Bioinformatik für Tier- und Pflanzenwissenschaften

VL-5 Sequenzalignment - Teil 1

Dr. Paula Korkuć

Fachgebiet Züchtungsbiologie und molekulare Tierzüchtung

Humboldt-Universität zu Berlin

16.11.2023

Overview for today

HUMBOLDT-UNIVERSITÄT ZU BERLIN

- Conservation and homology
- Sequence alignments
  - Excurse proteins and protein sequence
  - Pairwise alignments
  - Visualization of alignments → dot plot
  - Scoring of alignments
  - Optimal alignment using Needleman-Wunsch algorithm

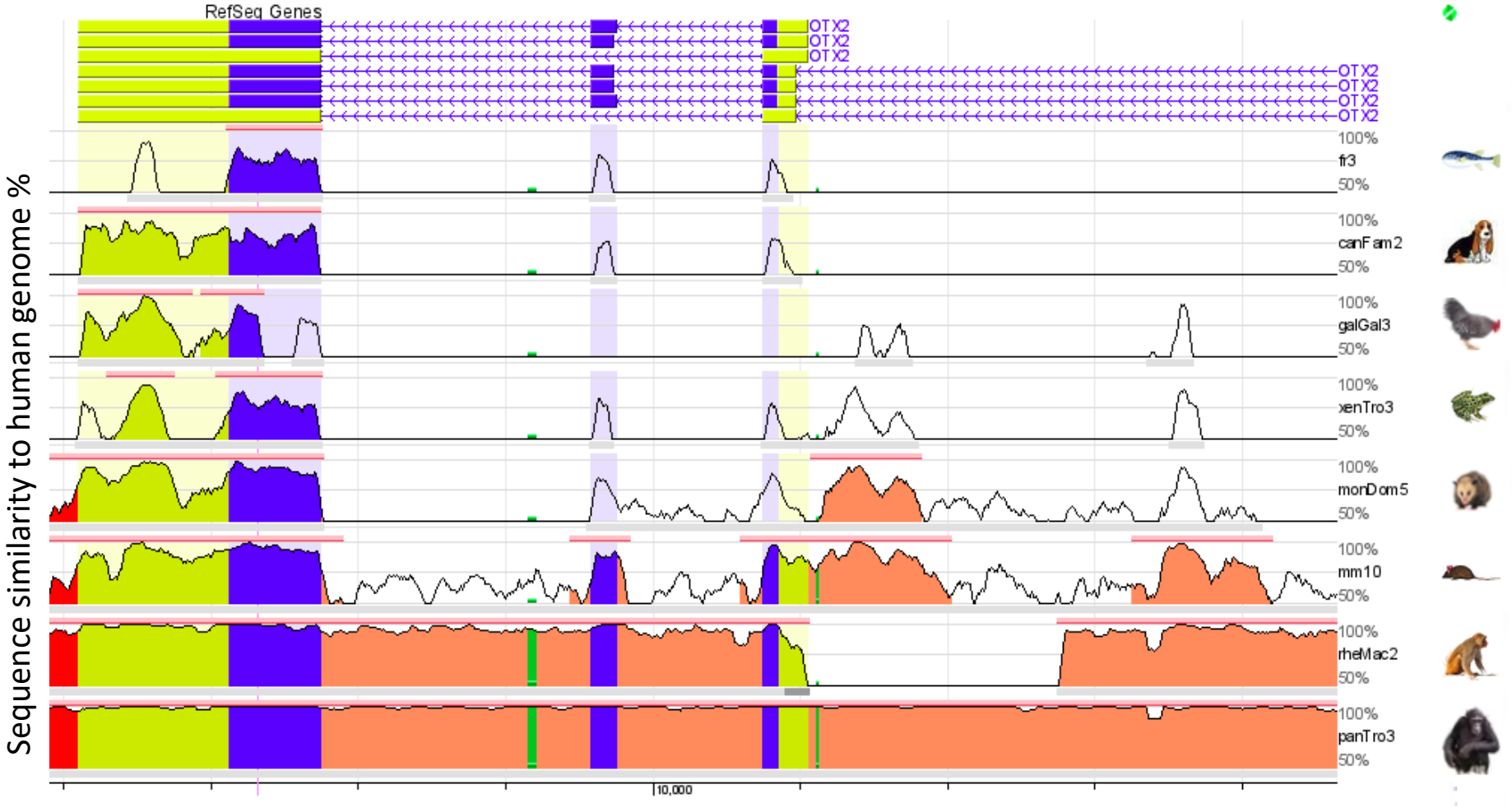
Differences between organisms

HUMBOLDT-UNIVERSITÄT ZU BERLIN

- DNA sequences change in the course of evolution:
  - Mutations, insertions, deletions
  - Chromosomal rearrangement: Duplications, inversions, translocations
- Genes are highly conserved (less variation)

→**BUT: Even same gene (or resulting protein) from two closely related species are rarely identical**

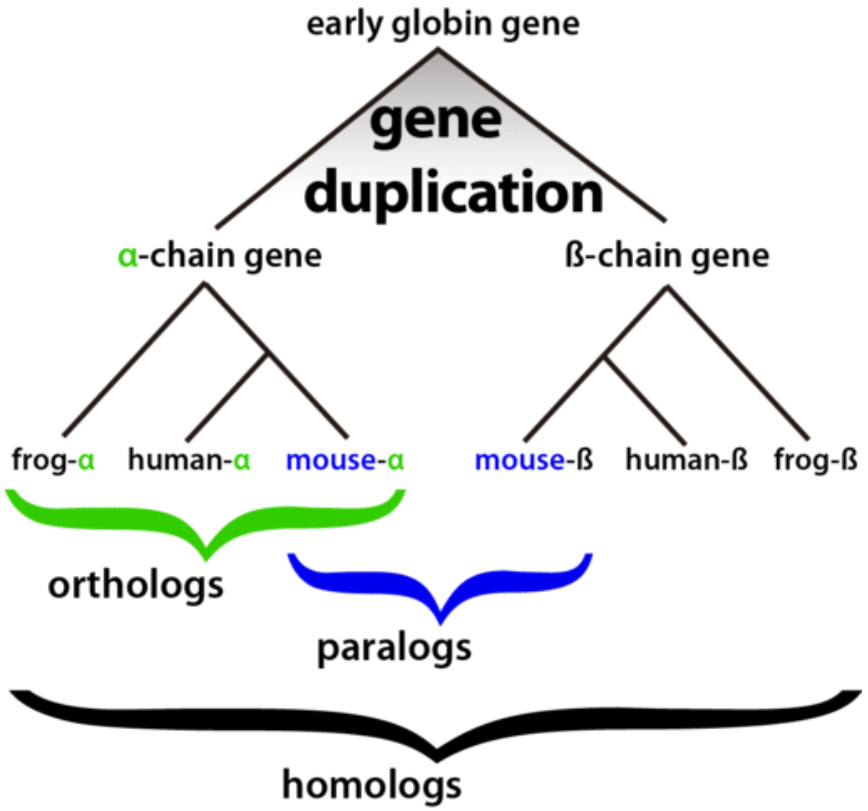
Conservation of sequences - OTX2 gene



https://en.wikipedia.org/wiki/Conserved_sequence#/media/File:ECR_browser_showing_conserved_OTX2_gene_in_vertebrates.png

Sequence homology

- Shared ancestry of sequences because of
  - Speciation event (orthologs)
  - Duplication event (paralogs)



https://bitesizebio.com/26762/homology-terminology-never-say-wrong-word/

But how to align homolog sequences?

