MA3259 Lecture 3

Scoring Matrices
For Aligning Protein Sequences

LX Zhang
Department of Mathematics
National University of Singapore
matzlx@nus.edu.sg

Introduction

- A simple scoring scheme works well for comparing DNA sequences.
- For comparing protein sequences, the scoring matrices are usually complicated. They reflect the frequency of an amino acid replacing another and the frequency of each amino acid in homologues protein sequences.

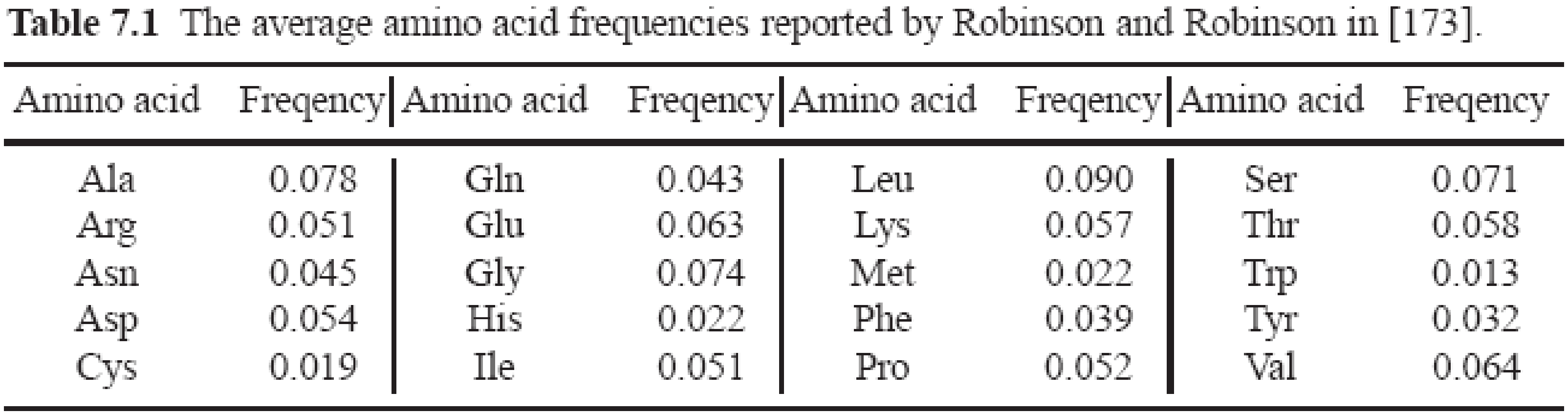
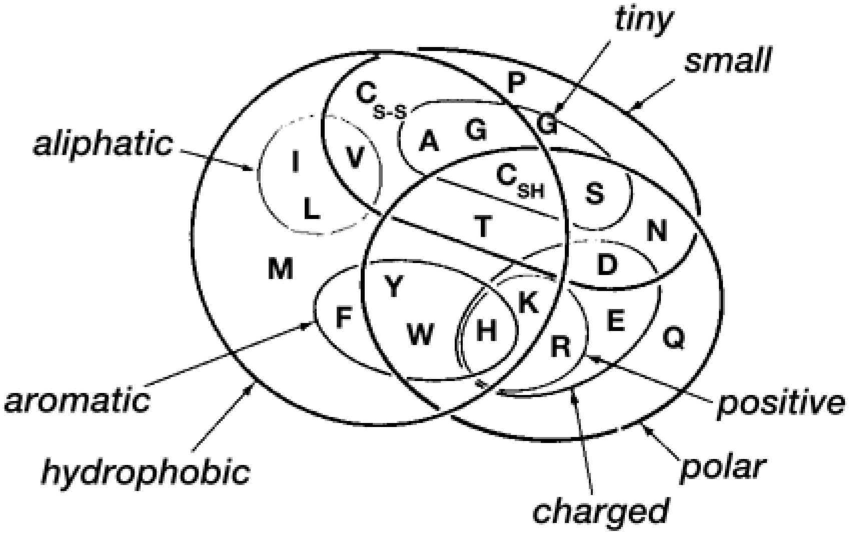


Table 7.1 The average amino acid frequencies reported by Robinson and Robinson in [173].

