第三章 序列分析与联配

序列分析是生物信息学最主要的研究内容之一，它可以分为两个主要部分：一是序列组成（特别是涉及到基因组层次上）分析，二是序列之间的比较分析。两条序列或多条序列间的比对或联配(alignment)的目的，是对它们的序列相似性进行评估，找出这些序列中结构或功能相似性区域等。通过联配未知序列与已知序列(其功能或结构等已知)的相似程度，我们可以判断或推测未知序列的结构与功能。

第一节 序列组成及单一序列分析[1]

一．碱基组成

DNA 序列一个显而易见的特征是四种碱基类型的分布。尽管四种碱基的频率相等时对数学模型的建立可能是方便的，但几乎所有的研究都证明碱基是以不同频率分布的。表 3.1 包含了 9 条完整 DNA 分子序列的资料，表 3.2 的数据来自两个胎儿球蛋白基因(Gr 和 Ar)，每个基因具有三个外显子和两个内含子(shen 等 1981)。这两个例子说明序列内和序列间碱基具有不同的频率。在基因每一侧的 500 个任意碱基区域被称为“侧翼”，基因间区域是指两个基因间的其余序列。

表 3.1 九条完整 DNA 序列的碱基组成

序 列	名 称*	碱基频率				总 计
		A	C	G	T	
噬菌体						
λ	LAMCG	0.25	0.24	0.25	0.26	48502
T_7	PT7	0.27	0.23	0.24	0.26	39936
ØX174	PX1CG	0.24	0.22	0.31	0.23	5386
病毒						
花椰菜镶病毒	MCACGDH	0.37	0.21	0.23	0.19	8016
人类乳头多瘤空泡病毒 BK	PVBMM	0.30	0.20	0.30	0.20	4936
肝炎 B	HPBAYW	0.28	0.22	0.23	0.27	3182
线粒体						
人类	HUMMT	0.31	0.31	0.25	0.13	16569
牛	BOVMT	0.33	0.26	0.27	0.14	16338
鼠	MUSMT	0.35	0.24	0.29	0.12	16295

*取自 GenBank 数据库

[1]部分内容取自Weir B．S．(徐云碧等译)．遗传学数据分析—群体遗传学离散型数据分析方法，北京：中国农业出版社，1996

表 3.2 人类胎儿球蛋白基因不同区段的碱基组成

区 段	长 度	A	C	G	T
5′例翼(2)	1000	0.33	0.23	0.22	0.22
3′例翼(2)	1000	0.29	0.15	0.26	0.30
内含子(4)	1996	0.27	0.17	0.27	0.29
外显子(6)	882	0.24	0.25	0.28	0.22
基因间(1)	2487	0.32	0.19	0.18	0.31

*数据来自 EMBL 数据库 HSGLBN 基因

二．碱基相邻频率

分析 DNA 序列的主要困难之一是碱基相邻的频率不是独立的。碱基相邻的频率一般不等于单个碱基频率的乘积：如果 P_u 是序列中碱基 u 的频率，且 P_{uv} 为两个相邻碱基 u 和 v 的频率，则

$$p_{uv} \neq p_u p_v$$

Nussinov(1984)研究了两碱基相邻的频率(表 3.3)。数据来自 166 个脊椎动物的 DNA 序列，总长 136731 个碱基。表中的比值为 16 种二个碱基相邻的频率除以相应的单个碱基频率的乘积。

表 3.3 脊椎动物中两碱基的相邻频率

相邻碱基对	观测频率/期望频率*
TG	1.29
CT	1.26
CC	1.18
AG	1.16
AA	1.15
CA	1.15
GG	1.14
TT	1.07
GA	10.4
TC	1.00
GC	0.99
AT	0.85
AC	0.84
GT	0.82
TA	0.65
CG	0.42

*期望频率为相应两个单个碱基频率的乘积

作为一个特别的例子，图 3.1 给出了鸡血红蛋白 β 链的 mRNA 编码区的 438 个碱基。表 3.4 列出了 4 种碱基和 16 种两碱基的数目。将该表看作 4×4 的表，

计算行列独立性的卡方统计量，得到 $\chi^2 = 59.3(\chi^2_{0.05,9} = 16.92)$，表明行(第一碱基)列(第二碱基)之间存在明显的关联。

```
GTGCACTGGA CTGCTGAGGA GAAGCAGCTC ATCACCGGCC TCTGGGCAA  GGTCAATGTG 60
GCCGAATGTG GGGCCGAAGC CCTGGCCAGG CTGCTGATCG TCTACCCCTG GACCCAGAGG 120
TTCTTTGCGT CCTTTGGGAA CCTCTCCAGC CCCACTGCCA TCCTTGGCAA CCCCATGGTC 180
CGCGCCCACG GCAAGAAAGT GCTCACCTCC TTTGGGGATG CTGTGAAGAA CCTGGACAAC 240
ATCAAGAACA CCTTCTCCCA ACTGTCCGAA CTGCATTGTG ACAAGCTGCA TGTGGACCCC 300
GAGAACTTCA GGCTCCTGGG TGACATCCTC ATCATTGTCC TGGCCGCCCA CTTCAGCAAG 360
GACTTCACTC CTGAATGCCA GGCTGCCTGG CAGAAGCTGG TCCGCGTGGT GGCCCATGCC 420
CTGGCTCGCA AGTACCAC
```

图 3.1 鸡 β 球蛋白基因编码区的 DNA 序列
(GenBank：CHKHBBM，记录号 J00860)

表 3.4 图 3.1 鸡 β 球蛋白基因序列的相邻碱基分布

		第二碱基				
		A	C	G	T	总计
第一碱基	A	23	26	23	15	87
	C	37	51	14	41	143
	G	25	38	36	19	118
	T	2	29	41	14	89
总计		87	144	117	89	437

在编码区，存在某种约束来限制 DNA 序列编码氨基酸。在密码子水平上，这一约束与碱基相邻频率有关。表 3.5 列出了遗传密码和图 3.1 序列中各密码子数量。尽管数目很小，难以作出有力的统计结论，但编码同一氨基酸的不同密码子(同义密码子)好像不是等同存在的。这种密码子偏倚必定与两碱基相邻频率水平有关。表 3.5 还清楚地表明，由于密码子第 3 位置上碱基的改变常常不会改变氨基酸的类型，因而对第 3 位置上碱基的约束要比第 2 位碱基小得多。

表 3.5 64 种可能的碱基三联体密码子及相应的氨基酸数（据图 3.1 序列）

UUU Phe 3	UCU Ser 0	UAU Tyr 0	UGU Cys 2
UUC Phe 5	UCC Ser 5	UAC Tyr 2	UGC Cys 1
UUA Leu 0	UCA Ser 0	UAA Stop 0	UGA Stop 0
UUG Leu 0	UCG Ser 0	UAG Stop 0	UGG Trp 4
CUU Leu 1	CCU Pro 1	CAU His 3	CGU Arg 0
CUC Leu 6	CCC Pro 4	CAC His 4	CGC Arg 3
CUA Leu 0	CCA Pro 0	CAA Gln 1	CGA Arg 0
CUG Leu 11	CCG Pro 0	CAG Gln 0	CGG Arg 0
AUU Ile 1	ACU Thr 3	AAU Asn 1	AGU Sre 0
AUC Ile 6	ACC Thr 4	AAC Asn 6	AGC Ser 2
AUA Ile 0	ACA Thr 0	AAA Lys 1	AGA Arg 0
AUG Met 1	ACG Thr 0	AAG Lys 9	AGG Arg 3
GUU Val 0	GCU Ala 4	GAU Asp 1	GGU Gly 1
GUC Val 5	GCC Ala 11	GAC Asp 5	GGC Gly 4
GUA Val 0	GCA Ala 0	GAA Glu 4	GGA Gly o
GUG Val7	GCG Ala 1	GAG Glu 3	GGG Gly 3

