbg.fr/pub/ClustalW
modeller	building model	http://salilab.org/modeller/download_installation.html
phylip	phylogenetics	http://evolution.gs.washington.edu/phylip.html
pymol	good viewer	http://pymol.sourceforge.net/
rasmol	fast viewer	http://www.umass.edu/microbio/rasmol/
molscript	nice images	http://www.avatar.se/molscript/obtain_info.html

Software

bioinformatics:

strap	java toolbox	http://www.charite.de/bioinf/strap/
tinker	mol. dyn., fortran	http://www.es.embnet.org/Services/MolBio/tinker/
biodesigner	mol. dynamics	http://www.pirx.com/biodesigner/download.html
threader	threading	http://bioinf.cs.ucl.ac.uk/threader/threader.html
Loopp	threading	http://folding.chmcc.org/loopp/loopp.html
prospect	threading	http://compbio.ornl.gov/structure/prospect/

Software

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles

bioinformatics:

sspro4	sec. struc.	http://contact.ics.uci.edu/download.html
psipred	sec. struc.	ftp://bioinf.cs.ucl.ac.uk/pub/psipred/
prof	sec. struc.	http://www.aber.ac.uk/compsci/Research/bio/dss/prof/
jnet	sec. struc.	http://www.compbio.dundee.ac.uk/~www-jpred/jnet/download.html
PHD	sec. struc.	http://www.embl-heidelberg.de/predictprotein/predictprotein.html
DSSP	sec. struc. f. 3D	http://swift.cmbi.ru.nl/gv/dssp/
whatif	mol. modelling	http://swift.cmbi.kun.nl/whatif/
hmmer	alignment HMM	http://hmmer.wustl.edu/
Prosall	struc. verf.	http://www.came.sbg.ac.at/Services/prosa.html
CE	struc. alig.	ftp://ftp.sdsc.edu/pub/sdsc/biology/CE/src/
DALI	struc. alig.	http://www.ebi.ac.uk/dali/

Articles

- „PubMed“ http://www.ncbi.nlm.nih.gov/
  entrez/query.fcgi?db=PubMed

- machine learning and computer science:
  http://scholar.google.com/

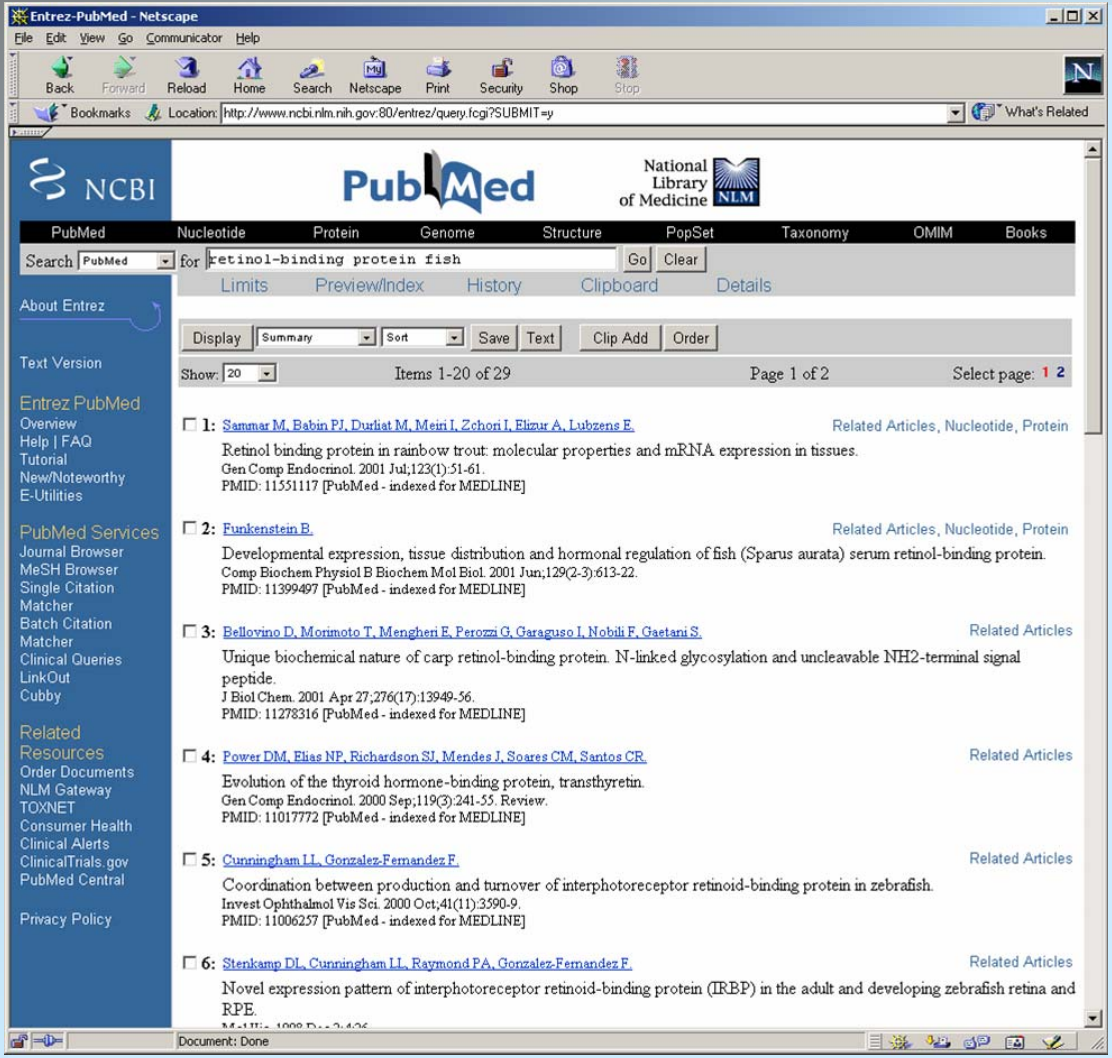
Articles

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles



☐ 1: Sammar M, Babin PJ, Durliat M, Meiri I, Zchori I, Elizur A, Lubzens E. — Related Articles, Nucleotide, Protein

Retinol binding protein in rainbow trout: molecular properties and mRNA expression in tissues.
Gen Comp Endocrinol. 2001 Jul;123(1):51-61.
PMID: 11551117 [PubMed - indexed for MEDLINE]

☐ 2: Funkenstein B. — Related Articles, Nucleotide, Protein

Developmental expression, tissue distribution and hormonal regulation of fish (Sparus aurata) serum retinol-binding protein.
Comp Biochem Physiol B Biochem Mol Biol. 2001 Jun;129(2-3):613-22.
PMID: 11399497 [PubMed - indexed for MEDLINE]

☐ 3: Bellovino D, Morimoto T, Mengheri E, Perozzi G, Garaguso I, Nobili F, Gaetani S. — Related Articles

Unique biochemical nature of carp retinol-binding protein. N-linked glycosylation and uncleavable NH2-terminal signal peptide.
J Biol Chem. 2001 Apr 27;276(17):13949-56.
PMID: 11278316 [PubMed - indexed for MEDLINE]

☐ 4: Power DM, Elias NP, Richardson SJ, Mendes J, Soares CM, Santos CR. — Related Articles

Evolution of the thyroid hormone-binding protein, transthyretin.
Gen Comp Endocrinol. 2000 Sep;119(3):241-55. Review.
PMID: 11017772 [PubMed - indexed for MEDLINE]

☐ 5: Cunningham LL, Gonzalez-Fernandez F. — Related Articles

Coordination between production and turnover of interphotoreceptor retinoid-binding protein in zebrafish.
Invest Ophthalmol Vis Sci. 2000 Oct;41(11):3590-9.
PMID: 11006257 [PubMed - indexed for MEDLINE]

☐ 6: Stenkamp DL, Cunningham LL, Raymond PA, Gonzalez-Fernandez F. — Related Articles

Novel expression pattern of interphotoreceptor retinoid-binding protein (IRBP) in the adult and developing zebrafish retina and RPE.

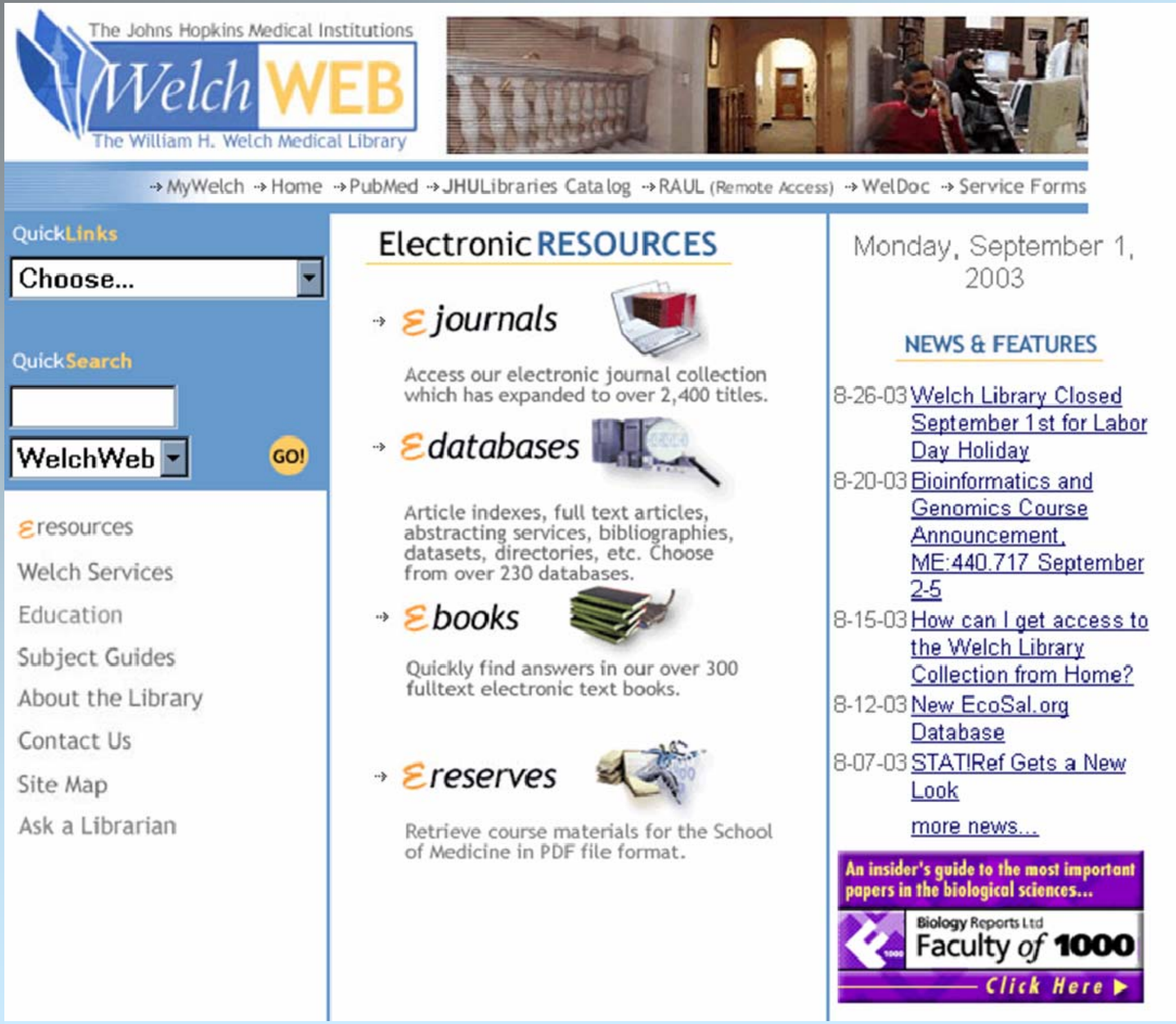
Articles

2 Bioinformatics Rescources

2.1 Data Bases

2.2 Software

2.3 Articles



The Johns Hopkins Medical Institutions
Welch WEB
The William H. Welch Medical Library

→ MyWelch → Home → PubMed → JHULibraries Catalog → RAUL (Remote Access) → WelDoc → Service Forms

QuickLinks
Choose...

QuickSearch
WelchWeb
GO!

- Eresources
- Welch Services
- Education
- Subject Guides
- About the Library
- Contact Us
- Site Map
- Ask a Librarian

Electronic RESOURCES

→ Ejournals

Access our electronic journal collection which has expanded to over 2,400 titles.

→ Edatabases

Article indexes, full text articles, abstracting services, bibliographies, datasets, directories, etc. Choose from over 230 databases.

→ Ebooks

Quickly find answers in our over 300 fulltext electronic text books.

→ Ereserves

Retrieve course materials for the School of Medicine in PDF file format.

Monday, September 1, 2003

NEWS & FEATURES

8-26-03 Welch Library Closed September 1st for Labor Day Holiday
8-20-03 Bioinformatics and Genomics Course Announcement, ME:440.717 September 2-5
8-15-03 How can I get access to the Welch Library Collection from Home?
8-12-03 New EcoSal.org Database
8-07-03 STATIRef Gets a New Look

more news...

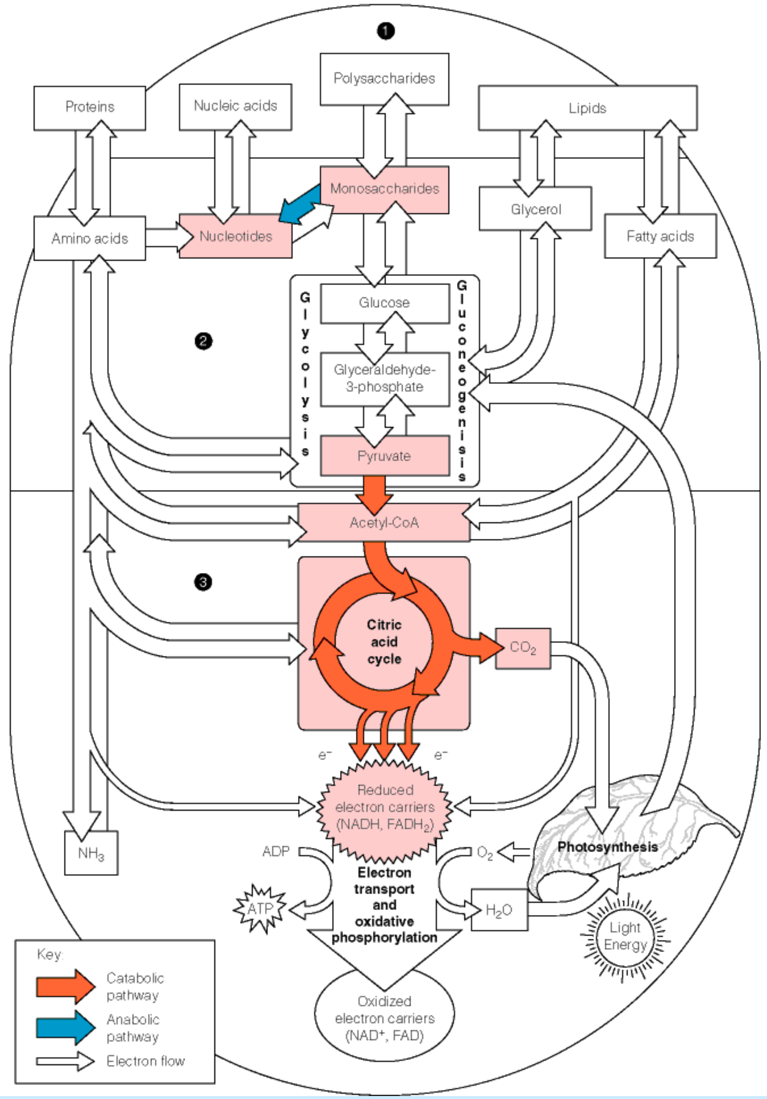
Contents

Motivation

Remark: we mostly focus on sequences of amino acids but all concepts are valid for sequences of nucleotides

↪ proteins of different species are very similar to one another because they must perform the same tasks
(glucose cycles, DNA repair, carrying molecules, membrane proteins, chromosome building proteins)

Motivation


❶
Proteins
Nucleic acids
Polysaccharides
Lipids
Monosaccharides
Glycerol
Amino acids
Nucleotides
Fatty acids
Glucose
Glycolysis
Gluconeogenisis
❷
Glyceraldehyde-3-phosphate
Pyruvate
Acetyl-CoA
❸
Citric acid cycle
CO_2
e^-
e^-
Reduced electron carriers (NADH, $FADH_2$)
NH_3
ADP
O_2
Photosynthesis
ATP
Electron transport and oxidative phosphorylation
H_2O
Light Energy
Oxidized electron carriers (NAD^+, FAD)
Key:
Catabolic pathway
Anabolic pathway
Electron flow

Motivation

➥ a new sequence is first compared to known sequences

➥ if a similar sequence is found: function/structure is also similar

➥ only 1% of the human genes do not match mouse genes (average similarity is 85%)

➥ cells possess a common ancestor cell (a mother cell)

➥ mutations change the genes

➥ difficult to find similarities when many mutations occurred

➥relationship at the structural basis

➥similarities search: **comparative genomics** or **homology search**

Motivation

As in machine learning the optimality criterion, i.e. scoring (penalty, error, energy, cost), is separated from the optimization algorithm.