HUMBOLDT-UNIVERSITÄT ZU BERLIN

- Example #1:

```
THISISLECTURE
||--|||||||||
THATISLECTURE
```

→2 mismatches

- Example #2: Mutation lead to insertion of three letters

```
THISISISALECTURE
||--||-------
THATISLECTURE
```

→9 mismatches

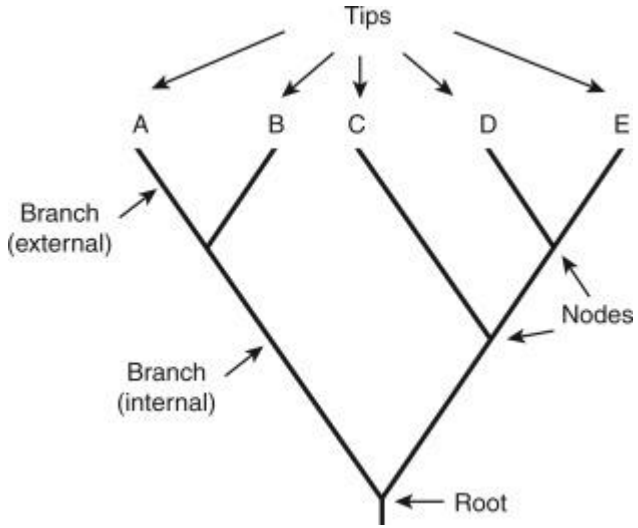
```
THISISISALECTURE
||----|||||||||||
THATIS---LECTURE
```

→ introducing gaps

Sequence alignment

→Arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences

- Application:
  - Uncharacterized sequences → comparison in a database (gene, protein, family)
  - Characterized sequences → Phylogenetic trees
  - Measuring similarity

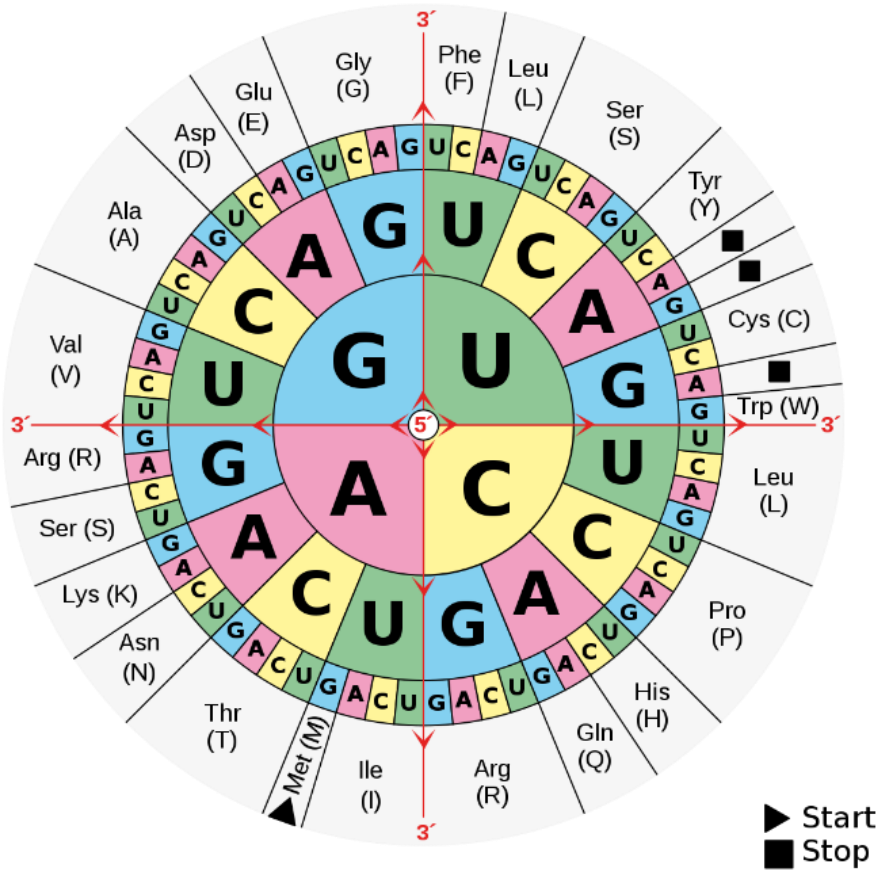


https://ars.els-cdn.com/content/image/3-s2.0-B9780128000496002031-f00203-01-9780128000496.jpg

Alignments at DNA or protein level

- Three amino acids are forming a codon
- 64 different codons
  - 3 codons are reserved for stop
  - 61 codons for amino acids

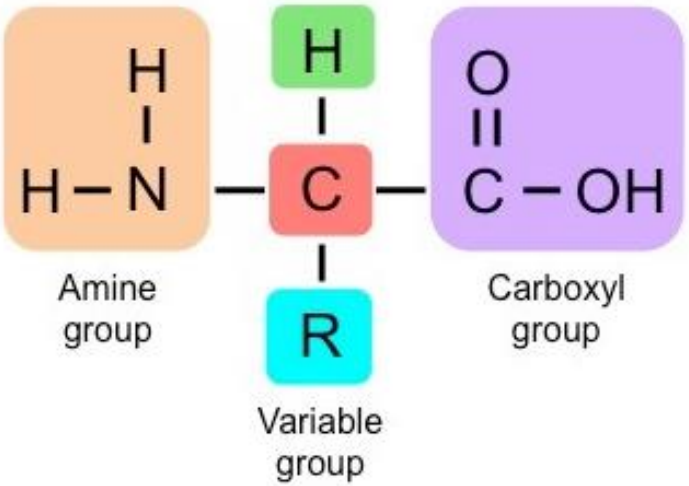
- BUT: < 45 different tRNAs are produced
- →3rd position in codon: “wobble base”



