Proc. Natl. Acad. Sci. USA
Vol. 89, pp. 10915–10919, November 1992
Biochemistry

Amino acid substitution matrices from protein blocks

(amino acid sequence/alignment algorithms/data base searching)

STEVEN HENIKOFF* AND JORJA G. HENIKOFF

Howard Hughes Medical Institute, Basic Sciences Division, Fred Hutchinson Cancer Research Center, Seattle, WA 98104

Communicated by Walter Gilbert, August 28, 1992 (received for review July 13, 1992)

ABSTRACT **Methods for alignment of protein sequences typically measure similarity by using a substitution matrix with scores for all possible exchanges of one amino acid with another. The most widely used matrices are based on the Dayhoff model of evolutionary rates. Using a different approach, we have derived substitution matrices from about 2000 blocks of aligned sequence segments characterizing more than 500 groups of related proteins. This led to marked improvements in alignments and in searches using queries from each of the groups.**

Among the most useful computer-based tools in modern biology are those that involve sequence alignments of proteins, since these alignments often provide important insights into gene and protein function. There are several different types of alignments: global alignments of pairs of proteins related by common ancestry throughout their lengths, local alignments involving related segments of proteins, multiple alignments of members of protein families, and alignments made during data base searches to detect homology. In each case, competing alignments are evaluated by using a scoring scheme for estimating similarity. Although several different scoring schemes have been proposed (1–6), the mutation data matrices of Dayhoff (1, 7–9) are generally considered the standard and are often the default in alignment and searching programs. In the Dayhoff model, substitution rates are derived from alignments of protein sequences that are at least 85% identical. However, the most common task involving substitution matrices is the detection of much more distant relationships, which are only inferred from substitution rates in the Dayhoff model. Therefore, we wondered whether a better approach might be to use alignments in which these relationships are explicitly represented. An incentive for investigating this possibility is that implementation of an improved matrix in numerous important applications requires only trivial effort.

METHODS

Deriving a Frequency Table from a Data Base of Blocks. Local alignments can be represented as ungapped blocks with each row a different protein segment and each column an aligned residue position. Previously, we described an automated system, PROTOMAT, for obtaining a set of blocks given a group of related proteins (10). This system was applied to a catalog of several hundred protein groups, yielding a data base of >2000 blocks. Consider a single block representing a conserved region of a protein family. For a new member of this family, we seek a set of scores for matches and mismatches that best favors a correct alignment with each of the other segments in the block relative to an incorrect alignment. For each column of the block, we first count the number of matches and mismatches of each type between the new sequence and every other sequence in the block. For example, if the residue of the new sequence that aligns with the first column of the first block is A and the column has 9 A residues and 1 S residue, then there are 9 AA matches and 1 AS mismatch. This procedure is repeated for all columns of all blocks with the summed results stored in a table. The new sequence is added to the group. For another new sequence, the same procedure is followed, summing these numbers with those already in the table. Notice that successive addition of each sequence to the group leads to a table consisting of counts of all possible amino acid pairs in a column. For example, in the column consisting of 9 A residues and 1 S residue, there are $8 + 7 + \ldots 1 = 36$ possible AA pairs, 9 AS or SA pairs, and no SS pairs. Counts of all possible pairs in each column of each block in the data base are summed. So, if a block has a width of w amino acids and a depth of s sequences, it contributes $ws(s-1)/2$ amino acid pairs to the count [$(1 \times 10 \times 9)/2 = 45$ in the above example]. The result of this counting is a frequency table listing the number of times each of the $20 + 19 + \ldots 1 = 210$ different amino acid pairs occurs among the blocks. The table is used to calculate a matrix representing the odds ratio between these observed frequencies and those expected by chance.

Computing a Logarithm of Odds (Lod) Matrix. Let the total number of amino acid i, j pairs ($1 \le j \le i \le 20$) for each entry of the frequency table be f_{ij}. Then the observed probability of occurrence for each i, j pair is

$$q_{ij} = f_{ij} / \sum_{i=1}^{20} \sum_{j=1}^{i} f_{ij}.$$

For the column of 9 A residues and 1 S residue in the example, where $f_{AA} = 36$ and $f_{AS} = 9$, $q_{AA} = 36/45 = 0.8$ and $q_{AS} = 9/45 = 0.2$. Next we estimate the expected probability of occurrence for each i, j pair. It is assumed that the observed pair frequencies are those of the population. For the example, 36 pairs have A in both positions of the pair and 9 pairs have A at only one of the two positions, so that the expected probability of A in a pair is $[36 + (9/2)]/45 = 0.9$ and that of S is $(9/2)/45 = 0.1$. In general, the probability of occurrence of the ith amino acid in an i, j pair is

