CSE 549: Computational Biology

Substitution Matrices

Stony Brook University

How should we score alignments

So far, we've looked at "arbitrary" schemes for scoring mutations. How can we assign scores in a more meaningful way?

Are these scores better than these scores?

	A	C	G	T
A	5	-5	-3	-5
C	-5	5	-5	-3
G	-3	-5	5	-5
T	-5	-3	-5	5

	A	C	G	T
A	4	-1	-1	-1
C	-1	4	-1	-1
G	-1	-1	4	-1
T	-1	-1	-1	4

How should we score alignments

So far, we've looked at "arbitrary" schemes for scoring mutations. How can we assign scores in a more meaningful way?

Are these scores

	A	C	G	T
A	5	-5	-3	-5
C	-5	5	-5	-3
G	-3	-5	5	-5
T	-5	-3	-5	5

better than these scores?

	A	C	G	T
A	4	-1	-1	-1
C	-1	4	-1	-1
G	-1	-1	4	-1
T	-1	-1	-1	4

One option — "learn" the substitution / mutation rates from real data

How should we score alignments

Main Idea: Assume we can obtain (through a potentially burdensome process) a collection of high quality, high confidence sequence alignments.

We have a collection of sequences which, presumably, originated from the same ancestor — differences are mutations due to divergence.

Learn the frequency of different mutations from these alignments, and use the frequencies to derive our scoring function.

BLOSUM62 matrix

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

Brick, Kevin, and Elisabetta Pizzi. "A novel series of compositionally biased substitution matrices for comparing Plasmodium proteins." BMC bioinformatics 9.1 (2008): 236.

Probabilities to Scores

Assuming we have a reasonable process by which to compute frequencies, how can we use this to obtain a score?

Probabilities to Scores

Assuming we have a reasonable process by which to compute frequencies, how can we use this to obtain a score?

Hypothesis we wish to test; two amino acids are correlated because they are homologous.

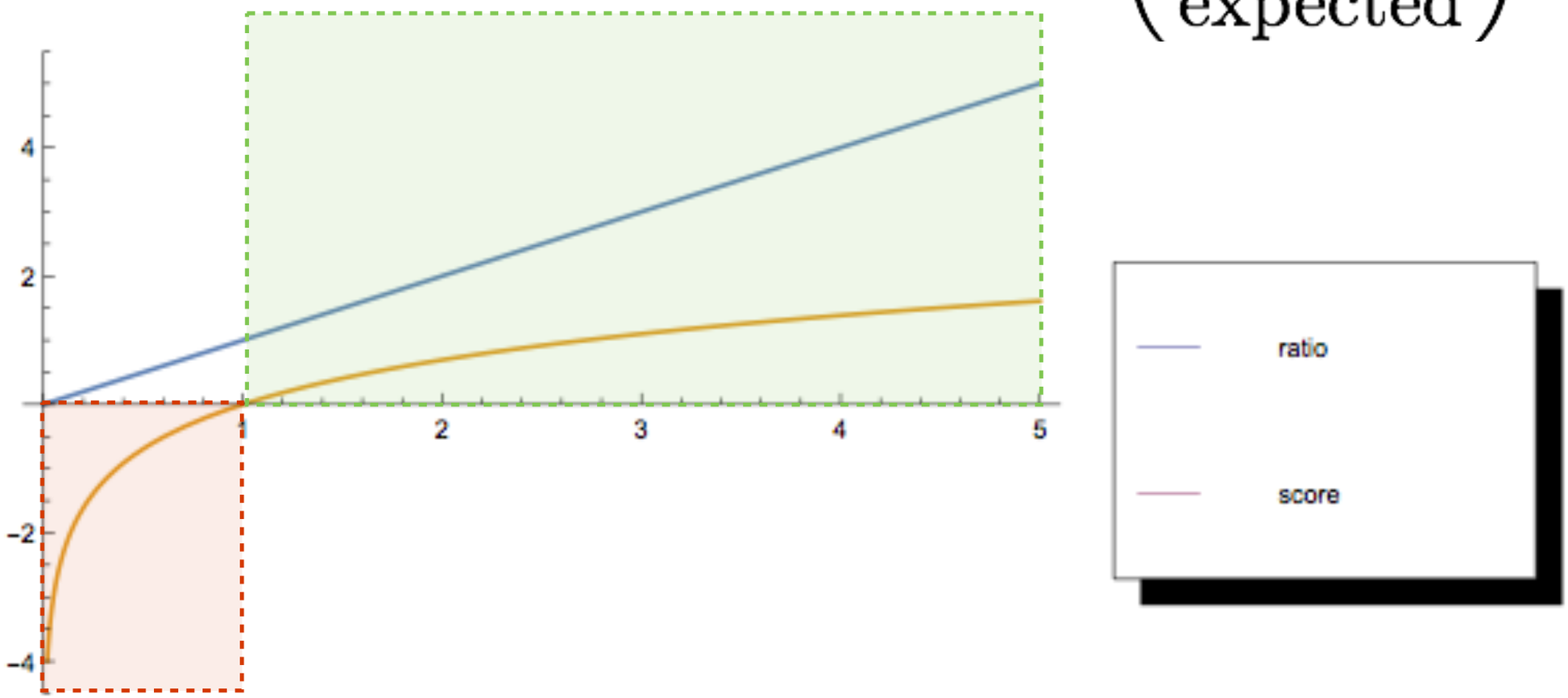
$$\text{score} = \text{log odds ratio} = s_{AB} \propto \log\left(\frac{\text{observed}}{\text{expected}}\right)$$