相邻碱基之间的关联将导致更远碱基之间的关联，这些关联延伸距离的估计可以从马尔科夫链(Markov chain)理论得到(Javare 和 Giddings，1989)。在不援引任何生物学机制的情况下，第 k 阶马尔科夫链假定在序列中某一位置上碱基的存在只取决于前面 k 个位置上的碱基。一阶链假定一个特定碱基存在于位置 i 的概率只取取决于在位置 i-1 的 4 种碱基概率。相互独立的碱基所组成的序列将与 0 阶马尔科夫链相对应。阶可以通过似然法估计。同时，马尔科夫链分析更适应于基因组水平，而非单一序列(基因)。相关内容可参见第四章第 2 节。

三．同向重复序列分析

除了分析整个序列碱基关联程度的特征外，我们常对寻找同向重复序列(direct repeats)之类的问题感兴趣。Karlin等(1983)给出了完成这一分析的有效算法。该法采用由特定的几组碱基字母组成的不同亚序列或称为字码(word)。只需要对整个序列搜索一次。给一碱基赋以值 α，例如A、C、G、T的值为 0、1、2、3。由X_1、X_2、…、X_k共k个字母组成的每一种不同的字码按$1+\sum_{i=1}^{k}\alpha_i 4^{k-i}$ 计算字码值。这些值的取值范围为 1 到 4^k。例如，5 字码TGACC的值为 $1+3\times4^4+2\times4^3+0\times4^2+1\times4^1+1\times4^0=459$。可先从低k值的字码开始搜索。记录序列中每一个位置k字码的字码值。只有在发现k字码长度重复的那些位置考虑进行长度大于k的字码搜索。

表 3.6 列出了序列 TGGAAATAAAACGTAAGTAG 中所有碱基 2 字码(k=2)的初始位置和字码值。对于完全重复、长度大于 2 的同向重复或亚序列的搜索可只限于 2 字码重复的初始位置。在本例中只有 4 个重复的 2 碱基重复序列。例如，在位置 4、5、8、9、10 和 15 均发现了字码值为 1 的碱基重复序列。从有重复的第 2 个碱基为起点的 3 字码值及位置列于表 3.7，其中发现字码值为 1、45 和 49 的序列有重复。以每一重复的 3 碱基为起点的 4 字码搜索未能发现更长的重复序列。

因此最长的同向重复为 4、8、9 位置上的 AAA，13、17 位置上的 GTA 以及 7、14 位置上的 TAA。同样对图 3.1 鸡 β 球蛋白 DNA 序列进行同向重复序列搜索，一些最长同向重复序列列于表 3.8。

表 3.6 序列 TGGAAATAAAACGTAAGTAAGTAG 的 2 字码值和位置(Karlin，1983)

字码值	碱基位置	字码值	碱基位置
1	4,5,8,9,10,15	9	3
2	11	10	-
3	16,19	11	2
4	6	12	13,17
5	-	13	7,14,18
6	-	14	-
7	12	15	1
8	-	16	1

表 3.7 序列 TGGAAATAAAACGTAAGTAG 的 3 字码值和位置(Karlin，1983)

字码值	碱基位置
1	4,8,9
2	10
3	15
4	5
45	13,17
49	7,14
51	18

表 3.8 鸡 β 球蛋白 DNA 序列中(图 3.1)长度为 8 或 8 以上的碱基重复序列

长度	重复序列	起始位置
8	GCCCTGGC	79，418
	GCCAGGCT	85，377
	CCAGGCTG	86，378
	CAGGCTGC	87，379
	TCCTTTGG	130，208
	CCTTTGGG	131，209
	TGGTCCGC	176，398
	GGTCCGCG	177，399
9	GCCAGGCTG	85，377
	CCAGGCTGC	86，378
	TCCTTTGGG	130，208
	TGGTCCGCG	176，398
10	GCCAGGCTGC	85，377

Karlin等(1983)提出了序列内存在的最长同向重复序列的统计显著性评价

方法。在核苷酸的位置为独立的假定下(相当于阶次为 0 的马尔科夫链)，长度为n的序列中，最长同向重复*Ln*的期望长度和方差为：

$$\mu_L = \frac{0.6359 + 2\ln n + \ln(1-p)}{\ln(1/p)} - 1$$

$$\sigma_L^2 = \frac{1.645}{(\ln P)^2} \qquad (3.1)$$

其中，*P*为序列中碱基频率的平方和：

$$P = \sum_{i=1}^{4} P_i^2$$

用尽可能接近最大长度的期望均值的字码(即R→μ_L) 来开始同向重复序列的搜索计算可能节省计算量。

可以用一个近似方法来验证以上统计假说。假定同向重复序列的长度呈正态分布。对于图 3.1 鸡β蛋白序列，A、C、G、T 四个碱基的次数分别为 87、144、118 和 89，因而 P=0.2614，最长重复序列的期望长度为 8.13 且具有期望方差 0.9138。根据 95%的正态分布概率，理论上可以预期最长同向重复序列不超过 10。

四. DNA 序列的几何学分析—Z 曲线

DNA 序列实际上是一种用 4 种字母表达的“语言”，只是其“词法”和“语法”规则目前还没有搞清楚。人类的语言有文字、声音两种基本表现形式，此外还有手语、旗语甚至图画语等特殊表达形式。同样，DNA 序列作为一种语言，其表达形式也不是唯一的。传统上，DNA 序列是用 4 种字母符号表达的一维序列。这是一种抽象形式，适合于存储、印刷和代数算法的处理，包括比较、排列和查找特殊序列等。我国学者张春霆等开展了 DNA 序列三维空间曲线表示形式，即 DNA 序列几何表示形式的研究。几何形式虽然与符号形式完全等价，但显示了 DNA 序列的新特征。两种形式各有其特点，相互补充。这一新方法，为解读 DNA 序列信息提供了崭新的手段。

他们的研究始于对 4 种碱基对称性的观察，提出了用正面体表示碱基对称性。1994 年，他们利用这种形式来表示任意长度的 DNA 序列。现将这种序列表示方法简述如下。

考察一个长为L的单股DNA序列，方向(5'→3'或 3'→5')不限。从第一个碱基开始，依次考察此序列，每次只考察一个碱基。当考察到第n个碱基时(n=1,2，…,L)，数一下从 1 到n这个子序列中四种碱基各自出现的次数。设 4 种碱基A、C、G、T出现的次数分别以A_n、C_n、G_n、T_n表示之，这里下标“n”是表明这些整数是从 1 到n这个子序列中数出来的，如图 3.2 所示。显然，它们都是正整数。根据正四面体的对称性可以证明，在正面体内存在唯一的一个点P_n与这四个正整数对应。点P_n构成了四个正整数的一一对应映射。点P_n坐标可用四正整数表达：

$$\begin{aligned} x_n &= 2(A_n+G_n)-n, \\ y_n &= 2(A_n+G_n)-n, \\ z_n &= 2(A_n+T_n)-n, \end{aligned} \qquad (3.2)$$