https://de.wikipedia.org/wiki/Genetischer_Code#/media/Datei:Aminoacids_table.svg

Amino acid structure

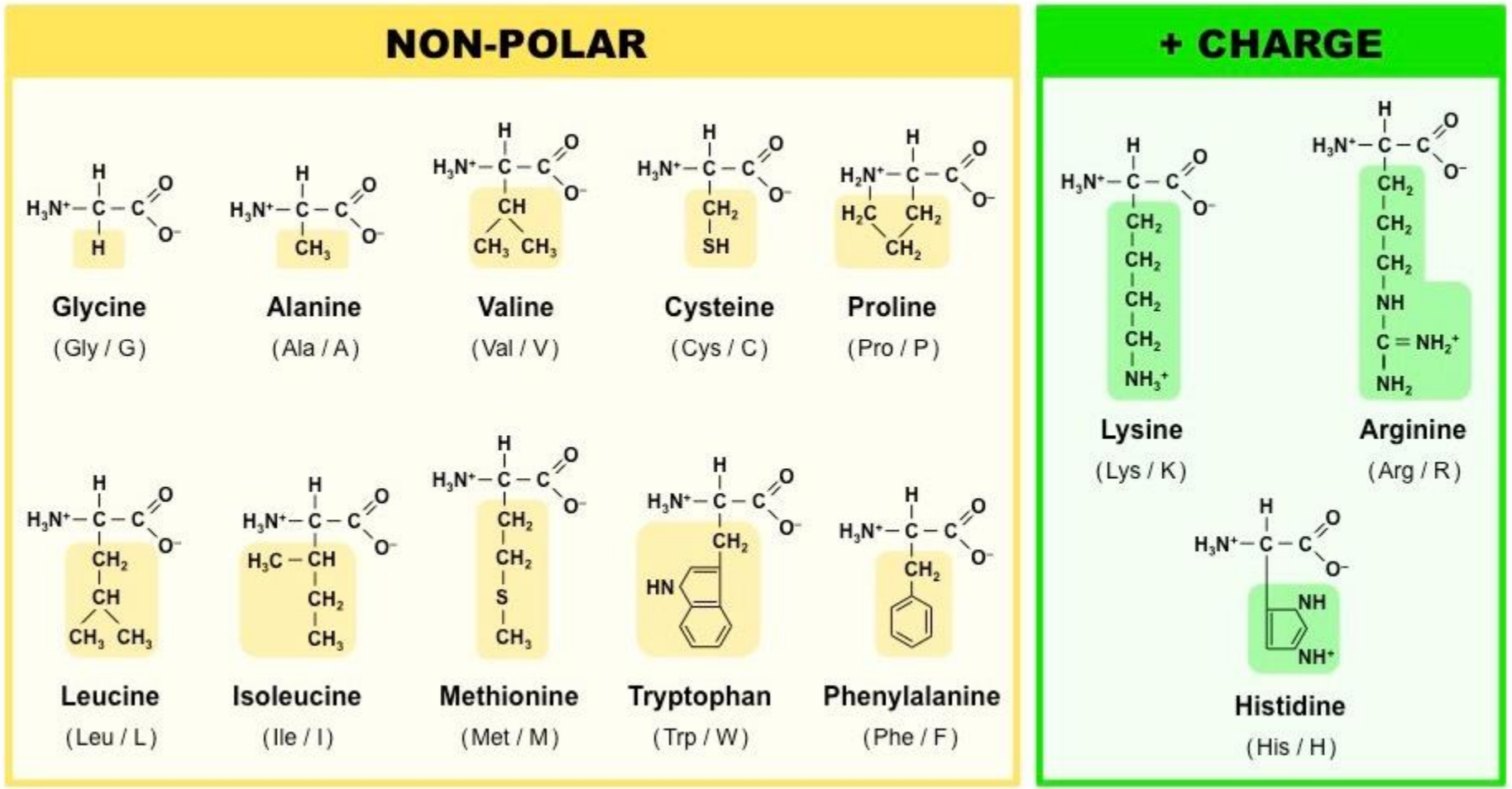
- Structure of a Generalised Amino Acid
  - An amine group (NH_2)
  - A carboxylic acid group (COOH)
  - A hydrogen atom (H)
  - A variable side chain (R) → distinct chemical properties



- Many amino acids together form polypeptides, which makeup proteins
- Most natural polypeptide chains contain between 50 and 2000 amino acid residues

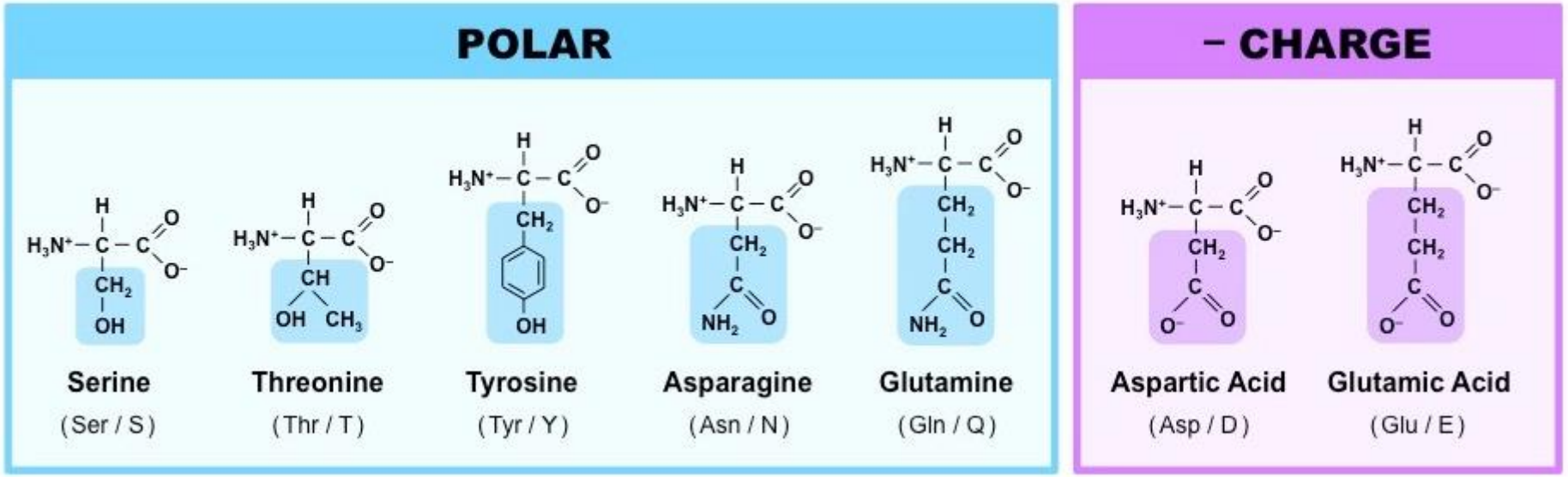
http://ib.bioninja.com.au/standard-level/topic-2-molecular-biology/24-proteins/amino-acids.html

20 universal amino acids



http://ib.bioninja.com.au/standard-level/topic-2-molecular-biology/24-proteins/amino-acids.html

20 universal amino acids



Properties: size, charge, polarity, hydrophobicity, flexibility

- Polar→ more soluble in water; hydrophilic → outside of proteins
- Non-polar → hydrophobic → core of proteins

http://ib.bioninja.com.au/standard-level/topic-2-molecular-biology/24-proteins/amino-acids.html

