



# Introduction to Bioinformatics

## Topic Seven Protein Alignment

### Lecturers

Gregory R. Grant

Fall 2023

Gregory R. Grant

Genetics Department

[ggrant@pennmedicine.upenn.edu](mailto:ggrant@pennmedicine.upenn.edu)

### Teaching Assistants

Chetan Vadali

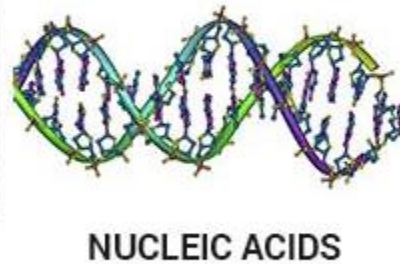
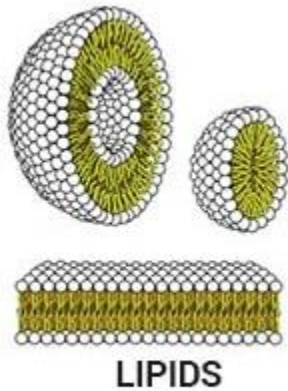
Jianing Yang

*ITMAT Bioinformatics Laboratory*

*University of Pennsylvania*

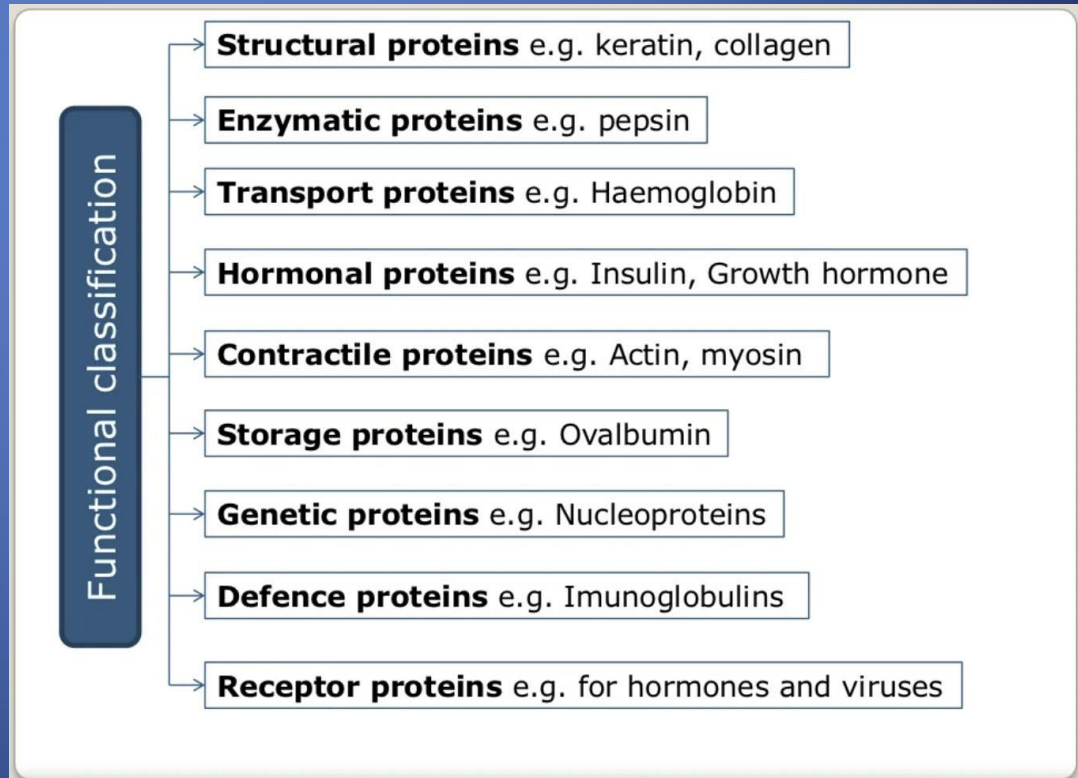
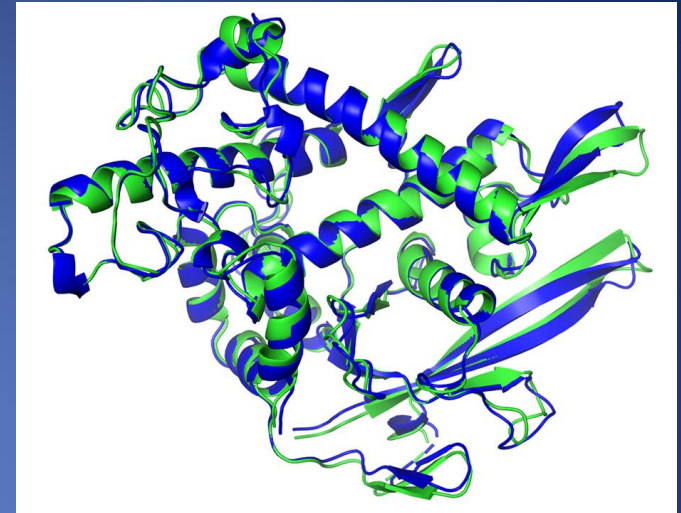
# Biomolecules

- Biological molecules fall into four major categories.
- We've talked about nucleic acid; we turn our attention now to Proteins.



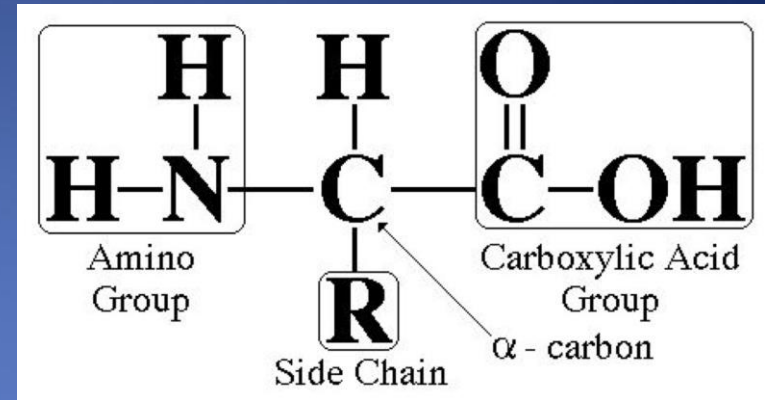
# Protein

- Proteins are the most abundant organic molecules in an organism.
  - Making up about 50% of the dry weight of a cell.
- Proteins are involved in both structure and function.



# Amino Acids

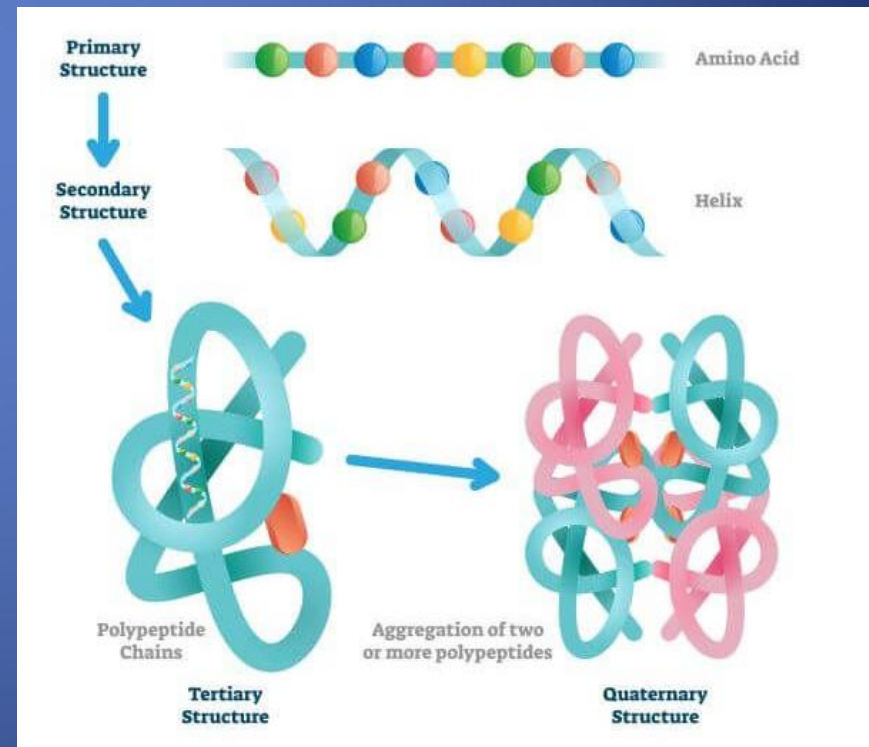
- Proteins are constructed by attaching amino together in a chain.
  - There are 20 amino acids.
  - Some obscure organisms have 21
- Amino acids have various chemical properties that determine the protein's shape and function.



Nature	Amino acids
<b>NEUTRAL</b> : Amino acids with 1 amino and 1 carboxyl group	Glycine (Gly), Alanine (Ala), Valine (Val), Leucine (Leu), Isoleucine (Ile)
<b>ACIDIC</b> : 1 extra carboxyl group	Aspartic acid (Asp), Asparagine (Asn), Glutamic acid (Glu), Glutamine (Gln)
<b>BASIC</b> : 1 extra amino group	Arginine (Arg), Lysine (Lys)
<b>S - CONTAINING</b> : Amino acids have sulphur	Cysteine (Cys), Methionine (Met)
<b>ALCOHOLIC</b> : Amino acids having -OH group	Serine (Ser), Threonine (Thr), Tyrosine (Tyr)
<b>AROMATIC</b> : Amino acids having cyclic structure	Phenylalanine (Phe), Tryptophan (try)
<b>HETEROCYCLIC</b> : amino acids having N in ring structure	Histidine (His), Proline (Pro)

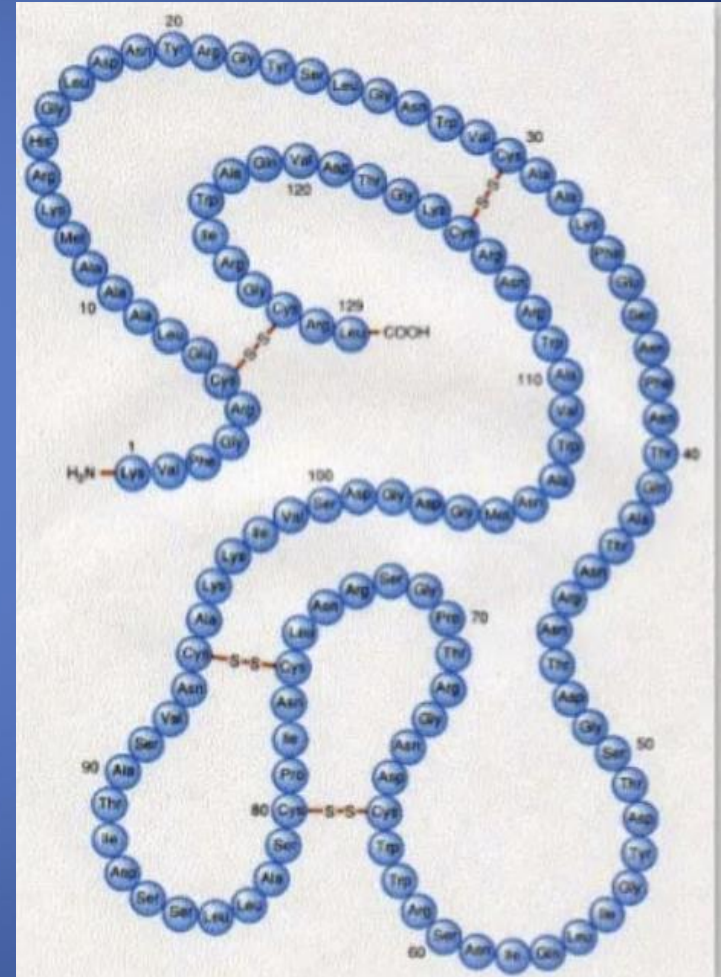
# Structure

- A protein's amino acid sequence is called its “primary structure”.
- But they fold up and combine into complex shapes.
- The ultimate structure is described hierarchically.
  - There are four levels in the hierarchy.