$x_n, y_n, z_n \in [-n, n]$, $n=1, 2, \cdots, L$,

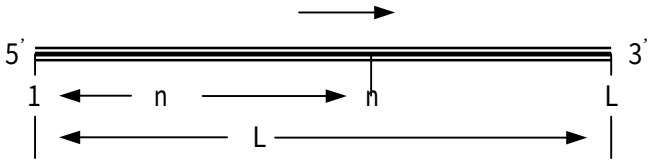


图 3.2 DNA 序列示意图

其中x_n，y_n和z_n为点P_n的三个坐标分量。当n从 1 跑到L时，我们依次得到P_1，P_2，P_3，…，P_L共L个点。将相邻两点用适当的曲线连接所得到的整条曲线，就称为表示DNA序列的Z曲线。可以证明，Z曲线与所表示的DNA序列是一一对应的，即给定一DNA序列，存在唯一的一条Z曲线与之对应；反之，给定一条Z曲线，可找到唯一的一个DNA序列与之对应。换言之，Z曲线包含了DNA序列的全部信息。Z曲线是与符号DNA序列等价的另一种表示形式，一种几何形式。可以通过Z曲线对DNA序列进行研究。

Z曲线的三个分量(方程 3.2)具有明确的生物学意义：x_n表示嘌呤/嘧啶碱基沿序列的分布。当从 1 到n的这个子序列中(图 3.2)嘌呤碱基多于嘧啶碱基时，$x_n>0$，否则，$x_n<0$，当两者相等时$x_n=0$。同样，y_n表示氨基/酮基碱基沿序列的分布。当在子序列中氨基碱基多于酮基碱基时，$y_n>0$，否则，$y_n<0$，当两者相等时$y_n=0$。z_n表示强/弱氢键碱基沿序列的分布。当弱氢键碱基多于强氢键碱基时，$z_n>0$，否则$z_n<0$，当两者相等时，$z_n=0$。这三种分布是相互独立的，表现在以下事实上：任何一种分布不能由其它两种分布的线性叠加表示出来。给定的DNA序列唯一地决定了这三种分布；三种分布唯一地描述了DNA序列。对DNA序列的研究就是通过对这三种分布的研究来进行。从方法学的角度来看，这是DNA序列的一种几何学研究途径。

图 3.3 给出了大肠杆菌 ayoP 基因族序列 Z 曲线的三个分量，即三种分布图。该基因族包含了大肠杆菌 5 个基因 aroP,A,aceFE,aceF 和 lpd,总长度为 9501bp，分别编码芳香族氨基酸运输蛋白 aroP，蛋白质 A(功能不详)和三种酶，即丙酮酸脱氢酶，二氢硫辛酰基转移酶和二氢硫辛酰脱氢酶。它们位于此序列的 0039-1406，1947-2654，2870-5527，5545-7434，7759-9183 区间。在图中 X 轴的下方的基因排列图上已分别用阴影标出相应基因。在这些基因之间有三个启动子区（*pm1*、*pm2* 和 *pm3*），其中 aceE 和 aceF 基因属于 ace 操纵子，共用一个启动子。三个启动子区亦在图中标出。非常令人感兴趣的是，在 5 个编码区，Z 曲线的 z 分量基本上都是单调下降的，而在三个启动子区基本上都是单调上升的。x，y 分量亦有变化，但不如 z 分量明显。在上升、下降的交界处，Z 曲线均发生了重大的转折，据此有可能用 Z 曲线识别这些位置。由此图可见，用 Z 曲线这种几何方法显示 DNA 序列不仅直观，而且作为一种识别序列中的不同基因和功能区的新方法，展现了广阔的应用前景。

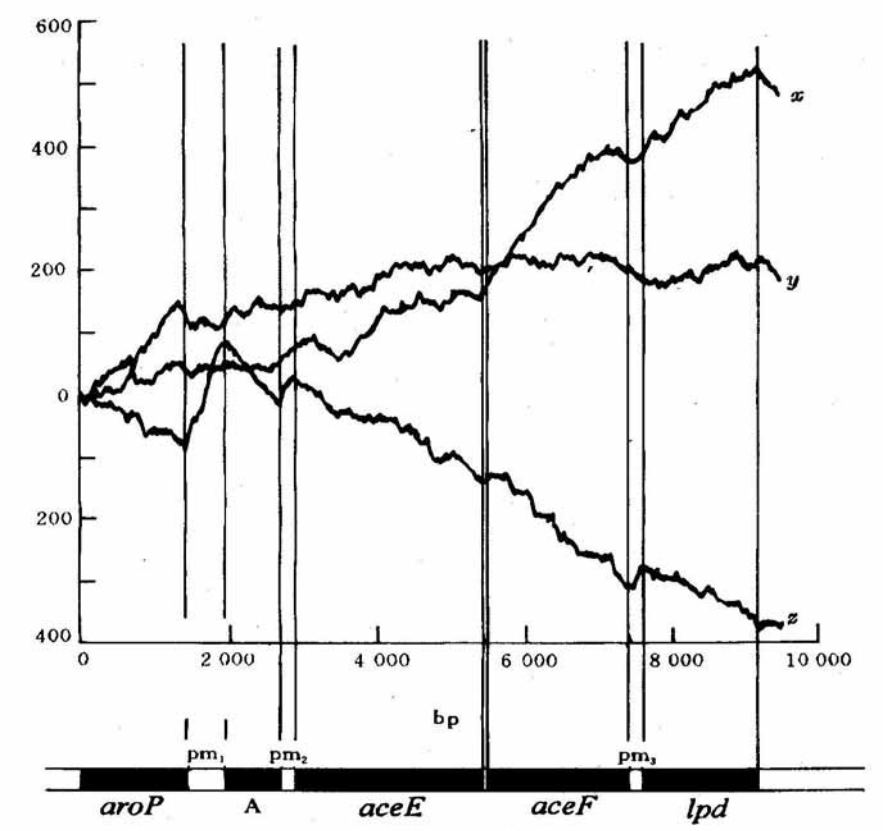


图 3.3 大肠杆菌 ayoP 基因族序列 Z 曲线的三个分量（三种分布图）

第二节 序列联配[2]

一. Needleman-Wunsch 算法

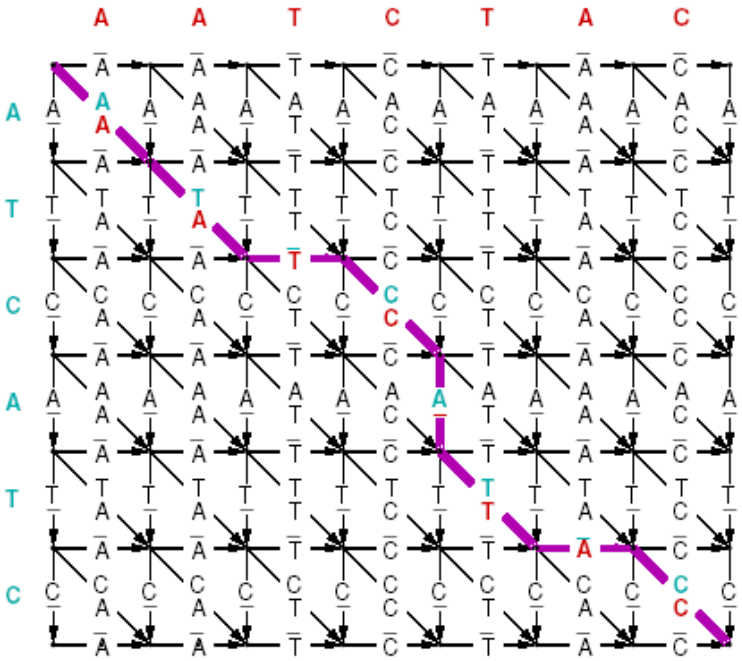
有 2 种经典方法可以计算两条序列间的最适联配。Needleman-Wunsch 算法是一种整体联配(global alignment)算法，最佳联配中包括了全部的最短匹配序列。Smith-Wateman 算法是在 Needleman-Wunsch 算法基础上发展而来的，它是一种局部联配(Local alignment)算法。这二种算法均可以用于核酸和蛋白质序列。在给定空位罚值和替换矩阵情况下，它们总是能给出具有最高（优）联配值的联配。但是，这个联配并不需要达到生物学意义上的显著水平。GCG 软件包中，BESFIT 和 GAP 程序，EMBOSS 的 needle 等可用于该联配。一些网站可以通过递交序列进行两条序列的联配分析。