Sequence alignment

HUMBOLDT-UNIVERSITÄT ZU BERLIN

Pairwise sequence alignment

- Global or local alignment
- Comparatively simple algorithms
  - Find out conserved regions between the two sequences
  - Similarity searches in a database

Multiple sequence alignment

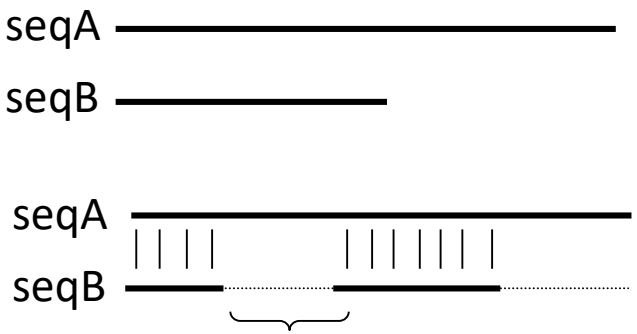
- Generally a global alignment
- Complex sophisticated algorithm
  - To detect regions of variability or conservation in a family of proteins
  - Phylogenetic analysis
  - Detection of homology between a newly sequenced gene and an existing gene family prediction of protein structure
  - Demonstration of homology in multigene families

Global versus local pairwise alignment

HUMBOLDT-UNIVERSITÄT ZU BERLIN

Global

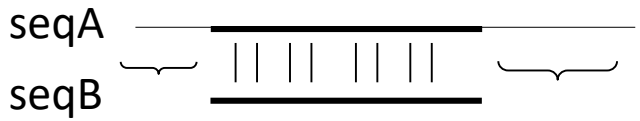
- Comparing sequences over their entire length
- Usually sequences are equally long
- Needleman-Wunsch algorithm



Insert gaps in seqB

Local

- Comparing sequences with partial homology
- Alignments describing most similar region(s)
- Possible to compare short vs. longer sequences
- Smith-Waterman algorithm

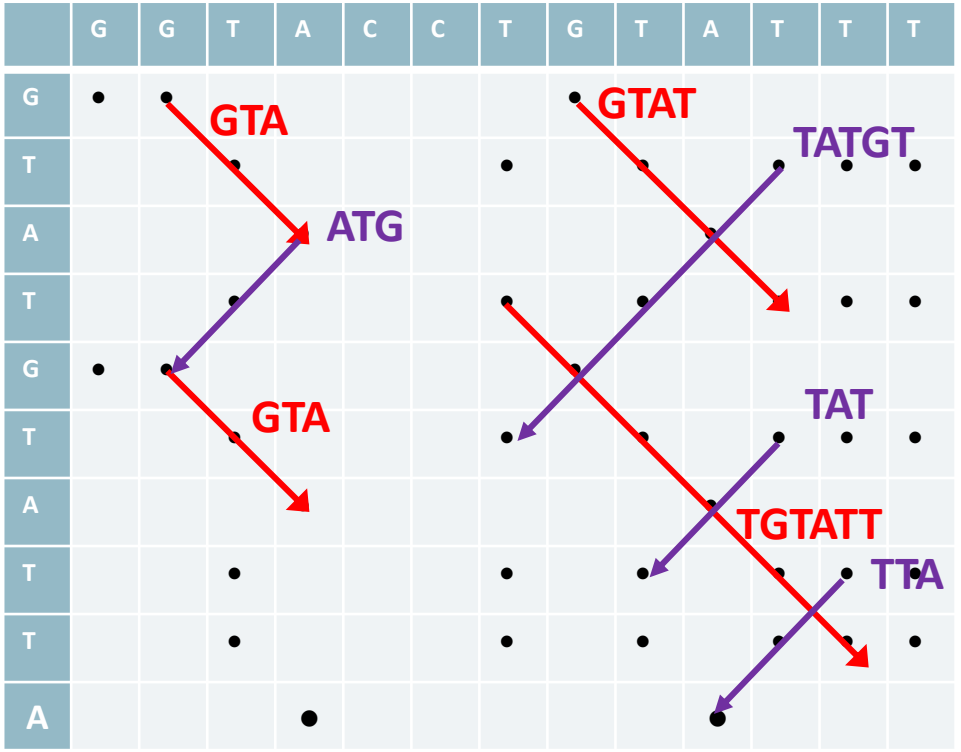


Ignore terminals in seqA

Pairwise alignments

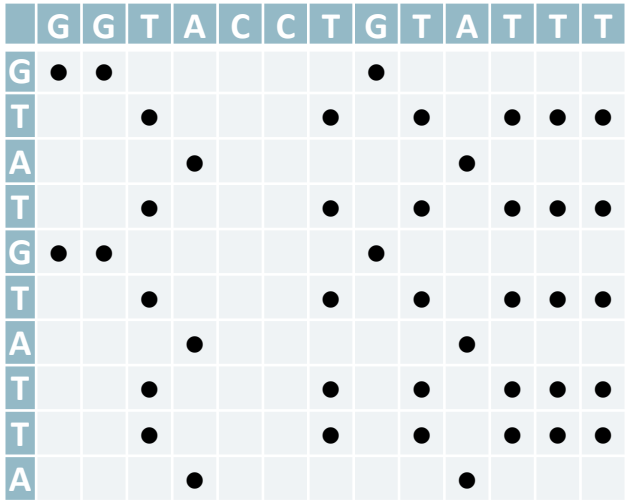
- Gene sequence of interest: `GGTACCTGTATTT`
- Gene sequence with known function: `GTATGTATTA`

- Dotplot
  - Graphic representation
  - Simplest detection method
  - Reveal complex patterns
  - 2-dimensional table:
    - rows= sequence 1
    - columns= sequence 2
    - mark with ● if identical