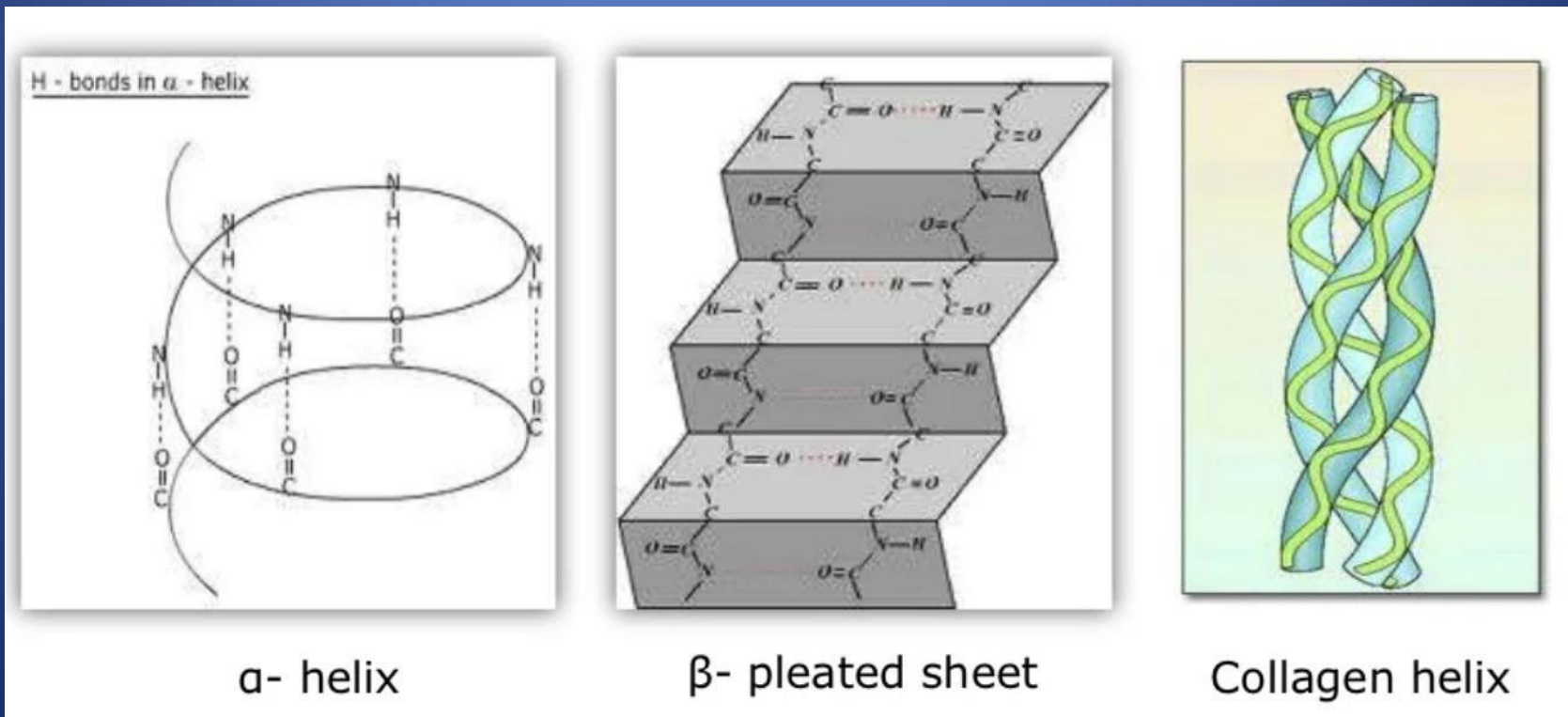
# Primary Structure

- The primary structure refers to the number and linear sequence of amino acids in the polypeptide chain and the location of the disulphide bridges.
  - Disulfide bridges are covalent links between the Sulphur atoms of two cysteine amino acids.
  - Their formation stabilizes the tertiary and higher order structure of proteins.



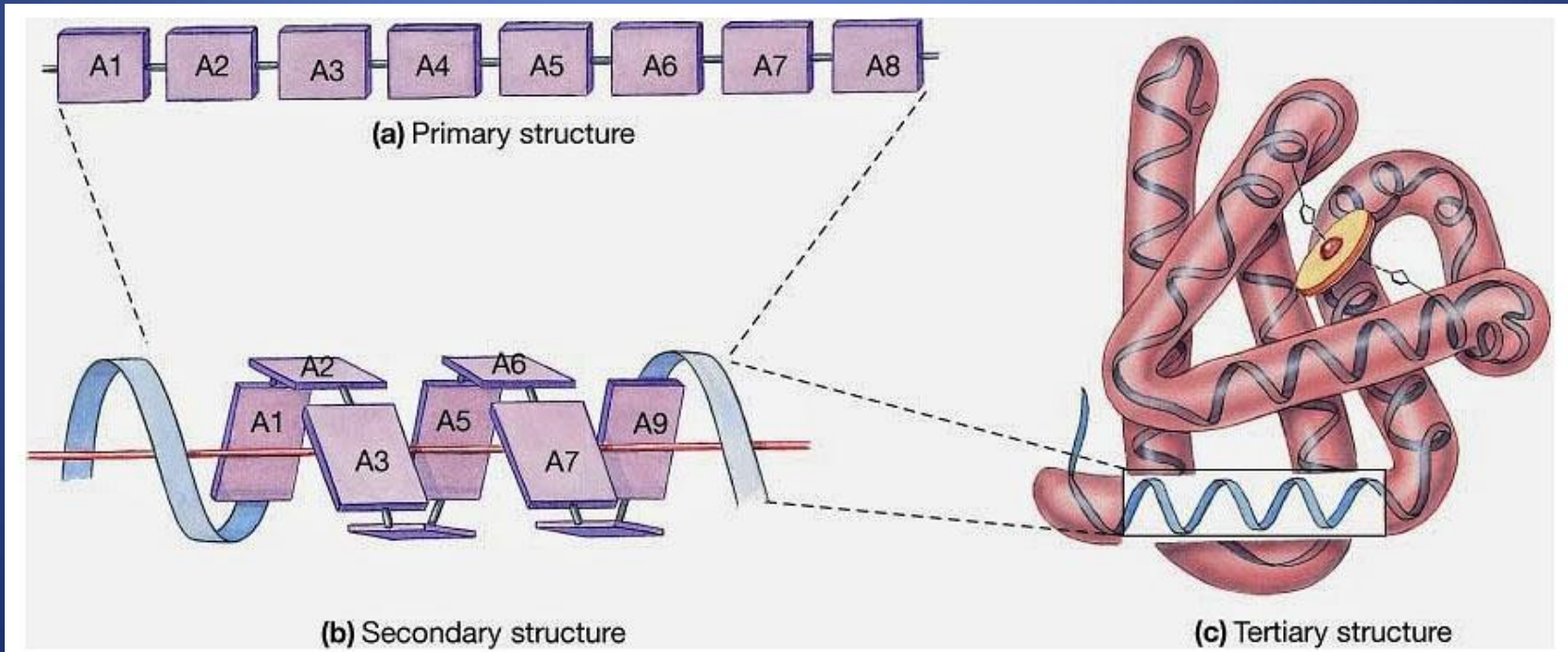
# Secondary Structure

- The linear chain folds into three types of coiled structures:
  - $\alpha$ -helix,  $\beta$ -sheet, Collagen-helix



# Tertiary Structure

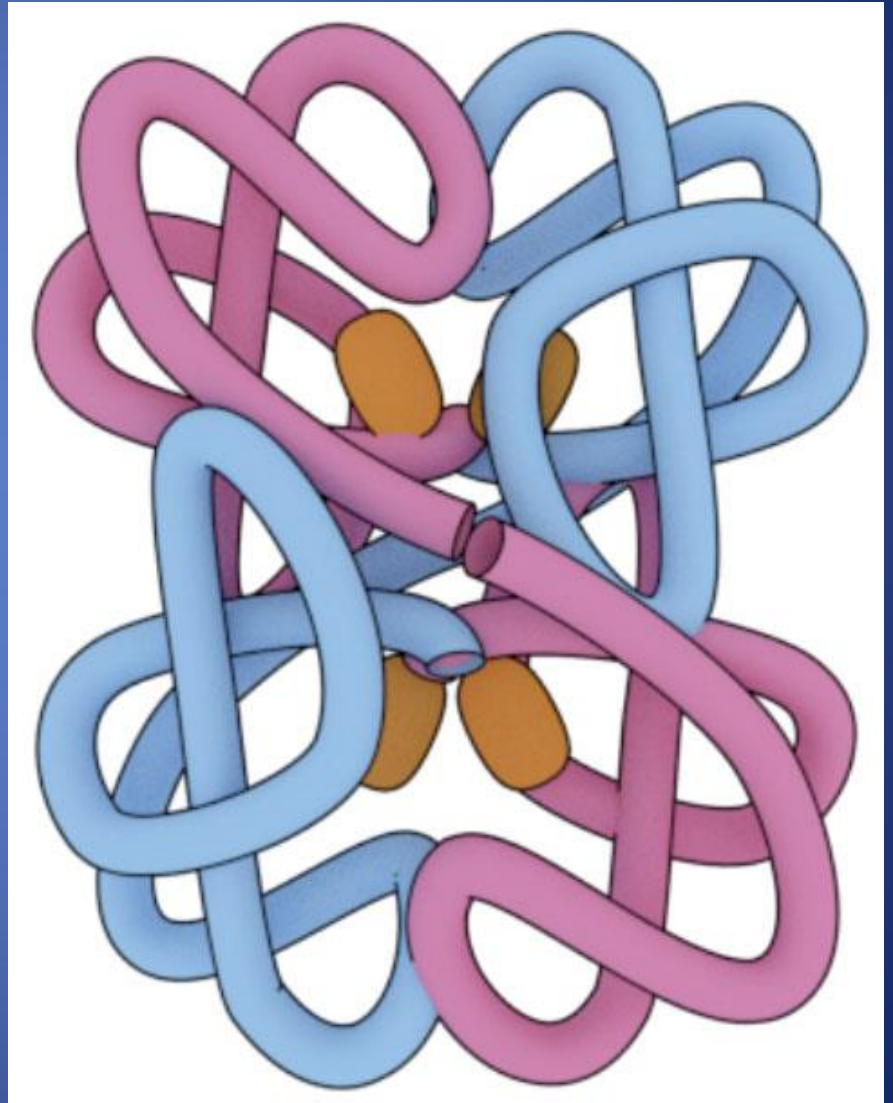
- The helical polypeptide folds in upon itself to assume a complex, but specific, form.





