Nucleic Acid Matrices

Abhinav Mishra
IIIT-H

Motivation

Scoring matrices => based on models appropriate to the analysis of molecular sequencing errors or biological mutation processes are presented.

Alignment information (available) using an optimal scoring system is compared with that obtained using the BLASTN default scoring.

Results of searches performed using BLASTN's default score matrix are compared with those using scores based on a mutational model in which transitions are more prevalent than transversions.

Introduction

BLASTN => **Match (+5)/Mismatch(-4)**

(Source Code)

Natural mutations do not interconvert the various bases uniformly: **transitions are favored over transversions by a factor of ~= 3.**

Scoring of closely related sequences should differ from that of sequences known to be distantly related => **Markov transition model** [Basis of PAM matrices].

Such matrices can be derived for nucleotide sequence comparisons as well.

Log odds score matrices

Mutation --> Random & Independent

A matrix **M** of probabilities for substituting base **i** by base **j** after any given amount of evolution can be calculated by *successive iteration* of a reference mutation matrix : $M_n = (M_1)^n$

M_1 = matrix reflecting 99% sequence conservation and one point accepted mutation (1 PAM) per 100 bases.

M_n = substitution probabilities after n PAMs.

n-PAM log-odds score for aligning a given pair of bases is the log (base 2) of the relative chance of that pair occurring as a result of evolution as opposed to that occurring from a random alignment of two bases:

$$S_{i,j} = \log (p_i * M_{ni,j} / p_i * p_j)$$

p_i = frequency of base **i**

For $p_A = p_C = p_G = p_T$, $\boldsymbol{S_{i,j} = \log (4 * M_{ni,j})}$

If the base of the logarithm used in formula is taken to be 2, then scores can be thought of as being expressed in bits.

At any given PAM distance, *the expected information **H** (in bits) per alignment position* can be calculated as described by Altschul:

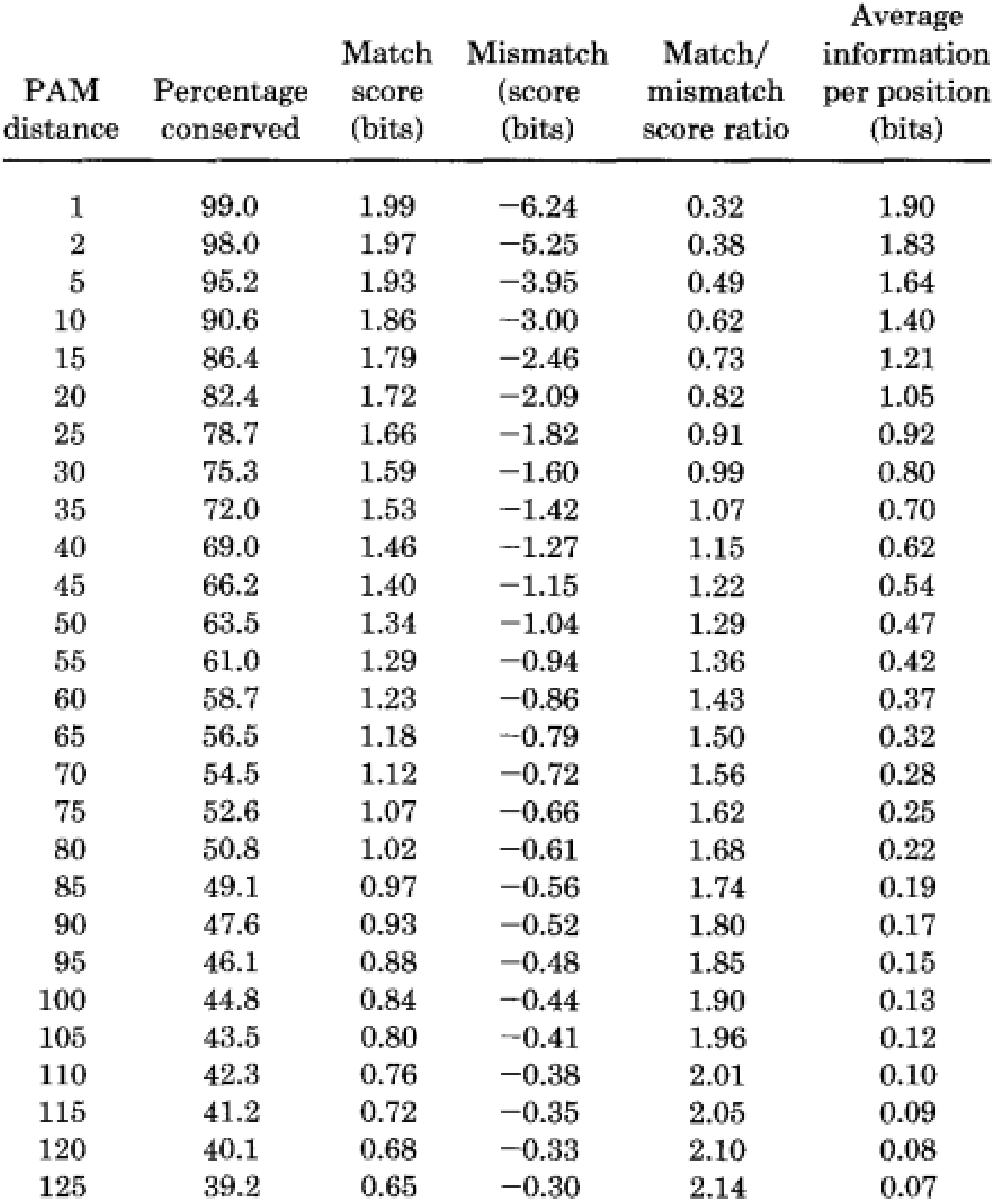
$$H = \Sigma\, p_i * p_j * S_{i,j} * 2^{S_{i,j}}$$