A scoring scheme can be optimized in different ways.

Many algorithms can be used with different scoring schemes.

The next sections are 3.2 Scoring and 3.3 Algorithms

Sequence Similarity and Scoring

Sequence similarity: trend (stock market), pattern (text), frequencies (speech)

DNA and amino acid sequences: pattern (mutations)

Mutations:

➥point mutations (one nucleotide or amino acid is changed)

➥deletions (one nucleotide or amino acid or a whole subsequence is deleted)

➥insertions (one nucleotide or amino acid or a whole subsequence is inserted)

Sequence Similarity and Scoring

3 Pairwise Alignment
3.1 Motivation
3.2 Scoring
3.2.1 Identity
3.2.2 PAM
3.2.3 BLOSUM
3.2.4 Gap Penalties
3.3 Algorithms
3.3.1 Global
3.3.2 Local
3.3.3 FASTA, BLAST
3.4 Significance
3.4.1 HSPs
3.4.2 Matches

Kind of mutations for a DNA example

- point mutations:

CCGTCAGTTACGCCGTATCGTCTAGCT
CCGCCAGTTTCGCCGTAGCGTCTACCT

- deletion:

CCGTCAGTTACGCCGTATCGTCTAGCT
CCGTCAGTT---CCGTATCGTCTAGCT

- insertion:

CCGTCAGTTACGTATCGTCTAGCT
CCGTCAGTT---TATCGTCTAGCT

Deletion and insertion are indistinguishable: „Indel“

Goal: optimal position of the blanks (max. score)

Sequence Similarity and Scoring

first approach: similarity of two sequences is minimal number of mutations

However: point mutations are more likely then indels

Solution: length of insertions and deletions count

Result: counting the number of matching amino acids

$0.5\ (l1 + l2 - \text{indels} - 2\ \text{mismatches})$

Sequence Similarity and Scoring

```
BIOINFORMATICS                BIOI--N-FORMATICS
                      →
BOILING FOR MANICS            B-OILINGFORMANICS
                                              ↑
```

12 identical letters out of the 14 letters of `BIOINFORMATICS`

Mutations:

(1) delete `I`	`BOINFORMATICS`
(2) insert `LI`	`BOILINFORMATICS`
(3) insert `G`	`BOILINGFORMATICS`
(4) change `T` into `N`	`BOILINGFORMANICS`

Is `I` deleted form the first string or inserted in the second?
Indels are denoted by "`-`"

Identity Matrix

Sequence alignment: arrangement of the two strings so that the number of mutations is minimal

Score (optimality value): number matches (match +1 and mismatch 0)

Pairwise amino acid score: identity matrix

An alignment algorithm searches for the arrangement of two sequences such that a criterion is optimized.

Arrangement: inserting “–” into the strings and moving them horizontally against each other

Identity Matrix

Dot matrix:

- one sequence on the top and the other vertically
- letters of the sequences are paired (all pairs)
- each matching pair of letters receives a dot

Identity Matrix

	B	I	O	I	N	F	O	R	M	A	T	I	C	S
B	●													
O			●				●							
I		●		●								●		
L														
I		●		●								●		
N					●									
G														
F						●								
O			●				●							
R								●						
M									●					
A										●				
N														
I												●		
C													●	
S														●

Identity Matrix

Which pairs correspond to the optimal alignment?

Each path through the matrix is an alignment and vice versa
Goal: search path with most dots

A simple game, where you can move

- horizontally “→” (a ”-” in the vertical sequence)
- vertically “↓” (a “-” in the horizontal sequence)
- diagonal “↘” only if you at the position of a dot (matches)

Task: hit as many dots as possible if you run from the upper left corner to the lower right corner.

Identity Matrix

	B	I	O	I	N	F	O	R	M	A	T	I	C	S
B	↘													
O		→	↘				●							
I		●		↘								●		
L					↓									
I		●		●	↓							●		
N					↘									
G						↓								
F						↘								
O			●				↘							
R								↘						
M									↘					
A										↘				
N											↓			
I											→	↘		
C													↘	
S														↘

Identity Matrix

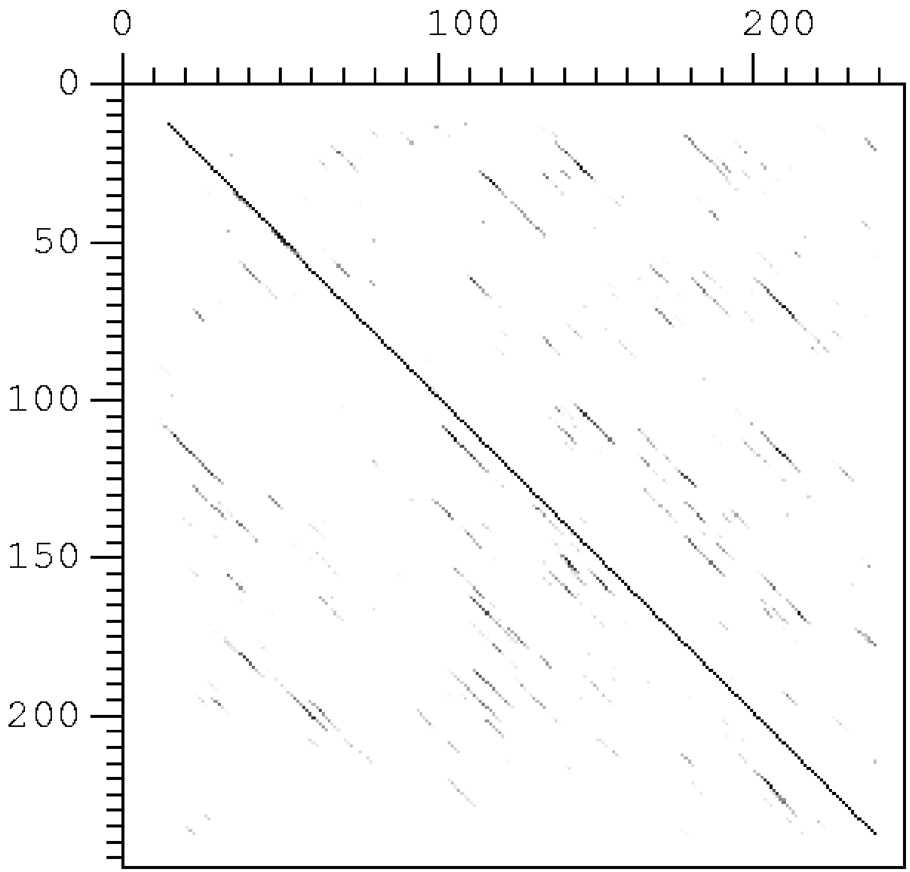
dots on diagonals correspond to matching regions

Example: triosephosphate isomerase (TIM)

- human
- Yeast
- E. coli (bacteria)
- archaeon

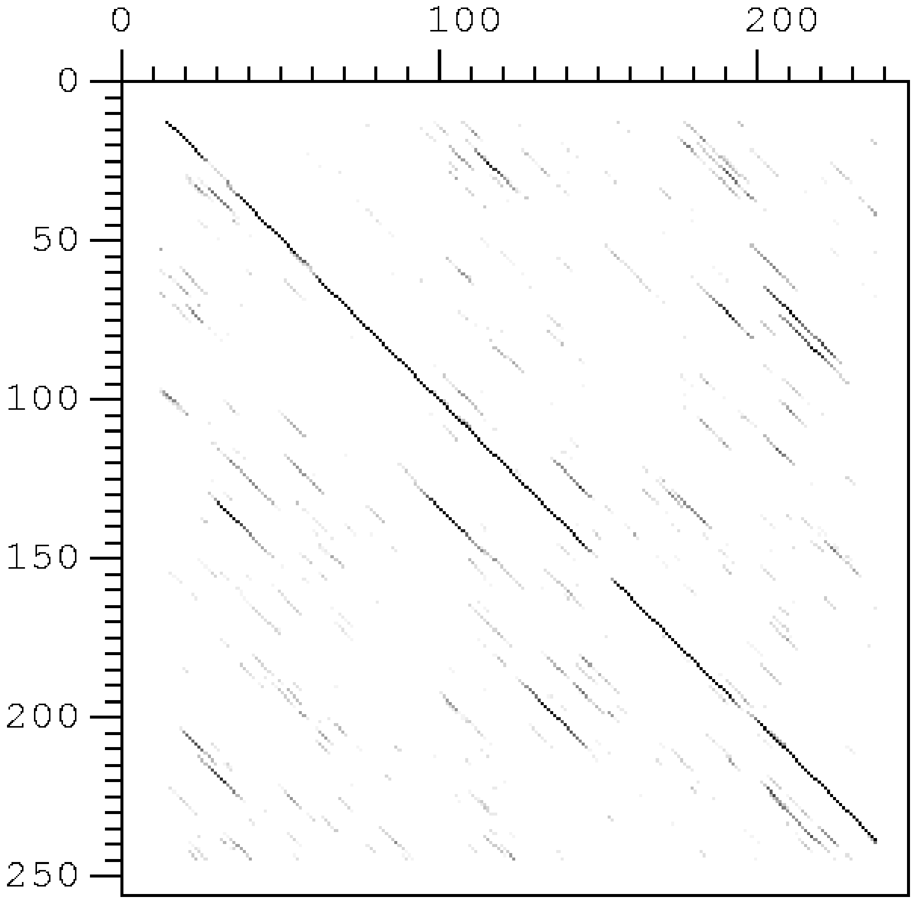
Identity Matrix

TIM-Human (horizontal) vs. TIM-Yeast (vertical)



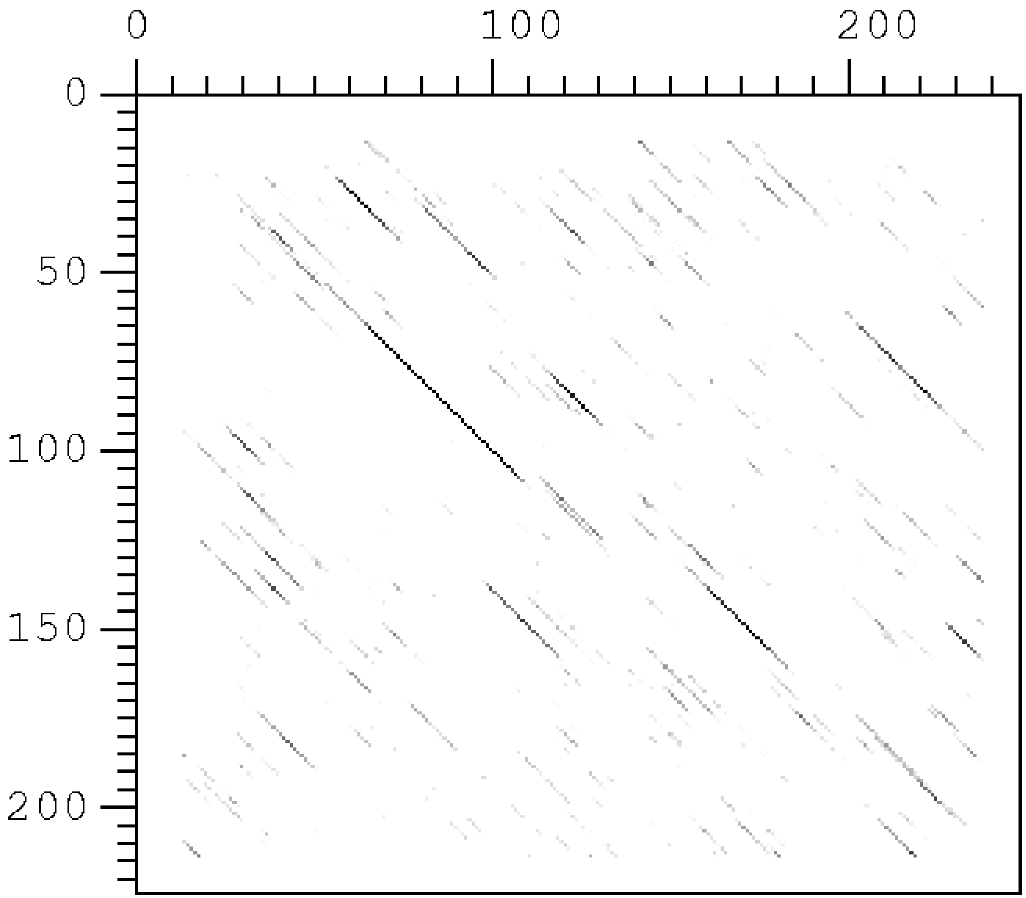
Identity Matrix

TIM-Human (horizontal) vs. TIM-Ecoli (vertical)



Identity Matrix

TIM-Human (horizontal) vs. TIM-Archaeon (vertical)



Identity Matrix

Other scoring schemes also judge the mismatches

Amino acids are more likely to mutate into another amino acid with similar chemical properties (scoring by mutation

Also indels may be differently scored

Identity Matrix

Example scoring without identity (gap is higher penalized)

Position *l*:

$$s(x_l, y_l) = \begin{cases} +2 & \text{for} \quad x_l = y_l \\ -1 & \text{for} \quad x_l \neq y_l \\ -2 & \text{for} \quad x_l \text{ or } y_l \ \text{blank} \end{cases}$$

Scoring

x_l	T	C	A	G	A	C	A
y_l	T	–	–	G	A	T	–
$s(x_l, y_l)$	2	–2	–2	2	2	–1	–2

Whole scoring

$$\sum_l s(x_l, y_l) = -1$$

PAM Matrices

Point Accepted Mutation (PAM) matrices: Dayhoff et. al (1978)

nPAM:
- n% mutations on average per position
- 1 PAM = 1 point mutation/100 amino acids
- nPAM = $(1\text{PAM})^n$
- n is time unit (measured in the time interval to obtain 1 mutation)

Symmetric Markov matrices:

$$P = \begin{pmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,20} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,20} \\ \vdots & \vdots & \ddots & \vdots \\ p_{20,1} & p_{20,2} & \cdots & p_{20,20} \end{pmatrix},$$

where $p_{i,j} \geq 0$ and $\sum_j p_{i,j} = 1$.

PAM Matrices

Construction of PAM:

- 71 subsequences (blocks) with >85% mutual identity
- 1,572 changes
- phylogenetic tree for each block
- $C_{i,j}$: number transition i ➔ j with $C_{i,j} = 0.5\ (C_{i,j} + C_{j,i})$ in the tree (direction of point mutations is ambiguous)
- Mutation probability i ➔ j approximated

$$c_{i,j} = \frac{C_{i,j}}{\sum_l C_{i,l}}$$

PAM Matrices

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A																				
R	30																			
N	109	17																		
D	154	0	532																	
C	33	10	0	0																
Q	93	120	50	76	0															
E	266	0	94	831	0	422														
G	579	10	156	162	10	30	112													
H	21	103	226	43	10	243	23	10												
I	66	30	36	13	17	8	35	0	3											
L	95	17	37	0	0	75	15	17	40	253										
K	57	477	322	85	0	147	104	60	23	43	39									
M	29	17	0	0	0	20	7	7	0	57	207	90								
F	20	7	7	0	0	0	0	17	20	90	167	0	17							
P	345	67	27	10	10	93	40	49	50	7	43	43	4	7						
S	772	137	432	98	117	47	86	450	26	20	32	168	20	40	269					
T	590	20	169	57	10	37	31	50	14	129	52	200	28	10	73	696				
W	0	27	3	0	0	0	0	0	3	0	13	0	0	10	0	17	0			
Y	20	3	36	0	30	0	10	0	40	13	23	10	0	260	0	22	23	6		
V	365	20	13	17	33	27	37	97	30	661	303	17	77	10	50	43	186	0	17	

Table 1: Cumulative Data for computing PAM with 1572 changes.

PAM Matrices

$$p_{i,j} = m_i \; c_{i,j} = m_i \, \frac{C_{i,j}}{\sum_{l,l \neq i} C_{i,l}}$$

$$f_i \; p_{i,j} = f_j \; p_{j,i}$$

Steady state: amino acid distribution remains constant

$$f_i \; m_i \, \frac{C_{i,j}}{\sum_{l,l \neq i} C_{i,l}} = f_j \; m_j \, \frac{C_{i,j}}{\sum_{l,l \neq j} C_{j,l}}$$

$$m_i \, \frac{f_i}{\sum_{l,l \neq i} C_{i,l}} = m_j \, \frac{f_j}{\sum_{l,l \neq j} C_{j,l}} := c$$

$$m_i = c \, \frac{\sum_{l,l \neq i} C_{i,l}}{f_i}$$

PAM Matrices

$$p_{i,j} = c \, \frac{\sum_{l,l\neq i} C_{i,l}}{f_i} \, \frac{C_{i,j}}{\sum_{l,l\neq i} C_{i,l}} = c \, \frac{C_{i,j}}{f_i}$$

$$\sum_i f_i \, (1 - p_{i,i}) = \sum_i \sum_{j\neq i} f_i \, p_{i,j} =$$

$$c \sum_i \sum_{j\neq i} f_i \, \frac{C_{i,j}}{f_i} = c \sum_i \sum_{j\neq i} C_{i,j} = 1/100$$

$$c = 1/\left(100 \sum_i \sum_{j\neq i} C_{i,j}\right)$$

Choose c to obtain 1 mutation per 100 amino acids

$$p_{i,j} = \frac{C_{i,j}}{100 \; f_i \; \sum_l \sum_{k\neq l} C_{l,k}}$$

PAM Matrices

Gly	0.089	Val	0.065	Arg	0.041	His	0.034
Ala	0.087	Thr	0.058	Asn	0.040	Cys	0.033
Leu	0.085	Pro	0.051	Phe	0.040	Tyr	0.030
Lys	0.081	Glu	0.050	Gln	0.038	Met	0.015
Ser	0.070	Asp	0.047	Ile	0.037	Trp	0.010

Table 1: Amino acid frequencies according to Dayhoff et. al (1978).