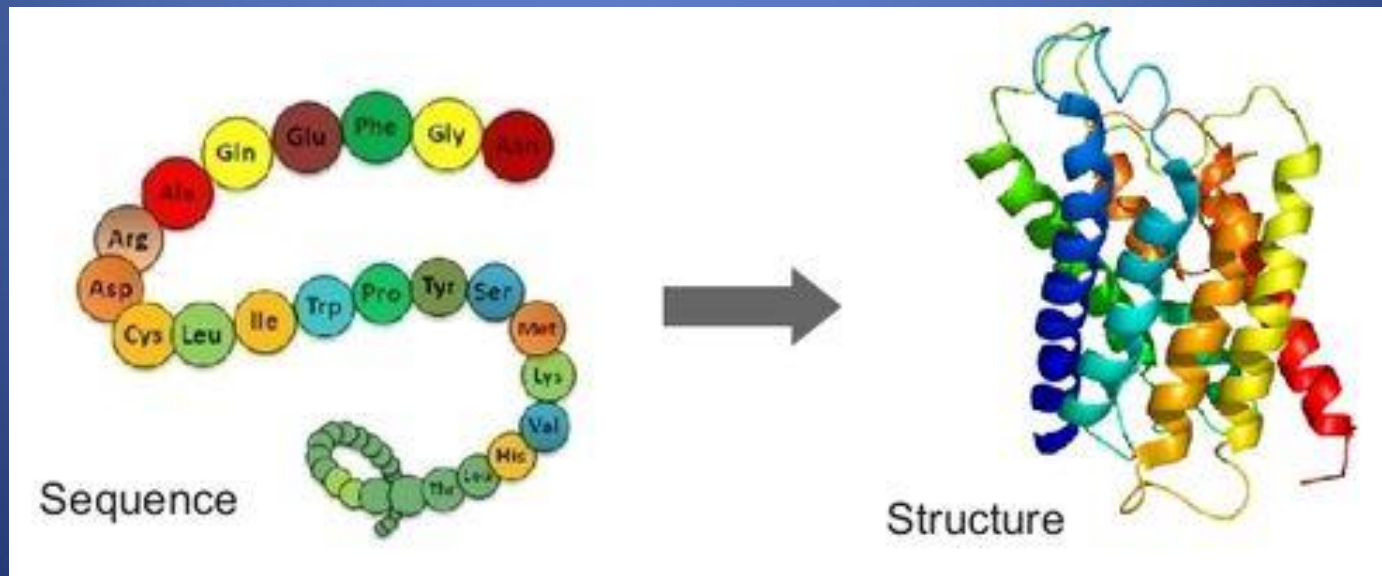
# Quaternary Structure

- Quaternary structure is when two or more proteins (polypeptide chains) combine into a complex.
- Hemoglobin, for example, consists of four proteins.



# Protein Structure Inference

- An active domain of computational biology is the inference of protein structure.
- Given just the sequence of amino acids, determine the protein's secondary and tertiary structures.



# CASP

- A competition is held biannually (once every two years).
- Contestants are given only the amino acid sequence of a protein.
  - Whose structure, which has been determined the hard way, is known only to the judges.
- Contestants submit an algorithm and the one that achieves the closest structure to the truth wins.
- This is extremely competitive and winning is extremely good for your career.

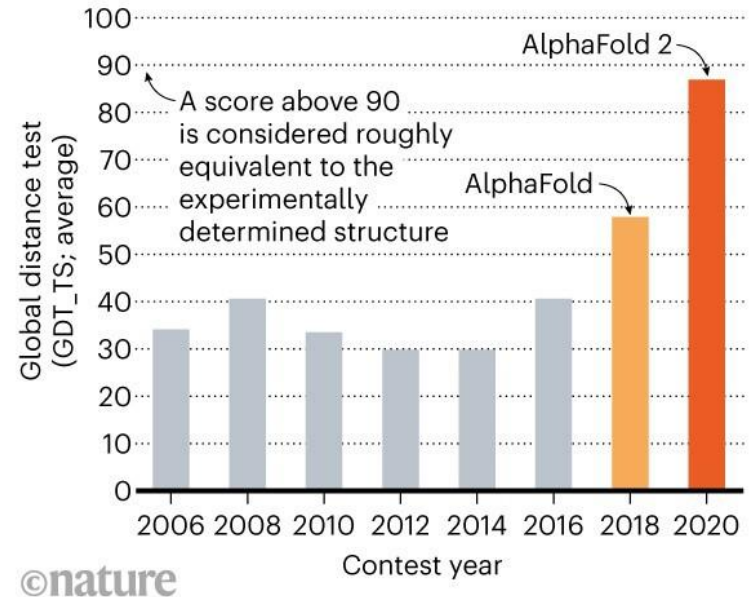
# CASP 2020

- CASP 2020 was won by Google.
  - “An artificial intelligence (AI) network developed by Google AI offshoot DeepMind has made a gargantuan leap in solving one of biology’s grandest challenges — determining a protein’s 3D shape from its amino-acid sequence.”

- “DeepMind’s program, called AlphaFold, outperformed around 100 other teams in a biennial protein-structure prediction challenge called CASP.”

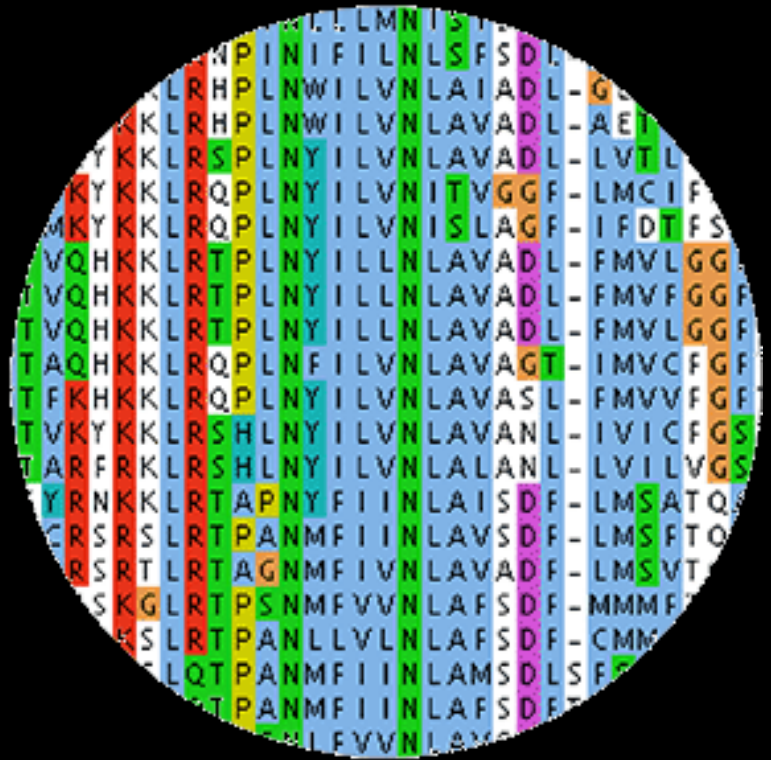
## STRUCTURE SOLVER

DeepMind’s AlphaFold 2 algorithm significantly outperformed other teams at the CASP14 protein-folding contest — and its previous version’s performance at the last CASP.



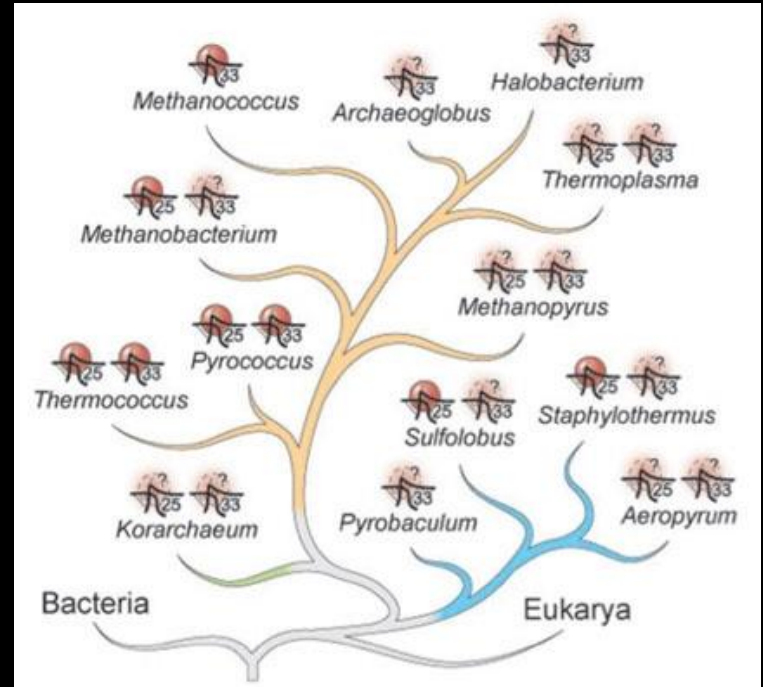
# Protein Alignment

- Our focus in protein space will be on alignment.
- On its face, alignment of proteins is a similar problem to alignment of nucleic acids.
  - Just a different alphabet, 20 letters instead of 4.
- In some ways it's simpler.
  - Proteins are always relatively short.
  - Proteins are less burdened by low complexity or repeat sequence.
  - Shorter alignments can be significant.
    - Since there's an alphabet of 20.
    - There are 25,600,000,000 possible peptides of length 8.