Table 1 :

PAM scores for

Uniform Mutational

Model



PAM distance	Percentage conserved	Match score (bits)	Mismatch (score (bits)	Match/ mismatch score ratio	Average information per position (bits)
1	99.0	1.99	−6.24	0.32	1.90
2	98.0	1.97	−5.25	0.38	1.83
5	95.2	1.93	−3.95	0.49	1.64
10	90.6	1.86	−3.00	0.62	1.40
15	86.4	1.79	−2.46	0.73	1.21
20	82.4	1.72	−2.09	0.82	1.05
25	78.7	1.66	−1.82	0.91	0.92
30	75.3	1.59	−1.60	0.99	0.80
35	72.0	1.53	−1.42	1.07	0.70
40	69.0	1.46	−1.27	1.15	0.62
45	66.2	1.40	−1.15	1.22	0.54
50	63.5	1.34	−1.04	1.29	0.47
55	61.0	1.29	−0.94	1.36	0.42
60	58.7	1.23	−0.86	1.43	0.37
65	56.5	1.18	−0.79	1.50	0.32
70	54.5	1.12	−0.72	1.56	0.28
75	52.6	1.07	−0.66	1.62	0.25
80	50.8	1.02	−0.61	1.68	0.22
85	49.1	0.97	−0.56	1.74	0.19
90	47.6	0.93	−0.52	1.80	0.17
95	46.1	0.88	−0.48	1.85	0.15
100	44.8	0.84	−0.44	1.90	0.13
105	43.5	0.80	−0.41	1.96	0.12
110	42.3	0.76	−0.38	2.01	0.10
115	41.2	0.72	−0.35	2.05	0.09
120	40.1	0.68	−0.33	2.10	0.08
125	39.2	0.65	−0.30	2.14	0.07

Table 1 shows the log-odds scores (expressed in bits) derived using the uniform substitution model for various PAM distances.

For computational purposes, the scores may be multiplied by any positive number.

At 30 PAMs (about 75% sequence conservation when back mutations are considered) the magnitudes of the match and mismatch scores are nearly identical, and at 47 PAMs the ratio is ~= 5 / 4.

Scaled by a constant factor, these are the scores incorporated into BLASTN

PAM distance corresponding to an alignment cannot be known before the alignment is found, but the information H(D) available at various PAM distances D is acheived when appropriate scores are used.

Since one does not want to use hundreds of different scoring systems, an important question is over **what range of actual PAM distances a given set of scores is nearly optimal** ?