Null hypothesis; two amino acids occur independently (and are uncorrelated and unrelated).

Eddy, Sean R. "Where did the BLOSUM62 alignment score matrix come from?." Nature biotechnology 22.8 (2004): 1035-1036.

Probabilities to Scores

$$\text{score} = \text{log odds ratio} = s_{AB} \propto \log\left(\frac{\text{observed}}{\text{expected}}\right)$$



Positive scores mean we find “conservative substitutions”

Negative scores mean we find “nonconservative substitutions”

Eddy, Sean R. "Where did the BLOSUM62 alignment score matrix come from?." Nature biotechnology 22.8 (2004): 1035-1036.

BLOSUM matrix

Introduced by Henikoff & Henikoff in 1992

Start with the BLOCKS database (H&H '91)

1. Look for conserved (gapless, >=62% identical) regions in alignments.

2. Count all pairs of amino acids in each column of the alignments.

3. Use amino acid pair frequencies to derive "score" for a mutation/replacement

Henikoff, S.; Henikoff, J.G. (1992). "Amino Acid Substitution Matrices from Protein Blocks". PNAS 89 (22): 10915–10919.

BLOSUM matrix

Start with the BLOCKS database (H&H '91)

1. Look for conserved (gapless) regions in alignments.

sequences too similar are "clustered" & represented by either a single sequence, or a weighted combination of the cluster members

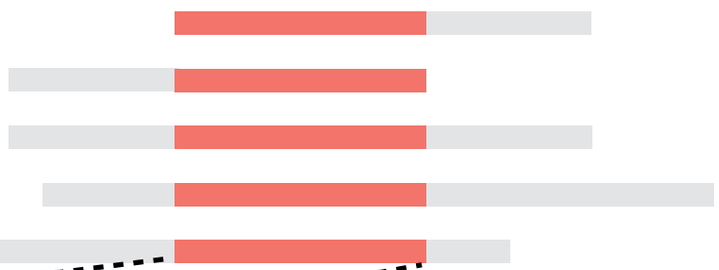
BLOSUM r: the matrix built from blocks with no more than r% of similarity – e.g., BLOSUM62 is the matrix built using sequences with no more than 62% similarity.*

Henikoff, S.; Henikoff, J.G. (1992). "Amino Acid Substitution Matrices from Protein Blocks". PNAS 89 (22): 10915–10919.