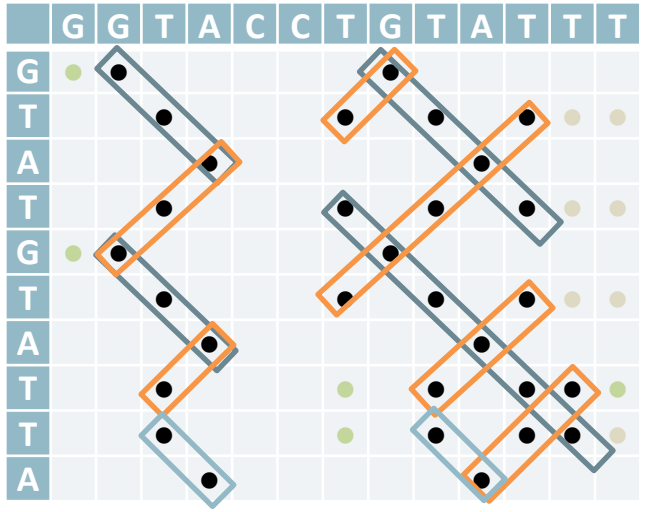


Dot plots - remove noise

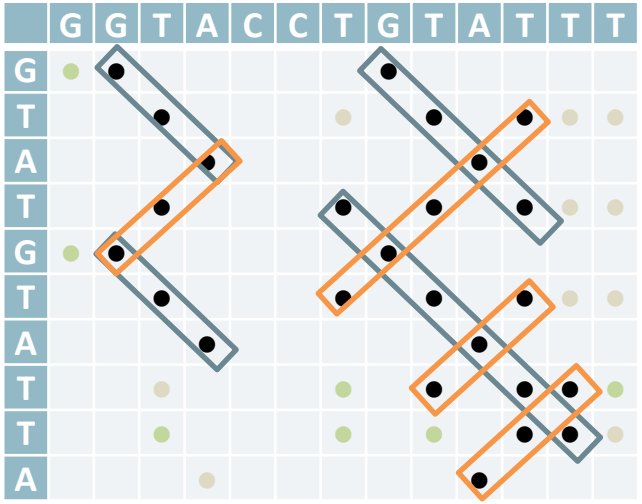
window size = 1



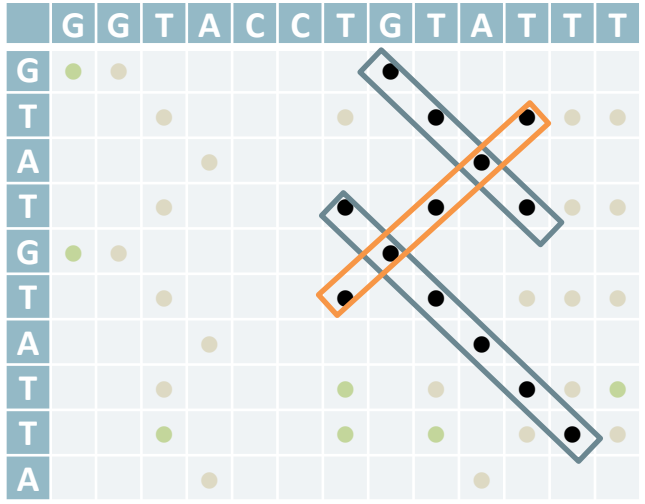
window size = 2



window size = 3



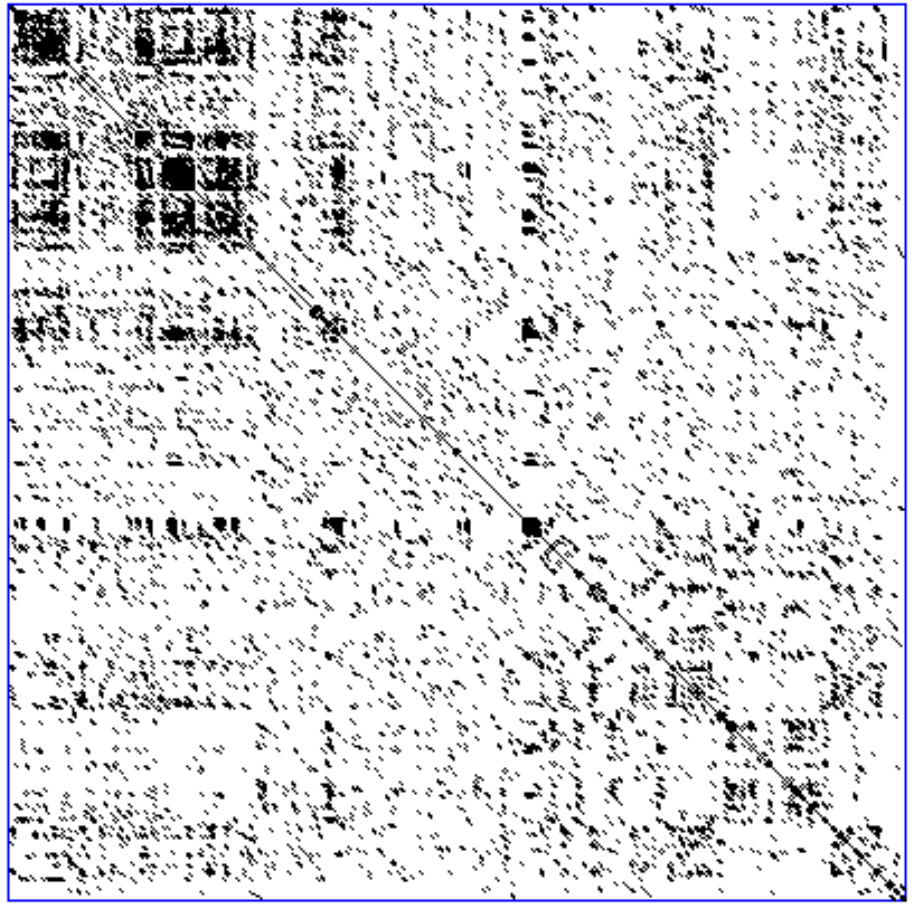
window size = 4



Dot plot example

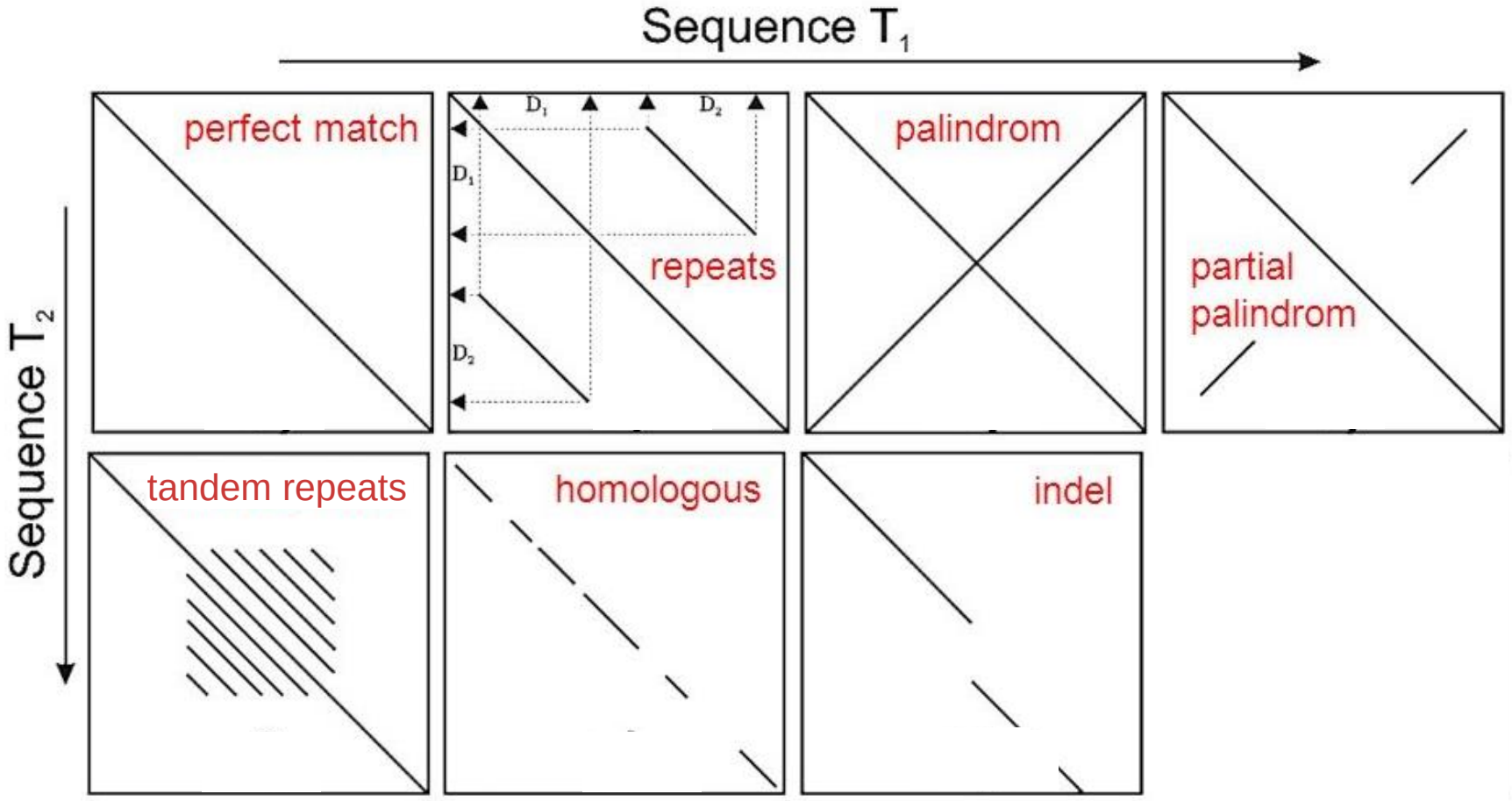
Self-alignment

- Comparing a sequence with itself
  - Repeated domains
  - Motifs repeated many times
  - Mirror regions (palindromes) in nucleic acids



https://en.wikipedia.org/wiki/Dot_plot_(bioinformatics)#/media/File:Zinc-finger-dot-plot.png

Interpretation of dot plots



http://www.code10.info/index.php?option=com_content&view=article&id=64:inroduction-to-dot-plots&catid=52:cat_coding_algorithms_dot-plots&Itemid=76

Scoring of alignments

- In order to asses the quality of an alignment a **scoring function** is needed
- A very simple score: percentage of matches

Alignment:

```
seqA  GGTACCTGTATTT
seqB  -GTA--TGTATTA
```

9/13 = 69.2 %

Scoring of alignments

HUMBOLDT-UNIVERSITÄT ZU BERLIN