# Evolution

- When we align proteins, most of the time it is because we are interested in their evolutionary relationship.
- Unlike nucleotides in nucleic acid, amino acids tend to be more interchangeable.
  - For example, two amino acid sequences could be determined to be related, even if every one of the individual amino acids have changed.



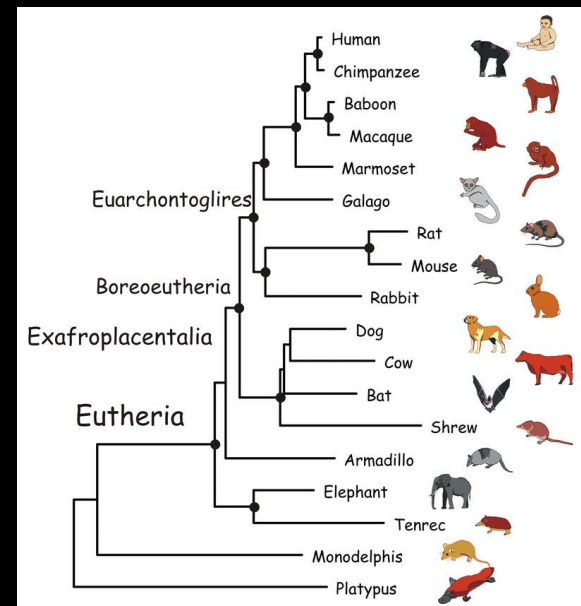
# Amino Acids

- Amino acids with the same chemical properties, can often substitute for each other without changing the function of the protein.

Nature	Amino acids
<b>NEUTRAL</b> : Amino acids with 1 amino and 1 carboxyl group	Glycine (Gly), Alanine (Ala), Valine (Val), Leucine (Leu), Isoleucine (Ile)
<b>ACIDIC</b> : 1 extra carboxyl group	Aspartic acid (Asp), Asparagine (Asn), Glutamic acid (Glu), Glutamine (Gln)
<b>BASIC</b> : 1 extra amino group	Arginine (Arg), Lysine (Lys)
<b>S – CONTAINING</b> : Amino acids have sulphur	Cysteine (Cys), Methionine (Met)
<b>ALCOHOLIC</b> : Amino acids having –OH group	Serine (Ser), Threonine (Thr), Tyrosine (Tyr)
<b>AROMATIC</b> : Amino acids having cyclic structure	Phenylalanine (Phe), Tryptophan (try)
<b>HETEROCYCLIC</b> : amino acids having N in ring structure	Histidine (His), Proline (Pro)

# Impact on Alignment

- How do we account for this extra flexibility in alignment?
  - It's a problem that is handled by the choice of a good scoring scheme.
- We must define the right scoring scheme that reflects the different substitution probabilities of the 20 amino acids.
- But these probabilities depend on the evolutionary distance between the two sequences being aligned.
  - This creates somewhat of a catch-22 situation.
  - But don't worry, we will bootstrap our way out of that problem, as best as we can, later.





# Substitution Matrices

- A substitution matrix is a 20x20 matrix which tabulates scores for each of the possible substitutions.
  - A scoring scheme in general is a substitution matrix together with a gap penalty function.
  - This includes things which ‘substitute’ for themselves (to be found on the main diagonal of the matrix).
  - There is no directionality however, so the matrix is symmetric across the diagonal.
    - This means we assume it’s equally likely that amino acid *A* substitutes for *B* as it is that amino acid *B* substitutes for *A*.

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

# The PAM250 Substitution Matrix

- Notice it's as likely a B becomes a D as it is to stay a B.
  - That is why a true evolutionarily correct alignment can be largely or entirely mismatches.

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

# Impact on Needleman-Wunsch

- The only difference is the score of a mismatch is taken from the amino-acid substitution matrix.
  - The score for D/L is -4.

		D	Q
	0	-1	-2
L	-1		
D	-2		
Q	-3		

# Impact on Needleman-Wunsch

- The score for D/L is -4.
  - Assume a gap penalty of -1

- Therefore, along the diagonal we get:

$$0 - 4 = -4$$

- And coming from the horizontal or vertical we get

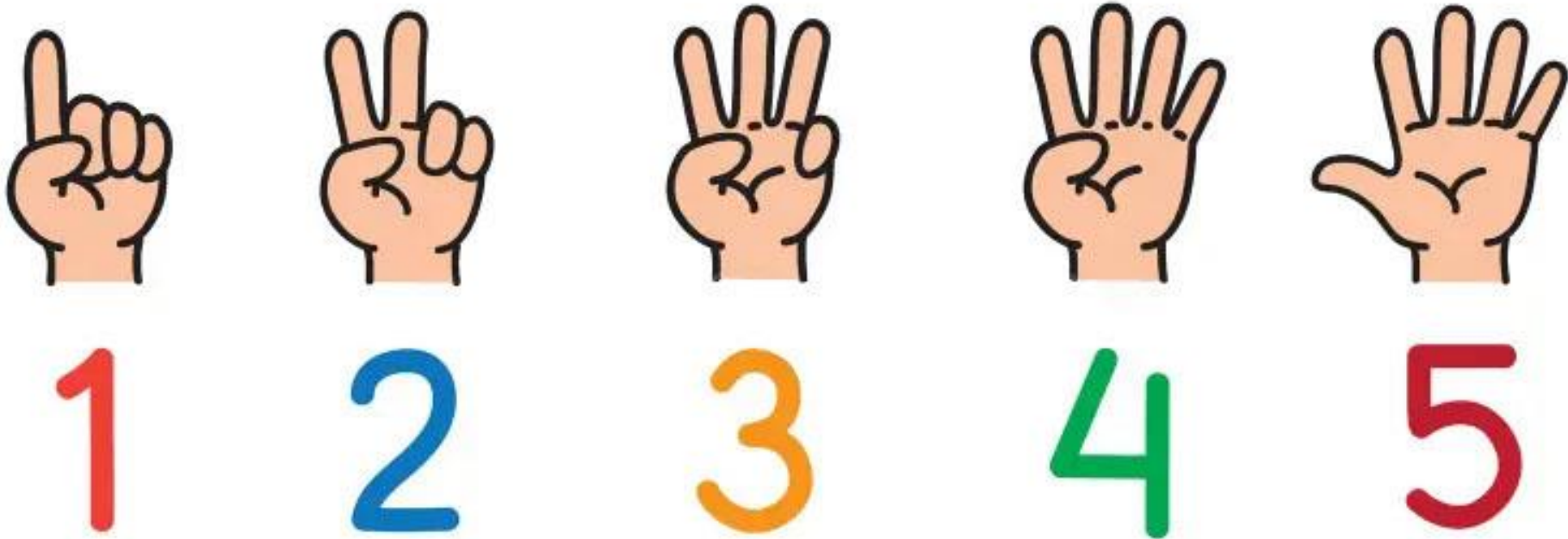
$$-1 - 1 = -2$$

		D	Q
	0	-1	-2
L	-1		
D	-2		
Q	-3		

# Impact on Needleman-Wunsch

- The maximum of -2 and -4 is -2.

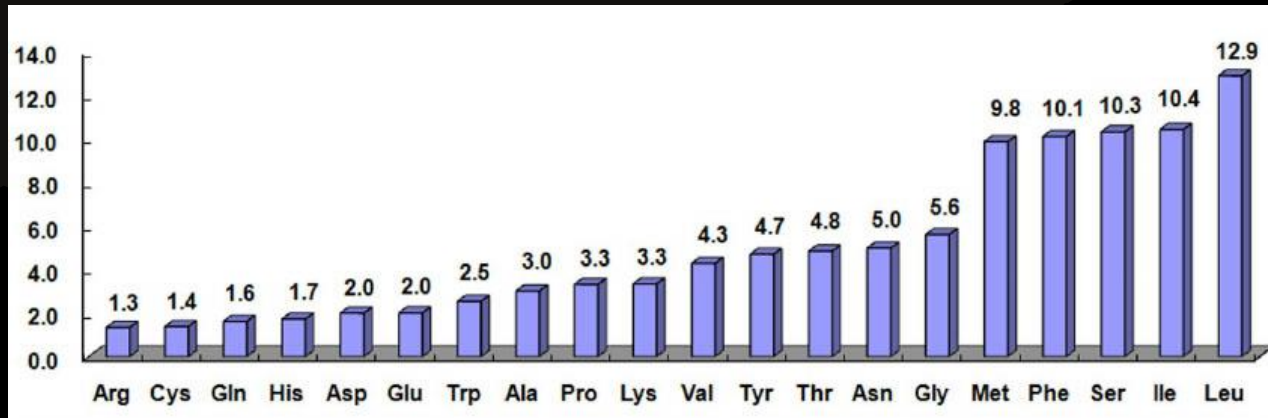
		D	Q
	0	-1	-2
L	-1	-2	
D	-2		
Q	-3		



## Counting

- Suppose there are just two amino acids  $A$  and  $B$ .
- In creating substitution matrices, we must estimate the probability  $p_{AB}$  that  $A$  changes into  $B$  *in one generation*.
- Suppose we have data
  - E.g., a protein of length 1000 in parent and offspring.
- Suppose  $A$  changed into  $B$  a total of 17 times.
- Is that enough information to estimate the probability  $p_{AB}$  ?

# Probabilities



- It is not enough information to estimate the probability  $p_{AB}$  ?
  - We also need to know how many *As* were in the 1000 positions to begin with.
- If there were 200 *As*, then
$$p_{AB} = 17 / 200$$
- Let  $p_A = 200 / 1000 = 0.2$ , the frequency of *A*'s
- And  $p_B = 800 / 1000 = 0.8$ , the frequency of *B*'s
  - We call these the 'background frequencies' of *A* and *B*.

# Null Hypothesis Probabilities

- Consider lining up two random sequences of 1000 *A*s and *B*s (without gaps) generated according to the background frequencies.
  - Assume  $p_A$  is the background frequency of *A*'s in the first sequence and  $p_B$  is the background frequency of *B*'s in the second sequence.
- Then the probability that an *A* in the first sequence is aligned with a *B* in the second, at any given location, is  $2 p_A p_B$
- This is the null hypothesis probability of *A*'s aligning with *B*'s, assuming there is no actual relationship between the sequences.