BLOSUM matrix

Start with the BLOCKS database (H&H ’91)

2. Count all pairs of amino acids in each column of the alignments.



```
FPTADAGGRS
FVTADALGRS
FPTPDAGLRN
FVTAEAGIRQ
FPTAEAGGRS
```

$$c_{AB}^{(i)} = \begin{cases} \binom{c_A^{(i)}}{2} & \text{if } A = B \\ c_A^{(i)} \times c_B^{(i)} & \text{otherwise} \end{cases}$$

$$c_A^{(i)} = \text{num. of occurrences of } A \text{ in column } i$$

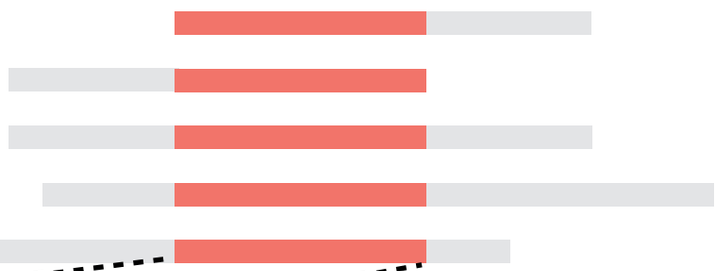
What is the intuition behind this expression?

Henikoff, S.; Henikoff, J.G. (1992). "Amino Acid Substitution Matrices from Protein Blocks". PNAS 89 (22): 10915–10919.

BLOSUM matrix

Start with the BLOCKS database (H&H ’91)

2. Count all pairs of amino acids in each column of the alignments.



```
FPTADAGGRS
FVTADALGRS
FPTPDAGLRN
FVTAEAGLRQ
FPTAEAGGRS
```

Example:

$$c_{GG}^{(i)} = \binom{3}{2} = 3$$

$$c_{GL}^{(i)} = 3 \times 2$$

$$c_{LL}^{(i)} = \binom{2}{2} = 1$$

In this column, there are 3 ways to pair G with G transitions, 6 potential ways to pair G with L and 1 potential way to pair L with L.

Henikoff, S.; Henikoff, J.G. (1992). "Amino Acid Substitution Matrices from Protein Blocks". PNAS 89 (22): 10915–10919.

Computing Scores

3. Use amino acid pair frequencies to derive "score" for a mutation/replacement

Total # of potential align. between A & B: $c_{AB} = \sum_i c_{AB}^{(i)}$

Total number of pairwise char. alignments: $T = \sum_{A \geq B} c_{AB}$

Normalized frequency of aligning A & B: $q_{AB} = \frac{c_{AB}}{T}$

BLOSUM matrix

FPTADAGGRS

FVTADALGRS

FPTPDAGLRN

FVTAEAGLRQ

FPTAEAGGRS

In our example, we get

$$q_{GL} = \frac{0+0+0+0+0+0+4+6+0+0}{10\frac{(5)(4)}{2}} = \frac{10}{100}$$

Henikoff, S.; Henikoff, J.G. (1992). "Amino Acid Substitution Matrices from Protein Blocks". PNAS 89 (22): 10915–10919.

Computing Scores

3. Use amino acid pair frequencies to derive “score” for a mutation/replacement

Probability of occurrence of amino acid A in any {A,B} pair:

$$p_A = q_{AA} + \sum_{A \neq B} \frac{q_{AB}}{2}$$

Expected likelihood of each {A,B} pair, assuming independence:

$$e_{AB} = \begin{cases} (p_A)(p_B) = (p_A)^2 & \text{if } A = B \\ (p_A)(p_B) + (p_B)(p_A) = 2(p_A)(p_B) & \text{otherwise} \end{cases}$$

Computing Scores

Recall the original idea (likelihood → scores)

$$\text{score} = \text{log odds ratio} = s_{AB} \propto \log\left(\frac{\text{observed}}{\text{expected}}\right)$$

score = log odds ratio = $s_{AB} = \text{Round}\left(\left(\frac{1}{\lambda}\right)\log_2\left(\frac{q_{AB}}{e_{AB}}\right)\right)$

Scaling factor used to produce scores that can be rounded to integers; set to 0.5 in H&H '92.

Henikoff, S.; Henikoff, J.G. (1992). "Amino Acid Substitution Matrices from Protein Blocks". PNAS 89 (22): 10915–10919.