- **Additive scoring with linear gap penalty:**
  - +1 for match
  - -1 for mismatch
  - -1 gap penalty

- **Gap penalty (GAP):**
  - Introducing gaps is often needed for alignment
  - Minimizing gaps in an alignment is important to create a useful alignment

- **Score** = $\sum_{i=1}^{n} SIM(seqA_i, seqB_i) + \#gaps * GAP$

- Our alignment:

```
GGTACCTGTATTT
-GTA--TGTATTA
-+++--+++++++-
```

→ Score = 9 matches * 1 + 1 mismatches *-1 + 3 gaps * -1 = 5

Scoring of alignments

- Additive scoring with affine gap penalty
- GOP is called "**gap opening penalty**" → not too many small gaps (e.g. -2)
- GEP is called "**gap extension penalty**" → costs less (e.g. -0.5)

- $\sum_{i=1}^{n} SIM(seqA_i, seqB_i) + \#gap\_openings * GOP + \#gap\_extension * GAP$

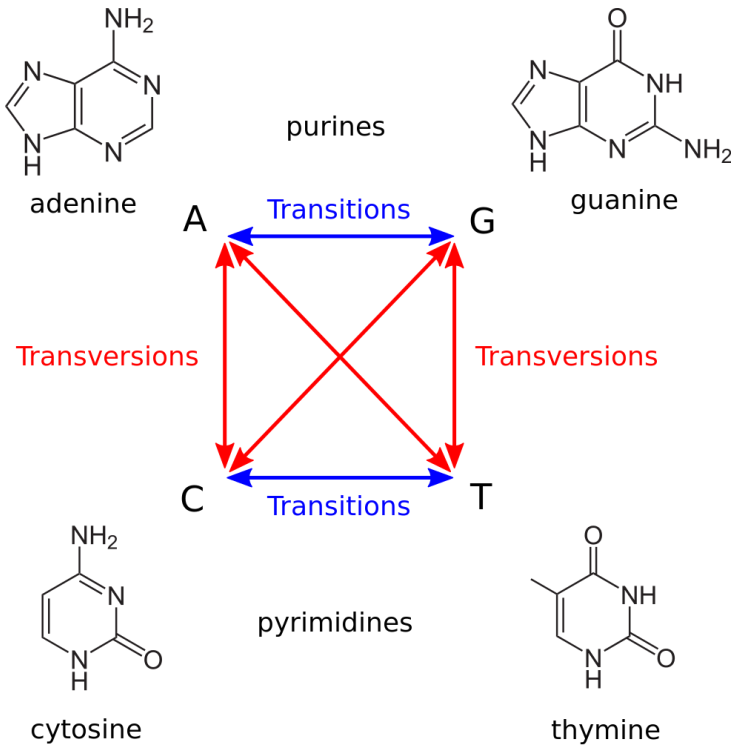
Our alignment:

```
GGTACCTGTATTT
-GTA--TGTATTA
-+++--+++++++-
```

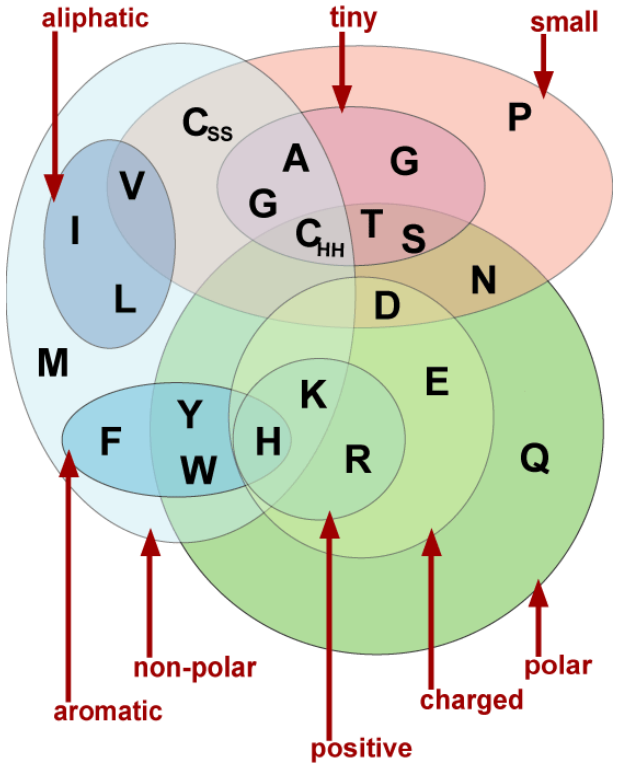
→Score: 9 matches *1 +
1 mismatches * -1 +
2* gaps_opening * -2 +
1 *gaps_extension *-0.5 = 3.5