PAM Matrices

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	9867	2	9	10	3	8	17	21	2	6	4	2	6	2	22	35	32	0	2	18
R	1	9913	1	0	1	10	0	0	10	3	1	19	4	1	4	6	1	8	0	1
N	4	1	9822	36	0	4	6	6	21	3	1	13	0	1	2	20	9	1	4	1
D	6	0	42	9859	0	6	53	6	4	1	0	3	0	0	1	5	3	0	0	1
C	1	1	0	0	9973	0	0	0	1	1	0	0	0	0	1	5	1	0	3	2
Q	3	9	4	5	0	9876	27	1	23	1	3	6	4	0	6	2	2	0	0	1
E	10	0	7	56	0	35	9865	4	2	3	1	4	1	0	3	4	2	0	1	2
G	21	1	12	11	1	3	7	9935	1	0	1	2	1	1	3	21	3	0	0	5
H	1	8	18	3	1	20	1	0	9912	0	1	1	0	2	3	1	1	1	4	1
I	2	2	3	1	2	1	2	0	0	9872	9	2	12	7	0	1	7	0	1	33
L	3	1	3	0	0	6	1	1	4	22	9947	2	45	13	3	1	3	4	2	15
K	2	37	25	6	0	12	7	2	2	4	1	9926	20	0	3	8	11	0	1	1
M	1	1	0	0	0	2	0	0	0	5	8	4	9874	1	0	1	2	0	0	4
F	1	1	1	0	0	0	0	1	2	8	6	0	4	9946	0	2	1	3	28	0
P	13	5	2	1	1	8	3	2	5	1	2	2	1	1	9926	12	4	0	0	2
S	28	11	34	7	11	4	6	16	2	2	1	7	4	3	17	9840	38	5	2	2
T	22	2	13	4	1	3	2	2	1	11	2	8	6	1	5	32	9871	0	2	9
W	0	2	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	9976	1	0
Y	1	0	3	0	3	0	1	0	4	1	1	0	0	21	0	1	1	2	9945	1
V	13	2	1	1	3	2	2	3	3	57	11	1	17	1	3	2	10	0	2	9901

Table 1: 1 PAM evolutionary distance (times 10000).

PAM Matrices

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	13	6	9	9	5	8	9	12	6	8	6	7	7	4	11	11	11	2	4	9
R	3	17	4	3	2	5	3	2	6	3	2	9	4	1	4	4	3	7	2	2
N	4	4	6	7	2	5	6	4	6	3	2	5	3	2	4	5	4	2	3	3
D	5	4	8	11	1	7	10	5	6	3	2	5	3	1	4	5	5	1	2	3
C	2	1	1	1	52	1	1	2	2	2	1	1	1	1	2	3	2	1	4	2
Q	3	5	5	6	1	10	7	3	7	2	3	5	3	1	4	3	3	1	2	3
E	5	4	7	11	1	9	12	5	6	3	2	5	3	1	4	5	5	1	2	3
G	12	5	10	10	4	7	9	27	5	5	4	6	5	3	8	11	9	2	3	7
H	2	5	5	4	2	7	4	2	15	2	2	3	2	2	3	3	2	2	3	2
I	3	2	2	2	2	2	2	2	2	10	6	2	6	5	2	3	4	1	3	9
L	6	4	4	3	2	6	4	3	5	15	34	4	20	13	5	4	6	6	7	13
K	6	18	10	8	2	10	8	5	8	5	4	24	9	2	6	8	8	4	3	5
M	1	1	1	1	0	1	1	1	1	2	3	2	6	2	1	1	1	1	1	2
F	2	1	2	1	1	1	1	1	3	5	6	1	4	32	1	2	2	4	20	3
P	7	5	5	4	3	5	4	5	5	3	3	4	3	2	20	6	5	1	2	4
S	9	6	8	7	7	6	7	9	6	5	4	7	5	3	9	10	9	4	4	6
T	8	5	6	6	4	5	5	6	4	6	4	6	5	3	6	8	11	2	3	6
W	0	2	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	55	1	0
Y	1	1	2	1	3	1	1	1	3	2	2	1	2	15	1	2	2	3	31	2
V	7	4	4	4	4	4	4	4	5	4	15	10	4	10	5	5	5	72	4	17

Table 1: 250 PAM evolutionary distance (times 100).

PAM Matrices

How do we score?

The likelihood ratio compares a random pairing to a pairing resulting from mutations:

$$\frac{f_i \ p_{i,j}}{f_i \ f_j} = \frac{p_{i,j}}{f_j} = \frac{p_{j,i}}{f_i}$$

$$\begin{aligned} p_{i,j} &\cong p(j \mid i) \\ f_i &\cong p(i) \\ f_i \ p_{i,j} &\cong p(i,j) \end{aligned}$$

Positions are independent from each other then for the whole sequence:

$$\prod_k \frac{f_{i_k} \ p_{i_k,j_k}}{f_{i_k} \ f_{j_k}} = \prod_k \frac{p_{i_k,j_k}}{f_{j_k}}$$

and taking the logarithm:

$$\sum_k \log\left(\frac{p_{i_k,j_k}}{f_{j_k}}\right)$$

PAM Matrices

The values $\log\left(\frac{p_{i_k,j_k}}{f_{j_k}}\right)$ are called “log-odd-scores”

Multiplied by a constant and rounded gives the PAM “log-odd-scores”

Positive “log-odd-scores”: pair of amino acids appears more often in aligned homologous sequences than by chance

PAM Matrices

3 Pairwise Alignment
3.1 Motivation
3.2 Scoring
3.2.1 Identity
3.2.2 PAM
3.2.3 BLOSUM
3.2.4 Gap Penalties
3.3 Algorithms
3.3.1 Global
3.3.2 Local
3.3.3 FASTA, BLAST
3.4 Significance
3.4.1 HSPs
3.4.2 Matches

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

PAM250 “log-odd-scores”

BLOSUM Matrices

3 Pairwise Alignment
3.1 Motivation
3.2 Scoring
3.2.1 Identity
3.2.2 PAM
3.2.3 BLOSUM
3.2.4 Gap Penalties
3.3 Algorithms
3.3.1 Global
3.3.2 Local
3.3.3 FASTA, BLAST
3.4 Significance
3.4.1 HSPs
3.4.2 Matches

PAM matrices: very similar sequences and generalized to less similar by matrix multiplication

BLOSUM (BLOck SUbstitution Matrix, Henikoff and Henikoff, 1992) is based on the Blocks database

BLOSUM directly determines the similarity

BLOSUM p: p% identity of the blocks

BLOSUM 62 (62% identity) is most popular

BLOSUM Matrices

Calculation of the BLOSUM matrices:

1. Sequences with at least p% identity are clustered. Each cluster provides a sequence of frequencies. In the following we only consider the case without frequencies.
2. Frequency sequences are compared and pairs (i,j) counted by $c_{i,j}$ according to

column k: n_i^k amino acids i and n_j^k amino acids j

$$c_{i,j}^k = \begin{cases} \binom{n_i^k}{2} & \text{for} \quad i = j \\ n_i^k n_j^k & \text{for} \quad i > j \end{cases}$$

$$\binom{n_i^k}{2} = \frac{1}{2}\left(n_i^k \, n_i^k - n_i^k\right)$$

where the factor 1/2 accounts for symmetry and $-n_i^k$ subtracts the original sequence (no mutation)

BLOSUM Matrices

3. For N sequences of length L compute

$$c_{i,j} = \sum_k c^k_{i,j} \ , \quad Z = \sum_{i,j<i} c_{i,j} = \frac{L \ N \ (N-1)}{2}$$

$$q_{i,j} = \frac{c_{i,j}}{Z} \ , \quad q_{j,i} = q_{i,j} \quad \forall i > j \qquad \boxed{q_{i,j} = 2p(i,j) \ , \quad p_{i,j} = p(j|i)}$$

4. probability to observe amino acid i is $q_i = q_{i,i} + \sum_{j \neq i} \frac{q_{i,j}}{2}$

$q_{i,j}$ is divided by 2: mutations from i to j and j to i in step 2.

5. Likelihood ratios and the log-odd ratios

$$\frac{q_{i,i}}{q_i^2} \ , \quad \frac{q_{i,j}/2}{q_i \ q_j} \ , \quad \mathrm{BLOSUM}_{i,j} = \begin{cases} 2\log_2 \frac{q_{i,i}}{q_i^2} & \text{for} \quad i = j \\ 2\log_2 \frac{q_{i,j}}{2 \ q_i \ q_j} & \text{for} \quad i \neq j \end{cases}$$

BLOSUM values are rounded to integers

BLOSUM Matrices

example for computing BLOSUM100 matrix

1	NFHV
2	DFNV
3	DFKV
4	NFHV
5	KFHR

$$\begin{array}{cccccccc} & \mathrm{R} & \mathrm{N} & \mathrm{D} & \mathrm{H} & \mathrm{K} & \mathrm{F} & \mathrm{V} \\ \mathrm{R} & 0 & - & - & - & - & - & - \\ \mathrm{N} & 0 & 1 & - & - & - & - & - \\ \mathrm{D} & 0 & 4 & 1 & - & - & - & - \\ \mathrm{H} & 0 & 3 & 0 & 3 & - & - & - \\ \mathrm{K} & 0 & \textcircled{3} & 2 & 3 & 0 & - & - \\ \mathrm{F} & 0 & 0 & 0 & 0 & 0 & 10 & - \\ \mathrm{V} & 4 & 0 & 0 & 0 & 0 & 0 & 6 \end{array}$$

1x2+1x1

$$Z = 4 \cdot \frac{5 \cdot 4}{2} = 40 = \sum_{i \geq j} c_{i,j}$$

BLOSUM Matrices

	R	N	D	H	K	F	V
R	0	0	0	0	0	0	0.1
N	0	0.025	0.1	0.075	0.075	0	0
D	0	0.1	0.025	0	0.05	0	0
H	0	0.075	0	0.075	0.075	0	0
K	0	0.075	0.05	0.075	0	0	0
F	0	0	0	0	0	0.25	0
V	0.1	0	0	0	0	0	0.15

R	0.05
N	0.15
D	0.1
H	0.15
K	0.1
F	0.25
V	0.2

$$\texttt{N:}\ 0.025 + \frac{1}{2}(0.1 + 0.075 + 0.075) = 0.15$$

BLOSUM Matrices

3 Pairwise Alignment
3.1 Motivation
3.2 Scoring
3.2.1 Identity
3.2.2 PAM
3.2.3 BLOSUM
3.2.4 Gap Penalties
3.3 Algorithms
3.3.1 Global
3.3.2 Local
3.3.3 FASTA, BLAST
3.4 Significance
3.4.1 HSPs
3.4.2 Matches

	R	N	D	H	K	F	V
R	-	-	-	-	-	-	5
N	-	1.1	3.3	1.7	2.5	-	-
D	-	3.3	2.5	-	2.5	-	-
H	-	1.7	-	3.3	2.5	-	-
K	-	2.5	2.5	2.5	-	-	-
F	-	-	-	-	-	4	-
V	5	-	-	-	-	-	3.8

likelihood ratio

	R	N	D	H	K	F	V
R	-	-	-	-	-	-	4.6
N	-	0.3	3.5	1.5	2.6	-	-
D	-	3.5	3.4	-	2.6	-	-
H	-	1.5	-	3.4	2.6	-	-
K	-	2.6	2.6	2.6	-	-	-
F	-	-	-	-	-	4	-
V	4.6	-	-	-	-	-	3.8

log-odd ratios

BLOSUM Matrices

Now we consider clusters and frequencies

$f^k_{i,l}$: frequency (amino acid i, kth column, lth cluster)

$$
\begin{aligned}
c^k_{i,j} &= \sum_{l,m:l\neq m} f^k_{i,l}\ f^k_{j,m} = \\
&\sum_{l} f^k_{i,l} \sum_{m:m\neq l} f^k_{j,m} = \\
&n^k_i\ n^k_j - \sum_{l} f^k_{i,l}\ f^k_{j,l}\ , \qquad n^k_i = \sum_{l} f^k_{i,l}
\end{aligned}
$$

$$
c^k_{i,i} = \frac{1}{2}\left(\left(n^k_i\right)^2 - \sum_{l}\left(f^k_{i,l}\right)^2\right)
$$

Other computations remain the same

BLOSUM Matrices

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

BLOSUM62

Scoring matrix

BLOSUM Matrices

BLOSUM and PAM compared:

- PAM100 ≈ BLOSUM90
- PAM120 ≈ BLOSUM80
- PAM160 ≈ BLOSUM60
- PAM200 ≈ BLOSUM52
- PAM250 ≈ BLOSUM45

PAM:
- context dependent, dependency between substitutions
- low probability mutations are not as well observed
- subsequences of very similar sequences (bias to mutation)

BLOSUM:
- not model based
- evolutionary relationships not considered

BLOSUM Matrices

sequence similarities pointwise

more complex scores?

simple scores lead to efficient algorithms

Gap Penalties

```
BIOINFORMATICS              BIOI--N-FORMATICS
                      ⟶
BOILING FOR MANICS          B-OILINGFORMANICS
                                         ↑
```

Gap: maximal substring of „-“

gaps contribute negatively to the score but how?

linear gap penalty: - l d (l is gap length, d is cost)

Gap Penalties

However: neighboring indels may result from a single mutation and are statistically not independent

Sequence with introns and exons may be compared to a measured sequence (x-ray, NMR) --> missing introns

Affine gap penalty: $-d-(l-1)e$

d: *gap open penalty*
e: *gap extension penalty*

Gap Penalties

Examples with BLOSUM62 as scoring matrix

➥ d=20 and e=1:

```
RKFFVGGNWKMNGDKKSLNGAKLSADTEVVCGAPSIYLDF
|.||||||:|       ||.|.:..:.|||...|:.|||:
RTFFVGGNFK-------LNTASIPENVEVVICPPATYLDY
```

➥ d=1 and e=1 (linear gap penalty):

```
RKFFVGGNWKMNGDKKSL--NGAKLSADTEVV-CGAPSIYLDF
|.||||||:|:|  ..|:  |   :    ||| | .|:.|||:
RTFFVGGNFKLN--TASIPEN---V----EVVIC-PPATYLDY
```

➥ d=4 and e=4 (linear gap penalty):

```
RKFFVGGNWKMNGDKKSLNGAKLSADTEVVCGAPSIYLDF
|.||||||:|:|  ..|:  .: :.:. |:| .|:.|||:
RTFFVGGNFKLN--TASI--PE-NVEV-VIC-PPATYLDY
```

Few gaps: e<d; Few “-”: high gap penalty compared to BLOSUM

Gap Penalties

no amino acid has a preference for a gap (is that true?).