Scores are data-dependent

distribution of amino acids matter

GG	GW
GA	GA
WG	GW
WA	GA
NG	GN
GA	GA
GA	GA

$p_G = 0.5$

$e_{GG} = 0.25$

$q_{GG} = 0.214$

$s_{GG} = \text{Round}[(2)\log_2(0.214 / 0.25)]$
$= \text{Round}[(2)(-0.22)] = 0$

$p_G = 0.5$

$e_{GG} = 0.25$

$q_{GG} = 0.5$

$s_{GG} = \text{Round}[(2)\log_2(0.5 / 0.25)]$
$= \text{Round}[(2)(1)] = 2$

adopted from: http://www.bioinfo.rpi.edu/bystrc/courses/biol4540/lecture5.pdf

Scores are data-dependent

{G,W} observed a lot

GG
GA
WG
AW
NG
GA

GA

$p_G = 0.5$ $p_W = 0.143$

$e_{GW} = 0.143$

$q_{GW} = 0.167$

$s_{GW} = \text{Round}[(2)\log_2(0.167 / 0.143)]$
$= \text{Round}[(2)(0.224)] = 0$

{G,W} observed rarely

GW
GA
GW
GA
GN
GA

AG

$p_G = 0.5$ $p_W = 0.143$

$e_{GW} = 0.143$

$q_{GW} = 0.048$

$s_{GW} = \text{Round}[(2)\log_2(0.048 / 0.143)]$
$= \text{Round}[(2)(-1.575)] = -3$

adopted from: http://www.bioinfo.rpi.edu/bystrc/courses/biol4540/lecture5.pdf

Example

FPTADAGGRS

FVTADALGRS

FPTPDAGLRN

FVTAEAGLRQ

FPTAEAGGRS

$$c_{AB} = \sum_i c_{AB}^{(i)}$$

Matrix of c_{AB} values

	A	D	E	F	G	L	N	P	Q	R	S	T	V
A	16												
D		3											
E		6	1										
F				10									
G					9								
L					10	1							
N							0						
P	4							3					
Q							1		0				
R										10			
S							3		3		3		
T												10	
V								6					1

Example

C_{AB}

$$q_{AB} = \frac{c_{AB}}{T}$$

Matrix of q_{AB} values

	A	D	E	F	G	L	N	P	Q	R	S	T	V
A	0.16												
D		0.03											
E		0.06	0.01										
F				0.1									
G					0.09								
L					0.1	0.01							
N							0						
P	0.04							0.03					
Q							0.01		0				
R										0.1			
S							0.03		0.03		0.03		
T												0.1	
V								0.06					0.01

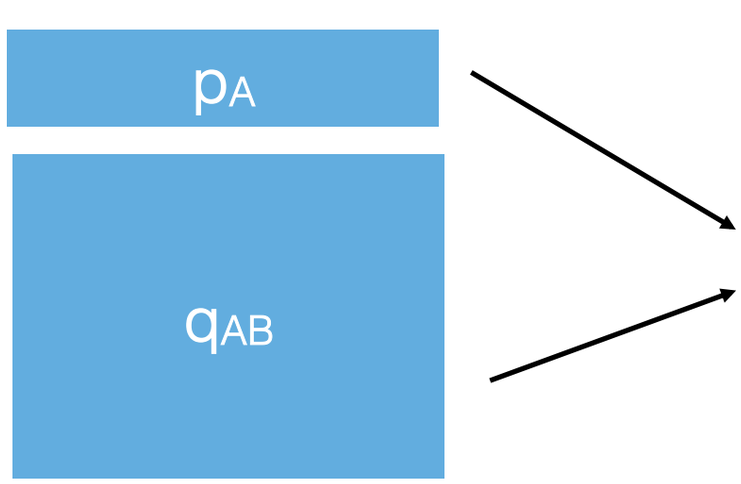
$$p_A = q_{AA} + \sum_{A \neq B} \frac{q_{AB}}{2}$$