Amino acid	Freqency	Amino acid	Freqency	Amino acid	Freqency	Amino acid	Freqency
Ala	0.078	Gln	0.043	Leu	0.090	Ser	0.071
Arg	0.051	Glu	0.063	Lys	0.057	Thr	0.058
Asn	0.045	Gly	0.074	Met	0.022	Trp	0.013
Asp	0.054	His	0.022	Phe	0.039	Tyr	0.032
Cys	0.019	Ile	0.051	Pro	0.052	Val	0.064

Similarity scoring matrices might be constructed from any property of amino acids that can be quantified

- -partition coefficients between hydrophobic and hydrophilic phases
- -charge
- -molecular volume, etc.



Rationale for Scoring Alignment

- Consider an ungapped alignment A between two sequences of equal length:

$$S = s_1 s_2 \cdots s_k \quad T = t_1 t_2 \cdots t_k$$

We want to determine to see if A is due to their homology (hypothesis I: they are homologous) or A is by chance (hypothesis II: they have evolved independently).

A popular method for this kind of hypothesis test is to consider the log odds ratio

Let p_a denote the probability that the amino acid a appears in these two sequences.

The probability that A is seen under hypothesis II is

$$\Pr[\text{ A}|\text{ hypothesis II}) = \prod_i (p_{si}\, p_{ti})$$

Let $f_{a,b}$ be the probability of replacing a with b given that they are homologous.

Then,

$$\Pr[A \mid \text{S and T are homologous}] = \prod_i f_{si,ti}$$

The likelihood ratio of A under the two hypotheses is

$$\frac{\Pr[A \mid \text{S and T are homolougs}]}{\Pr[A \mid \text{hypothesis II}]} = \prod_i \frac{f_{s_i t_i}}{p_{s_i} p_{t_i}}$$

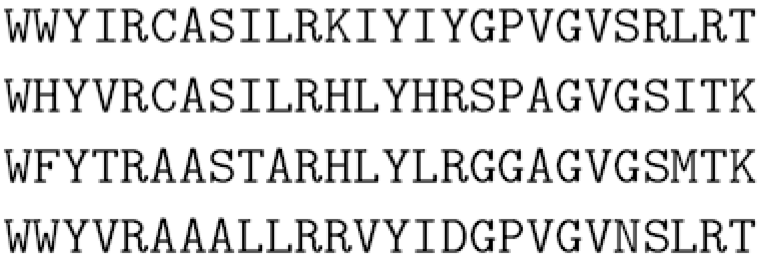
$$\log_2 \frac{\Pr[\text{ A} \mid \text{S and T are homolougs }]}{\Pr[\text{A} \mid \text{hypothesis II}]} = \sum_{i=1}^{n} \log_2 \left(\frac{f_{s_i t_i}}{p_{s_i} p_{t_i}} \right)$$

Column score of aligning s_i and t_i

2. BLOSUM matrices

- BLOSUM matrices were first proposed by Henikoff and Henikoff for aligning protein sequences.

- They were derived from conserved blocks in a database named BLOCKS. A conserved block is a ungapped multiple sequence alignment in which sequences match each other at some predefined level of similarity like



```
WWYIRCASILRKIYIYGPVGVSRLRT
WHYVRCASILRHLYHRSPAGVGSITK
WFYTRAASTARHLYLRGGAGVGSMTK
WWYVRAAALLRRVYIDGPVGVNSLRT
```

S. Henikoff and J. Henikoff. Amino Acid Substitution Matrices from Protein Blocks, *Proc. of Nat. Aca. Sci. USA*, 89(1992), 10915-10919.