$$p_i = q_{ii} + \sum_{j \ne i} q_{ij}/2.$$

The expected probability of occurrence e_{ij} for each i, j pair is then $p_i p_j$ for $i = j$ and $p_i p_j + p_j p_i = 2 p_i p_j$ for $i \ne j$. In the example, the expected probability of AA is $0.9 \times 0.9 = 0.81$, that of AS + SA is $2 \times (0.9 \times 0.1) = 0.18$, and that of SS is $0.1 \times 0.1 = 0.01$. An odds ratio matrix is calculated where each entry is q_{ij}/e_{ij}. A lod ratio is then calculated in bit units as $s_{ij} = \log_2(q_{ij}/e_{ij})$. If the observed frequencies are as expected, $s_{ij} = 0$; if less than expected, $s_{ij} < 0$; if more than expected, $s_{ij} > 0$. Lod ratios are multiplied by a scaling factor of 2 and then rounded to the nearest integer value to produce

The publication costs of this article were defrayed in part by page charge payment. This article must therefore be hereby marked "*advertisement*" in accordance with 18 U.S.C. §1734 solely to indicate this fact.

Abbreviation: lod, logarithm of odds.

*To whom reprint requests should be addressed.

BLOSUM (blocks substitution matrix) matrices in half-bit units, comparable to matrices generated by the PAM (percent accepted mutation) program (11). For each substitution matrix, we calculated the average mutual information (12) per amino acid pair H (also called relative entropy), and the expected score E in bit units as

$$H = \sum_{i=1}^{20} \sum_{j=1}^{i} q_{ij} \times s_{ij}; \qquad E = \sum_{i=1}^{20} \sum_{j=1}^{i} p_i \times p_j \times s_{ij}.$$

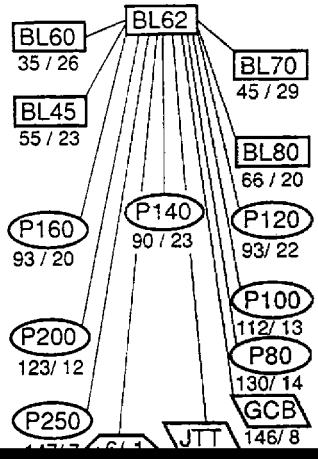
Clustering Segments Within Blocks. To reduce multiple contributions to amino acid pair frequencies from the most closely related members of a family, sequences are clustered within blocks and each cluster is weighted as a single sequence in counting pairs (13). This is done by specifying a clustering percentage in which sequence segments that are identical for at least that percentage of amino acids are grouped together. For example, if the percentage is set at 80%, and sequence segment A is identical to sequence segment B at ≥80% of their aligned positions, then A and B are clustered and their contributions are averaged in calculating pair frequencies. If C is identical to either A or B at ≥80% of aligned positions, it is also clustered with them and the contributions of A, B, and C are averaged, even though C might not be identical to both A and B at ≥80% of aligned positions. In the above example, if 8 of the 9 sequences with

matrix construction. Frequency tables, matrices, and programs for UNIX and DOS machines are available over Internet by anonymous ftp (sparky.fhcrc.org).

Constructing Blocks Data Bases. For this work, we began with versions of the blocks data base constructed by PROTOMAT (10) from 504 nonredundant groups of proteins catalogued in Prosite 8.0 (14) keyed to Swiss-Prot 20 (15). PROTOMAT employs an amino acid substitution matrix at two distinct phases of block construction (16). The MOTIF program uses a substitution matrix when individual sequences are aligned or realigned against sequence segments containing a candidate motif (16). The MOTOMAT program uses a substitution matrix when a block is extended to either side of the motif region and when scoring candidate blocks (10). A unitary substitution matrix (matches = 1; mismatches = 0) was used initially, generating 2205 blocks. Next, the BLOSUM program was applied to this data base of blocks, clustering at 60%, and the resulting matrix was used with PROTOMAT to construct a second data base consisting of 1961 blocks. The BLOSUM program was then applied to this second data base, clustering at 60%. This matrix was used to construct version 5.0 of the BLOCKS data base from 559 groups in Prosite 9.00 keyed to Swiss-Prot 22. The BLOSUM program was applied to this final data base of 2106 blocks, using a series of clustering percentages to obtain a family of lod substitution matrices. This series of matrices is very similar to the series derived from the second data base. Approximately similar matrices

PAM matrix, PAM 200 (Fig. 3). In this range, each BLOSUM matrix missed 12–25 fewer members than the PAM matrix with similar relative entropy. Therefore, BLOSUM improved detection of members of this family regardless of the searching program used.