# Likelihood Ratios

- We consider a substitution as “likely” if it is occurring more often than by chance, where “by chance” is the null hypothesis probability as defined on the previous slide.
  - And unlikely if it is occurring less often than chance.
- This is quantified in the ratio

$$e_{xy} = \begin{cases} \frac{2p_x p_y}{p_{xy}} & \text{if } x \neq y, \\ \frac{p_x p_y}{p_{xy}} & \text{if } x = y. \end{cases}$$

# Scores



- We convert the likelihood ratio into a score by taking -2 times the log base 2 and rounding to the nearest integer.  
$$\text{round}(-2 \log_2 LR)$$
- In this way pairs that are more likely than chance will have positive scores, and those less likely will have negative scores.
- The use of the log of the likelihood ratio is more than just intuitive.
  - It can be justified rigorously by using theory from statistical hypothesis testing and random walks.
    - Which is unfortunately beyond the scope of this class.

# PAM and BLOSUM

Substitution matrices come in families.

- Indexed by natural numbers (e.g. PAM250)
- We will see why shortly.

There are two standard approaches that utilize different mathematical procedures.

- PAM (Accepted Point Mutation)
- BLOSUM (BLOcks SUBstitution Matrices)

Both methods start from a set of 'training' data that can (hopefully) be trusted.

The data and statistical methods are used to infer the likelihood ratios which are then translated into 'scores'.

- These scores are the elements of the matrix.

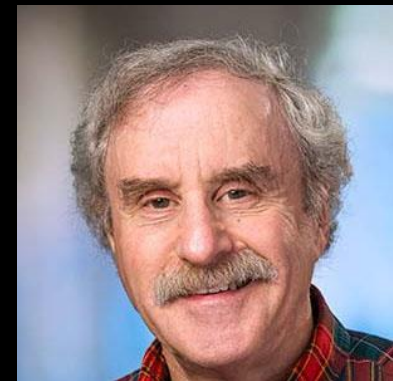
# History

---

- PAM
  - Accepted Point Mutation
  - Dayhoff 1978
  
- BLOSUM
  - **B**LOcks **S**Ubstitution **M**atrices
  - Henikoff and Henikoff 1992

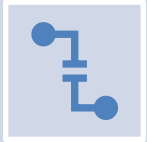


Margaret Dayhoff



Steven Henikoff

# BLOSUM



BLOSUM substitution matrices are more recent than PAM but use a more straightforward approach which we will describe in some detail.

BLOSUM, unlike PAM, does not use the machinery of Markov Chains.



BLOSUM starts with a set of with a set of protein sequences from public databases that have been grouped into related families.

- A block is the un-gapped alignment of a relatively highly conserved region of a family of proteins.
  - We shall see later how to obtain these blocks when we study multiple sequence alignment.

```
A B A C C A
A B B C C A
A A B C A A
A A A C A A
```

# Example of some Real BLOCKS

WWYIR	CASILRKIYIYGPV	GVSRLRTAYGGRK	NRG
WFYVR	CASILRHLYHRSPA	GVGSI TKIYGGRR	RNG
WYYVR	AAAVARHIYLRKTV	GVGRLRKVHGSK	NRG
WYFIR	AASICRHLYIRSPA	GIGSFEKIYGGRR	RRG
WYYTR	AASIARKIYLRQGI	GVGGFQKIYGGRR	RNG
WFYKR	AASVARHIYMRKQV	GVGKLNKLYGGAK	SRG
WFYKR	AASVARHIYMRKQV	GVGKLNKLYGGAK	SRG
WYYVR	TASIARRLYVRSPT	GVDALRLVYGGSK	RRG
WYYVR	TASVARRLYIRSP	GVGALRRVYGGNK	RRG
WFYTR	AASTARHLYLRGGA	GVGSM TKIYGGRR	RNG
WFYTR	AASTARHLYLRGGA	GVGSM TKIYGGRR	RNG
WWYVR	AAALLRRVYIDGPV	GVNSLRTHYGGKK	DRG

A set of four blocks from the Blocks database

# Circularity

- Algorithms used to construct aligned blocks employ substitution matrices.
- Henikoff and Henikoff broke this circularity as follows:
  - They started by using a simple “unitary” substitution matrix where the score is 1 for a match, 0 for a mismatch.
  - They then constructed only those blocks that they could obtain with this simple matrix.
  - This procedure has the effect of generating a conservative set of blocks; that is, it tends to omit blocks with low sequence identity



# Blocks

- This restricts the blocks to ones that are reasonably trustworthy and not biased toward any specific scoring scheme.
- Using the blocks so constructed, Henikoff and Henikoff then count:
  1. The number of occurrences of each amino acid
    - To estimate the ‘background’ probabilities.
  2. The number of occurrences of each pair of amino acids aligned in the same column.
    - To estimate the ‘foreground’ probabilities.





# Simple Example

- Suppose there are only three amino acids  $A$ ,  $B$ , and  $C$ .
- And only one block
- In this block there are 24 amino acids:
  - 14 are  $A$
  - 4 are  $B$
  - 6 are  $C$
- Thus, the observed proportions are

$B$	$A$	$B$	$A$
$A$	$A$	$A$	$C$
$A$	$A$	$C$	$C$
$A$	$A$	$B$	$A$
$A$	$A$	$C$	$C$
$A$	$A$	$B$	$C$

amino acid	proportion of times observed
$A$	$14/24$
$B$	$4/24$
$C$	$6/24$

# BLOSUM Example 1

- There are  $4\binom{6}{2} = 60$  aligned pairs of amino acids in the block.
- These 60 pairs occur with proportions as given in the following table:

aligned pair	proportion of times observed
<i>A</i> to <i>A</i>	26/60
<i>A</i> to <i>B</i>	8/60
<i>A</i> to <i>C</i>	10/60
<i>B</i> to <i>B</i>	3/60
<i>B</i> to <i>C</i>	6/60
<i>C</i> to <i>C</i>	7/60

# BLOSUM Example 1

- We next compare these observed proportions to:
  - The *expected* (null background probability) proportion of times that each amino acid pair is aligned given two random sequences.
    - generated with the estimated background frequencies.
  - The expected proportion of pairs in which A is aligned with A is  $\frac{14}{24} \cdot \frac{14}{24}$
  - The expected proportion of pairs in which A is aligned with B is  $2 \cdot \frac{14}{24} \cdot \frac{4}{24}$
  - and so on.

# BLOSUM Example 1

- These fractions are now used to calculate “estimated likelihood ratios” as shown in the following table:

aligned pair	proportion observed	proportion expected	$2 \log_2 \left( \frac{\text{proportion observed}}{\text{proportion expected}} \right)$
<i>A to A</i>	26/60	196/576	0.70
<i>A to B</i>	8/60	112/576	-1.09
<i>A to C</i>	10/60	168/576	-1.61
<i>B to B</i>	3/60	16/576	1.70
<i>B to C</i>	6/60	48/576	0.53
<i>C to C</i>	7/60	36/576	1.80

# BLOSUM Example 1

- the respective elements in the BLOSUM substitution matrix are found by rounding the numbers in the 4<sup>th</sup> column to the nearest integer.
- In this simplified example, the substitution matrix would thus be

	<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	1	-1	-2
<i>B</i>	-1	2	1
<i>C</i>	-2	1	2

# Bias in the Simple Approach

- This rudimentary approach does yield a useful scoring scheme.
  - It is certainly better than one that merely scores 1 for a match and 0 for a mismatch.
- But it overlooks an important factor that can bias the results.
- The substitution matrix will depend significantly on which sequences of each family happen to be in the database used to create the blocks.
  - If there are many very closely related proteins in one block, and only a few others that are less closely related, then the contribution of that block will be biased toward closely related proteins.

# Bias Example

- Suppose the data in one block are as follows

<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>
<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>
<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>
<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>
<i>A</i>	<i>A</i>	<i>B</i>	<i>D</i>
<i>A</i>	<i>C</i>	<i>B</i>	<i>A</i>
<i>D</i>	<i>A</i>	<i>B</i>	<i>A</i>

- The first four sequences possibly derive from closely related species and the last three from three more distant species.
- Since *A* occurs with high frequency in the first four sequences, the observed number of pairings of *A* with *A* will be higher than is appropriate if we are comparing distantly related sequences.

# Strategy to Overcome the Bias

- Ultimately, it would be preferable to include sequences in each block so that any pair of them have roughly the same amount of “evolutionary distance” between them.
- We could throw out three of them and just keep one.
  - But we would lose some information if the sequences were close but not identical.
- Instead, the first four rows will be “clustered” and treated as one ‘unit’

<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>
<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>
<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>
<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>
<i>A</i>	<i>A</i>	<i>B</i>	<i>D</i>
<i>A</i>	<i>C</i>	<i>B</i>	<i>A</i>
<i>D</i>	<i>A</i>	<i>B</i>	<i>A</i>



# Clustering in Blocks

- to overcome the bias -

- The solution employed by Henikoff and Henikoff is to group, or cluster, those sequences in each block that are “sufficiently close” to each other.
  - Then, treat the resulting cluster as a single sequence.
- First, we need a definition of “sufficiently close.”

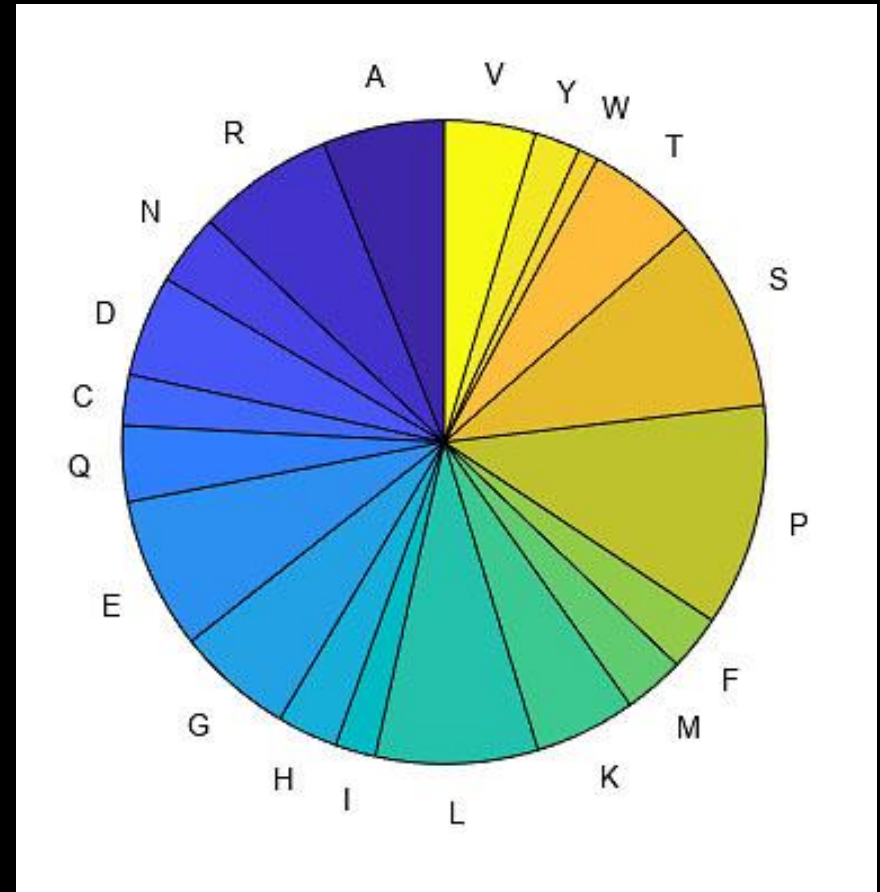
# “Sufficiently Close”

- We specify a cut-off proportion, say 85%, and then group the sequences in each block into clusters in such a way that:
  - each sequence in any cluster has 85% or higher sequence identity to *at least one other* sequence in the cluster in that block.