Computing BLOSUM-x matrix

- Eliminate sequences which are identical in more than $x\%$ (e.g. 62%) positions. This can be done in the following two ways.

  — Remove all but one sequence from the block,

  — Find a cluster of similar sequences and weight each seuqnece so that the whole cluster of sequences looks like a sequence.

- Compute

  (a) the observed frequency f_{ij} of the aligned pair A_i to A_j. f_{ij} is an estimate of the probability that we expect to see A_i and A_j aligned in the given alignments.

  (b) the observed frequency f_i of A_i appearing in the block.

- The (i, j)-entry of BLOSUM-x matrix is the closest integer of $\frac{1}{0.347}s_{ij} \approx 2s_{ij}$, where

$$s_{ij} = \log_2 \frac{f_{ij}}{p_{ij}}, \quad p_{ij} = \begin{cases} f_i f_i & i = j \\ 2f_i f_j & i \neq j. \end{cases}$$

Example of Deriving A Scoring Matrix

There are three letters and only one block

B	A	B	A
A	A	C	A
A	A	C	C
A	A	B	A
A	A	B	C
A	A	B	B

Step 2: Compute the frequency of each aligned pairs and each letter.

aligned pairs	*observed frequencies* f_{ij}
A to A	28/60
A to B	8 /60
A to C	6/60
B to B	6/60
B to C	10/60
C to C	2/60

amino acids	*observed frequencies* f_i
A	14/24
B	6/24
C	4/24

Step 3: Compute s_{ij} and round $2s_{ij}$ to the closest integers:

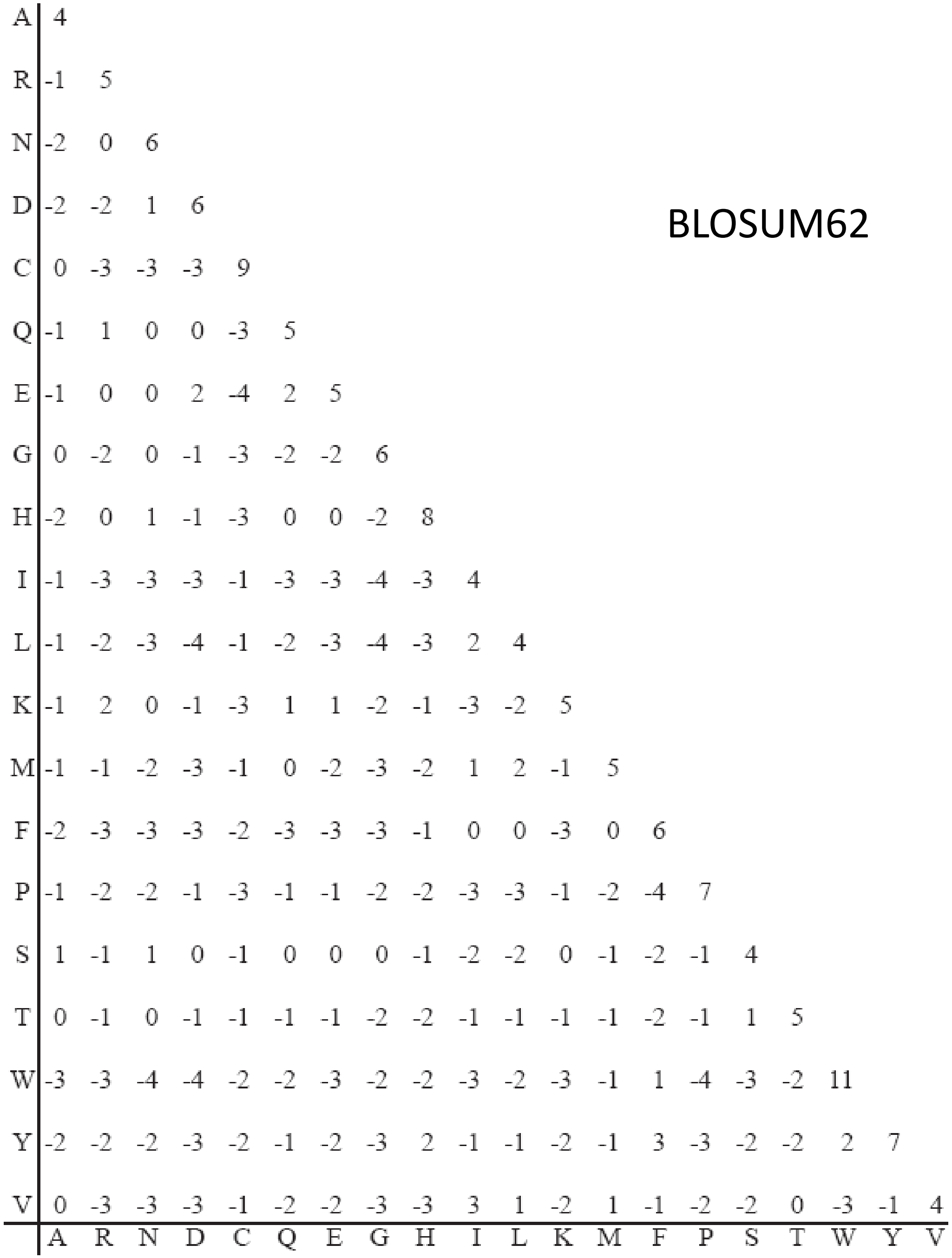
$$s_{ij} = \log_2 \frac{f_{ij}}{p_{ij}}, \quad p_{ij} = \begin{cases} f_i f_i & i = j \\ 2 f_i f_j & i \neq j. \end{cases}$$