Likelihood: opening 2^{-d} , extending 2^{-e}, then the $\log_2$-likelihood is affine gap penalty (log-odds framework)

Alignment Algorithms

alignment algorithms: global or local

Local alignment: finds similarities

↪ if (alternative) splicing occurs

↪ if domains are glued together

↪ if conserved regions exist in remote homologous sequences

Global Alignment – Needleman-Wunsch

How did you solve the simple game?
try out all alignments, compute their scores and choose the best

Two sequences of length n:
↪ $\binom{n}{i}$ sequences of length i for each sequence

↪ $\binom{n}{i}^2$ pairs compared in i steps

↪ to compare all subsequences:

$$\sum_i \binom{n}{i}^2 i \ge \sum_i \binom{n}{i}^2 = \binom{2n}{n} \approx$$

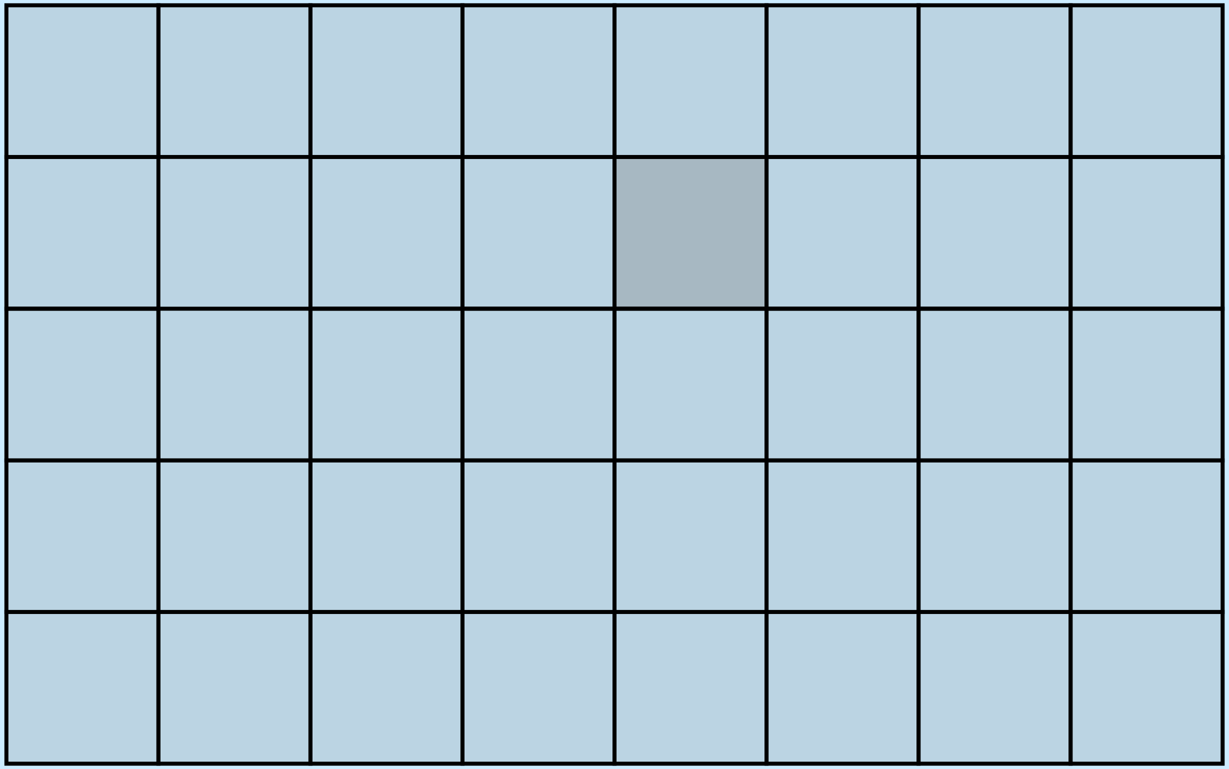
Stirling's formula

$$\sqrt{4\,\pi\,n}\;(2n/e)^{2n} / \left(\sqrt{2\pi\,n}\;(n/e)^n\right)^2 = 2^{2n}/\sqrt{\pi\,n}$$

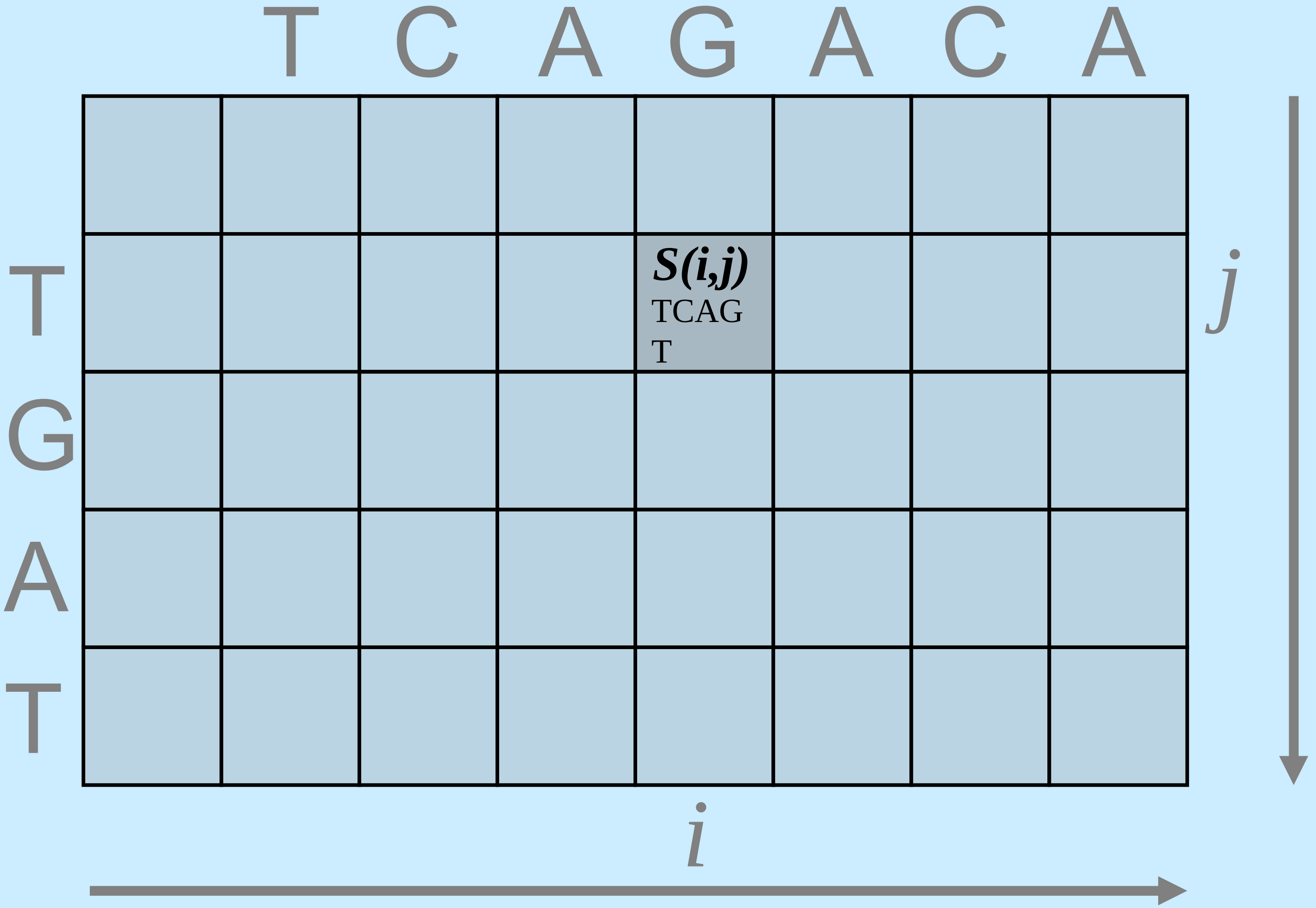
number of operation exponentially with the sequence length

Global Alignment – Needleman-Wunsch

Matrix of optimal score



Global Alignment – Needleman-Wunsch



Global Alignment – Needleman-Wunsch

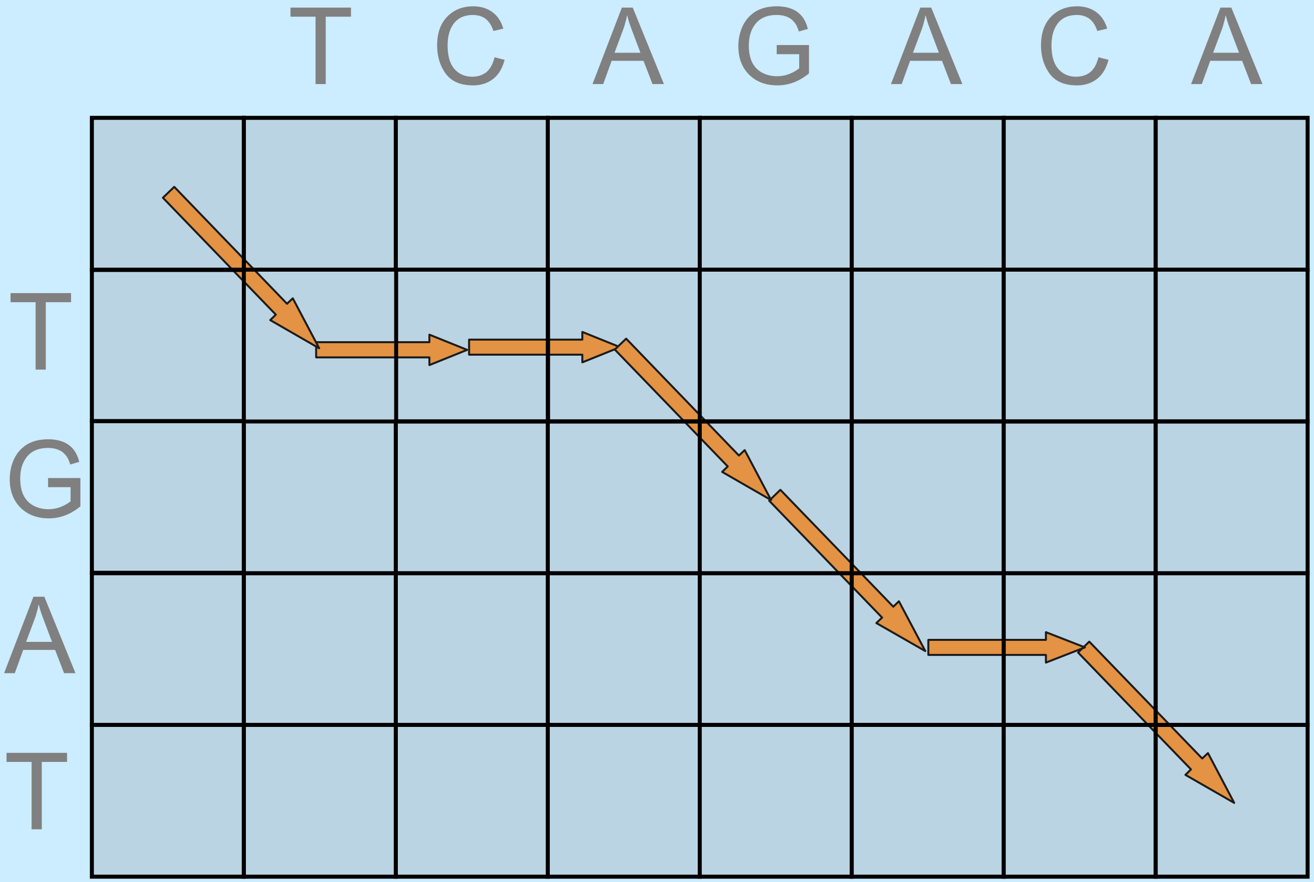
3 Pairwise Alignment
3.1 Motivation
3.2 Scoring
3.2.1 Identity
3.2.2 PAM
3.2.3 BLOSUM
3.2.4 Gap Penalties
3.3 Algorithms
3.3.1 Global
3.3.2 Local
3.3.3 FASTA, BLAST
3.4 Significance
3.4.1 HSPs
3.4.2 Matches

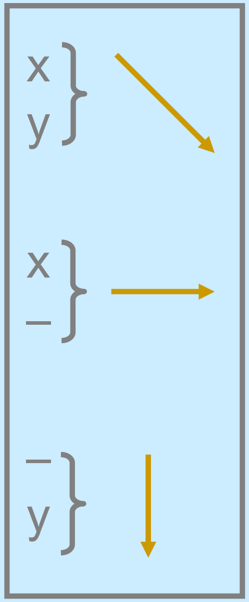
		T	C	A	G	A	C	A
T								
G								
A								
T								

Each path through matrix is an alignment

Goal: Optimal score and path of the score

Global Alignment – Needleman-Wunsch

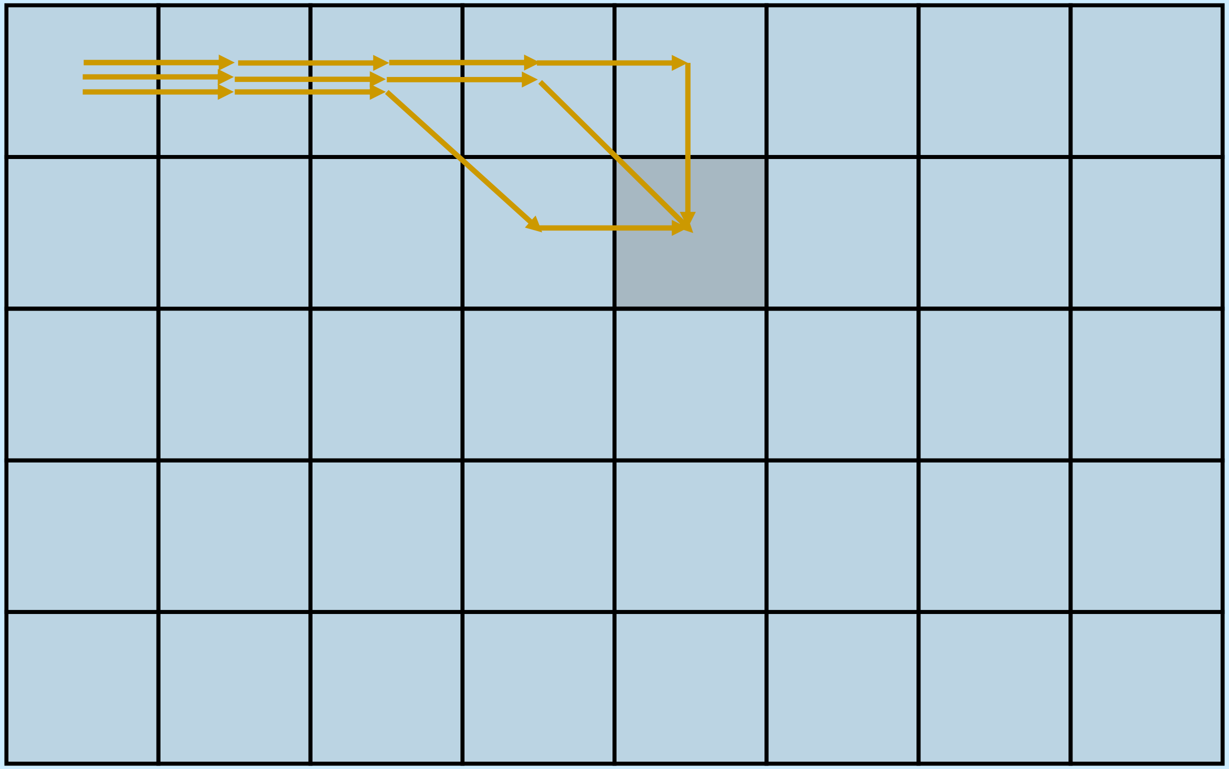




y T C A G A C A
x T – – G A – T
(i,j) (0,0) (1,1) (2,1) (3,1) (4,2) (5,3) (6,3) (7,4)

Global Alignment – Needleman-Wunsch

Recursion to compute optimal score plus path



Problem: multiple computations ⇒ exponential complexity

Global Alignment – Needleman-Wunsch

1970 Needleman and Wunsch: Dynamic Programming

alignment of two sequences of length n and m can be reduced to the alignment of two shorter sequences

match	gap	gap
x	?x	x−
y	y−	?y

x = T C A G A C A   **y** = T G A T

```
T C A G A C A      T C A G A C A      T C A G A C A −
T − − G A − T      T − − G A T −      T − − G A − − T
```

Either the ends match or the end of one sequence is more to the right then the end of the other sequence.