# Counting individual amino acids:

- The count of each amino acid is found by dividing each occurrence by the number of sequences in the cluster containing that occurrence.
  - Then summing over all occurrences.
  - Don't worry, this will be clear once we do an example.
- These weighted counts are then used as before



# Counting pairs of amino acids:

- If in any block two sequences are in the same cluster, then:
  - In that block no counts are taken between amino acids in those two sequences.
- For any aligned amino acids in sequences *in two different clusters*, then:
  - The count for any amino acid pair is divided by  $nm$ , where  $n$  and  $m$  are the sizes of the two clusters from which the amino acids are taken.
- These weighted counts are then used as before

# BLOSUM Example 2

- We will work a simple example with two blocks

<i>B</i>	<i>A</i>	<i>B</i>	<i>A</i>
<i>B</i>	<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>	<i>C</i>

<i>C</i>	<i>B</i>	<i>B</i>
<i>C</i>	<i>B</i>	<i>B</i>
<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>

- We'll set the identity for clustering to be 0.75.
  - Thus, we cluster the first two sequences in each block together.

<i>B</i>	<i>A</i>	<i>B</i>	<i>A</i>
<i>B</i>	<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>	<i>C</i>

<i>C</i>	<i>B</i>	<i>B</i>
<i>C</i>	<i>B</i>	<i>B</i>
<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>

# BLOSUM Example 2

- Total Symbol Count -

<i>B</i>	<i>A</i>	<i>B</i>	<i>A</i>
<i>B</i>	<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>	<i>C</i>

<i>C</i>	<i>B</i>	<i>B</i>
<i>C</i>	<i>B</i>	<i>B</i>
<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>

- We've grouped the first two rows of block 1.
  - Thus, each column contributes 2 to the total counts of the symbols.
- Similarly, each column of the second block contributes 3.
- Thus, the total count of symbols is 17

# BLOSUM Example 2

## - Counting individual Symbols -

<i>B</i>	<i>A</i>	<i>B</i>	<i>A</i>
<i>B</i>	<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>	<i>C</i>

<i>C</i>	<i>B</i>	<i>B</i>
<i>C</i>	<i>B</i>	<i>B</i>
<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>

- The *As* are counted as follows.
- Block 1:
  - The first column has one *A*.
  - In the second column, since the first two sequences are clustered, the top two *As* contribute  $\frac{1}{2}$  each. So, from column 2 we count  $\frac{1}{2} + \frac{1}{2} + 1 = 2$  *As*.
  - The fourth column contributes  $\frac{1}{2}$  of an *A*.
- Block 2:
  - There are three *As*, each occurrence occurs in a cluster of size one.
- So, in total there are  $13/2$  *As*.

# BLOSUM Example 2

<i>B</i>	<i>A</i>	<i>B</i>	<i>A</i>
<i>B</i>	<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>	<i>C</i>

<i>C</i>	<i>B</i>	<i>B</i>
<i>C</i>	<i>B</i>	<i>B</i>
<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>

- So, the proportion of A's is  $(13/2)/17 = 13/34$ .
- We record the proportions for all symbols in the following table:

amino acid	proportion of times observed
<i>A</i>	13/34
<i>B</i>	5/17
<i>C</i>	11/34



# BLOSUM Example 2

## - Counting Pairs -

<i>B</i>	<i>A</i>	<i>B</i>	<i>A</i>
<i>B</i>	<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>	<i>C</i>

<i>C</i>	<i>B</i>	<i>B</i>
<i>C</i>	<i>B</i>	<i>B</i>
<i>A</i>	<i>B</i>	<i>C</i>
<i>A</i>	<i>A</i>	<i>C</i>

- We turn next count *A/B* pairs.
- There are two occurrences in the first column of the first block which contribute  $\frac{1}{2}$  each, and in the second column of the second block the contribution is  $\frac{1}{2} + \frac{1}{2} + 1$ 
  - So, the total *A/B* count from the two blocks is 3.
- There are a total of 13 pairs in the blocks, four in the first block (each column contributes one pair, or more precisely, two half pairs) and nine in the second block.
- Thus, the proportion of *A/B* pairs is  $\frac{3}{13}$ .

# BLOSUM Example 2 – counting pairs

- We record the proportions for all pairs of symbols in the following table:

aligned pair	proportion of times observed
<i>A to A</i>	$2/13$
<i>A to B</i>	$3/13$
<i>A to C</i>	$5/26$
<i>B to B</i>	$1/13$
<i>B to C</i>	$3/13$
<i>C to C</i>	$3/26$

- The procedure is then carried out as before.

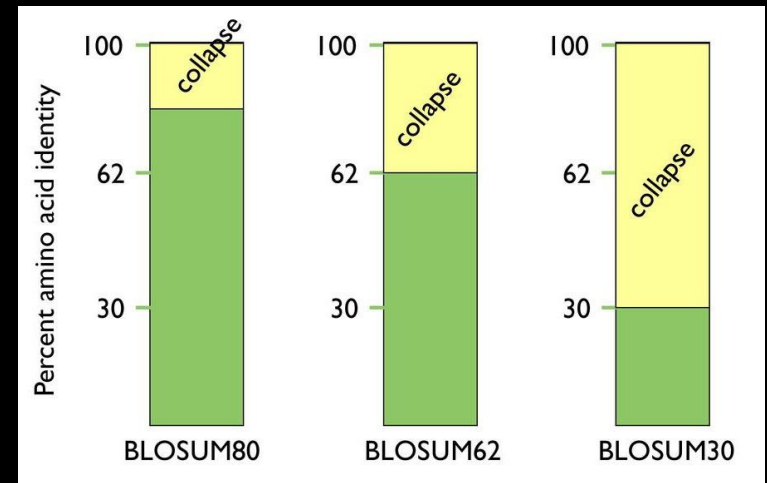


## Iterative Refinement

- After obtaining a BLOSUM substitution matrix as just described:
  - The matrix obtained is then used instead of the conservative “unitary” matrix to construct a second, less conservative, set of blocks.
- A new substitution matrix is then obtained from these blocks.
- Then the process is repeated a third time.
  - Henikoff and Henikoff derive the final family of BLOSUM matrices from this third set of blocks.

# BLOSUM Family of Matrices

- If the 0.85 similarity score criterion is adopted, the final matrix is called a BLOSUM85 matrix.
- In general, if clusters with  $X\%$  identity are used, then the resulting matrix is called BLOSUM $X$ .
  - The BLOSUM matrices typically used are BLOSUM45, BLOSUM62, and BLOSUM80.
- Note that the larger-numbered matrices correspond to more recent divergence, and the smaller-numbered matrices correspond to more distantly related sequences.
  - This is in contrast with PAM matrices where larger numbers correspond to more distantly related sequences.



# BLOSUM62

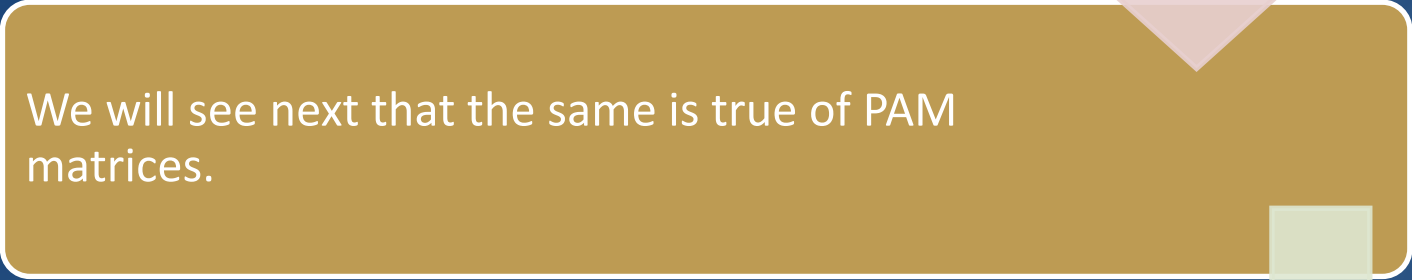
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2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

# Likelihood Ratios

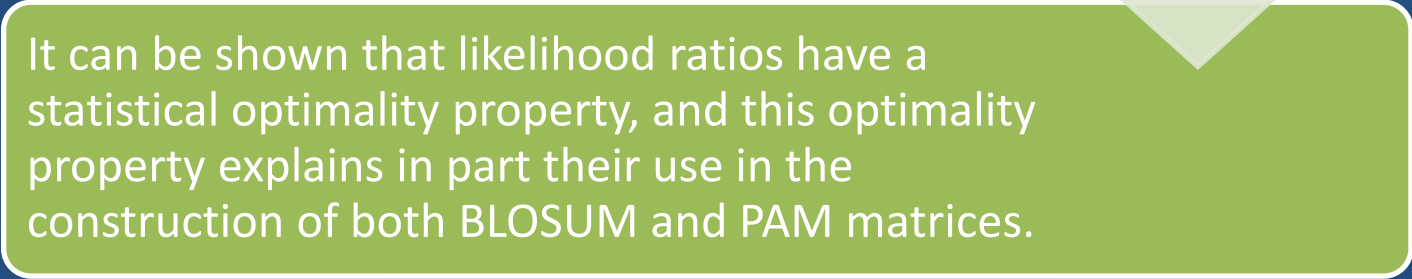
A central feature of the BLOSUM substitution matrix calculation is the use of (estimated) likelihood ratios.



We will see next that the same is true of PAM matrices.



It can be shown that likelihood ratios have a statistical optimality property, and this optimality property explains in part their use in the construction of both BLOSUM and PAM matrices.