从整体上分析两个序列的关系，即考虑序列总长的整体比较，用类似于使整体相似 (global similarity)最大化的方式，对序列进行联配。两个不等长度序列的联配分析必需考虑在一个序列中圈掉一些碱基或在另一序列作空位(gap)处理。Needleman 和 Wunsch(1970)的法则为这些步骤提供了实例。这一算法是为氨基酸序列发展的，但也可以用于核苷酸序列。算法最初寻求的是使两条序列间

[2]部分内容取自Weir B. S. Genetic Data Analysis Ⅱ—Methods for Discrete Population Genetic Data, Sunderland：Sinauer Associates Inc. Publishes, 1996

的距离最小。尽管这类距离的元素是以一种特定的方式定义的，但该算法的良好特性在于它确定了最短距离。这是一个动态规划(dynamic programming)的方法。

将两条联配的序列沿双向表的轴放置，两条序列的所有可能的联配方式都将在它们所形成的方形图中（见下图）。从任一碱基对，即表中的任一单元开始，联配可延三种可能的方式延伸：如果碱基不匹配，则每一序列加上一个碱基，并给其增加一个规定的距离权重；或在一个序列中增加一个碱基而在另一序列中增加一个空位或反之亦然。引入一个空位时也将增加一个规定的距离权重。因此，表中的一个单元可以从(至多)三个相邻的单元达到。我们把达左上角单元距离最小的方向看作相似序列延伸的方向。等距离时意味着存在两种可能的方向。将这些方向记录下来，并在研究了所有的单元之后，沿着记录的方向就有一条路径可从右下角(两个序列的末端)追踪到左上角（两个序列的起点)。由此所产生的路径将给出具有最短距离的序列联配。



Alignment corresponding to the colored path:

A T – C A T – C
A A T C – T A C

以两个短序列 CTGTATC 和 CTATAATCCC 为例，将上述过程说明于图 3.4。设碱基错配时距离权重为 1，引入一个空位时距离权重为 3。该图边缘的行和列作为起始条件增加到表中。在单元 5 行 3 列，即相应较短序列(第二序列)的第 2 个 T 碱基和较长序列(第一序列)的第 1 个 T 碱基位置，有三种可能的距离增量。设在各序列中增加碱基 T 时（从 4 行 2 列移动)对距离的贡献为 0。从 5 行 2 列的位置作水平移动(等价于增加第二序列的碱基 T 而在第一序列引入一个空位)，在本例中增加一个罚值 3。从 3 列 4 行向该单元作垂直移动，使第一序列增加碱基 T 而第二序列引入一个空位，结果也得到一个罚值 3。因此从该单元(5 行 3

列)所得到的最小距离的延伸方向是沿对角线和水平方向。在表中这两个方向用箭头表示。这两种最短方向都使从左上角到该单元的距离为 6。沿箭头所指方向在表中从右下角向左上角追踪，得到 6 种可能的联配：

```
CTATAATCCC      CTATAATCCC      CTATAATCCC
CTGTA-TC--      CTGTA-T-C-      CTGTA-T--C

CTATAATCCC      CTATAATCCC      CTATAATCCC
CTGT-ATC--      CTGT-AT-C-      CTGT-AT--C
```

在上述 6 种联配中，距离均为 10，即在较短序列中有 6 个匹配碱基、1 个错配碱基和 3 个空位。

	0		C		T		A		T		A		A		T		C		C		C
0	0	←	3	3	←	3	6	←	3	9	←	3	12	←	3	15	←	3	18	←	3	21	←	3	24	←	3	27	←	3	30
	↑			↑																										
C	3		0	3		1	3		1	3		1	3		1	3		1	3		1	3		0	3		0	3		0	3
	3		3	0	←	3	3	←	3	6	←	3	9	←	3	12	←	3	15	←	3	18	←	3	21	←	3	24	←	3	27
	↑			↑																										
T	3		1	3		0	3		1	3		0	3		1	3		1	3		0	3		1	3		1	3		1	3
	6		3	3		3	0	←	3	3	←	3	6	←	3	9	←	3	12	←	3	15	←	3	18	←	3	21	←	3	24
	↑			↑			↑							←																
G	3		1	3		1	3		1	3		1	3		1	3		1	3		1	3		1	3		1	3		1	3
	9		3	6		3	3		3	1	←	3	4	←	3	7	←	3	10	←	3	13	←	3	16	←	3	19	←	3	22
	↑			↑			↑			↑																				
T	3		1	3		0	3		1	3		0	3		1	3		1	3		0	3		1	3		1	3		1	3
	12		3	9		3	6		3	4		3	1	←	3	4	←	3	7	←	3	10	←	3	13	←	3	16	←	3	19
	↑			↑			↑						↑																	
A	3		1	3		1	3		0	3		1	3		0	3		0	3		1	3		1	3		1	3		1	3
	15		3	12		3	9		3	6		3	4		3	1	←	3	4	←	3	7	←	3	10	←	3	13	←	3	16
	↑			↑			↑			↑						↑														
T	3		1	3		0	3		1	3		0	3		1	3		1	3		0	3		1	3		1	3		1	3
	18		3	15		3	12		3	9		3	6		3	4		3	2		3	4	←	3	7	←	3	10	←	3	13
	↑			↑			↑			↑			↑			↑														
C	3		0	3		1	3		1	3		1	3		1	3		1	3		1	3		0	3		0	3		0	3
	21		3	18		3	15		3	12		3	9		3	7		3	5		3	3		3	4	←	3	7	←	3	10

图 3.4 Needleman-Wusch 算法实例。设定碱基错配的距离权重为 1，单个碱基缺失或插入时距离权重为 3

该算法可以用代数形式来描述。设具有碱基a_i和b_j的两个序列a和b，这两个序列间距离为d(a,b)。通过评价序列a中前i个位置和序列b前j位置的距离$d(a^i,b^j)$，递归地得到距离d(a,b)。如果a和b的长度为m和n，则其期望距离为$d(a^m,b^n)$。上表中引入的第1行1列单元的距离为0(相当于空序列),在单元(i,j)内，使到达该单元距离增加的三种可能事件为：

1.从单元(i-1,j)向(i,j)的垂直移动，相当于在b序列中插入一个空位使相似序列延伸。换言之，b序列由a序列中a_i的缺失所产生，这一事件的权重记作$w_(a_i)$。

2.从单元(i-1,j-1)向(i,j)的对角线移动，相当于增加碱基a_i和b_j使相似序列延伸。换言之，b序列由a序列中的a_i被b_j取代所产生，这一事件的权重记为$w_(a_i,b_j)$。

3.从单元(i,j-1)向(i,j)的水平移动，相当于在序列b中插入一个空位使相似序列延伸。换言之，b序列由b_j插入a序列所产生，这一事件的权重记为$w_+(b_j)$。

因此，单元(i,j)的距离$d(a^i,b^j)$可看成三个相邻单元的距离加上相应权重后的最小者，即

$$d(a^i,b^j)=\min\begin{cases}d(a^{i-1},b^j)+w\_(a_i)\\ d(a^{i-1},b^{j-1})+w(a_i,b_j)\\ d(a^i,b^{j-1})+w_+(b_i)\end{cases} \qquad (3.3)$$