Global Alignment – Needleman-Wunsch

3 Pairwise Alignment
3.1 Motivation
3.2 Scoring
3.2.1 Identity
3.2.2 PAM
3.2.3 BLOSUM
3.2.4 Gap Penalties
3.3 Algorithms
3.3.1 Global
3.3.2 Local
3.3.3 FASTA, BLAST
3.4 Significance
3.4.1 HSPs
3.4.2 Matches

optimal score $S(n, m)$

$\boldsymbol{x}$: 1.sequence with $x_i,\ 1 \leq i \leq n$
$\boldsymbol{y}$: 2. sequence with $y_i,\ 1 \leq i \leq m$
d : gap penalty (linear gap)
s : scoring function

Recursion for S:

$$S(i,j) \;=\; \max \begin{cases} S(i-1,j-1) \;+\; s(x_i,y_j) \\ S(i-1,j) \;-\; d \\ S(i,j-1) \;-\; d \end{cases}$$

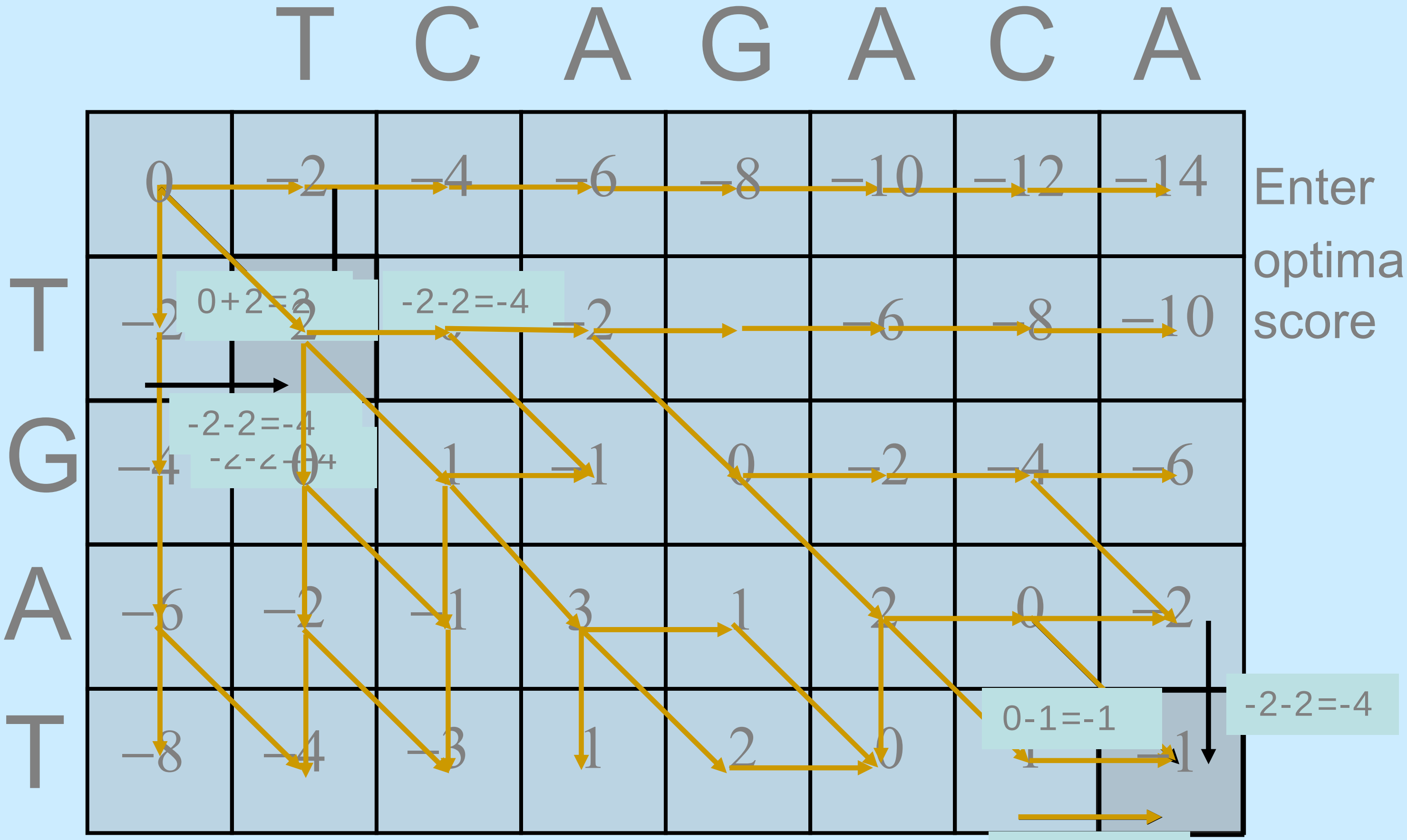
$$S(0,0) \;=\; 0 \;\text{ and }\; S(-1,j) \;=\; S(i,-1) \;=\; -\infty$$

$$\Rightarrow S(0,j) = -\; j\; d \quad \text{and} \quad S(i,0) = -\; i\; d$$

Global Alignment – Needleman-Wunsch

$$
\begin{array}{r|cccccccc|}
 & 0 & x_1 & \dots & x_{i-1} & & x_i & \dots & x_n \\
\hline
0 & S(0,0) & S(1,0) & \dots & & & & \dots & S(n,0) \\
y_1 & S(0,1) & S(1,1) & \dots & & & & \dots & S(n,1) \\
y_2 & & & \dots & & & & \dots & \\
y_3 & & & \dots & & & & \dots & \\
\vdots & \vdots & \vdots & \ddots & \vdots & & \vdots & \ddots & \vdots \\
y_{j-1} & & & & S(i-1,j-1) & & S(i,j-1) & & \\
 & & & & & \searrow & \downarrow & & \\
y_j & & & & S(i-1,j) & \rightarrow & S(i,j) & & \\
\vdots & \vdots & \vdots & \ddots & \vdots & & \vdots & \ddots & \vdots \\
y_m & S(0,m) & S(1,m) & \dots & & & & \dots & S(n,m) \\
\hline
\end{array}
$$

Global Alignment – Needleman-Wunsch



		T	C	A	G	A	C	A
	0	−2	−4	−6	−8	−10	−12	−14
T	−2	2	0	−2		−6	−8	−10
G	−4	0	1	−1	0	−2	−4	−6
A	−6	−2	−1	3	1	2	0	−2
T	−8	−4	−3	1	2	0	1	−1

Enter optimal score

$$s(x_l,y_l) = \begin{cases} +2 \text{ for } x_l = y_l: \searrow \\ -1 \text{ for } x_l \neq y_l: \searrow \\ -2 \text{ for } x_l, y_l = „–“: \downarrow \text{ and } \rightarrow \end{cases}$$

$$u = u_{shortend} + s(x_l, y_l)$$

maximum

Global Alignment – Needleman-Wunsch

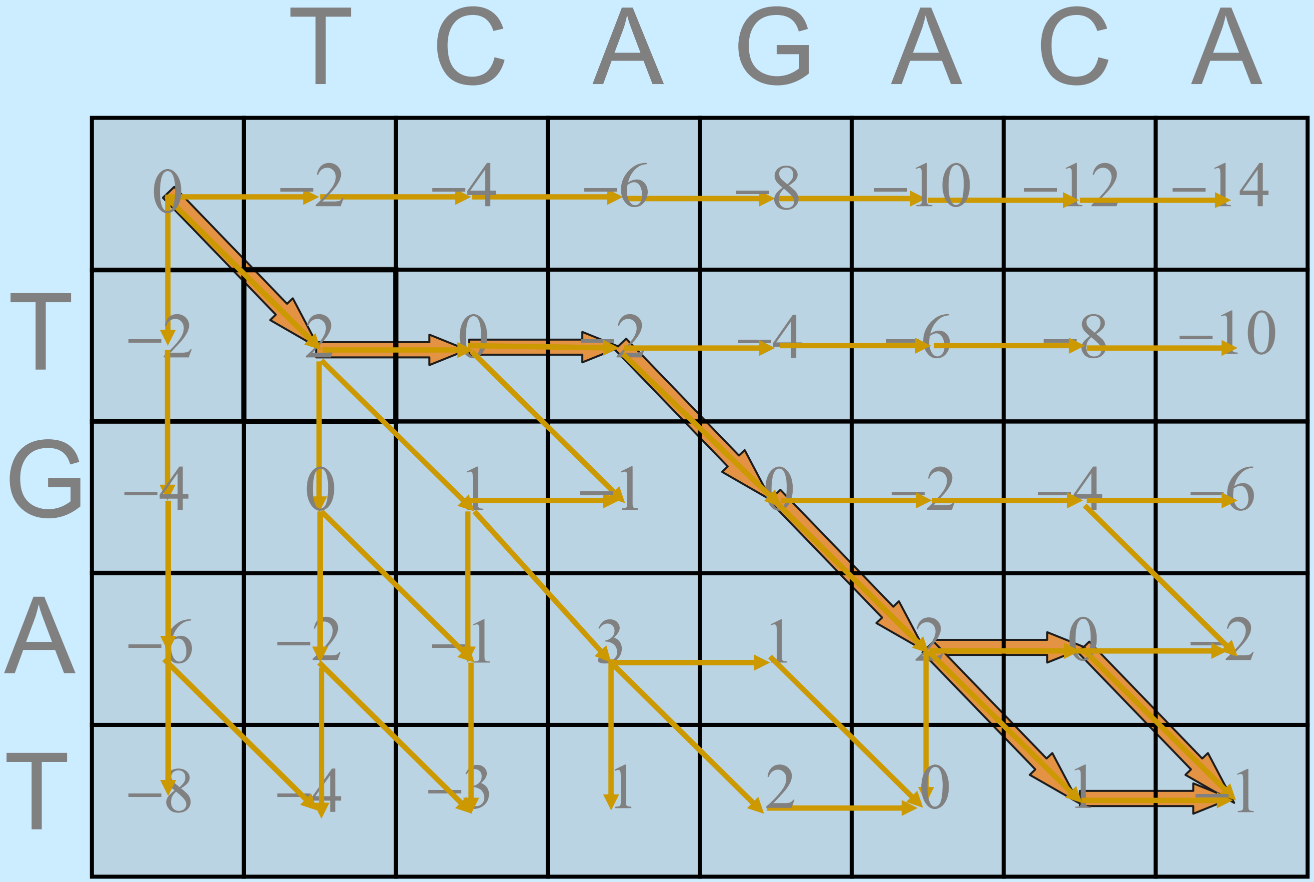
During filling the matrix the path must be memorized. The $S(i-1,j-1), S(i-1,j), S(i,j-1)$ from which $S(i,j)$ was computed must be stored in a variable B:

$$B(i,j) = (i-1,j-1) \quad \text{or} \quad (i-1,j) \quad \text{or} \quad (i,j-1)$$

This variable allows to generate the alignment through backtracking starting from (n,m):

$$\textbf{if} \quad B(i,j) \;=\; \begin{cases} (i-1,j-1) & \textbf{then print} \quad \begin{array}{c} x_i \\ y_j \end{array} \\[2ex] (i-1,j) & \textbf{then print} \quad \begin{array}{c} x_i \\ - \end{array} \\[2ex] (i,j-1) & \textbf{then print} \quad \begin{array}{c} - \\ y_j \end{array} \end{cases}$$

Global Alignment – Needleman-Wunsch



		T	C	A	G	A	C	A
	0	−2	−4	−6	−8	−10	−12	−14
T	−2	2	0	−2	−4	−6	−8	−10
G	−4	0	1	−1	0	−2	−4	−6
A	−6	−2	−1	3	1	2	0	−2
T	−8	−4	−3	1	2	0	1	−1

„back-tracking“

x
y } ↘

x
– } →

–
y } ↓

T C A G A C A
T – – G A – T

T C A G A C A
T – – G A T –

Global Alignment – Needleman-Wunsch

Algorithm 1 Needleman-Wunsch with linear gap

Input: two sequences $\boldsymbol{x}$ and $\boldsymbol{y}$ with length n and m, respectively; scoring matrix s, gap penalty d

Output: optimal global alignment and its score

```
BEGIN INITIALIZATION
  $S(0,0) = 0,\ S(0,j) = -\ j\ d,\ 1 \leq j \leq m$, and $S(i,0) = -\ i\ d,\ 1 \leq i \leq n$
END INITIALIZATION

BEGIN PROCEDURE
  for $1 \leq i \leq n$ do
    for $1 \leq j \leq m$ do
      $a(i-1,j-1) = S(i-1,j-1) + s(x_i,y_j)$ , $a(i-1,j) = S(i-1,j) - d$,
      $a(i,j-1) = S(i,j-1) - d$
      $S(i,j) = \max\{a(i-1,j-1), a(i-1,j), a(i,j-1)\}$
      $B(i,j) = \arg\max\{a(i-1,j-1), a(i-1,j), a(i,j-1)\}$
    end for
  end for
  print "Score: " $S(n,m)$

  $(i,j) = (n,m)$
  while $(i,j) \neq (0,0)$ do
      if $B(i,j) = \begin{cases} (i-1,j-1) & \textbf{then print } \begin{smallmatrix} x_i \\ y_j \end{smallmatrix} \\ (i-1,j) & \textbf{then print } \begin{smallmatrix} x_i \\ - \end{smallmatrix} \\ (i,j-1) & \textbf{then print } \begin{smallmatrix} - \\ y_j \end{smallmatrix} \end{cases}$
    $(i,j) = B(i,j)$
  end while
END PROCEDURE
```

Global Alignment – Needleman-Wunsch

	G	A	K	L	S	A	D	T	E	V	V	C	G	A	P	S	I	Y	L	D	F
	-20	-21	-22	-23	-24	-25	-26	-27	-28	-29	-30	-31	-32	-33	-34	-35	-36	-37	-38	-39	-40
R	-14	-15	-16	-17	-18	-19	-20	-21	-22	-23	-24	-25	-26	-27	-28	-29	-30	-31	-32	-33	-34
T	-12	-13	-14	-15	-16	-17	-18	-15	-16	-17	-18	-19	-20	-21	-22	-23	-24	-25	-26	-27	-28
F	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19	-20	-21	-22	-23	-21	-22	-23	-21
F	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19	-17
V	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15
G	12	11	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8
G	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	-1
N	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6
F	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	13	12	11	13
K	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14
L	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19
N	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26
T	46	46	45	44	44	43	42	45	44	43	42	41	40	39	38	37	36	35	34	33	32
A	47	50	49	48	47	48	47	46	45	44	43	42	41	44	43	42	41	40	39	38	37
S	49	49	50	49	52	51	50	49	48	47	46	45	44	43	43	47	46	45	44	43	42
I	52	51	50	52	51	51	50	49	48	51	50	49	48	47	46	46	51	50	49	48	47
P	51	51	50	51	51	50	50	49	48	50	49	48	47	47	54	53	52	51	50	49	48
E	52	51	52	51	51	50	52	51	54	53	52	51	50	49	53	54	53	52	51	52	51
N	←**57**	←**56**	←**55**	54	53	52	51	52	53	51	50	49	51	50	52	54	53	52	51	52	51
V	55	57	56	↖**56**	←**55**	←**54**	←**53**	←**52**	52	57	56	55	54	53	52	53	57	56	55	54	53
E	55	56	58	57	56	55	56	55	↖**57**	56	55	54	53	53	52	52	56	55	54	57	56
V	54	55	57	59	58	57	56	56	56	↖**61**	60	59	58	57	56	55	55	55	56	56	56
V	53	54	56	58	57	58	57	56	55	60	↖**65**	64	63	62	61	60	59	58	57	56	55
I	52	53	55	58	57	57	55	56	55	59	↑**64**	64	63	62	61	60	64	63	62	61	60
C	51	52	54	57	57	57	56	55	54	58	63	↖**73**	←**72**	71	70	69	68	67	66	65	64
P	50	51	53	56	56	56	56	55	54	57	62	72	71	↖**71**	78	77	76	75	74	73	72
P	49	50	52	55	55	55	55	55	54	56	61	71	70	70	↖**78**	77	76	75	74	73	72
A	50	53	52	54	56	59	58	57	56	55	60	70	71	74	77	↖**79**	78	77	76	75	74
T	49	52	52	53	55	58	58	63	62	61	60	69	70	73	76	78	↖**78**	77	76	75	74
Y	48	51	51	52	54	57	57	62	61	61	60	68	69	72	75	77	77	↖**85**	84	83	82
L	47	50	50	55	54	56	56	61	60	62	62	67	68	71	74	76	79	84	↖**89**	88	87
D	47	49	49	54	55	55	62	61	63	62	61	66	67	70	73	75	78	83	88	↖**95**	94
Y	45	48	48	53	54	54	61	60	62	62	61	65	66	69	72	74	77	85	87	94	↖**98**

Global Alignment – Needleman-Wunsch

		R	K	F	F	V	G	G	N	W	K	M	N	G	D	K	K	S	L	N	G
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19	-20
R	-1	↖**5**	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14
T	-2	4	↖**4**	3	2	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-9	-10	-11	-12
F	-3	3	3	↖**10**	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
F	-4	2	2	9	↖**16**	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V	-5	1	1	8	15	↖**20**	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5
G	-6	0	0	7	14	19	↖**26**	25	24	23	22	21	20	19	18	17	16	15	14	13	12
G	-7	-1	-1	6	13	18	25	↖**32**	31	30	29	28	27	26	25	24	23	22	21	20	19
N	-8	-2	-1	5	12	17	24	31	↖**38**	37	36	35	34	33	32	31	30	29	28	27	26
F	-9	-3	-2	5	11	16	23	30	37	↖**39**	38	37	36	35	34	33	32	31	30	29	28
K	-10	-4	2	4	10	15	22	29	36	38	↖**44**	43	42	41	40	39	38	37	36	35	34
L	-11	-5	1	3	9	14	21	28	35	37	43	↖**46**	45	44	43	42	41	40	41	40	39
N	-12	-6	0	2	8	13	20	27	34	36	42	45	↖**52**	←**51**	←**50**	49	48	47	46	47	46
T	-13	-7	-1	1	7	12	19	26	33	35	41	44	51	50	50	↖**49**	48	49	48	47	46
A	-14	-8	-2	0	6	11	18	25	32	34	40	43	50	51	50	49	↖**48**	49	48	47	47
S	-15	-9	-3	-1	5	10	17	24	31	33	39	42	49	50	51	50	49	↖**52**	51	50	49
I	-16	-10	-4	-2	4	9	16	23	30	32	38	41	48	49	50	49	48	51	↖**54**	53	52
P	-17	-11	-5	-3	3	8	15	22	29	31	37	40	47	48	49	49	48	50	↑**53**	52	51
E	-18	-12	-6	-4	2	7	14	21	28	30	36	39	46	47	50	50	50	49	↑**52**	53	52
N	-19	-13	-7	-5	1	6	13	20	27	29	35	38	45	46	49	50	50	51	51	↖**58**	←**57**
V	-20	-14	-8	-6	0	5	12	19	26	28	34	37	44	45	48	49	49	50	52	57	55
E	-21	-15	-9	-7	-1	4	11	18	25	27	33	36	43	44	47	49	50	49	51	56	55
V	-22	-16	-10	-8	-2	3	10	17	24	26	32	35	42	43	46	48	49	48	50	55	54
V	-23	-17	-11	-9	-3	2	9	16	23	25	31	34	41	42	45	47	48	47	49	54	53
I	-24	-18	-12	-10	-4	1	8	15	22	24	30	33	40	41	44	46	47	46	49	53	52
C	-25	-19	-13	-11	-5	0	7	14	21	23	29	32	39	40	43	45	46	46	48	52	51
P	-26	-20	-14	-12	-6	-1	6	13	20	22	28	31	38	39	42	44	45	45	47	51	50
P	-27	-21	-15	-13	-7	-2	5	12	19	21	27	30	37	38	41	43	44	44	46	50	49
A	-28	-22	-16	-14	-8	-3	4	11	18	20	26	29	36	37	40	42	43	45	45	49	50
T	-29	-23	-17	-15	-9	-4	3	10	17	19	25	28	35	36	39	41	42	44	44	48	49
Y	-30	-24	-18	-14	-10	-5	2	9	16	19	24	27	34	35	38	40	41	43	43	47	48
L	-31	-25	-19	-15	-11	-6	1	8	15	18	23	26	33	34	37	39	40	42	47	46	47
D	-32	-26	-20	-16	-12	-7	0	7	14	17	22	25	32	33	40	39	39	41	46	48	47
Y	-33	-27	-21	-17	-13	-8	-1	6	13	16	21	24	31	32	39	38	38	40	45	47	45