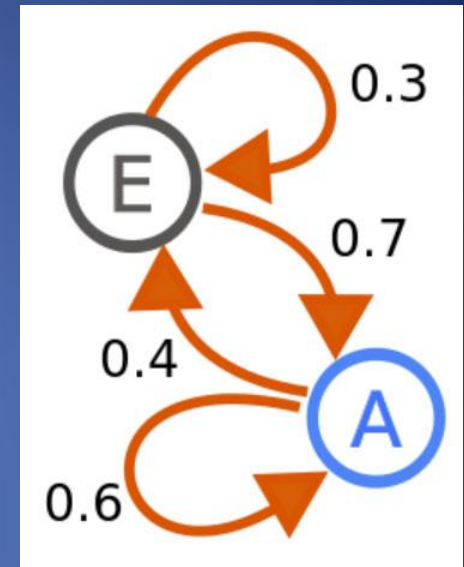
# Markov Models

- PAM matrices are based on Markov Models.
  - We will not have time to discuss Markov Models in detail, but they're extremely important in biology.
- A Markov Model models a process that evolves in time.
- Time proceeds in discrete steps.
  - $t=1,2,3,\dots$
- At each time there's a 'current state'.
- As time increments one unit, the current state changes according to fixed (time independent) probabilities.



# Markov Models

- We represent a Markov Model with a graph.
  - States are represented by nodes.
  - Transitions are represented by directed edges (arrows).
- And we record the transition probabilities between states with a matrix.
- These are used to model evolution of an amino acid in a protein.
  - Where the graph has 20 nodes
  - Time in this case is 'generation'



Example Markov chain with two states

$$\begin{array}{c} E \\ A \end{array} \begin{array}{cc} E & A \\ \left[ \begin{array}{cc} 0.3 & 0.7 \\ 0.4 & 0.6 \end{array} \right] \end{array}$$

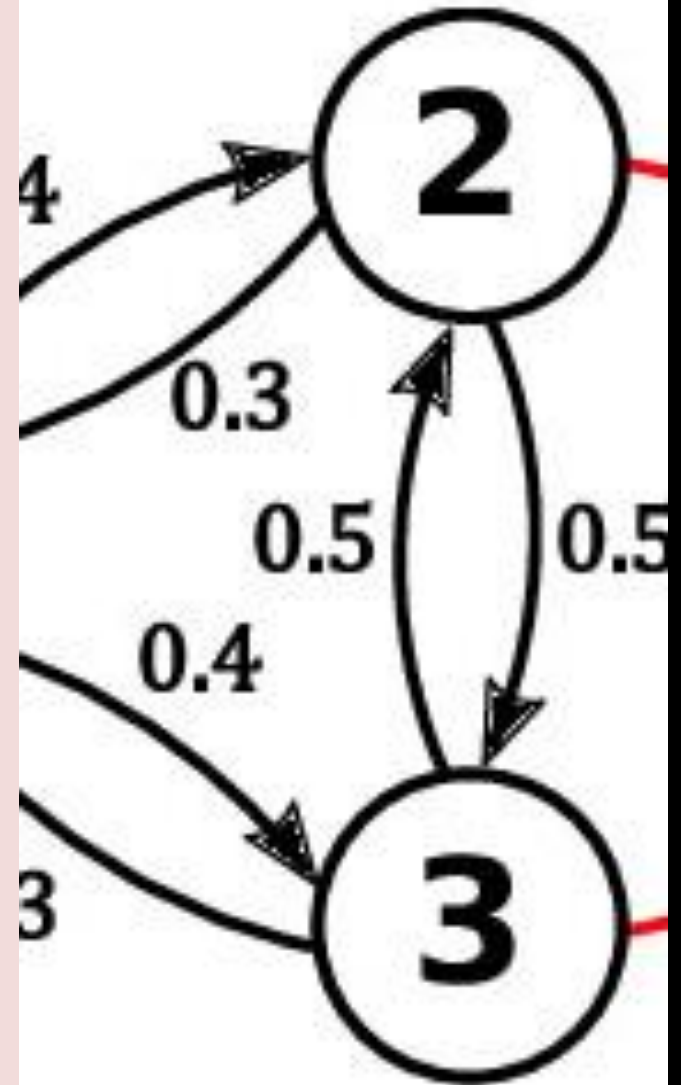
Probability Transition Matrix



# Markov Models

## - basic facts -

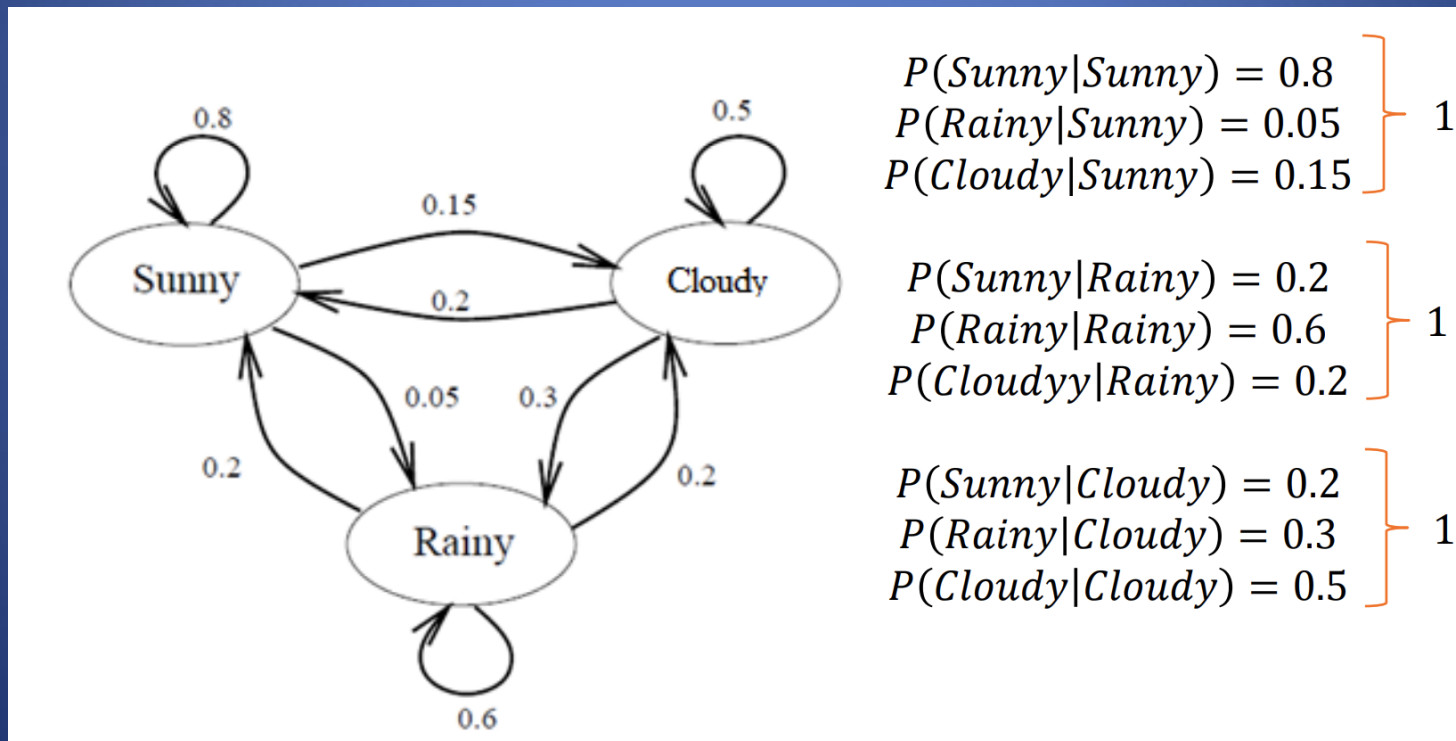
- At each time there's a current state.
- The process starts in some default state at time  $t=0$ , or it chooses an initial state according to an 'initial state probability distribution'.
- The current state updates as time increments according to fixed probabilities.
  - The process is 'memoryless' meaning the probability of being in a state at time  $t+1$  only depends on what state it's in at time  $t$  and does not depend on what state it was in at any previous time points.



# Markov Example

- weather -

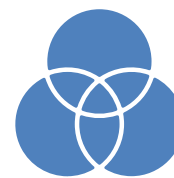
- Simple model to predict the weather each day from the previous day.



# Markov Models

## - more facts -

- **Notation:** We denote the element in the  $i^{\text{th}}$  row and  $j^{\text{th}}$  column of the matrix  $M$  by  $p_{ij}$ .
- If the Markov model has transition matrix  $M$ , then the probabilities of transitioning between states under two increments of time is  $M^2$ .
  - And similarly, the probability of transitioning between states under  $n$  increments of time is  $M^n$ .
- You learn about matrix multiplication in linear algebra.
  - Don't worry if it's new to you, just know it can be done.
  - $M^n$  is a matrix the same size as  $M$  and is probably *not* defined the way you would guess – in other words, it's not just the matrix whose entries are the  $n^{\text{th}}$  powers of the entries in the original matrix.



# Scores from a Markov Model


- Given the Markov transition matrix, we turn it into a score by the formula

$$C \cdot \log \left( \frac{p_{jk}}{p_k} \right)$$

- This is again a log likelihood ratio
  - It may not look like one, but it's derived from:  $p_j p_{jk} / p_j p_k$ 
    - The numerator is the (estimated) probability of finding amino acid  $j$  aligned to amino acid  $k$  in real data.
    - The denominator is the (estimated) probability of finding amino acid  $j$  aligned to amino acid  $k$  in random data.
  - $p_j$  and  $p_k$  are the background frequencies before the transition
    - These give the initial state distribution.
  - $p_{jk}$  is the probability amino acid  $j$  turns into amino acid  $k$  in one time step.
- $C$  is a constant, which is used to scale the numbers so that when we round to the nearest integer, the scores spread out nicely.

# Derivations - PAM

The Markov transition matrix  $M$  is inferred from the data by building phylogenetic trees and inferring ancestral states.



We've talked a bit about phylogenetics when we did multiple sequence alignment.

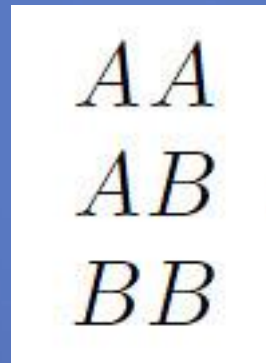
The model is constructed from closely related sequences that can be easily aligned without gaps and where it is unlikely that two mutations happened at the same location.

Using closely related sequences, we need a large data set to represent all possible changes, enough times each, to estimate their probabilities.

Fortunately, the world is full of data for us to use for this purpose.

# Example PAM

- We'll construct PAM matrices assuming the data consists of one simple alignment of 3 sequences of length 2.