Substitution probabilities

Transition are more likely
Transversion are rare



Amino acids with similar properties are also more likely to be exchanged



Taylor diagram

Scoring matrices

- The purpose of the scoring matrix is to score one nucleotide against another, e.g. A matched to G is worse than A matched to T

	A	C	G	T	-
A	1	-1	-1	-0.5	-0.5
C	-1	1	-0.5	-1	-0.5
G	-1	-0.5	1	-1	-0.5
T	-0.5	-1	-1	1	-0.5
-	-0.5	-0.5	-0.5	-0.5	NA

Our alignment:

```
GGTACCTGTATTT
-GTA--TGTATTA
-+++--++++++-
```

→Score:

9 matches * 1 +
1 mismatch(A/T) * -0.5 +
3 gaps * -0.5 = 7

(no gap openings)

Scoring matrices

HUMBOLDT-UNIVERSITÄT ZU BERLIN

- Scoring matrices are created based on biological evidence
- Alignments can be thought of as two sequences that differ due to mutations

- Therefore substitution matrices for proteins:
  - BLOSUM (Blocks Substitution Matrix)
  - PAM (Point Accepted Mutation) matrix

Scoring matrices - BLOSUM

- BLOSUM (Blocks Substitution Matrix)
  - Amino acids in the table grouped according to the chemistry of the side chain
  - Each value in the matrix is calculated by dividing the frequency of occurrence of the amino acid pair in the BLOCKS database
    - Score = 0 : Frequency with which two amino acids were found aligned in the database was as expected by chance
    - Score >0 : Alignment was found more often than by chance
    - Score <0 : Alignment was found less often than by chance

  - BLOSUM r: the matrix built from blocks with no more than r% of similarity
    - BLOSUM62 is the matrix built using sequences with no more than 62% similarity

    - BLOSUM62: moderate related proteins
    - BLOSUM80: more related proteins
    - BLOSUM45: distantly related proteins

Fun fact about BLOSUM62

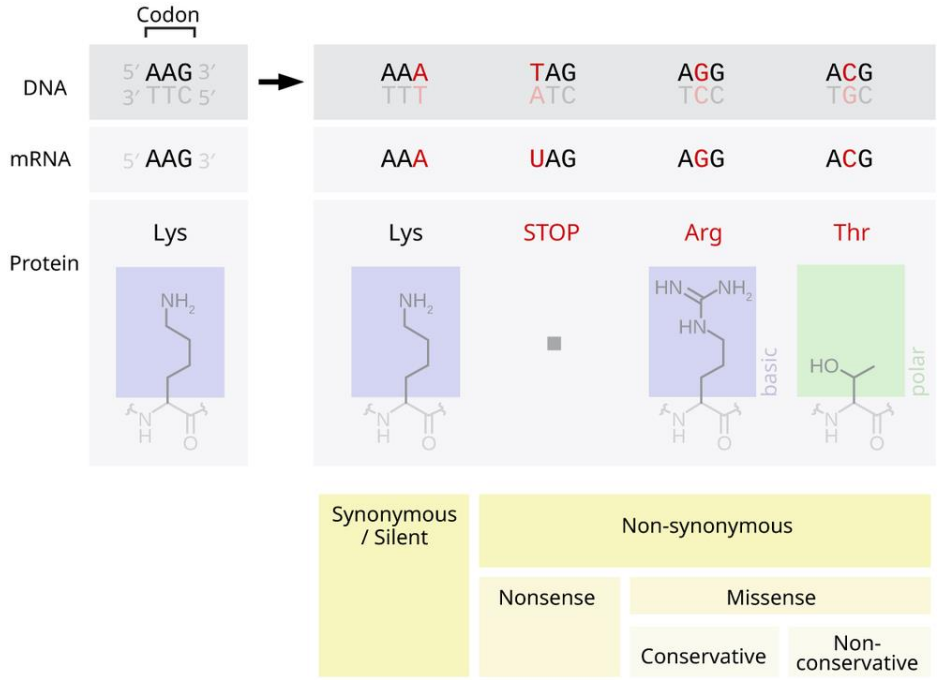
- BLOSUM 62 is the default matrix for protein BLAST
- BLOSUM62 used for so many years as a standard is not exactly accurate according to the algorithm described by Henikoff and Henikoff
- Surprisingly, the miscalculated BLOSUM62 improves search performance

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

https://de.wikipedia.org/wiki/BLOSUM#/media/Datei:Blosum62-dayhoff-ordering.svg

Scoring matrices - PAM

- **PAM (Point Accepted Mutation)**
  - Replacement of a amino acid with another amino acid, which is accepted by the processes of natural selection
  - Example: PAM for lysine→ Missense mutations may be classed as PAMs if the mutated protein is not rejected by natural selection
- **Substitution matrix PAMn:**
  - PAM1 matrix indicates the rate at which substitution would be expected if 1% of the amino acids had changed, thus corresponding to 99% similarity
  - Not quite correct, but good to remember: Percentage of allowed mutations



https://en.wikipedia.org/wiki/Point_accepted_mutation#/media/File:Point_mutations-en.png

PAM250

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

- PAM250 is commonly used for sequence comparison
- Probabilities in a PAM matrix are multiplied by 10000 for the sake of clarity

https://en.wikipedia.org/wiki/Point_accepted_mutation#/media/File:PAM250.png

Differences between PAM and BLOSUM

HUMBOLDT-UNIVERSITÄT ZU BERLIN

PAM

- Based on global alignments of closely related proteins
- PAM1 is the matrix calculated from comparisons of sequences with no more than 1% divergence
- Other PAM matrices are extrapolated from PAM1
- Higher numbers in matrices naming scheme denote larger evolutionary distance