Global Alignment – Needleman-Wunsch

Affine Gap Penalty

Problem: long term dependencies

Introducing a gap implies a gap opening event earlier in the sequence and all earlier events must be considered:

$$S(i,j) \;=\; \max \begin{cases} S(i-1,j-1) \;+\; s(x_i,y_j) \\ S(i-k,j) \;-\; d \;-\; (k-1)\,e\;, \quad 1 \leq k \leq i \\ S(i,j-k) \;-\; d \;-\; (k-1)\,e\;, \quad 1 \leq k \leq j \end{cases}$$

two sequences of length n: complexity $O\left(n^3\right)$

because all $S(i,j)$ must considered ($O\left(n^2\right)$) and $O(n)$ for checking all previous gap openings

Global Alignment – Needleman-Wunsch

Idea: propagate 3 matrices

- best score up to position (i,j): $\boxed{G_d(i,j)}$
- best score up to position (i,j) with an opened gap in $\boldsymbol{x}$ at position i: $\boxed{G_x(i,j)}$
- best score up to position (i,j) with an opened gap in $\boldsymbol{y}$ at position j: $\boxed{G_y(i,j)}$

For G_x and G_y it must be checked whether extending an existing gap or to introduce a new gap gives a better score

Global Alignment – Needleman-Wunsch

recursion equations:

$$G_y(i,j) = \max \begin{cases} G_d(i-1,j) & - \; d \\ G_y(i-1,j) & - \; e \end{cases},$$

$$G_x(i,j) = \max \begin{cases} G_d(i,j-1) & - \; d \\ G_x(i,j-1) & - \; e \end{cases} \quad \text{and}$$

$$\begin{aligned} G_d(i,j) &= \max\{G_d(i-1,j-1), G_y(i-1,j-1), \\ &\qquad G_x(i-1,j-1)\} + s(x_i,y_j) \end{aligned}$$

initialization:

$$\begin{aligned} &S(0,0) = 0, G_y(0,0) = -\infty \text{ and } G_x(0,0) = -\infty, \\ &S(i,0) = G_y(i,0) = -d - (i-1)\, e,\; G_x(i,0) = -\infty, \\ &S(0,j) = G_x(0,j) = -d - (j-1)\, e,\; G_y(0,j) = -\infty \end{aligned}$$

Global Alignment – Needleman-Wunsch

Algorithmus 1 Needleman-Wunsch with affine gap

Input: two sequences $\boldsymbol{x}$ and $\boldsymbol{y}$ with length n and m, respectively; scoring matrix s, gap opening penalty d and gap extend penalty e

Output: optimal global alignment and its score

```
BEGIN INITIALIZATION
    $G_d(0,0) = 0,\ G_x(0,0) = -d - (n+m)\ e,\ G_y(0,0) = -d - (n+m)\ e$
    for $1 \leq j \leq m$
        $G_x(0,j) = -d\ -\ (j-1)\ e$
        $G_y(0,j) = G_d(0,j) = -d - (n+m)\ e$
        $B_x(0,j) = \text{“x”}$
    for $1 \leq i \leq n$
        $G_y(i,0) = -d\ -\ (i-1)\ e$
        $G_x(i,0) = G_d(i,0) = -d - (m+n)\ e$
        $B_y(i,0) = \text{“y”}$
END INITIALIZATION
```

Global Alignment – Needleman-Wunsch

```
BEGIN PROCEDURE
  for $1 \leq i \leq n$ do
    for $1 \leq j \leq m$ do
      $G_x(i,j) = \max\{G_d(i,j-1) - d \, , \, G_x(i,j-1) - e\}$
      if $G_x(i,j) = G_d(i,j-1) - d$ then $B_x(i,j) =$ “d” else $B_x(i,j) =$ “x”
      $G_y(i,j) = \max\{G_d(i-1,j) - d \, , \, G_y(i-1,j) - e\}$
      if $G_y(i,j) = G_d(i-1,j) - d$ then $B_y(i,j) =$ “d” else $B_y(i,j) =$ “y”
      $G_d(i,j) = \max\{G_d(i-1,j-1), G_y(i-1,j-1), G_x(i-1,j-1)\} + s(x_i,y_j)$
      if $G_d(i,j) = G_d(i-1,j-1) + s(x_i,y_j)$ then $B_d(i,j) =$ “d”
      if $G_d(i,j) = G_y(i-1,j-1) + s(x_i,y_j)$ then $B_d(i,j) =$ “y”
      if $G_d(i,j) = G_x(i-1,j-1) + s(x_i,y_j)$ then $B_d(i,j) =$ “x”
    end for
  end for
  $S = \max\{G_d(n,m) \, , \, G_x(n,m) \, , \, G_y(n,m)\}$
  print “Score: ” $S$

  if $G_d(n,m) = S$ then $t =$ “d”
  if $G_x(n,m) = S$ then $t =$ “x”
  if $G_y(n,m) = S$ then $t =$ “y”
  $(i,j) = (n,m)$
  while $(i,j) \neq (0,0)$ do
    if $t = \begin{cases} \text{“d” then print } \frac{x_i}{y_j}; \; i = i-1, \; j = j-1, \; t = B_d(i,j) \\ \text{“y” then print } \frac{x_i}{-}; \; i = i-1, \; t = B_y(i,j) \\ \text{“x” then print } \frac{-}{y_j}; \; j = j-1, \; t = B_x(i,j) \end{cases}$
  end while
END PROCEDURE
```

Time and memory complexity of O(n m)

Global Alignment – Needleman-Wunsch

	G	A	K	L	S	A	D	T	E	V	V	C	G	A	P	S	I	Y	L	D	F
	-39	-40	-41	-42	-43	-44	-45	-46	-47	-48	-49	-50	-51	-52	-53	-54	-55	-56	-57	-58	-59
R	-33	-34	-35	-36	-37	-38	-39	-40	-41	-42	-43	-44	-45	-46	-47	-48	-49	-50	-51	-52	-53
T	-33	-33	-35	-36	-35	-37	-39	-34	-41	-41	-42	-44	-45	-45	-47	-46	-49	-50	-51	-52	-53
F	-26	-27	-28	-29	-30	-31	-32	-33	-34	-35	-36	-37	-38	-39	-40	-41	-42	-43	-44	-45	-46
F	-19	-20	-21	-22	-23	-24	-25	-26	-27	-28	-29	-30	-31	-32	-33	-34	-35	-36	-37	-38	-39
V	-14	-15	-16	-17	-18	-19	-20	-21	-22	-23	-24	-25	-26	-27	-28	-29	-30	-31	-32	-33	-34
G	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19	-20	-21	-22	-23	-24	-25	-26	-27
G	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19	-20
N	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13
F	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-6	-9	-10	-6
K	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5
L	18	17	16	17	14	13	12	11	10	9	8	7	6	5	4	3	2	1	2	-1	-2
N	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5
T	↖**26**	25	23	22	23	21	19	24	17	17	16	14	12	13	11	12	9	7	7	6	4
A	27	↖**30**	24	22	23	27	19	19	23	17	17	16	14	16	12	12	11	7	6	5	4
S	25	28	↖**30**	22	26	24	27	20	19	21	15	16	16	15	15	16	10	9	6	6	4
I	26	24	25	↖**32**	20	25	21	26	17	22	24	14	12	15	12	13	20	9	11	3	6
P	23	25	23	22	↖**31**	19	24	20	25	15	20	21	12	11	22	11	10	17	6	10	2
E	31	22	26	20	22	↖**30**	21	23	25	23	13	16	19	11	10	22	8	8	14	8	7
N	27	29	25	24	23	22	↖**31**	21	23	22	20	16	16	17	13	12	19	10	9	15	7
V	44	27	27	26	22	23	19	↖**31**	19	27	26	19	13	16	15	11	15	18	11	6	14
E	25	43	28	24	26	21	25	18	↖**36**	17	25	22	17	12	15	15	9	13	15	13	5
V	27	25	41	29	22	26	18	25	16	↖**40**	21	24	19	17	16	15	18	13	14	12	12
V	22	27	23	42	27	22	23	19	23	20	↖**44**	24	23	22	21	20	19	18	17	16	15
I	21	21	24	25	40	26	19	22	17	26	24	↖**43**	23	22	21	20	24	18	20	16	16
C	20	21	19	23	24	40	23	19	18	18	25	33	↖**40**	23	19	20	19	22	17	17	14
P	20	19	20	20	22	23	39	22	18	17	22	22	31	↖**39**	30	18	17	16	19	16	13
P	19	19	18	19	19	21	22	38	21	17	21	21	20	30	↖**46**	29	25	24	23	22	21
A	20	23	18	18	20	23	19	22	37	21	20	21	21	24	29	↖**47**	28	26	25	24	23
T	17	20	22	17	19	20	22	24	21	37	21	19	19	21	25	30	↖**46**	26	25	24	23
Y	15	15	18	21	15	17	17	20	22	20	36	19	16	17	24	26	29	↖**53**	33	32	31
L	14	14	13	22	19	15	15	16	17	23	21	35	15	15	23	25	28	33	↖**57**	37	36
D	15	13	13	14	22	17	21	14	18	15	20	18	34	14	22	24	24	32	37	↖**63**	43
Y	15	13	11	13	12	20	14	19	13	17	15	18	15	32	21	23	23	31	36	43	↖**66**

Global Alignment – Needleman-Wunsch

		R	K	F	F	V	G	G	N	W	K	M	N	G	D	K	K	S	L	N	G
	0	-20	-21	-22	-23	-24	-25	-26	-27	-28	-29	-30	-31	-32	-33	-34	-35	-36	-37	-38	-39
R	-20	↖**5**	-15	-16	-17	-18	-19	-20	-21	-22	-23	-24	-25	-26	-27	-28	-29	-30	-31	-32	-33
T	-21	-15	↖**4**	-16	-17	-17	-19	-20	-20	-22	-23	-24	-24	-26	-27	-28	-29	-28	-31	-31	-33
F	-22	-16	-16	↖**10**	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19	-20	-21	-22	-23	-24	-25	-26
F	-23	-17	-17	-10	↖**16**	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14	-15	-16	-17	-18	-19
V	-24	-18	-18	-11	-4	↖**20**	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13	-14
G	-25	-19	-19	-12	-5	0	↖**26**	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
G	-26	-20	-20	-13	-6	-1	6	↖**32**	12	11	10	9	8	7	6	5	4	3	2	1	0
N	-27	-21	-20	-14	-7	-2	5	12	↖**38**	18	17	16	15	14	13	12	11	10	9	8	7
F	-28	-22	-22	-14	-8	-3	4	11	18	↖**39**	19	18	17	16	15	14	13	12	11	10	9
K	-29	-23	-17	-16	-9	-4	3	10	17	19	↖**44**	←**24**	←**23**	←**22**	←**21**	←**20**	←**19**	←**18**	17	16	15
L	-30	-24	-24	-17	-10	-5	2	9	16	18	24	46	26	25	24	23	22	21	↖**22**	19	18
N	-31	-25	-24	-18	-11	-6	1	8	15	17	23	26	52	32	31	30	29	28	27	↖**28**	25
T	-32	-26	-26	-19	-12	-7	0	7	14	16	22	25	32	50	31	30	29	30	27	27	↖**26**
A	-33	-27	-27	-20	-13	-8	-1	6	13	15	21	24	31	32	48	30	29	30	29	25	27
S	-34	-28	-27	-21	-14	-9	-2	5	12	14	20	23	30	31	32	48	30	33	28	30	25
I	-35	-29	-29	-22	-15	-10	-3	4	11	13	19	22	29	28	28	29	45	28	35	25	26
P	-36	-30	-30	-23	-16	-11	-4	3	10	12	18	21	28	27	27	27	28	44	25	33	23
E	-37	-31	-29	-24	-17	-12	-5	2	9	11	17	20	27	26	29	28	28	28	41	25	31
N	-38	-32	-31	-25	-18	-13	-6	1	8	10	16	19	26	27	27	29	28	29	25	47	27
V	-39	-33	-33	-26	-19	-14	-7	0	7	9	15	18	25	24	24	25	27	26	30	27	44
E	-40	-34	-32	-27	-20	-15	-8	-1	6	8	14	17	24	23	26	25	26	27	23	30	25
V	-41	-35	-35	-28	-21	-16	-9	-2	5	7	13	16	23	22	21	24	23	24	28	25	27
V	-42	-36	-36	-29	-22	-17	-10	-3	4	6	12	15	22	21	20	21	22	21	25	25	22
I	-43	-37	-37	-30	-23	-18	-11	-4	3	5	11	14	21	20	19	20	18	20	23	23	21
C	-44	-38	-38	-31	-24	-19	-12	-5	2	4	10	13	20	19	18	19	17	17	19	22	20
P	-45	-39	-39	-32	-25	-20	-13	-6	1	3	9	12	19	18	18	18	18	16	14	21	20
P	-46	-40	-40	-33	-26	-21	-14	-7	0	2	8	11	18	17	17	17	17	17	13	20	19
A	-47	-41	-41	-34	-27	-22	-15	-8	-1	1	7	10	17	18	15	16	16	18	16	19	20
T	-48	-42	-42	-35	-28	-23	-16	-9	-2	0	6	9	16	15	17	15	15	17	17	18	17
Y	-49	-43	-43	-36	-29	-24	-17	-10	-3	0	5	8	15	14	13	15	13	13	16	17	15
L	-50	-44	-44	-37	-30	-25	-18	-11	-4	-2	4	7	14	13	12	13	13	11	17	16	14
D	-51	-45	-45	-38	-31	-26	-19	-12	-5	-3	3	6	13	13	19	12	12	13	8	18	15
Y	-52	-46	-46	-39	-32	-27	-20	-13	-6	-3	2	5	12	11	10	17	10	10	12	14	15

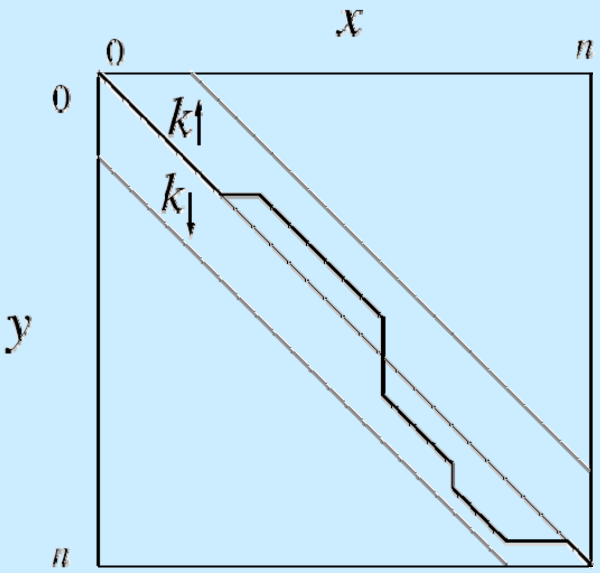
Global Alignment – Needleman-Wunsch

KBand Global Alignment

high sequence similarity: speed up

↪ backtracking paths on the main diagonal

↪ solutions within a band (width k) around the main diagonal



Global Alignment – Needleman-Wunsch

k: estimated gaps in the alignment (difference of length)

to leave the band with linear gap penalty costs -2(k+1) d

iterative algorithm with increasing k: running time depends on the similarity

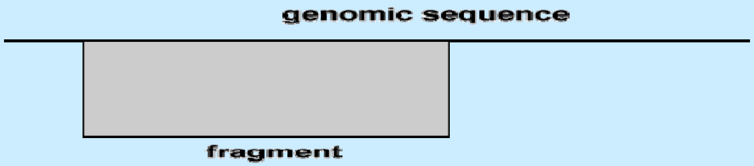
Local Alignment – Smith-Waterman

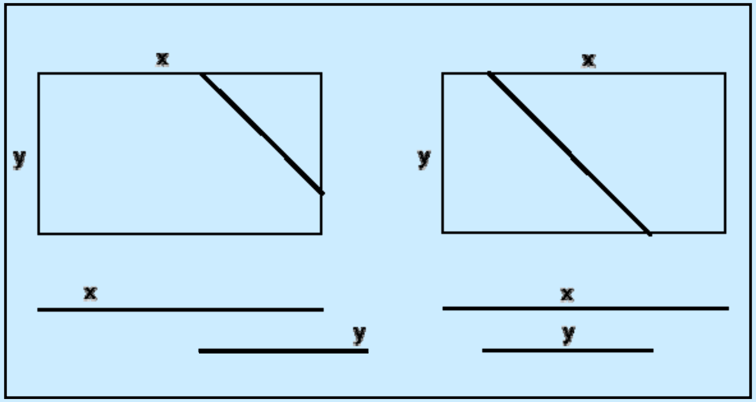
proteins which are remote related (homologous) share subsequences which have much higher similarity than random sequences