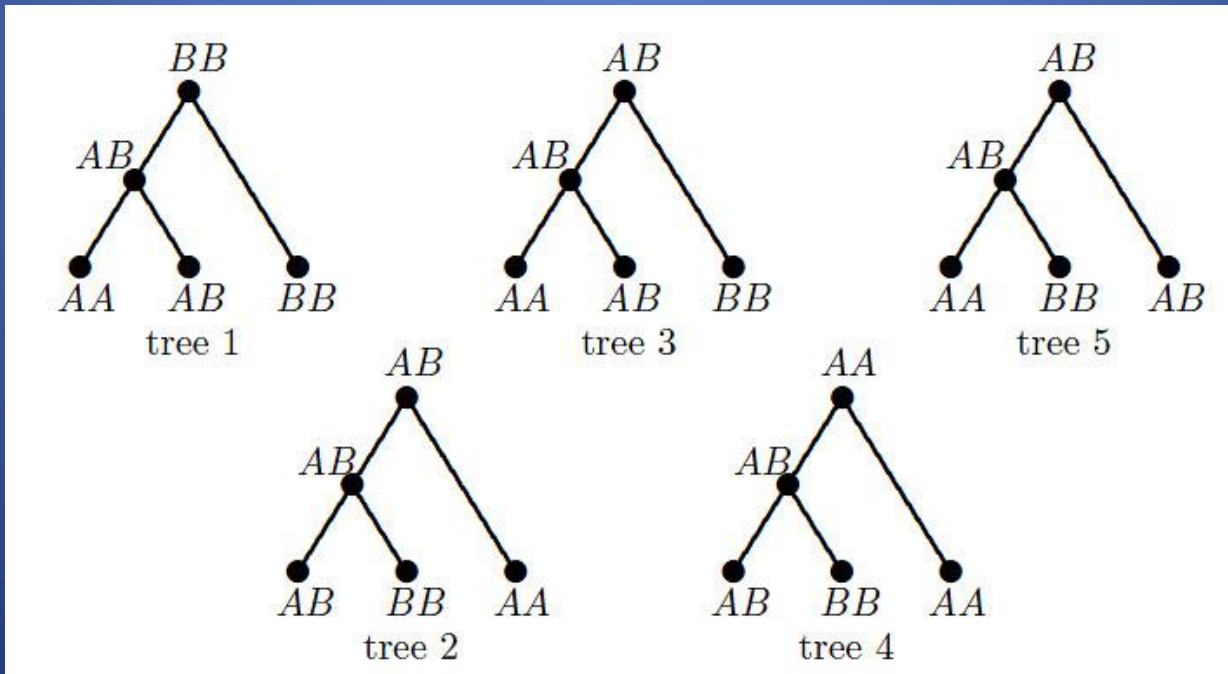


*AA*  
*AB*  
*BB*

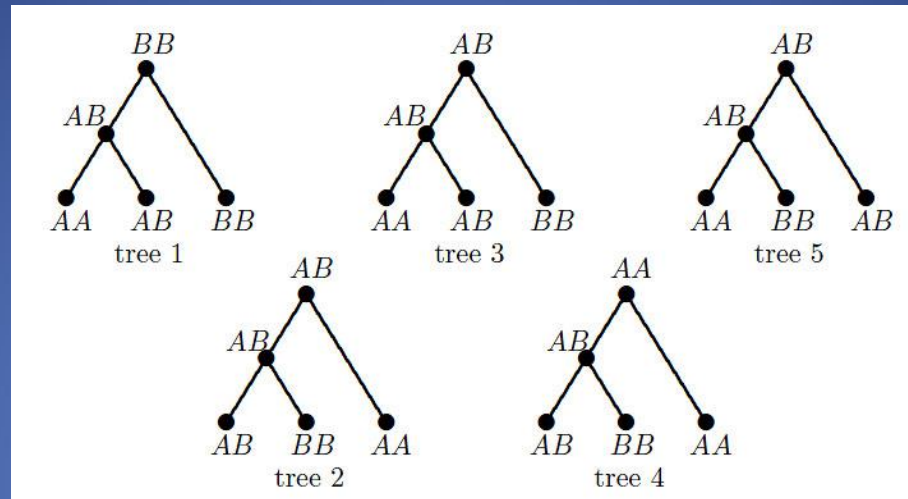
# Trees

- There are  $n = 5$  “most parsimonious” trees leading to these three sequences at the leaves of the trees,

*AA*  
*AB*  
*BB*



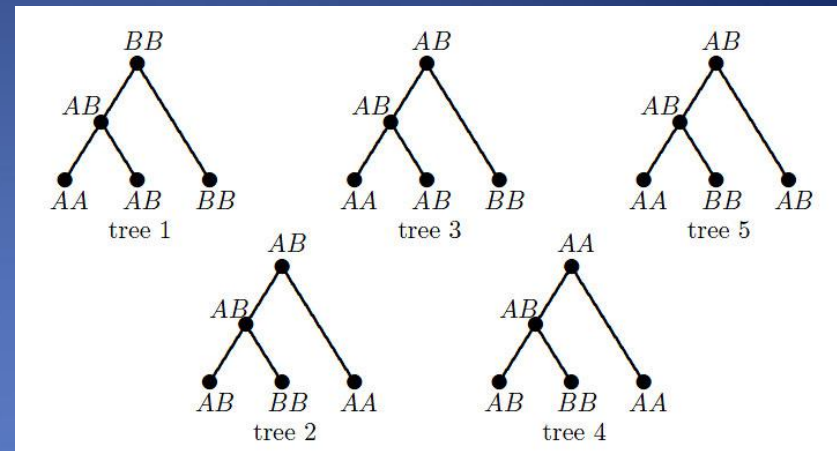
# Trees *en route* to PAM



- Among these trees *A* is aligned with, and substituted for, *B* (or conversely) twice in each tree, leading to a total A/B count of 10.
- Division by the number of trees (5) leads to a final contribution of 2 from this to the A/B count.

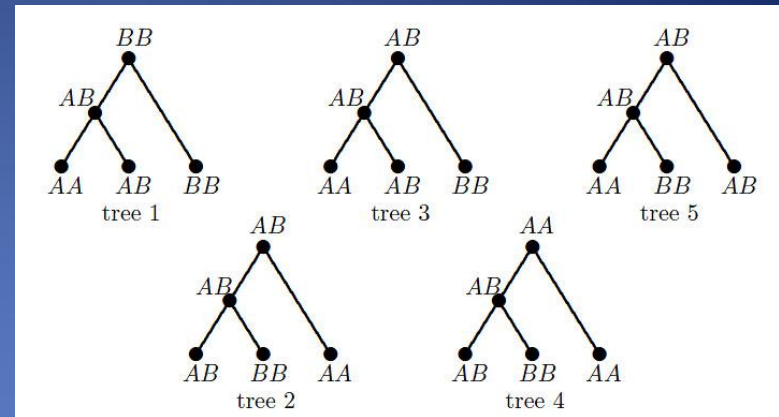


# PAM Example



- A is aligned with A two times in tree 1, three times in tree 2, three times in tree 3, four times in tree 4, and three times in tree 5.
  - Leading to a total of 15 A/A alignments.
- Each A/A alignment contributes 2 to the count, so that the total A/A count is 30.
- Division by the number of trees for this block leads to a final overall A/A count of 6.

# PAM Example



- Similar calculations show that the contribution to the B/B count from this block is also 6.
- The final matrix of counts is then

	A	B
A	6	
B	2	6

- In general, there will be more than one block in the data set. Each gives its own tree (or set of equally parsimonious trees).
  - If so, we simply add the counts from the different blocks to obtain an overall count matrix.

# Transition Matrix

- Suppose that the amino acids are numbered from 1 to 20 and denote the  $j,k$  entry in the overall count matrix by  $A_{jk}$ .
- The next task is to use this count matrix to construct an estimated Markov chain transition matrix.
  - *This is the tricky part; the next slide might take a few passes to get a handle on. Don't get frustrated.*
- For any  $j$  and  $k$  (not necessarily distinct), define  $a_{jk}$  by

$$a_{jk} = \frac{A_{jk}}{\sum_m A_{jm}}$$

# Transition Matrix

$$a_{jk} = \frac{A_{jk}}{\sum_m A_{jm}}$$

copied from previous slide  
for reference

- Fix  $j$
- For  $k \neq j$  let  $p_{jk} = c \cdot a_{jk}$ 
  - $c$  is a positive scaling constant (to be determined shortly)
- Let  $p_{jj} = 1 - \sum_{k \neq j} c a_{jk}$
- It follows from these definitions that  $\sum_k p_{jk} = 1$
- Now  $c$  can be chosen to be sufficiently small so that each  $p_{jj}$  is non-negative.
- The matrix of  $p_{ij}$ 's then has the properties of a Markov chain transition matrix.
- We can further calibrate with  $c$  so that the expected proportion of changes in one time step is 0.01

# Derivations

## - PAMn

- In general, the Markov transition matrix  $M$  is inferred from the data and calibrated so that there's approximately 1% change in one unit of time.
  - In Markov Chain parlance, the “weighted expected proportion” of amino acid changes is 0.01.
  - We will not have time to delve into the details of what that means, but intuitively it means there's a very small amount of change in one time unit.
- The substitution matrix thus defined is called PAM1.
- The matrix corresponding to  $M^n$  is called PAMn.

# Derivations

## - PAM

- This matrix that represents 1% change is said to correspond to an evolutionary distance of 1 PAM.
  - PAM stands for “Percent Accepted Mutation”
- Once we have the transition matrix  $M$ , we construct the matrix of scores as described earlier.
- The model with matrix  $M^2$  corresponds to two units of time, so 2 PAMs.
- In general, an evolutionary distance of  $n$  PAMs is given by the Markov model with transition matrix  $M^n$ .
- That is why PAM is a family of substitution matrices, indexed by the natural numbers.
  - In this case, *the larger the index, the more distant.*
    - Contrast with BLOSUM goes the opposite way.

# PAM250

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

# Choice of Matrix

- In theory, one should know the evolutionary distance between sequences to know which matrix to use.
  - But alignment of the sequences is the most immediate way to assess this distance.
    - And we need a substitution matrix to do that.
  - Also, we are not always aligning just two sequences.
    - We may align many together in a Multiple Sequence Alignment.
  - In this case there will be multiple evolutionary distances between pairs of sequences.
- All this adds up to the fact that we need to find good middle ground matrices that can be used as 'general purpose' across a range of evolutionary distances.
- And it also means we must evaluate and quantify exactly how bad things go if matrices are inferred at one distance and used for another.



# Choice of Matrix

---

- The BLOSUM62 is considered a good general-purpose matrix.
- According to Stephen Altschul
  - When using a local alignment method three matrices should ideally be used: PAM40, PAM120 and PAM250, the lower PAM matrices will tend to find short alignments of highly similar sequences, while higher PAM matrices will find longer, weaker local alignments.
  - When comparing sequences that were not known in advance to be related, for example when database scanning, a 120 PAM matrix was the best compromise.
- We'll discuss database scanning next.