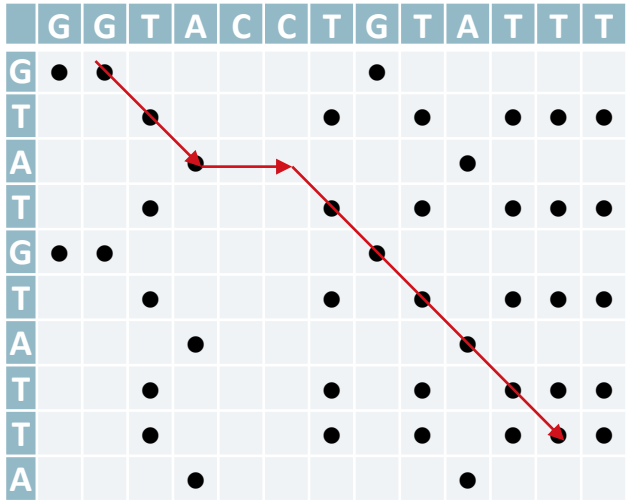
BLOSUM

- Based on local alignments
- BLOSUM 62 is a matrix calculated from comparisons of sequences with no more than 62% identical
- Based on observed alignments; they are not extrapolated from comparisons of closely related proteins
- Larger numbers in matrices naming scheme denote higher sequence similarity and therefore smaller evolutionary distance

Finding the optimal alignment

- **Optimal alignment** → Find the minimal path through the dot plot

- **Needleman- Wunsch algorithm (1970)**
  - First applications of dynamic programming to compare biological sequences
  - Algorithm divides a large problem (e.g. sequence) into smaller problems, and it uses those solutions to find an optimal solution to the larger problem
  - Finding optimal **global** alignment



- **Smith-Waterman algorithm (1981)**
  - Instead of looking at the entire sequence, the Smith-Waterman algorithm compares segments of all possible lengths and optimizes the similarity measure

Needleman-Wunsch algorithm

- Example:
  - Sequence A: GGCAG
  - Sequence B: GAGCTG

Score: +1 for match
-1 for mismatch
-2 for gap

- Recursion (Needleman-Wunsch):

$$D_{i,j} = max \begin{cases} D_{i-1,j-1} + score(a_i, b_j) \\ D_{i-1,j} + score(a_i, -) \\ D_{i,j-1} + score(-, b_j) \end{cases}$$

→diagonal
→von oben
→von links

$$D_{1,1} = max \begin{cases} D_{1-1,1-1} + score(a_1, b_1) \\ D_{1-1,1} + score(a_1, -) \\ D_{1,1-1} + score(-, b_1) \end{cases}$$

$$= max \begin{cases} 0 + 1 \\ -2 + -2 \\ -2 + -2 \end{cases}$$

B

A	D	-	G_1	A_2	G_3	C_4	T_5	G_6
	-	0	-2	-4	-6	-8	-10	-12
	G_1	-2	1					
	G_2	-4						
	C_3	-6						
	A_4	-8						
	G_5	-10						

Needleman-Wunsch algorithm

- Example:
  - Sequence A: GGCAG
  - Sequence B: GAGCTG

Score: +1 for match
-1 for mismatch
-2 for gap

- Recursion (Needleman-Wunsch):

$$D_{i,j} = max \begin{cases} D_{i-1,j-1} + score(a_i, b_j) \\ D_{i-1,j} + score(a_i, -) \\ D_{i,j-1} + score(-, b_j) \end{cases}$$

→diagonal
→von oben
→von links

$$D_{1,2} = max \begin{cases} D_{1-1,2-1} + score(a_1, b_2) \\ D_{1-1,2} + score(a_1, -) \\ D_{1,2-1} + score(-, b_2) \end{cases}$$

$$= max \begin{cases} -2 + -1 \\ -4 + -2 \\ 1 + -2 \end{cases}$$

B

A

D	-	G_1	A_2	G_3	C_4	T_5	G_6
-	0	-2	-4	-6	-8	-10	-12
G_1	-2	1	-1				
G_2	-4						
C_3	-6						
A_4	-8						
G_5	-10						

Needleman-Wunsch algorithm

- Example:
  - Sequence A: GGCAG
  - Sequence B: GAGCTG

Score: +1 for match
-1 for mismatch
-2 for gap

- Recursion (Needleman-Wunsch):

$$D_{i,j} = max \begin{cases} D_{i-1,j-1} + score(a_i, b_j) \\ D_{i-1,j} + score(a_i, -) \\ D_{i,j-1} + score(-, b_j) \end{cases}$$

→diagonal
→von oben
→von links

$$D_{1,3} = max \begin{cases} D_{1-1,3-1} + score(a_1, b_3) \\ D_{1-1,3} + score(a_1, -) \\ D_{1,3-1} + score(-, b_3) \end{cases}$$

$$= max \begin{cases} -4 + \quad 1 \\ -6 + \quad -2 \\ -1 + \quad -2 \end{cases}$$

B

A

D	-	G_1	A_2	G_3	C_4	T_5	G_6
-	0	-2	-4	-6	-8	-10	-12
G_1	-2	1	-1	-3			
G_2	-4						
C_3	-6						
A_4	-8						
G_5	-10						

Needleman-Wunsch algorithm

- Example:
  - Sequence A: GGCAG
  - Sequence B: GAGCTG

Score: +1 for match
-1 for mismatch
-2 for gap

- Recursion (Needleman-Wunsch):

$$D_{i,j} = max \begin{cases} D_{i-1,j-1} + score(a_i, b_j) & \rightarrow \text{diagonal} \\ D_{i-1,j} + score(a_i, -) & \rightarrow \text{von oben} \\ D_{i,j-1} + score(-, b_j) & \rightarrow \text{von links} \end{cases}$$

B

A	D	-	G_1	A_2	G_3	C_4	T_5	G_6
	-	0	-2	-4	-6	-8	-10	-12
	G_1	-2	1	-1	-3	-5	-7	-9
	G_2	-4	-1	0	0	-2	-4	-6
	C_3	-6	-3	-2	-1	1	-1	-3
	A_4	-8	-5	-2	-3	-1	0	-2
	G_5	-10	-7	-4	-1	-3	-2	1

→ proceed until completely filled matrix: Score = 1
→ trace back

Needleman-Wunsch algorithm

- Tracking back:
  - Starting in lower right corner
  - Going up, left or diagonal (up-left)
    - Diagonal → match or mismatch
    - Horizontal or vertical → indel

- Alignment:
  - Diagonal → the letters from two sequences are aligned
  - Left → a gap is introduced in the left sequence
  - Up → a gap is introduced in the top sequence
  - G_GCAG
    GAGCTG
    +−++−+

A

B

D	-	G_1	A_2	G_3	C_4	T_5	G_6
-	0	-2	-4	-6	-8	-10	-12
G_1	-2	1	-1	-3	-5	-7	-9
G_2	-4	-1	0	0	-2	-4	-6
C_3	-6	-3	-2	-1	1	-1	-3
A_4	-8	-5	-2	-3	-1	0	-2
G_5	-10	-7	-4	-1	-3	-2	1

Summary

- Conservation and homology
- Sequence alignments
  - Excurse proteins and protein sequence
  - Pairwise alignments
  - Visualization of alignments → dot plot
  - Scoring of alignments
  - Optimal alignment using Needleman-Wunsch algorithm

Assignment for today: see Moodle