Subsequences may be important for the function or for the folding

Most relations between proteins are found by local alignment methods

DNA sequencing: only fragments are matched





Local Alignment – Smith-Waterman

Main idea of local alignment : *negative scores are avoided*

Negative scores: subsequence is not homologous
Position i: prefix match to extend or start a new match $S(i,j) = 0$

linear gap penalty:

$$S(i,j) \;=\; \max \begin{cases} 0 \\ S(i-1,j-1) \;+\; s(x_i,y_j) \\ S(i-1,j) \;-\; d \\ S(i,j-1) \;-\; d \end{cases}$$

$$S(0,0) \;=\; 0 \;\text{ and }\; S(-1,j) \;=\; S(i,-1) \;=\; -\infty$$

$$\Rightarrow \quad S(i,0) = S(0,j) = 0$$

Backtracking starts at position with maximal score

Local Alignment – Smith-Waterman

Algorithm 1 Smith-Waterman with linear gap

```
Input: two sequences $\boldsymbol{x}$ and $\boldsymbol{y}$ with length $n$ and $m$, respectively; scoring matrix $s$, gap penalty $d$
Output: optimal local alignment and its score

BEGIN INITIALIZATION
  $S(i,0) = S(0,j) = 0$ for $0 \leq j \leq m$ and $0 \leq i \leq n$
END INITIALIZATION

BEGIN PROCEDURE
  for $1 \leq i \leq n$ do
    for $1 \leq j \leq m$ do
      $a(i-1,j-1) = S(i-1,j-1) + s(x_i,y_j)$ , $a(i-1,j) = S(i-1,j) - d$, $a(i,j-1) = S(i,j-1) - d$
      $S(i,j) = \max\{0, a(i-1,j-1), a(i-1,j), a(i,j-1)\}$
      if $S(i,j) > 0$ then $B(i,j) = \arg\max\{0, a(i-1,j-1), a(i-1,j), a(i,j-1)\}$ else $B(i,j) = (-1,-1)$
    end for
  end for
  $(i,j) = \arg\max\{S(i,j) \mid 1 \leq i \leq n,\ 1 \leq j \leq m\}$
  print "Score: " $S(i,j)$

  while $S(i,j) \neq 0$ do
        if $B(i,j) = \begin{cases} (i-1,j-1) & \textbf{then print } \begin{smallmatrix} x_i \\ y_j \end{smallmatrix} \\ (i-1,j) & \textbf{then print } \begin{smallmatrix} x_i \\ - \end{smallmatrix} \\ (i,j-1) & \textbf{then print } \begin{smallmatrix} - \\ y_j \end{smallmatrix} \end{cases}$
    $(i,j) = B(i,j)$
  end while
END PROCEDURE
```

Time and memory complexity of O(n m)

Local Alignment – Smith-Waterman

		R	K	F	F	V	G	G	N	W	K	M	N	G	D	K	K	S	L	N	G
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	0	↖**5**	2	0	0	0	0	0	0	0	2	0	0	0	0	2	2	0	0	0	0
T	0	0	↖**4**	0	0	0	0	0	0	0	0	1	0	0	0	0	1	3	0	0	0
F	0	0	0	↖**10**	6	0	0	0	0	1	0	0	0	0	0	0	0	0	3	0	0
F	0	0	0	6	↖**16**	5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
V	0	0	0	0	5	↖**20**	2	0	0	0	0	1	0	0	0	0	0	0	1	0	0
G	0	0	0	0	0	2	↖**26**	8	2	0	0	0	1	↖**6**	0	0	0	0	0	1	6
G	0	0	0	0	0	0	8	↖**32**	12	8	4	0	0	7	↖**5**	0	0	0	0	0	7
N	0	0	0	0	0	0	2	12	↖**38**	18	14	10	6	2	8	↖**5**	0	1	0	6	0
F	0	0	0	6	6	0	0	8	18	↖**39**	19	15	11	7	3	5	↖**2**	0	1	0	3
K	0	2	5	0	3	4	0	4	14	19	↖**44**	24	20	16	12	8	10	↖**2**	0	1	0
L	0	0	0	5	0	4	0	0	10	15	24	↖**46**	26	22	18	14	10	8	↖**6**	0	0
N	0	0	0	0	2	0	4	0	6	11	20	26	↖**52**	32	28	24	20	16	12	↖**12**	4
T	0	0	0	0	0	2	0	2	2	7	16	22	32	50	31	27	23	21	15	12	↖**10**
A	0	0	0	0	0	0	2	0	0	3	12	18	28	32	48	30	26	24	20	13	12
S	0	0	0	0	0	0	0	2	1	0	8	14	24	28	32	48	30	30	22	21	13
I	0	0	0	0	0	3	0	0	0	0	4	10	20	22	25	29	45	28	32	19	17
P	0	0	0	0	0	0	1	0	0	0	0	6	16	18	21	24	28	44	25	30	17
E	0	0	1	0	0	0	0	0	0	0	1	2	12	14	20	22	25	28	41	25	28
N	0	0	0	0	0	0	0	0	6	0	0	0	8	12	15	20	22	26	25	47	27
V	0	0	0	0	0	4	0	0	0	3	0	1	4	6	9	13	18	20	27	27	44
E	0	0	1	0	0	0	2	0	0	0	4	0	1	2	8	10	14	18	17	27	25
V	0	0	0	0	0	4	0	0	0	0	0	5	0	0	0	6	8	12	19	19	24
V	0	0	0	0	0	4	1	0	0	0	0	1	2	0	0	0	4	6	13	16	16
I	0	0	0	0	0	3	0	0	0	0	0	1	0	0	0	0	0	2	8	11	12
C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	7	8
P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	5
P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Y	0	0	0	3	3	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	3	4	0	0	0	0	0	2	0	0	0	0	0	0	4	0	0
D	0	0	0	0	0	0	3	0	1	0	0	0	3	0	6	0	0	0	0	5	0
Y	0	0	0	3	3	0	0	0	0	3	0	0	0	0	0	4	0	0	0	0	2

Local Alignment – Smith-Waterman

	G	A	K	L	S	A	D	T	E	V	V	C	G	A	P	S	I	Y	L	D	F
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T	0	0	0	1	1	0	0	5	0	0	0	0	0	0	0	1	0	0	0	0	0
F	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	1	3	0	0	6
F	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	4	3	0	6
V	0	0	0	1	0	0	0	0	0	4	5	0	0	0	0	0	3	0	5	0	0
G	6	0	0	0	1	0	0	0	0	0	1	2	6	0	0	0	0	0	0	4	0
G	7	6	0	0	0	1	0	0	0	0	0	0	8	6	0	0	0	0	0	0	1
N	0	5	6	0	1	0	2	0	0	0	0	0	0	6	4	1	0	0	0	1	0
F	3	0	2	6	0	0	0	0	0	0	0	0	0	0	2	2	1	3	0	0	7
K	0	2	5	0	6	0	0	0	1	0	0	0	0	0	0	2	0	0	1	0	0
L	0	0	0	9	0	5	0	0	0	2	1	0	0	0	0	0	4	0	4	0	0
N	4	0	0	0	10	0	6	0	0	0	0	0	0	0	0	1	0	2	0	5	0
T	↖**10**	4	0	0	1	10	0	11	0	0	0	0	0	0	0	1	0	0	1	0	3
A	12	↖**14**	3	0	1	5	8	0	10	0	0	0	0	4	0	1	0	0	0	0	0
S	13	13	↖**14**	1	4	2	5	9	0	8	0	0	0	1	3	4	0	0	0	0	0
I	17	12	10	↖**16**	0	3	0	4	6	3	11	0	0	0	0	1	8	0	2	0	0
P	17	16	11	7	↖**15**	0	2	0	3	4	1	8	0	0	7	0	0	5	0	1	0
E	28	16	17	8	7	↖**14**	2	1	5	1	2	0	6	0	0	7	0	0	2	2	0
N	27	26	19	15	11	7	↖**15**	2	1	2	0	0	0	4	0	1	4	0	0	3	0
V	44	27	24	20	13	11	4	↖**15**	0	5	6	0	0	0	2	0	4	3	1	0	2
E	25	43	28	21	20	12	13	3	↖**20**	0	3	2	0	0	0	2	0	2	0	3	0
V	24	25	41	29	19	20	9	13	1	↖**24**	4	2	0	0	0	0	5	0	3	0	2
V	16	24	23	42	27	19	17	10	11	5	↖**28**	8	4	0	0	0	3	4	1	0	0
I	12	15	21	25	40	26	16	16	8	14	8	↖**27**	7	3	0	0	4	2	6	0	0
C	8	12	13	20	24	40	23	16	12	8	13	17	↖**24**	7	0	0	0	2	1	3	0
P	5	7	11	14	19	23	39	22	15	11	7	10	15	↖**23**	14	0	0	0	0	0	0
P	1	4	6	10	13	18	22	38	21	14	10	6	8	14	↖**30**	13	6	2	0	0	0
A	0	5	3	6	11	17	16	22	37	21	14	10	6	12	13	↖**31**	12	7	3	0	0
T	0	0	4	2	7	11	16	21	21	37	21	13	9	6	11	14	↖**30**	10	6	2	0
Y	0	0	0	3	0	5	8	14	19	20	36	19	12	8	4	9	13	↖**37**	17	13	9
L	0	0	0	4	1	0	3	7	11	20	21	35	15	11	7	3	11	17	↖**41**	21	17
D	0	0	0	0	4	0	6	2	9	9	17	18	34	14	10	7	2	13	21	↖**47**	27
Y	2	0	0	0	0	2	0	4	1	8	8	15	15	32	12	8	6	9	17	27	↖**50**

Local Alignment – Smith-Waterman

affine gap penalty is analogous to Needleman-Wunsch

An example with affine gaps d=20 and e=4

best local alignment has a score of 52:

```
RKFFVGGNWKMN
|.||||||:|.|
RTFFVGGNFKLN
```

second best score is 50:

```
GDKKSLNGAKLSADTEVVCGAPSIYLDF
|....||.|.:..:.|||...|:.|||:
GGNFKLNTASIPENVEVVICPPATYLDY
```

FASTA, BLAST and BLAT

The algorithms possess O(nm) complexity except for high similarity

NR database (SS prediction, protein classification): > 3 Mio. entries

Speed up through approximations: *seed and extend*

↪ exact matches of small subsequences (2 to 5)

↪ extend these matches

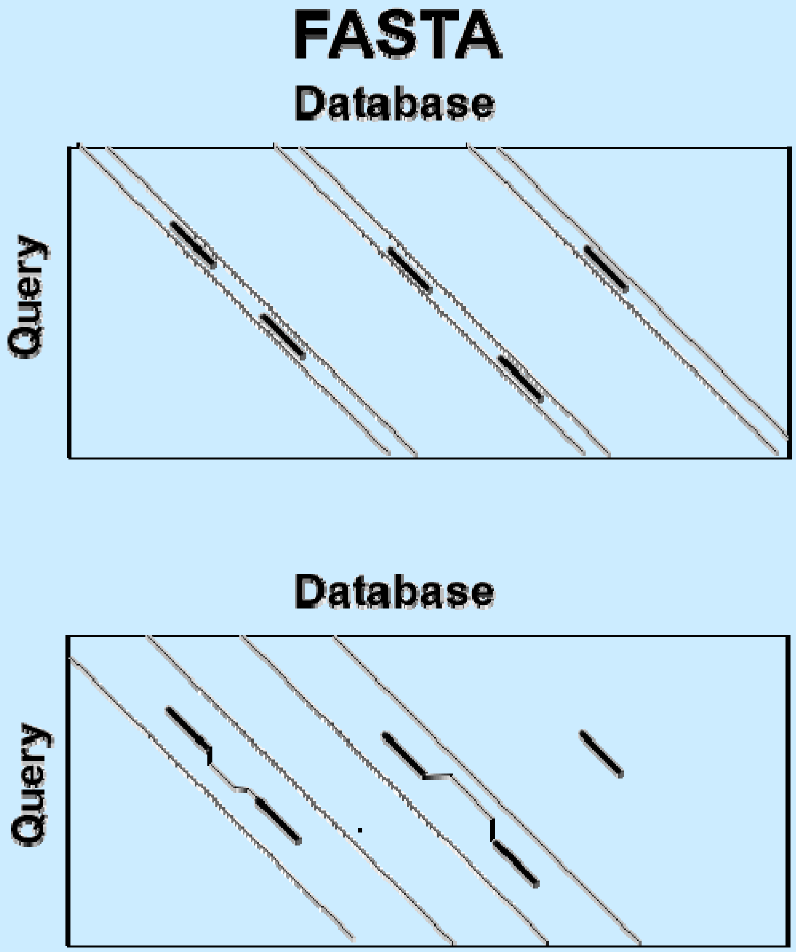
↪ query sequence preprocessed for the 2- to 7-mers matches

FASTA, BLAST and BLAT

FASTA (fast-aye, Lipman and Pearson, 1985, 1988)
www2.ebi.ac.uk/fasta3/

1. Searching *hot-spots* (matches of length k=2, k=6 nucleotides) and score with PAM250, lookup table of query words, ten best scoring regions (number of hot-spots and their distance)
2. regions are re-evaluated by diagonal runs accumulating the hot spots and matches shorter that k
3. best-scoring sub-alignments chained to a candidate alignment with gaps (penalty 20), threshold candidates
4. banded Smith-Waterman with k=23 → alignments around high-scoring regions. full Smith-Waterman

FASTA, BLAST and BLAT



FASTA misses:

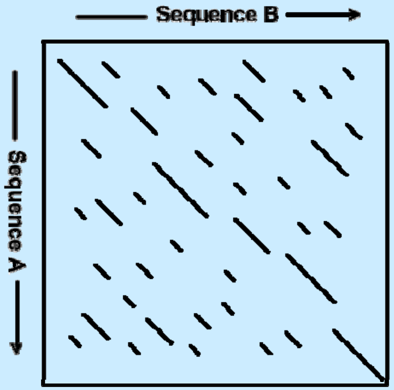
similarities at different positions

if pattern occur repeated

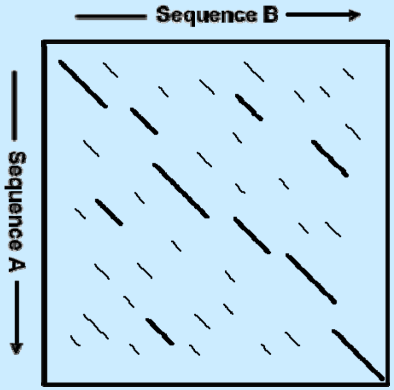
FASTA, BLAST and BLAT

3 Pairwise Alignment
3.1 Motivation
3.2 Scoring
3.2.1 Identity
3.2.2 PAM
3.2.3 BLOSUM
3.2.4 Gap Penalties
3.3 Algorithms
3.3.1 Global
3.3.2 Local
3.3.3 FASTA, BLAST
3.4 Significance
3.4.1 HSPs
3.4.2 Matches

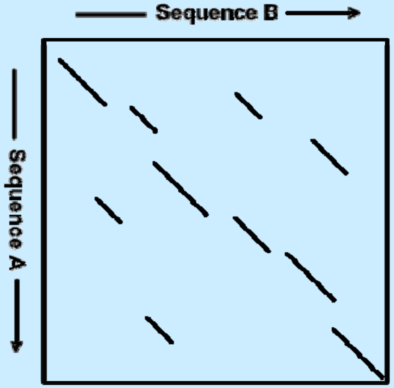
1.



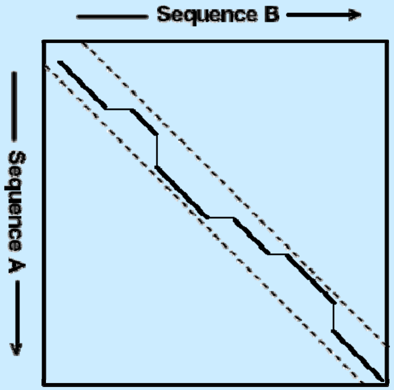
2.