P_A	P_D	P_E	P_F	P_G	P_L	P_N	P_P	P_Q	P_R	P_S	P_T	P_V
0.18	0.06	0.04	0.1	0.14	0.06	0.02	0.08	0.02	0.1	0.06	0.1	0.04

Example

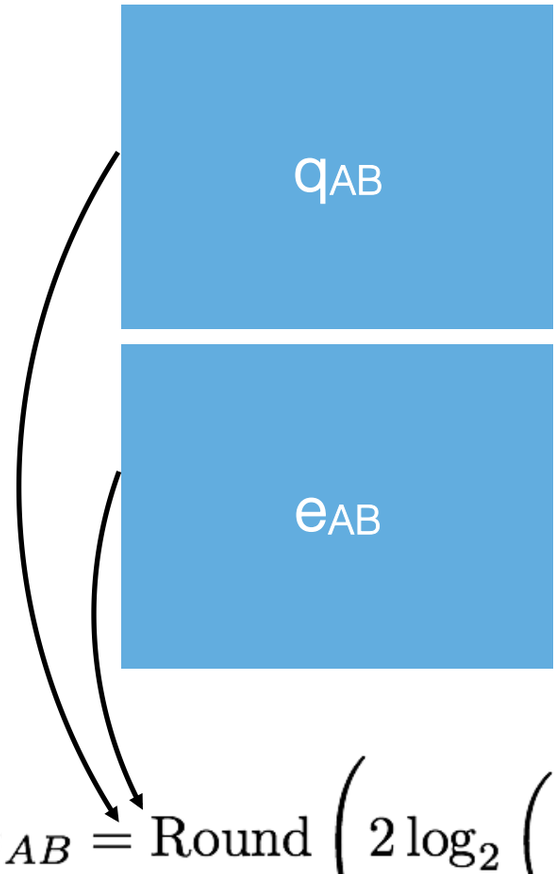
Matrix of e_{AB} values

	A	D	E	F	G	L	N	P	Q	R	S	T	V
A	0.0324												
D	0.0216	0.0036											
E	0.0144	0.0048	0.0016										
F	0.0360	0.0120	0.0080	0.0100									
G	0.0504	0.0168	0.0112	0.0280	0.0196								
L	0.0216	0.0072	0.0048	0.0120	0.0168	0.0036							
N	0.0072	0.0024	0.0016	0.0040	0.0056	0.0024	0.0004						
P	0.0288	0.0096	0.0064	0.0160	0.0224	0.0096	0.0032	0.0064					
Q	0.0072	0.0024	0.0016	0.0040	0.0056	0.0024	0.0008	0.0032	0.0004				
R	0.0360	0.0120	0.0080	0.0200	0.0280	0.0120	0.0040	0.0160	0.0040	0.0100			
S	0.0216	0.0072	0.0048	0.0120	0.0168	0.0072	0.0024	0.0096	0.0024	0.0120	0.0036		
T	0.0360	0.0120	0.0080	0.0200	0.0280	0.0120	0.0040	0.0160	0.0040	0.0200	0.0120	0.0100	
V	0.0144	0.0048	0.0032	0.0080	0.0112	0.0048	0.0016	0.0064	0.0016	0.0080	0.0048	0.0080	0.0016



$$e_{AB} = \begin{cases} (p_A)(p_B) = (p_A)^2 & \text{if } A = B \\ (p_A)(p_B) + (p_B)(p_A) = 2(p_A)(p_B) & \text{otherwise} \end{cases}$$

Example



$$s_{AB} = \text{Round}\left(2\log_2\left(\frac{q_{AB}}{e_{AB}}\right)\right)$$

Matrix of scores

	A	D	E	F	G	L	N	P	Q	R	S	T	V
A	5												
D		6											
E		7	5										
F				7									
G					4								
L					5	3							
N													
P	1							4					
Q							7						
R										7			
S							7		7		6		
T												7	
V								6					5

Blank cells are “missing data” (i.e. no observed values); wouldn’t happen with sufficient training data.