且初始条件为

$$d(a^0,b^0)=0$$

$$d(a^0,b^j)=\sum_{k=1}^{j}w_+(b_k)$$

$$d(a^i,b^0)=\sum_{k=1}^{i}w\_(a_k)$$

在图 3.4 的实例中

$$w\_(a_i)=3 \qquad (\text{对于每一个 i})$$

$$w(a_i,b_j)=\begin{cases}0 & (i=j)\\ 1 & (i\neq j)\end{cases}$$

$$w_+(b_j)=3 \qquad (\text{对于每一个 j})$$

当两个序列被联配时，通过计算其重排序列(shuffed version)的联配距离，可以得到这两个序列间的最小距离估计。如果实际得到的联配距离小于重排序列距离的 95%，则表明实际的联配距离达到了 5%的显著水平，是不可能由机误造成的。

二．Smith-Waterman 算法

由于亲缘关系较远的蛋白质序列可能只有一些相互独立的相同片段，所以进行局部相似性分析有时可能比整体相似性分析更合理。Smith和Waterman描述了一种查找具有最高相似性片段的算法。对于序列A=($a_1,a_2,...,a_m$)和B=($b_1,b_2,...,b_n$)，H_{ij}被定义为以a_i和b_j碱基对结束的片段(亚序列)的相似性值。与Needle-Wunsch算法一样，Smith-Waterman算法也要利用递推关系来确定H值，H的初始值为：

$$H_{i0}=0,\quad 0\le i\le n,\qquad H_{0j}=0,\quad 0\le j\le m$$

相似性计算中包括 2 个统计量：碱基对(序列因子) a_i,b_j 的相似性值 $S(a_i,b_j)$和空位权重 $w_k=v+uk$ (k 为空位长度)。Smith-Waterman 算法可以给出 2 条序列的最大相似性值。以 a_i,b_j 碱基对结束的片段可以由以 a_{i-1}和 b_{j-1}结束片段增加碱基(因子)来获得，或者 a_i 可以删除 k 长度的碱基片段，b_j 可删除 l 长度碱基片段。具体算法如下：

$$\begin{aligned}P_{ij}&=\max(H_{i-1,j}-w_1,P_{i-1,j}-u)\\Q_{ij}&=\max(H_{i,j-1}-w_1,P_{i,j-1}-u)\end{aligned}\qquad(3.4)$$

$$\text{则}\quad H_{ij}=\max\begin{cases}H_{i-1,j-1}+S(a_i,b_j)\\P_{ij}=\max_{1\le k\le i}(H_{i-k,j}-w_k)\\Q_{ij}=\max_{1\le l\le j}(H_{i,j-l}-w_l)\\0\end{cases},(1\le i\le m,1\le j\le n)\qquad(3.5)$$

其中 $P_{0,0}=P_{0,j}=Q_{0,0}=Q_{i,0}=0$

该算法可以确保具有最大H_{ij}值的序列片段是相似性最好的。从(a_i,b_j)为起点，向后追踪 H_{ij} 矩阵，直到到达某一负值。对于具有最大相似性片段以外部分的差异性不会影响到该片段的H值。

举例说明了这一算法。我们同样以上节 Needleman-Wunsch 算法中的两条短序列为例。两条序列(CTGTATC 和 CTATAATCCC)排于表 3.9 的两侧，相应的 H_{ij}，P_{ij} 和 Q_{ij} 值分别列入表中。本例的权重等根据 Smith 和 Waterman(1981)以前的例子设定为：

$$S(a_i,b_j)=\begin{cases}1 & a_i=b_j\\ -1/3 & a_i\neq b_j\end{cases}$$

w_k =1+k/3 (3.6)

对于 4 个碱基具有相同频率的随机长序列，$S(a_i,b_j)$ 值的平均值为零。w_k 值应至少不小于匹配与不匹配权重的差值。

表 3.9 的最大H_{ij}为 4.33(8 行与 7 列相交处)，星号(*)表示出具有最大相似性的片段匹配方式：

CTGTA-TC
CTATAATC

表 3.9 Smith-Waterman 算法例举

			j=0	j=1	j=2	j=3	j=4	j=5	j=6	j=7
			0	C	T	G	T	A	T	G
i=0	0	H_{ij}	0	0	0	0	0	0	0	0
		P_{ij}	0	0	0	0	0	0	0	0
		Q_{ij}	0	0	0	0	0	0	0	0
i=1	C	H_{ij}	0	1.00*	0.00	0.00	0.00	0.00	0.00	1.00
		P_{ij}	0	-0.33	-0.33	-0.33	-0.33	-0.33	-0.33	-0.33
		Q_{ij}	0	-0.33	-0.33	-0.67	-1.00	-1.33	-1.33	-1.33
i=2	T	H_{ij}	0	0.00	2.00*	0.67	1.00	0.00	1.00	0.00
		P_{ij}	0	-0.33	-0.67	-0.67	-0.67	-0.67	-0.67	-0.33
		Q_{ij}	0	-0.33	-0.67	0.67	0.33	0.00	-0.33	-0.33
i=3	A	H_{ij}	0	0.00	0.67	1.67*	0.33	2.00	0.67	0.67
		P_{ij}	0	-0.67	0.67	-0.67	-0.33	-1.00	-0.33	-0.67
		Q_{ij}	0	-0.33	-0.67	-0.67	0.33	0.00	0.67	0.33
i=4	T	H_{ij}	0	0.00	1.00	0.33	2.67*	1.33	3.00	1.67
		P_{ij}	0	-1.00	0.33	0.33	-0.67	0.67	-0.67	-0.67
		Q_{ij}	0	-0.33	-0.67	-0.33	-0.67	1.33	1.00	1.67
i=5	A	H_{ij}	0	0.00	0.00	0.67	1.33	3.67*	2.33	2.67
		P_{ij}	0	-1.33	0.00	0.00	1.33	0.00	1.67	0.33
		Q_{ij}	0	-0.33	-0.67	-1.00	-0.67	0.00	2.33	2.00
i=6	A	H_{ij}	0	0.00	0.00	0.00	1.00	2.33*	3.33	2.00
		P_{ij}	0	-1.33	-0.33	-0.33	1.00	2.33	1.33	1.33
		Q_{ij}	0	-0.33	-0.67	-1.00	-1.33	-0.33	1.00	2.00
i=7	T	H_{ij}	0	0.00	1.00	0.00	1.00	2.00	3.33*	3.00
		P_{ij}	0	-1.33	-0.67	-0.67	0.67	2.00	2.00	1.00
		Q_{ij}	0	-0.33	-0.67	-0.33	-0.67	-0.33	0.67	2.00
i=8	C	H_{ij}	0	1.00	0.00	0.67	0.33	1.67	2.00	4.33*
		P_{ij}	0	1.33	-0.33	-1.00	0.33	1.67	2.00	1. 67
		Q_{ij}	0	-0.33	-0.33	-0.67	-0.67	1.00	0.33	0. 67
i=9	C	H_{ij}	0	1.00	0.67	0.00	0.33	1.33	1.67	3. 00
		P_{ij}	0	-0.33	-0.67	-0.67	0.00	1.33	1.67	3. 00
		Q_{ij}	0	-0.33	-0.33	-0.67	-1.00	-1.00	0.00	0. 33
i=10	C	H_{ij}	0	1.00	0.67	0.33	0.00	1.00	1.33	2. 67
		P_{ij}	0	-0.33	-0.67	-1.00	-0.33	1.00	1.33	2. 67
		Q_{ij}	0	-0.33	-0.33	-0.67	-1.00	-1.33	-0.33	0. 00

三．序列相似性的统计特性[3]

到目前为止，对局部联配的统计学问题已基本搞清楚，特别是那些不含有空位(gap)的局部联配更是如此。我们不妨首先考虑不含有空位的局部联配问题，BLAST 最初的搜索程序便是以此为基础的。