aligned pairs	*observed frequencies* f_{ij}	*expected frequencies* p_{ij}	$2s_{ij}$
A to A	28/60	196/576	0.91
A to B	8/60	168/576	-2.26
A to C	6/60	112/576	-1.92
B to B	6/60	36/576	1.35
B to C	10/60	48/576	2.00
C to C	2/60	16/576	0.52

amino acids	*observed frequencies* f_i
A	14/24
B	6/24
C	4/24

	A	B	C
A	1	-2	-2
B		1	2
C			1

BLOSUM62



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

Remark 1

- f_{ij} is the observed frequency of amino acid A_i being aligned with amino acid A_j. p_{ij} is the probability that A_i and A_j align by chance between two random sequence in which A_i and A_j have frequency f_i and f_j.

  Hence, s_{ij} is an estimate of the log odds ratio of the aligned pair A_i and A_j.

  The score for a pair of amino acids in a BLOSUM matrix is positive if they are more likely than chance in alignment and negative if they are less likely.

Remark 2

- BLOSUM matrices are obtained for aligning protein sequence,

but they are obtained from conserved blocks, which are alignments! Then, how blocks are obtained? Does the block data bias the output scoring matrix

To break this circular argument. A three step iterative approach is used.

1. A unitary matrix which has 1 on diagonal and 0 elsewhere was first used, generating 2205 blocks;

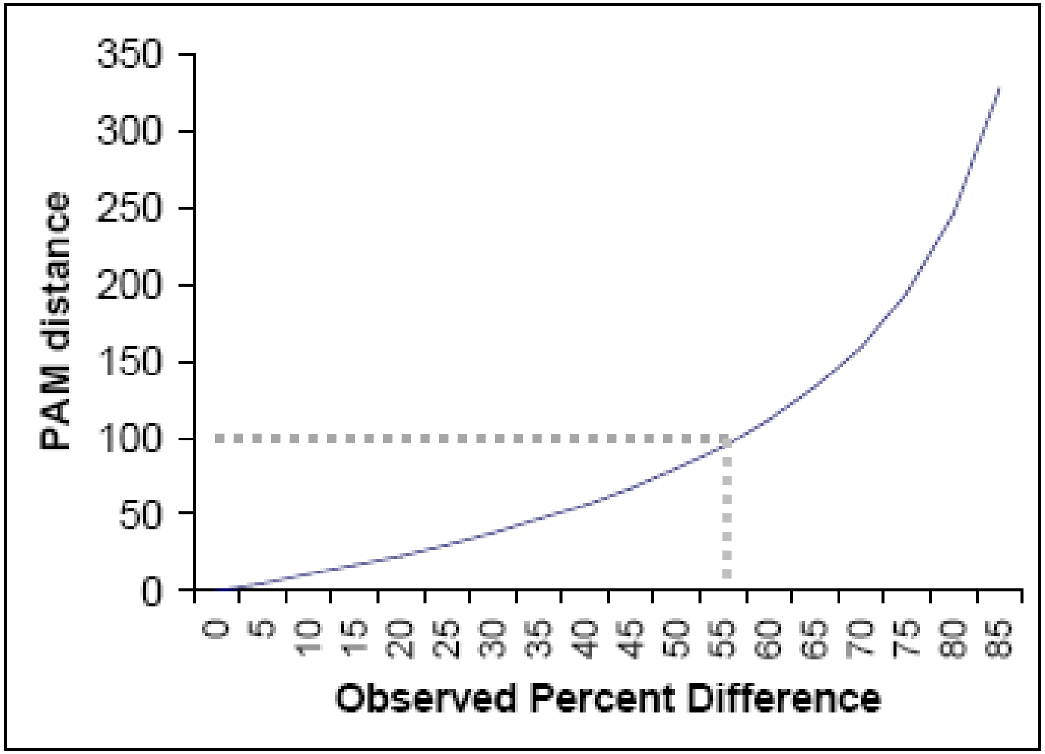
2. A scoring matrix was then obtained from these blocks by setting 60% as the similarity level, which was used to construct the 2nd set of 1961 blocks. Then, the second scoring matrix was reconstructed from these 1961 blocks.

3. Finally, the second scoring matrix was used to obtain the final version of the dataset of 2106 blocks. From this final dataset, various BLOSUM matrices were obtained.

2. PAM Matrices

- The PAM matrices are the first scoring matrices. They were constructed by Dayoff and coworkers in 1979.
- A point accepted mutation in a position is a substitution of one amino acid by another that is accepted by natural selection in the sense that the resulting protein has the same function as the previous one.
- A PAM unit is a time period over which 1% of the amino acids in a sequence are expected to undergo accepted mutations, some of which may occur at the same position.

That two sequences are 100 PAM unit diverged does not mean they are different in every position.