Using a set of scores optimized for PAM distance E, it is simple to calculate the average score achieved when segments actually separated by PAM distance D are aligned.

$$\text{Efficiency (E)} = S_{i,j} / H(D) \mathbf{<= 1}$$

PAM distances D from 0 to 100, using scores for the correct PAM distance but based on the uniform as opposed to biased mutational model yields an efficiency of about **(100 – D/5)% => Loss of information** (> 40 PAMs)**.**

Efficieny Curve for various PAM Matrices (E = 10,30,50,70,90)

At D = E,

max(E) = 1.0

Uniform

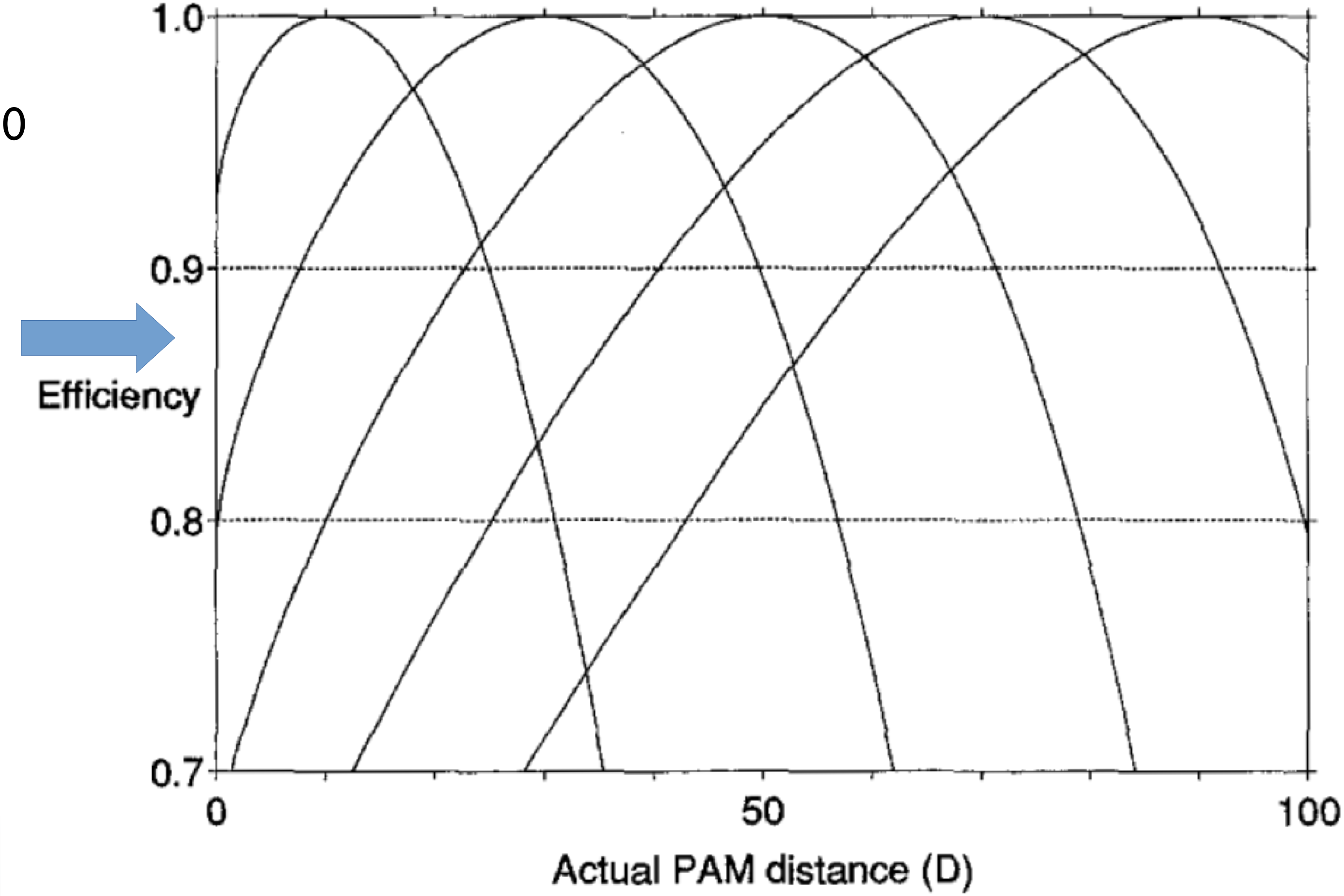
Mutational

Model

For biased,

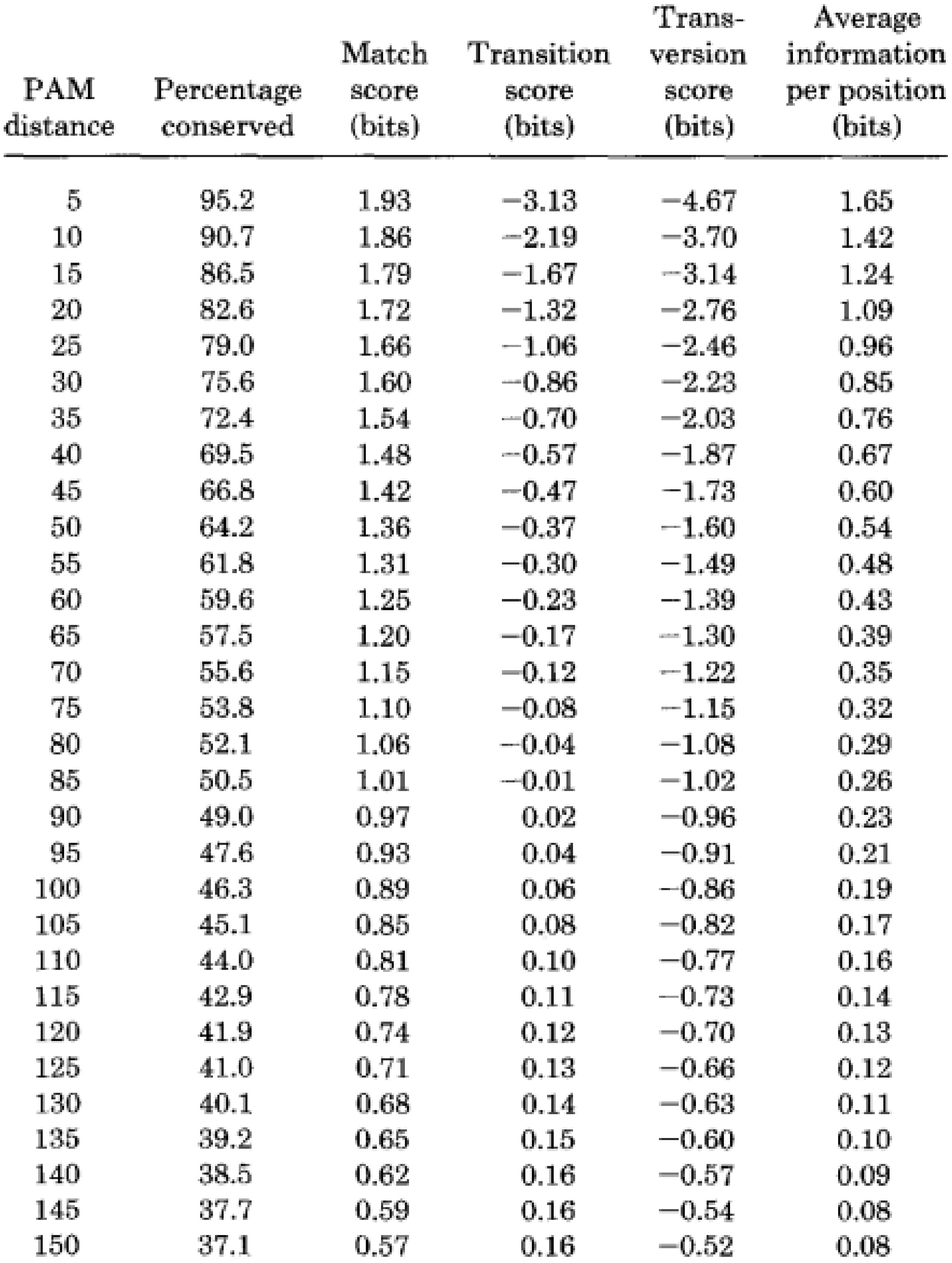
Curve will

get flatter.



Given a desired degree of efficiency and a desired range of actual PAM distances, one may calculate how many different PAM matrices need be employed and which ones should be used.