Dealing with sequence redundancy

E.g., for BLOSUM-80, group sequences that are >80% similar

```
TCMN_STRGA ( 331) IADLGGGDGWFLAQILRRHPHATGLLMDLPRVA  74
TCMO_STRGA ( 173) FVDLGGARGNLAAHLHRAHPHLRATCFDLPEME  81
ZRP4_MAIZE ( 204) LVDVGGGIGAAAQAISKAFPHVKCSVLDLAHVV  68

COMT_EUCGU ( 205) VVDVGGGTGAVLSMIVAKYPSMKGINFDLPHVI  42
CHMT_POPTM ( 204) LVDVGGGTGAVVNTIVSKYPSIKGINFDLPHVI  41
COMT_MEDSA ( 204) LVDVGGGTGAVINTIVSKYPTIKGINFDLPHVI  47

CRTF_RHOSH ( 205) LMDVGGGTGAFLAAVGRAYPLMELMLFDLPVVA  59
OMTA_ASPPA ( 250) VVDVGGGRGHLSRRVSQKHPHLRFIVQDLPAVI  47
```

1 sequence (1/3 for each)

- **Sequences are not independent because they are closely related, in this case COMT_EUCGU, CHMT_POPTM, and COMT_MEDSA are all >80 identical, and the others are more different**

- **BLOSUM approach accounts for this by treating the group of 3 as a count of 1**

- **One then gets a Weighted (BLOSUM 80) count of transitions for column 1:**

$c_{FF} = 0$ $\quad c_{FI} = 1$ $\quad c_{FL} = 2.67$ $\quad c_{FV} = 1.33$

$c_{II} = 0$ $\quad c_{IL} = 2.67$ $\quad c_{IV} = 1.33$

$c_{LL} = 2.33$ $\quad c_{LV} = 3.33$

$c_{VV} = 0.33$

(slide from Michael Gribskov)

Point Accepted Mutation Matrix

Introduced by Margaret Dayhoff in 1978

Observed mutation probabilities between amino acids over 71 families of closely related proteins (85% sequence identity within a family)

Based on a Markov mutation model; 1 PAM is the unit of time required for 1 mutation to occur per 100 amino acids. The PAM_1 matrix express the log odds ratio of the likelihood of a point accepted mutation from one amino acid to another to the likelihood that these amino acids were aligned by chance.

Picture from: http://en.wikipedia.org/wiki/Margaret_Oakley_Dayhoff#mediaviewer/File:Margaret_Oakley_Dayhoff_cropped.jpg

Other Scoring Matrices

PAM vs. BLOSUM

PAM	BLOSUM
To compare the closely related sequences, PAM matrices with lower numbers are created.	To compare the closely related sequences, BLOSUM matrices with higher numbers are created.
To compare the distantly related proteins, PAM matrices with high numbers are created.	To compare the distantly related proteins, BLOSUM matrices with low numbers are created.

PAM	BLOSUM
PAM100	BLOSUM90
PAM120	BLOSUM80
PAM160	BLOSUM60
PAM200	BLOSUM52
PAM250	BLOSUM45

from: http://en.wikipedia.org/wiki/BLOSUM, http://en.wikipedia.org/wiki/Point_accepted_mutation

Other Scoring Matrices

PAM vs. BLOSUM

PAM	BLOSUM
Based on global alignments of closely related proteins.	Based on local alignments of protein segments.
PAM1 is the matrix calculated from comparisons of sequences with no more than 1% divergent	BLOSUM 62 is calculated from comparisons of sequences no more than 62% identical
Other PAM matrices are extrapolated from PAM1	Other BLOSUM matrices are not extrapolated, but computed based on observed alignments at different identity percentage
Larger numbers in name denote larger evolutionary distance	Larger numbers in name denote higher sequence similarity (& therefore smaller evolutionary distance)
Based on explicit, Markovian, model of evolution	Not based on any explicit model of evolution, but learned empirically from alignments

from: http://en.wikipedia.org/wiki/BLOSUM

What about gap penalties?

Despite some work[+], the setting of gap penalties is still much more arbitrary than the selection of a substitution matrix.

> ★Gap penalty values are designed to reduce the score when an alignment has been disturbed by indels. The value should be small enough to allow a previously accumulated alignment to continue with an insertion of one of the sequences, but should not be so large that this previous alignment score is removed completely.

Changing the gap function can have significant effects on reported alignments. People often resort to “defaults” to avoid having to justify a choice.

eese, J. T., and William R. Pearson. "Empirical determination of effective gap penalties for sequence comparison." Bioinformatics 18.11 (2002): 1500-1507

★ http://en.wikipedia.org/wiki/Gap_penalty