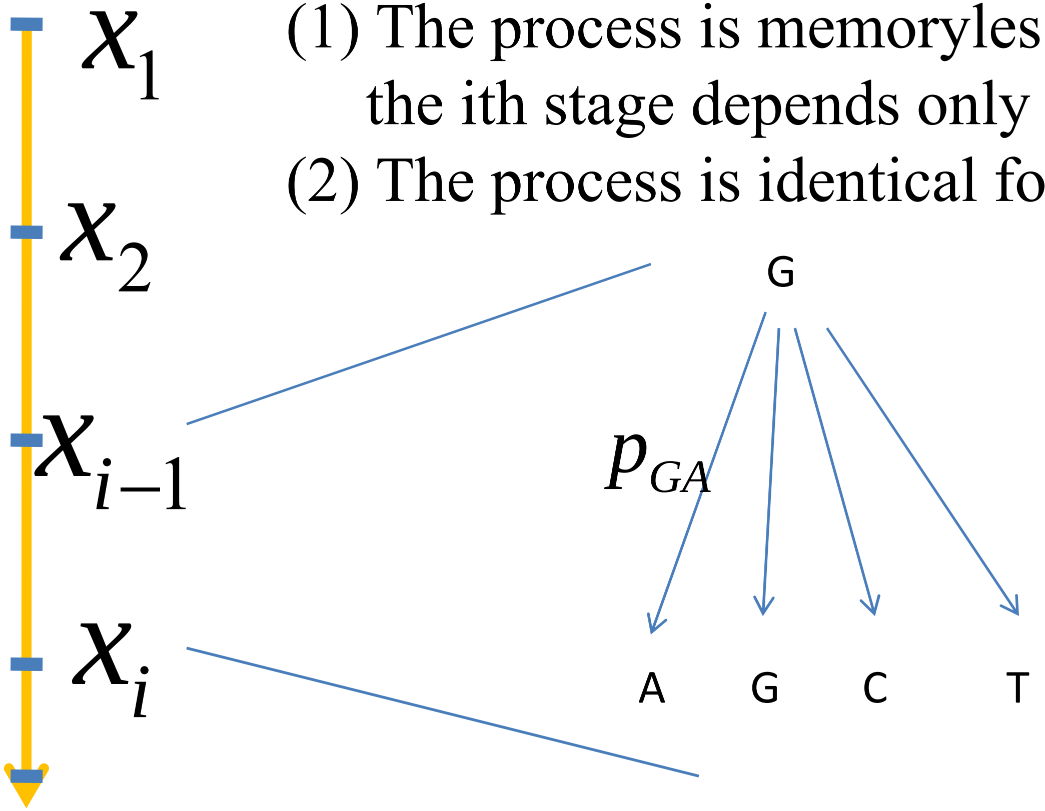


The PAM matrices are dervied based on Markovian evolutionary model:

- Different letters in a sequence have evolved independently;
- Substitution process at each site is modeled as a Markov chain process with the letters as its states;

(1) The process is memoryless. The state x_i after the ith stage depends only on x_i-1, not on x_j, j<i-1.

(2) The process is identical for each stage



$$\begin{bmatrix} p_{AA} & p_{AG} & p_{AC} & p_{AT} \\ p_{GA} & p_{GG} & p_{GC} & p_{GT} \\ p_{CA} & p_{CG} & p_{CC} & p_{CT} \\ p_{TA} & p_{TG} & p_{TC} & p_{TT} \end{bmatrix}$$

Transition matrix

Compute PAM1 Matrix

- Collected 71 blocks of aligned proteins sequences that are approximately 1-PAM unit diverged.

- Count substitutions occurring in the dataset.
  Use A_{ij} to denote the times Ai is substituted by A_j;
  and estimate the frequency f_i of amino acid $A_{i.}$

- PAM1 matrix M1 = (p_{ij}), where

$$p_{ij} = c\frac{A_{ij}}{\sum_k A_{ik}}, \qquad p_{ii} = 1 - \sum_{j \neq i} p_{ij}.$$

$$c = \frac{0.01}{\sum_i \sum_{j \neq i} f_i \left(A_{ij} / \sum_k A_{ik}\right)}.$$

Compute PAM-n matrices

We assume that evolution is a Markovian process.
Hence, M1=(pij) is the transition matrix of the process over one PAM unit period;

Therefore, $M_1{}^n = \left(m_{ij}^{(n)}\right)$ is the transition matrix of the process over *n* PAM unit period.

For any i, j, the probability that A_i mutates into A_j in n PAM units is $f_i m_{ij}^{(n)}$

The (i, j)-entry in the PAM-n matrix is defined as

$$\boxed{10\log\frac{f_i m_{ij}^{(n)}}{f_i f_j} = 10\log\frac{m_{ij}^{(n)}}{f_j}}$$

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

PAM 250

Hydrophobic group: M, I, L, V

Aromatic group: F, Y, W.