无空位局部联配涉及的是等长度的一对序列片段，两个片段的各部分彼此比较。一种 Smith-Waterman 或 Sellers 算法的改进算法可以找到所有高比值片段对(high-scoring segment pairs,HSPs)，即这些片段对的比较分值不会因片段的延伸而进一步升高。

为了分析上述分值随机性产生的几率大小，需要建立一个随机序列模型。对于蛋白质而言，最简单的序列模型可通过从一条序列中随机地选取氨基酸残基，当然这一条序列中各种残基的频率必需一定。另外，一对随机氨基酸的联配期望值必需为负值，否则不论联配片段是否相关的，都会得到高比值，统计理论也将派不上用场。

就象独立随机变量之和总是倾向于正态分布(normal distribution)一样，独立随机变量的最大值倾向于极值分布(extreme value distribution)。在研究最佳局部联配时，主要涉及的是后一种情况。在一定的序列长度 m 和 n 限定下，HSP 的统计值可由 2 个参数(k 和 λ)确定。最简单的形式，即不小于比较值为 S 的 HSP 个数，可由下列公式算得其期望值：

$$E = kmne^{-\lambda s} \qquad (3.7)$$

我们称该期望值为比值 S 的 E 值(E-Value)。

上述公式非常灵敏。在给定比值的情况下，将比较序列长度加倍，则 HSP 数(即 E 值)也将加倍，同样，S 值为 2X 的某个 HSP 长度必是 S 值为 X 的两倍，所以 E 值将随着 s 值的增大急剧减少。参数 K 和 λ 可分别被简单地视为搜索步长(search spacesize)和计分系统(scoring system)的特征数。

1.二进制值或标准比值(Bit score)

最初获得的比值(S)在没有计分系统或统计量 K 和 λ 的辅助下，没有什么意义。单独的比值就如同没有单位(米或者光年)的距离。可使比值按下式标准化：

$$S' = \frac{\lambda s - \ln k}{\ln 2} \qquad (3.8)$$

获得S′值就如同得到了具有标准单位的数值。

E 值因此可简化为：

$$E = mn2^{-S'} \qquad (3.9)$$

二进制值使所使用的计分系统赋予了统计学意义，使除了可以确定搜索步长外，同样可以计算相应的显著水平。

2.P 值(P-Value)(概率值)

具有大于或等于某一比值 S 的随机 HSP 数可由泊松分布(Poisson distribution)确定。由此可以计算出搜索到某一比值大于或等于 S 的 HSP 的机率为

[3]译自NCBI BLAST TURORIAL：The statistics of sequence similarity scores.

$$e^{-E}\frac{E^X}{X!} \qquad (3.10)$$

式中 E 由(3.7)式确定。

作为一个特例，搜索不到比值≥S的HSP概率为e^{-E}，所以至少发现一个HSP (比值≥S)的概率为

$$P = 1 - e^{-E} = 1 - \exp(-kmne^{-\lambda x}) \qquad (3.11)$$

这是与比值 S 相关的 P 值(概率值)。例如，在可能搜索到 3 个比值≥S 的 HSP 的情况下，至少发现一个 HSP 的机率为 0.95[可由(3.11)式算得]。BLAST 程序中使用了 E 值而非 P 值，这主要是从直观和便于理解的角度考虑。比如 E 值等于 5 和 10，总比 P 值等于 0.993 和 0.99995 更直观。但是当 E<0.01 时，P 值与 E 值接近相同。

3.数据库搜索策略

E 值计算公式[公式(3.7)]可以应用于 2 个蛋白质序列长度分别为 m 和 n 的比较，但是对于某一序列长度为 m 的蛋白序列，如何在那些长短不一的数据库序列中找到与之匹配良好的序列呢？一种思路是把数据库中的所有蛋白序列与待查序列的关系都视为相同重要，也就是说对于 E 值均较低的短和长序列，它们是等同重要的。FASTA 程序近期版本便是采用这一策略。另一种思路是把长序列视为比短序列更重要，因为长序列往往包括更多的特异功能域(domain)。如果对序列长度上进行相关优先处理，则在计算数据库序列长度为 n 的 E 值时，将乘以 N/n，其中 N 为数据库中序列的总长度。根据公式(3.7)，E 值的计算可简单地把整个数据库序列视为长度为 N 的单条序列。BLAST 程序采用了这一策略。FASTA 策略中 E 值的计算还需再乘上数据库的序列条数。如果考虑到核酸数据库的序列长度变化更大，则在 DNA 序列相似性搜索时，BLAST 的策略可能会是合理的选择。

一些数据库搜索程序，例如 FASTA 或其它基于 Smith-Waterman 算法的程序，在进行序列搜索时，会对数据库中的每条序列进行联配并给出联配值，这些值大部分与未知序列无关，但它们被用于了 K 和 λ 参数的估计。这一方法避免了随机序列模型因使用真实序列(real sequence)造成的随意性，但同时产生了使用相关序列估计参数的难题。BLAST 仅通过部分而不是全部无关序列计算最适联配值，这赢得了搜索速度。因此，对于某一选定的替换矩阵和空位罚值，必须进行 K 和 λ 参数的预先估计，估计中使用真实序列，而非通过随机序列模型产生的模拟序列。这一估计的结果看来非常准确。

4.空位联配(gapped alignment)的统计问题

根据统计理论，以上述及的统计方法只适用于不含有空位的局部联配(非空位联配)。但是，许多计算试验和分析结果充分证明，上述统计方法同样适用于空位联配。对于非空位联配，可用基于替换矩阵和比较序列的残基频率的办法估计统计参数；对于空位联配，参数的估计则必须根据“随机”序列的大尺度比较。

5.边际效应(edge effect)

以上统计学方法对于短序列来说有些偏差。这些统计方法的基础理论是一个渐近理论，该理论假设局部联配可以适用于任何规模的联配。但是，一个高比值联配必须有一定的长度，不能从接近二条序列末端的地方开始。这种边际效应可以通过计算序列的“效应长度”(effective length)来修正。BLAST 程序中包含了这一修正过程。对于长于 200 残基的序列可以不进行边际效应的修正。

6.替换矩阵的选择

局部联配的结果与所选用的替换矩阵紧密相关。没有任何一个计分方案(即替换矩阵)可以适用于所有研究目标，对于局部联配的计分基础理论的正确理解可以极大促进序列分析准确性。相关内容详见第 4 小节。

7．空位罚值(gap penalties)

联配中另一个重要问题是空位问题。空位处理是针对序列进化过程中可能发生的插入和缺失而设计的。插入和缺失可能只涉及 1 个或 2 个残基，也可能是整个功能域(domain)，所以，在进行空位罚值设计时必须反映这些情况。

有 2 个参数应用于空位罚值设定，一个与空位设置(gap opening)有关，另一个与空位扩展(gap extension)有关。任一空位的出现均处以空位设置罚值，而任一空位的扩大必须处于空位扩展罚值。对于一个空位长度为k的罚值W_k可用下式表示：

$$w_k = a + bk \qquad (3.12)$$

其中 a 是空位设置罚值，b 为空位扩展罚值。这两个参数值设置的变化对联配产生影响(表 3.10)．

表 3.10 空位设置和空位扩展罚值对联配的影响

空位设置罚值 (Gap opening penalty)	空位扩展罚值 (Gap extension penalty)	说 明 (Comment)
大	大	极少插入或缺失：适用于非常相关蛋白质间的联配；
大	小	少量大块插入：用于整个功能域可能插入的情况
小	大	大量小块插入：适用于亲缘关系较远的蛋白质同源性分析