Table 2: PAM scores for Biased Mutational Model



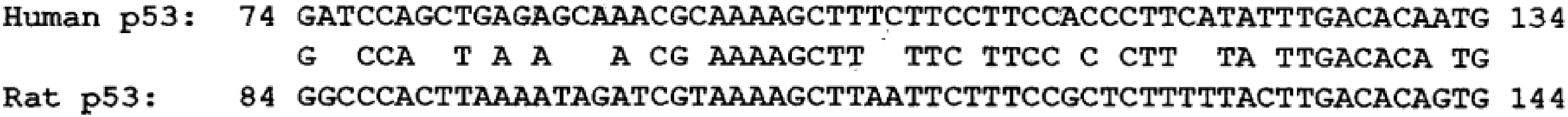
PAM distance	Percentage conserved	Match score (bits)	Transition score (bits)	Trans-version score (bits)	Average information per position (bits)
5	95.2	1.93	−3.13	−4.67	1.65
10	90.7	1.86	−2.19	−3.70	1.42
15	86.5	1.79	−1.67	−3.14	1.24
20	82.6	1.72	−1.32	−2.76	1.09
25	79.0	1.66	−1.06	−2.46	0.96
30	75.6	1.60	−0.86	−2.23	0.85
35	72.4	1.54	−0.70	−2.03	0.76
40	69.5	1.48	−0.57	−1.87	0.67
45	66.8	1.42	−0.47	−1.73	0.60
50	64.2	1.36	−0.37	−1.60	0.54
55	61.8	1.31	−0.30	−1.49	0.48
60	59.6	1.25	−0.23	−1.39	0.43
65	57.5	1.20	−0.17	−1.30	0.39
70	55.6	1.15	−0.12	−1.22	0.35
75	53.8	1.10	−0.08	−1.15	0.32
80	52.1	1.06	−0.04	−1.08	0.29
85	50.5	1.01	−0.01	−1.02	0.26
90	49.0	0.97	0.02	−0.96	0.23
95	47.6	0.93	0.04	−0.91	0.21
100	46.3	0.89	0.06	−0.86	0.19
105	45.1	0.85	0.08	−0.82	0.17
110	44.0	0.81	0.10	−0.77	0.16
115	42.9	0.78	0.11	−0.73	0.14
120	41.9	0.74	0.12	−0.70	0.13
125	41.0	0.71	0.13	−0.66	0.12
130	40.1	0.68	0.14	−0.63	0.11
135	39.2	0.65	0.15	−0.60	0.10
140	38.5	0.62	0.16	−0.57	0.09
145	37.7	0.59	0.16	−0.54	0.08
150	37.1	0.57	0.16	−0.52	0.08

Table 2 shows a series of scoring matrices derived from a biased mutational model in which each transition is three times more likely than each transversion.

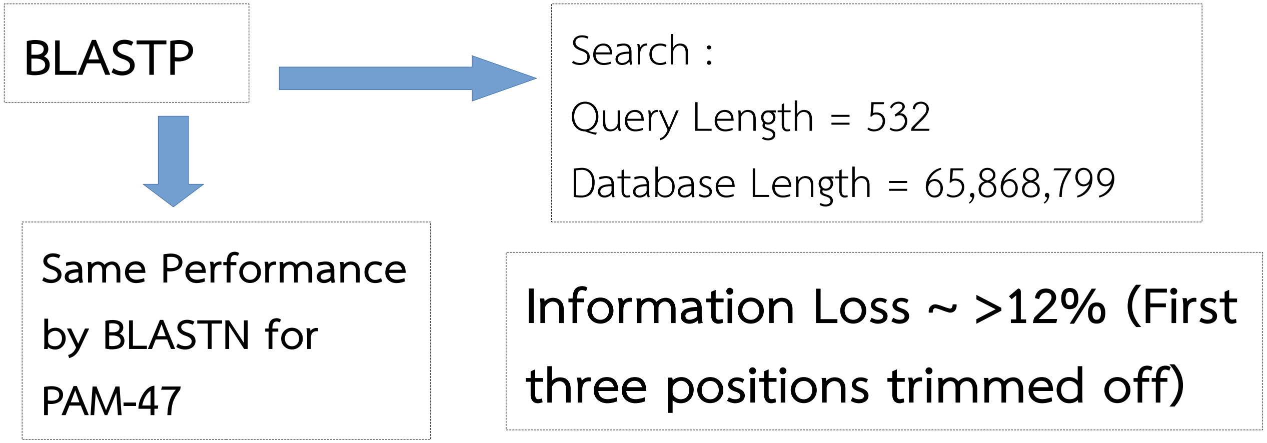
At greater than 87 PAMs, transitions score positively and are therefore, conservative substitutions.