PAM120

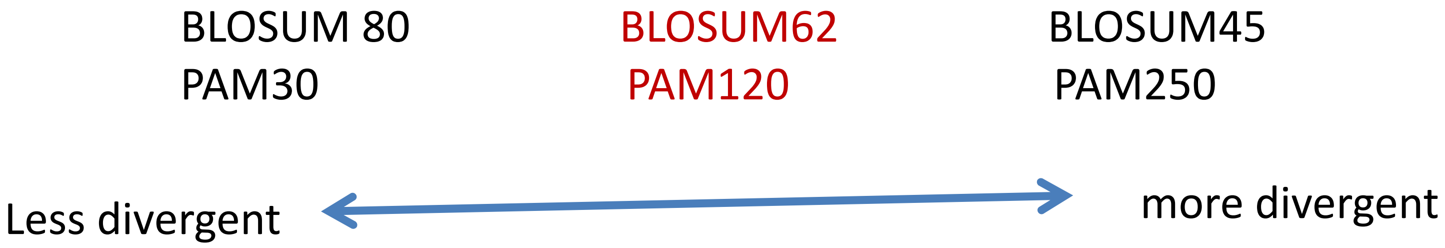
A	3																			
R	-3	6																		
N	-1	-1	4																	
D	0	-3	2	5																
C	-3	-4	-5	-7	9															
Q	-1	1	0	1	-7	6														
E	0	-3	1	3	-7	2	5													
G	1	-4	0	0	-4	-3	-1	5												
H	-3	1	2	0	-4	4	-1	-4	7											
I	-1	-2	-2	-3	-3	-3	-3	-4	-4	6							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

3. How to Select a Scoring Matrix?

- Different scoring matrices are only optimal for detecting different classes of alignments.
- One needs to consider the similarity and amino acid composition of compared sequences.

For general alignment purpose, BLOSUM62 or PAM120 are recommended.

For PAMx matrices, higher x detects more divergent sequences;
For BLOSUMn matrices, lower n detects more divergent sequences.



BLOSUM 80
PAM30

BLOSUM62
PAM120

BLOSUM45
PAM250

Less divergent

more divergent

- For database search, several complementary scoring matrices are suggested to be used, for example, a combination of PAM40, PAM120 and PAM250
- Protein sequence comparison contains more information.

4. Gap penalty models

Indel columns are introduced to bring up the matches that appear later. Thus, indels must be penalized like mismatches.

- Linear gap penalty model

Each indel column is penalized by constant δ. This is what we have used in this section.

In this model, the score of an alignment is equal to the sum of scores of match/mismatch columns and δ times the number of indel columns.

- Affine gap penalty model

A gap is defined to be a sequence of spaces '-' between two consecutive letters in a row of an alignment.

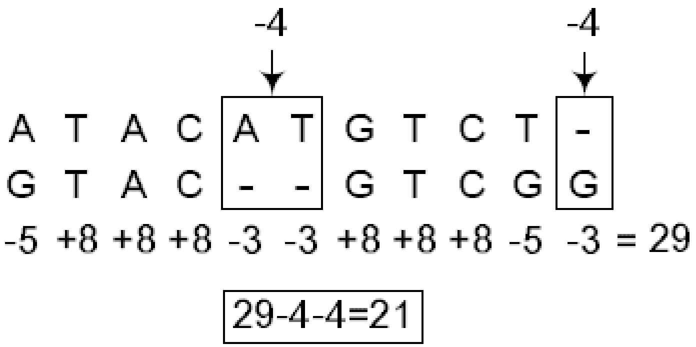
Two or more nucleotides are often removed or added in a mutational events. Hence, a gap of length k scores

- (o + k x e)

where o>0 is called the gap opening penalty and e>0 the gap extension penalty. Usually, o is large and e is small, say o=12, e=1.

Theorem: Alignment can be done in quadratic time even in affine gap penalty.

o=4, e=3



```
                  -4                      -4
A  T  A  C  A  T  G  T  C  T  -
G  T  A  C  -  -  G  T  C  G  G
-5 +8 +8 +8 -3 -3 +8 +8 +8 -5 -3 = 29
```

29-4-4=21