经过多年的试验，一个合适的空位罚值已经被确定下来。大多数联配程序均对特定的替换矩阵设定了空位罚值的缺略值(default)，如果使用者希望使用不同的替换矩阵，则原来的空位罚值设定不一定合适。如何设定罚值并无明确的理论可遁，但大的空位设置罚值配以很小的空位扩展罚值被普遍证实是最佳的设定思路。

四．替换矩阵[4]

1.替换矩阵的一般原理

我们并不能直接计算出两条序列的最佳联配，我们需要找到一个可以估计任何联配的某一统计数，使生物学关系匹配最显著的联配统计数最大。

[4]本部分内容主要取自Weir B．S．(徐云碧等译)．遗传学数据分析—群体遗传学离散型数据分析方法，北京：中国农业出版社，1996

先看以下 2 条氨基酸序列的联配情况。如果我们将各残基按相同的统计数处理，则 2 种联配(a 和 b)的得分将是相等的(9 个残基中 5 个匹配)：

```
(a)  TTYGAPPWCS          (b)  TTYGAPPWCS
      TGYAPPPWS                TGYAPPPWS
      *  ***  *                * *  ***
```

但是联配 a 是一些相对普通的残基(A、P、S 和 T)保持一致，而联配 b 则是一些相对稀有残基(W-色氨酸、Y-酪氨酸)相一致。我们需要一个更科学的赋分方法来反映匹配氨基酸间生物学和化学关系。

在联配中，C-C 匹配相对比 S-S 匹配更重要些，因为半胱氨酸(C)是具有非常特殊性质的相对稀有氨基酸，而丝氨酸(S)则相对普通。同样 D-E 匹配应取正值，因为这两个残基具有相同的化学性质，在两条联配的蛋白质序列中能起到相同的功用。但是，V-K 匹配则应被罚分，因为这两个残基毫无相似，不可能在两条序列中引到一样的作用。

替换矩阵(substitution matrices)包括了在联配中各种匹配方式如何赋分的信息，故替换矩阵又常被称为计分矩阵(scoring matrices)。

用于 DNA 序列联配的替换矩阵相对比较直观。以下是一个常被使用的替换矩阵：

	A	C	G	T
A	0.9	-0.1	-0.1	-0.1
C	-0.1	0.9	-0.1	-0.1
G	-0.1	-0.1	0.9	-0.1
T	-0.1	-0.1	-0.1	0.9

矩阵中每个匹配的碱基对均计为 0.9 分，每个不匹配的碱基对被罚 0.1 分，这样，下面一个联配的得分应为 4.3(=5×0.9+2×(-0.1))：

```
GCGCCTC
GCGGGTC
***  **
```

用于蛋白质联配的替换矩阵要复杂一些，因为没有一个矩阵可以适用各种情况。构建矩阵时应考虑不同的蛋白质家族在进化过程中，一种氨基酸突变成另一种氨基酸概率的差异，根据不同的蛋白质家族和预期的相似程度构建不同的替换矩阵。2 个最有名的蛋白质替换矩阵是 PAM 和 BLOSUM，它们分别是在 1979 年和 1992 年完成的。

最后，一个重要的概念必须明确。同源性(homology)和相似性(similarity)是不同的 2 个概念，不能混淆和混用。2 条序列具有同源性，意味着这两条序列有进化方面的关系，它们从一条共同的祖先序列进化而来；而相似性，只是表明一种相似程度。

2.PAM 氨基酸替换矩阵

在进行蛋白质序列联配时，必须通过一定的方法给联配的残基对赋予一定的分值，替换矩阵便是其中最重要的方法。

已故 Dayhoff 是蛋白质列序比较的先驱，她和她的同事们通过对蛋白质进化模式的研究，建立了一组被广泛应用的替换矩阵，这些矩阵常被称为 Dayhoff，MDM(Mutation Data Matrix)或 PAM(Percent Accepted Mutation)矩阵。

应用于DNA序列的许多算法最初是从氨基酸蛋白质序列的一些算法发展而来的。由于蛋白质最有可能是自然选择的目标，可以认为蛋白质序列的分析比DNA分析更具有生物学意义。蛋白质分析完全避免了几个三联体可能编码同一氨基酸的遗传密码简并问题。有必要进一步分析各种氨基酸间的同源性程度，以及在进化过程中一种氨基酸被另一种氨基酸替换的概率大小。也许把氨基酸按一定特性分成若干组更便于以上分析，例如氨基酸可分成中性疏水(G、A、V、L、I、F、P、M)、中性亲水(S、T、Y、W、N、E、C)、碱性(K、R、H)和酸性(D、E)氨基酸等。在比较许多具有相似性蛋白质序列的基础上，Dayhoff等于 1979 年构建了一个突变概率矩阵M(mutation probability matrix)。最初她们比较了许多对蛋白质序列，以确定进化过程中一种氨基酸被另一种氨基酸取代的经验资料。她们共观测到 1572 次取代“ 事件 ”。以此为基础，她们建立了表 3.11 的“ 可观测点突变矩阵 ”A(accepted point mutation matrix)(由于舍入误差使表中的数值相加不完全等于 1572)。氨基酸i被氨基酸j替换的经验次数(记作A_{ij})可从上表中找到。矩阵A可被称为原始PAM矩阵。

由矩阵A可以进一步获得突变概率矩阵M。矩阵M的元素M_{ij}表示经过一定的进化时期氨基酸j被氨基酸i所替换的经验频率。Dayhoff等进而把可观测突变百分率(percent accepted mutation或point accepted mutation per 100 residues)，即PAM作为一种时间度量单位。假设同一位点不会发生二次以上的突变，则 1PAM 等于 100 个氨基酸多肽链中预期发生一次替换所需的时间。

Dayhoff 提出了一个称为相对“ 突变力 ”(mutability)的概念，并将氨基酸 j 的相对突变力定义为观测到的氨基酸突变数除以联配序列中 j 氨基酸的频率，即：

$$m_j \propto \sum_{i \neq j} A_{ij} / f_j \qquad (3.13)$$

这里将氨基酸a_j所有可能的变化均考虑在内。各种氨基酸的m_j和f_j值(经标准化)列于表 3.12。

表 3.11 氨基酸替换次数表（Dayhof 等,1979）

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y
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

注：总计观测到 1572 次替换；表中次数均已乘 10；祖先序列不明时，次数以平分处理

表 3.12 根据可观测点突变资料得到的氨基酸相对突变力(m_i)和频率f_i (Dayhoff等，1979)

	m_i	f_i		m_i	f_i
A	100	0.087	L	40	0.085
R	65	0.041	K	56	0.081
N	134	0.040	M	94	0.015
D	106	0.047	F	41	0.040
C	20	0.033	P	56	0.051
Q	93	0.038	S	120	0.070
E	102	0.050	T	97	0.058
G	49	0.089	W	18	0.010
H	66	0.034	Y	41	0.030
I	96	0.037	V	20	0.065

氨基酸a_j发生变化的概率为$1-M_{jj}$，这必须与突变力相一致，即

$$1-M_{jj} \propto m_j$$

或按下式定义常数λ：

$$M_{jj} = 1-\lambda m_j \qquad (3.14)$$

同样

$$M_{ij} \propto m_j A_{ij}$$

由于M_{jj}和$\sum_{k\neq j} M_{kj}$之和必为 1

$$M_{ij} = \lambda m_j A_{ij} / \sum_{k \neq j} A_{kj} \qquad (3.15)$$

又因 1PAM 为 100 氨基酸中预期发生一次替换，则另外 99 个氨基酸不发生变化，有

$$99 = 100 \sum_i f_i M_{ii}$$

$$\lambda = \frac{1}{100 \sum_i m_i f_i} \qquad (3.16)$$

Schwarts 和 Dayhoff(1979)发现将突变概率矩阵 M 250 次方处理获得的 250PAM 矩阵(表 3.13)，对于研究远缘蛋白质之间进化关系是一个合适的时间单位。