If the mutations in the DNA sequences being compared are biased, then the **scores of Table 2 distinguish true relationships from random noise more efficiently than those of Table 1.**

Example [5’ upstream region]



```
Human p53:   74 GATCCAGCTGAGAGCAAACGCAAAAGCTTTCTTCCTTCCACCCTTCATATTTGACACAATG 134
                G  CCA  T A A   A CG AAAAGCTT  TTC TTCC C CTT  TA TTGACACA TG
Rat p53:     84 GGCCCACTTAAAATAGATCGTAAAAGCTTAATTCTTTCCGCTCTTTTTACTTGACACAGTG 144
```



BLASTP

Search :

Query Length = 532

Database Length = 65,868,799

Same Performance by BLASTN for PAM-47

Information Loss ~ >12% (First three positions trimmed off)

Matches = 41

Transitions = 13

Transversions = 7

Using PAM-50 scores,

Biased Model => Alignment score = 39.75 bits (Table 2), **p val = 0.013**

But,

Uniform Model => Alignment score = 34.88 bits (Table 1), p val = 0.31

BIASED MODEL IS MORE STATISTICALLY SIGNIFICANT THAN A UNIFORM MUTATIONAL MODEL

Proposed Changes

Nucleic acid database searches with application specific scores are easily implemented using the BLASTP program.

Minimal modification in source code:

‘fq’ <--- *uniform frequencies* for A, C, G, and T in the array

"QUERYLEN_MAX" <--- increase for long nucleotide sequences

“W” <--- *word size* increase to **6**, for the neighbourhood table
(increase in search speed!)

“T” <--- *score threshold for including a word* in the neighborhood table to a **large positive value** => Table containing only matches

“X” <--- *cutoff for extending word hits* to **1000** in score matrix (hundreths of bits) => reduce the probability of prematurely truncating an aligned segment to less than 0.1% (Default = Heurisitic Adjustment)

Computational cost to the increased flexibility achieved using BLASTP rather than BLASTN due to following reasons :

•restricted to a four-character alphabet

•employs hard-coded scores

•uses a long word size <= **significant loss of sensitivity for moderately diverged sequences.**

Alignment shown the example is missed altogether by BLASTN because it **lacks a run of 12 identities needed to generate a hit in the BLASTN neighborhood table**.

BLASTP vs BLASTN

For maximum specificity,

✓The sense and anti-sense strands are searched separately by BLASTP.

✓BLASTN automatically searches both strands of the query.

Protein Coding Regions

Consider two proteins that have diverged by **D** protein PAMs <= ***D non synonymous point mutations*** *at the DNA level.*

Broadly speaking, there tend to be over 1.5 synonymous point mutations (SPM) for every nonsynonymous point mutation (NSPM):