To determine whether the superiority of BLOSUM matrices over PAM matrices generalizes to other families, we carried out similar comparative tests for 504 groups of proteins catalogued in Prosite 8.0. For BLAST, BLOSUM 62 performed slightly better overall than BLOSUM 60 or 70, moderately better than BLOSUM 45, and much better than the best PAM matrix in this test, PAM 140 (Fig. 4). Specifically, BLOSUM 62 was better than PAM 140 for 90 groups, whereas it was worse in only 23 other groups. As a baseline for comparison, we used the simple +6/−1 matrix, which makes no distinction among matches or mismatches. Compared to +6/−1, BLOSUM 62 performance was better in 157 groups and was worse



DISCUSSION

We have found that substitution matrices based on amino acid pairs in blocks of aligned protein segments perform better in alignments and homology searches than those based on accepted mutations in closely related groups. Performance was improved overall in every test we have done, including multiple alignment (MULTALIN), detection of ungapped alignments (BLAST), detection of gapped alignments (FASTA and Smith–Waterman), and determination of the significance of an alignment (RDF2). The importance of such improved performance can be profound for weakly scoring alignments that are not detected in a search or are not trusted. For example, the alignment between predicted proteins encoded by mariner and Tc1 transposons improved by more than 4.5 SD above the mean of comparisons to shuffled sequences when BLOSUM 62 was used instead of PAM matrices.

There are fundamental differences between our approach and that of Dayhoff that could account for the superior performance of BLOSUM matrices in searches and alignments. Dayhoff estimated mutation rates from substitutions observed in closely related proteins and extrapolated those rates to model distant relationships. In our case, frequencies were obtained directly from relationships represented in the blocks, regardless of evolutionary distance. Since blocks were derived primarily from the most highly conserved regions of proteins, it is possible that many of the differences between BLOSUM and PAM matrices arise from different

automated PROTOMAT system. While the system itself uses a substitution matrix, iterative application soon leads to nearly the same set of scores, even starting with a unitary matrix or using a representative subset of the groups. Therefore, we do not expect that these substitution matrices will change significantly in the future.

The suggestion to make a substitution matrix from a blocks data base was made by Temple Smith at the 1991 Aspen Center for Physics workshop. We thank Scott Emmons and Jörg Heierhorst for independently pointing out the similarity between mariner and Tc1 predicted proteins, Bill Pearson for advice, and Domokos Vermes for discussions about information theory. This work was supported by a grant from the National Institutes of Health.

1. Dayhoff, M. O. & Eck, R. V., eds. (1968) *Atlas of Protein Sequence and Structure* (Natl. Biomed. Res. Found., Silver Spring, MD), Vol. 3, p. 33.
2. McLachlan, A. D. (1971) *J. Mol. Biol.* **61,** 409–424.
3. Feng, D. F., Johnson, M. S. & Doolittle, R. F. (1985) *J. Mol. Evol.* **21,** 112–125.
4. Rao, J. K. M. (1987) *Int. J. Pept. Protein Res.* **29,** 276–281.
5. Risler, J. L., Delorme, M. O., Delacroix, H. & Henaut, A. (1988) *J. Mol. Biol.* **204,** 1019–1029.
6. Smith, R. F. & Smith, T. F. (1990) *Proc. Natl. Acad. Sci. USA* **87,** 118–122.
7. George, D. G., Barker, W. C. & Hunt, L. T. (1990) *Methods Enzymol.* **183,** 333–351.
8. Dayhoff, M. (1978) *Atlas of Protein Sequence and Structure* (Natl. Biomed. Res. Found., Washington), Vol. 5, Suppl. 3, pp. 345–358.
9. Altschul, S. F. (1991) *J. Mol. Biol.* **219,** 555–565.