Dayhoff 等(1979)进一步定义了一个相对概率矩阵 R(relatedness odds matrix)，其元素 $R_{ij} = M_{ij} / f_i$。这一概率矩阵是对称的。该矩阵的元素已在类似 Needleman-Wunsch 算法中用作氨基酸 i 被氨基酸 j 替换的权重 w_{ij}，表 3.14 中各元素已经对数处理(故矩阵 R 又称为对数概率矩阵，Log-odds matrix)，并将最有可能发生相互替换的氨基酸归类排列。

表 3.13 250PAM 突变概率矩阵(Dayhoff 等,1979)

	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	5	4	15	10	4	10	5	5	5	7	2	4	17

*表中数值均乘以 100；舍入误差使本表结果与上二表计算结果不完全相等。

表 3.14 250PAM 的对数概率矩阵(Dayhoff 等,1979)

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

*表中数值均乘以 10

1PAM 相当于所有的氨基酸平均有 1%发生了变化，经过 100PAM 的进化，并非每个氨基酸的残基均发生变化：有一些可能突变多次，甚至又变成原来的氨基酸，而另一些氨基酸可能根本没有发生过变化。这使我们认识到，利用大于 100PAM 的时间间隔可能达到区分同源性蛋白质的目的。应该注意，PAM 与进化时间之间没有大致对应关系，因为不同的蛋白质家族的进化速率是不同的。当 2 条序列进行相似性比较时，事先不知道怎样的进化时间(PAM)是恰当的。对于相近的序列，比较容易选择，即使不太合适的矩阵也无妨。在很多年里，PAM250(矩阵后面数字，如 PAM250、PAM100 等，表示一种进化的距离，数字越大，距离越远)是应用最广的替换矩阵，因为该矩阵是唯一由 Dayhoff 最初发表的矩阵。

后来一些学者利用大量新出现的蛋白质序列数据来更新 Dayhoff 最初计算的频率数值，由此新构建的 PAM 矩阵与最初的 PAM 矩阵没有太大的差异。

3.BLOSUM 氨基酸替换矩阵

另外一种构建矩阵的方法是由 Henikoff 等于 1992 年提出的，建成的矩阵为 BLOSUM(Blocks Substitution Matrices)。他们直接利用多序列联配(multiple alignment)分析亲缘关系较远的蛋白质，而不是用相近的序列。这方法的优点是符合实际观测结果，不足之处是它不能和进化挂起钩来。大量的试验表明，BLOSUM 矩阵总体比 PAM 矩阵更适合于生物学关系的分析和局部相似性搜索。

假设f_{ij}为序列联配中氨基酸i和j对(忽略顺序)，则i，j对氨基酸所占比例为：

$$q_{ij} = f_{ij} / \sum_{i,j} f_{ij} \qquad (3.17)$$

在完全独立的状况下，该比例的期望值为：

$$e_{ij} = \begin{cases} p_i^2 & i = j \\ 2p_i p_j & i \neq j \end{cases}$$

$$p_i = q_{ii} + \frac{1}{2}\sum_{j \neq i} q_{ij}$$

则 BLOSUM 矩阵元素(i,j)定义为：

$$s_{ij} = 2\log_2(q_{ij} / e_{ij}) \qquad (3.18)$$

蛋白序列的高度保守区(highly conserved regions)或称为模块(block)数据被用于构建 BLOSUM 矩阵。BLOSUM 矩阵后的数字表示用于构建矩阵的模块的最小相似比例，例如 BLOSUM62 为用于构建矩阵的模块数据库中，序列片段的各联配点上至少 62%是相同的。矩阵后的数字越大，则表示关系越近。

4.DNA 替换矩阵

以上有关替换矩阵的讨论仅仅提及蛋白质序列的比较，但是，相关的原则同样适用于 DNA 序列的比较。在进行比较时应该意识到，用翻译而来的蛋白质序列总是好于直接用 DNA 序列。这是因为 DNA 序列的进化变化很少，在使用简单的 DNA 替换矩阵比较时，获得的同源性信息远少于蛋白质序列。

但是，有时我们希望比较一些非编码 DNA 序列。如前所述(见本小节第 1 部分)的 DNA 替换矩阵非常简单,所有 4 个碱基的匹配与不匹配的数值均设为相同，不同的只有匹配与否(0.9 和-0.1)。一个较复杂的模型是把转换(transition,两种嘧啶或两种嘌呤间的突变)频率设为高于颠换(tranversion,嘧啶与嘌呤间的突变)频率。

五．多序列联配

通过以上的两条序列算法，总是可以返回一个最佳匹配的联配结果。但是，当我们将两条以上的序列放在一起联配时，情况就就不一样了。现有实用的多序列联配方法还不能保证一定给出最优联配结果,只能给出一个近似值——往往人为的修正可以使联配结果更佳。人类(充满生物学智慧)的眼光在判断多序列联配方面远胜过目前的任何计算机。

同源序列的多序列联配是生物信息学一个重要课题。通过多序列联配结果，允许你观察残基可以改变到什么程度而蛋白质仍保持功能；它也可以使你得到围绕某一残基的三级结构信息。有关利用多序列联配预测蛋白结构的内容将在第六章讨论。有不少多序列联配程序可通过匿名 ftp 等服务获得，例如：ClustalW 等。

三条或三条以上序列的联配方法可分为几类，如用于两条序列联配的 Needleman-Munsch 等算法的改进算法、等级法(hierachical method)、片段法(segment method)、一致或区段法(consensus or regions'method)等。这些方法中，等级法是目前应用最为广泛的方法。

等级法又称为树法(tree method)，是由 Feng 和 Doolittge(1987)等人发展

的（ClustalW 程序)。由于两条序列的联配结果可以很容易地获得，多序列联配便可以在连续使用两条序列联配算法(如 Needleman-Wunsch 算法)基础上，通过先建“ 树 ”的思路来进行多序列联配。这一方法同样是一种动态规划方法。具体步骤如下：

①对所有序列进行两两联配分析，N 条序列应有 N×(N-1)/2 对；

②对两两联配的数据进行聚类分析，产生联配等级。该等级可用分叉树(binary tree)形式或简单的排序来表示；

③根据以上联配结果，首先从所有联配中相似性最好的两条序列开始，然后是剩余联配中相似性最好的两条序列……依次类推，直至多序列联配结束。一旦两条序列的联配被列入，则序列的位置就被固定下来。例如，对于序列 A、B、C、D，如果 A 与 C、B 与 D 分别是两两联配的最佳联配结果，则 A、B、C、D 四条序列的联配则通过比对 A-C 和 B-D 两个联配(每个联配位置取平均值)来确定。

这一组合方法对大量序列的多序列联配提供了实用的空位联配手段，除了最初的两序列间的联配过程，整个多序列联配过程是很快的。

可供同时联配多个序列的程序需要更多的计算机资源，而且不如前述的比联配序那么常用。在 GCG 的 PILEUP 程序中采用的 Feng 和 Doolittle 算法；NBRF 提供的 PIRAlign 是以 Needleman-Wunsch 算法的一个变通方案为基础。由 Greg Schuler 建立的 MACAW 程序则是适用于 Microsoft Windows(微软视窗)和 Macintosh 计算机的一个很有效的多序列联配软件。这些程序(见书后所附序列分析软件)都会从所有输入序列中找出共同区，然后以此为起点建立总体联配。如果将待比较的序列局限于它们的保守区域，这些程序一般较为有效。

应该指出，目前还没有一个最佳的多序列联配方法，自动联配程序给出的结果往往可以通过人为的分析而得到改进。