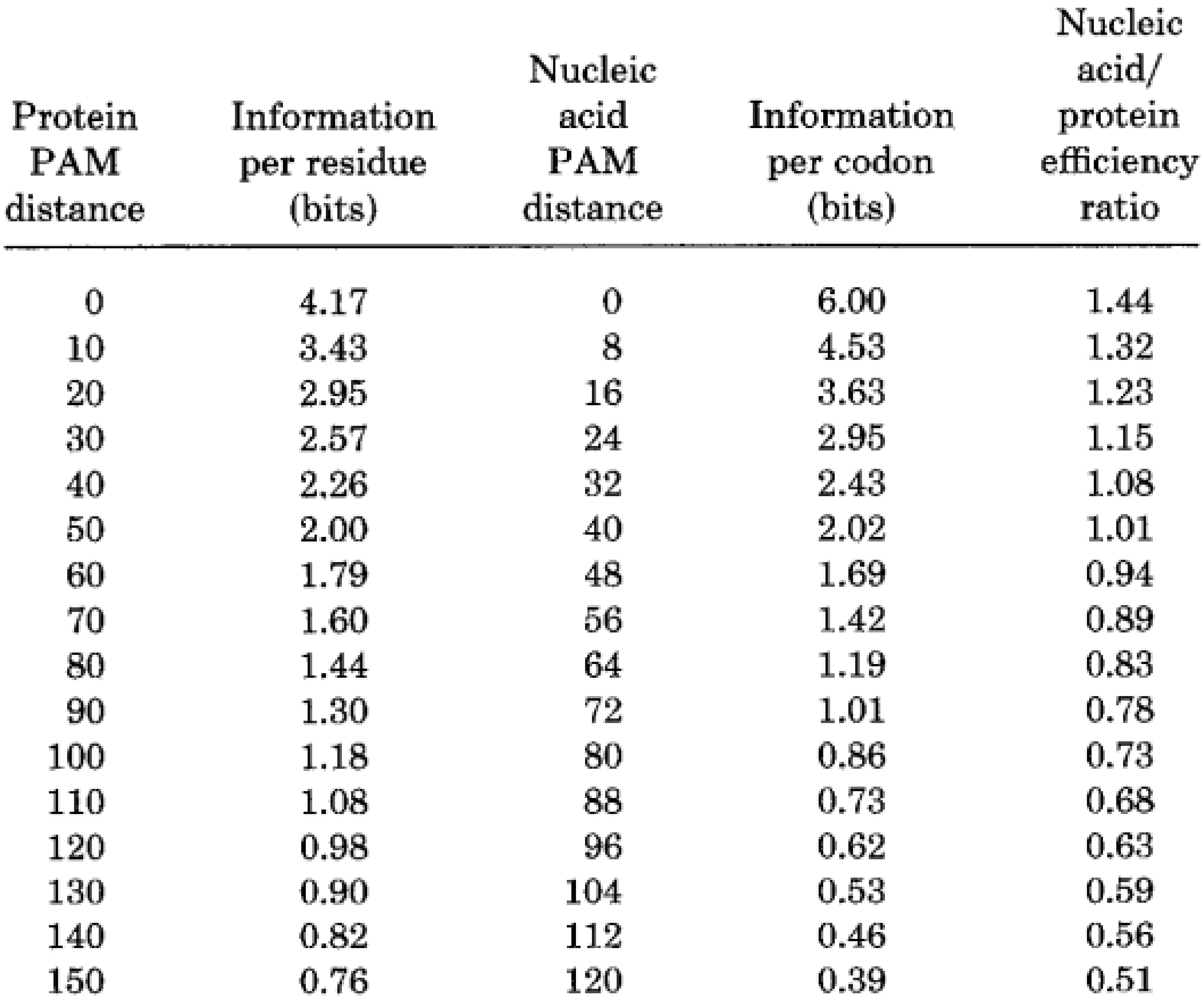
NSPM/SPM >= 1.5

1 codon = 3 nucleotides

Each amino acid PAM = (1 + 1.5)/3 ~ 0.8 Nucleic acid PAMs

Table 3:

Relative Information Available Using Protein and Nucleic Acid-Based PAM Scores



Protein PAM distance	Information per residue (bits)	Nucleic acid PAM distance	Information per codon (bits)	Nucleic acid/ protein efficiency ratio
0	4.17	0	6.00	1.44
10	3.43	8	4.53	1.32
20	2.95	16	3.63	1.23
30	2.57	24	2.95	1.15
40	2.26	32	2.43	1.08
50	2.00	40	2.02	1.01
60	1.79	48	1.69	0.94
70	1.60	56	1.42	0.89
80	1.44	64	1.19	0.83
90	1.30	72	1.01	0.78
100	1.18	80	0.86	0.73
110	1.08	88	0.73	0.68
120	0.98	96	0.62	0.63
130	0.90	104	0.53	0.59
140	0.82	112	0.46	0.56
150	0.76	120	0.39	0.51

Table 3 shows that at this distance, about 37% of the information available through an amino acid substitution matrix is lost using a nucleotide score matrix, even when a biased mutational model is employed.

Alignments of sequences that have diverged by fewer than 50 protein PAMs, the nucleic acid alphabet is more informative, while for more distant relationships the protein alphabet is superior.

While an alignment of two proteins diverged by fewer than 50 PAMs may be more significant when viewed using the nucleic acid alphabet, such an alignment in any case need be **no longer than 15 residues to yield 30 bits of information**.

Applications

1. DNA Sequencing projects => CONTIG

2. Evaluating Sequence segements :

- PCR

- Oligonucleotide Hybridization Primers

Conclusion

- To achieve optimal sensitivity, we must use scoring scheme according to specific requirements.

- Scores based on a biased mutational model may improve the search sensitivity for conserved elements in non-coding regions.

Reference

States, David J., Warren Gish, and Stephen F. Altschul. "Improved sensitivity of nucleic acid database searches using application-specific scoring matrices." Methods 3, no. 1 (1991): 66-70.