3.



4.



FASTA, BLAST and BLAT

3 Pairwise Alignment
3.1 Motivation
3.2 Scoring
3.2.1 Identity
3.2.2 PAM
3.2.3 BLOSUM
3.2.4 Gap Penalties
3.3 Algorithms
3.3.1 Global
3.3.2 Local
3.3.3 FASTA, BLAST
3.4 Significance
3.4.1 HSPs
3.4.2 Matches

BLAST (Basic Local Alignment Search Tool, Altschul 1990 – the most cited paper in bioinformatics) and Position-Specific-Iterative-BLAST (PSI-BLAST for data bases - is the *most used bioinformatics software*)

tutorial www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html
download: http://www.ncbi.nlm.nih.gov/BLAST/

BLAST searches *high-scoring segment pairs* (HSPs)

HSPs: local maximal segment pairs (matching subsequences) exceeding a scoring threshold

local maximal: shortening or extending has lower score

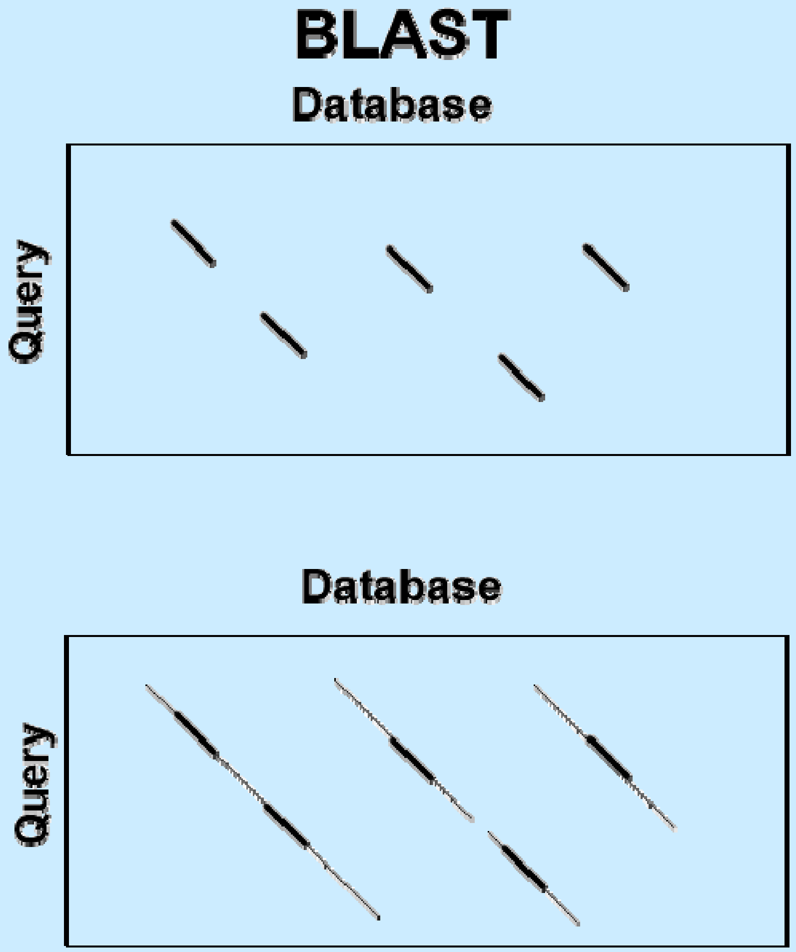
FASTA, BLAST and BLAT

BLAST

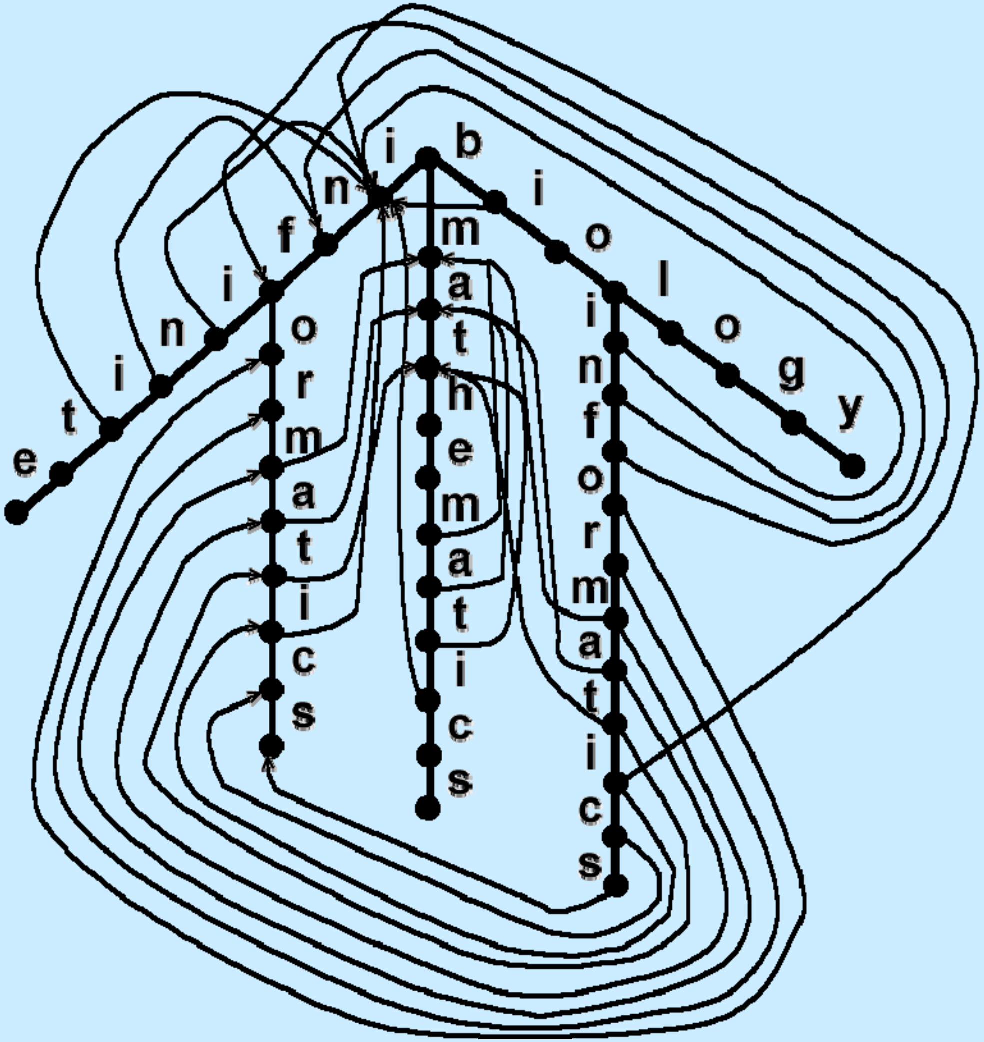
1. Query words as k-mers (k=3, k=11 nucleotides, about 50 words per amino acid), score threshold T (trade-off between speed and sensitivity)
2. data base (single sequence of length m) scanned for hits (Mealy automaton, Aho-Corasick algorithms, O(n+m+u) with u hits)
3. hits are gapless extended: locally maximal segment pairs, non-overlapping hits on the same diagonal with small distance are joined
4. KBand algorithm

BLAST can miss HSPs due to the value of k and the thresholds introduced in step 2 and 3.

FASTA, BLAST and BLAT



FASTA, BLAST and BLAT



FASTA, BLAST and BLAT

- BLASTP: compares amino acid sequence query to a protein sequence database
- BLASTN: compares nucleotide sequence query to a nucleotide sequence database
- BLASTX: nucleotide sequence query is translated and compared to a protein sequence database
- TBLASTN: compares amino acid sequence query to a nucleotide sequence database, where the later is translated
- TBLASTX: compares nucleotide sequence query to a nucleotide sequence database but the sequences are first translated

use amino acid sequence (translation)
e-value of 0.05: significance
repeated segments confuse BLAST (low complex / coiled-coil)

FASTA, BLAST and BLAT

PSI-BLAST: iterative BLAST with profile or position specific scoring matrix (PSSM) from a multiple alignment of highest scoring hits

scoring profile is used for new BLAST searches, profile is refined and sensitivity increased

PSI-BLAST profile: secondary structure prediction and for protein classification, original sequence is substituted by the PSSM

dirty profiles: coiled-coil or low complexity regions, hits which are not similar to the query

FASTA, BLAST and BLAT

	BLAST	FASTA
output	multiple HSPs	best alignment
better for	proteins	nucleotides
speed	faster	slower
sensitivity	low	higher
homologs	finds low	misses low
false positives	more	less
significance	e-values	estimation

\

FASTA, BLAST and BLAT

BLAT (BLAST-Like Alignment Tool): 50 times faster than BLAST at comparable sensitivity (500 times faster for nucleotides)

Difference to BLAST

- builds an index of the data base instead of the query
- goes linearly through the query
- uses near perfect matches (one mismatch)
- joins more than two hits
- chains the high scoring local alignments together

further methods: QUASAR (Q-gram Alignment based Suffix Arrays, Burkhardt et al., Recomb, 1999): data base as suffix array

Significance

How can be the result of an alignment be judged?

Is there a homology / similarity or not?

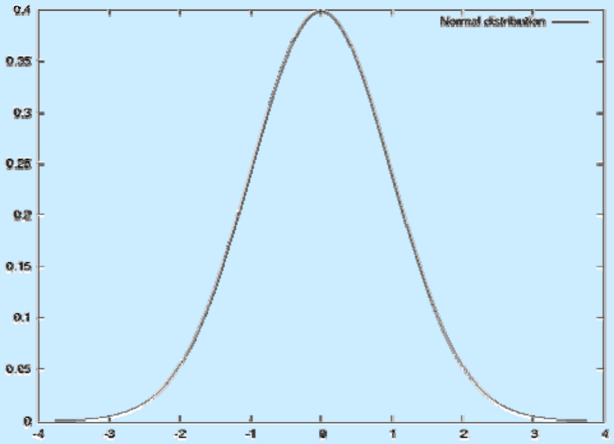
How likely would we find such an alignment randomly?

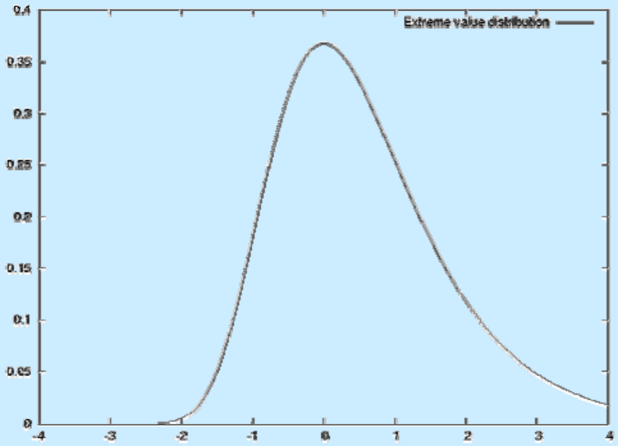
What do the significance measurements of the alignment tools mean?

HSP Significance

First we consider HSPs

distributions of the alignment score of random generated sequences?





$$\text{pdf: } p(x) = e^{-x}e^{-e^{-x}} \ , \quad \text{dist.: } P(x) = e^{-e^{-x}}$$

HSP Significance

Assumptions

- $\boldsymbol{x}$ and $\boldsymbol{y}$ are iid in their elements according to p_x and p_y
- $\boldsymbol{x}$ and $\boldsymbol{y}$ are long
- $\sum_{i,j} p_x(i)\, p_y(j) s(i,j)$ is negative
- it exists $s(i,j) > 0$ (existence of a positive score)

Karlin, Dembo, Kawabata, 1990, showed for maximal segment scores $S_{n,m}$ (n, m length' of the sequences):

$$S_{n,m} \propto \frac{\ln nm}{\lambda}$$

HSP Significance

Karlin and Altschul, 1990, and Altschul and Gish, 1996:

$$\tilde{S}_{n,m} \;=\; S_{n,m} \;-\; \frac{\ln nm}{\lambda}$$

$$P\left(\tilde{S}_{n,m} > S\right) \;\approx\; 1 - \exp\left(-K\ m\ n\ e^{-\lambda\ S}\right) \;\approx\; K\ m\ n\ e^{-\lambda\ S}$$

Approximation: $e^h \;=\; 1 + h + O(h^2)$

$$h \;=\; -K\ m\ n\ e^{-\lambda\ S}$$

h is small for large S

HSP Significance

log-odds score with target distributions $p_{i,j}$:

$$s(i,j) \;=\; \log\left(\frac{p_{i,j}}{p_x(i)\;p_y(j)}\right)\;/\;\lambda$$

expected contribution of a pair to the score:

$$\sum_{i,j} p_{i,j}\, s(i,j) \;=\; \sum_{i,j} p_{i,j} \log\left(\frac{p_{i,j}}{p_x(i)\;p_y(j)}\right)/\lambda \;=$$

$$\mathrm{KL}\left(p_{i,j} \,\|\, p_x(i)\;p_y(j)\right)/\lambda$$

“KL” denotes the Kullback-Leibler distance (number of additional bits needed to describe a pair i,j produced by target distribution if element distributions are given)

λ : scaling to bit scores

HSP Significance

determine λ:
two *random sequences* and delete one pair, then

$$E = P\left(\tilde{S}_{n,m} > S\right) \approx$$

$$\sum_{i,j} p_x(i)\; p_y(j)\; P\left(\tilde{S}_{n-1,m-1} > S - s(i,j)\right)$$

$P\left(\tilde{S}_{n,m} > S\right)$ is the probability of having score larger than S

By deleting a pair (i,j) it is reduced to a probability of sequences of length (n-1,m-1)
The probability of the score of shorter sequences larger than $S - s(i,j)$ is multiplied by observing the pair (i,j)

HSP Significance

approximate $\tilde{S}_{n-1,m-1}$ by $\tilde{S}_{n,m}$ then λ is obtained from

$$\sum_{i,j} p_x(i)\ p_y(j)\ \exp(\lambda\ s(i,j)) \ = \ 1$$

BLOSUM62: $\lambda = 0.254 \text{ and } K = 0.040$

K : how related amino acids are in the given context
λ: scale of the scoring matrix (change of the base of the logarithm)

HSP Significance

$P\left(\tilde{S}_{n,m} > S\right)$: **e-value** (BLAST)

$$\ln(E) \;=\; \ln\left(K\; m\; n\; e^{-\lambda\; S}\right) \;=\; \ln\left(m\; n\right) \;+\; \ln\left(K\right) \;-\; \lambda\; S$$

$$S \;=\; \frac{\bar{S}\;\ln 2 \;+\; \ln K}{\lambda}$$

$$\log_2(E) \;=\; \log_2\left(m\; n\right) \;-\; \bar{S}$$

$$E \;=\; m\; n\; 2^{-\bar{S}}$$

$\bar{S}$ is a score ("**bit-score**") independent of λ and K

HSP Significance

probability that l HSPs exceed S:

- ↪ Poisson distribution
- ↪ E is the average number of events

$$P(l) \;=\; e^{-E}\,\frac{E^l}{l!}$$

Perfect Matches Significance

significance of (near) perfect matches: BLAT

expected number of non-overlapping query k-mers for letters with same probability:

$$(n - k + 1) \frac{m}{k} \left(\frac{1}{a}\right)^k$$

- ↪ $(n - k + 1)$ number of k-mers of query
- ↪ $\frac{m}{k}$ number of non-overlapping k-mers in the data
- ↪ $\left(\frac{1}{a}\right)^k$ matching probability

Perfect Matches Significance

M: similarity between the query and the data base

$(M)^k$ matching probability of k-mer

$1 - \left(1 - M^k\right)^{m/k}$ probability of at least one matching k-mer in the data base to a certain k-mer in the query (one minus the probability of that all m/k k-mers do not match simultaneously)

probability of a match with 1 or less mismatches

$$k \, M^{k-1}(1 - M) + M^k$$

First tem is 1-mismatch and second perfect match
1-mismatch: (k-1) matches multiplied by prob. of mismatch (1-M),
k (k-1)-mers exist

Perfect Matches Significance

For more than one exact matches we introduce $p = M^k$

binomial distribution: match or non-match , l matches:

$$\binom{m/k}{l} p^l \; (1 - p)^{m/k-l}$$

binomial distribution $\mathcal{B}(m/k, p)$ approximated through Poisson or normal distribution (central limit theorem):

$$\mathcal{B}(m/k, p) \sim \begin{cases} \mathcal{P}(np) & \text{for} \quad m/k \; p \; \leq \; 5 \\ \mathcal{N}(np, \sqrt{np(1-p)}) & \text{for} \quad m/k \; p \; > \; 5 \end{cases}$$

Poisson probability-generating function: $G(s) = (p\; s + (1-p))^t$

$$(1 + l/t)^t \approx e^l$$

$$(p\; s + (1-p))^t = \left(1 + \frac{p\; n}{t}\,(s - 1)\right)^t \approx \exp(n\; p\; (s-1))$$