CABIOS

Vol 12 no. 2 1996
Pages 135–143

Using substitution probabilities to improve position-specific scoring matrices

Jorja G. Henikoff and Steven Henikoff[1]

Abstract

Each column of amino acids in a multiple alignment of protein sequences can be represented as a vector of 20 amino acid counts. For alignment and searching applications, the count vector is an imperfect representation of a position, because the observed sequences are an incomplete sample of the full set of related sequences. One general solution to this problem is to model unobserved sequences by adding artificial 'pseudo-counts' to the observed counts. We introduce a simple method for computing pseudo-counts that combines the diversity observed in each alignment position with amino acid substitution probabilities. In extensive empirical tests, this position-based method out-performed other pseudo-count methods and was a substantial improvement over the traditional average score method used for constructing profiles.

Introduction

Sequence databanks now contain representatives from a large percentage of all protein families (Green, 1994), so that most newly discovered sequences have homologs that are detectable in database searches (Koonin *et al.*, 1994). Such successes have fueled large-scale sequencing projects, including those involving single-pass sequencing of cDNAs (Adams *et al.*, 1991) and those of model organisms with high gene densities (Oliver *et al.*, 1992). As a result of these activities, alignment methods based on multiple related sequences have become increasingly important. One such method is the multiple alignment equivalent of a standard database search, in which a set of related sequences is used to query a sequence database (Dodd and Egan, 1987; Gribskov *et al.*, 1987; Barton and Sternberg, 1990; Henikoff *et al.*, 1990; Krogh *et al.*, 1994; Tatusov *et al.*, 1994), or else a sequence is used to query a database of multiple alignments (Henikoff and Henikoff, 1991; Sonnhammer and Kahn, 1994; Attwood and Beck, 1994). The most effective strategies convert each alignment into a position-specific scoring matrix (PSSM), weight matrix or profile (Gribskov *et al.*, 1987; Henikoff *et al.*, 1990; Tatusov *et al.*, 1994) in which each position in the alignment is represented in such a way that all of the available information is efficiently used. In a PSSM, an aligned position is represented as a vector of 20 scores derived from counts, one for each amino acid in the alignment column. Because the alignment is a sample drawn from a much larger distribution and most counts are zero, the count vector might not be a complete representation of the position, and several solutions to this problem have been described. The most popular solution is the one embodied in the original profile method of Gribskov et al. (1987), which generalizes pairwise alignment scores by using counts to weight scores from an amino acid substitution matrix.

A different approach adds 'pseudo-counts' to the sample counts in an attempt to model the underlying count vector for a position (Dodd and Egan, 1987; Brown *et al.*, 1993; Lawrence *et al.*, 1993; Claverie, 1994; Tatusov *et al.*, 1994). Evidence that pseudo-counts can be used effectively in PSSMs was provided by Tatusov et al. (1994) who compared the average score method and various methods for modelling pseudo-counts with a maximum pairwise segment score for several blocks of multiply aligned protein sequences. In their tests, all methods out-performed pairwise segment scoring, and a Dirichlet mixture method provided the best discrimination between true positive and true negative sequences. Their 'data-dependent' method (we refer to it here as a 'substitution probability' method), which models pseudo-counts on probabilities underlying an amino acid substitution matrix, performed less well than the Dirichlet mixture method.

Here we introduce a substitution probability method in which the total number of pseudo-counts is position-specific, a feature that is inherent to the Dirichlet mixture method. These position-based pseudo-counts are simple to compute and easy to understand. In extensive empirical tests involving PSSMs from 1673 different alignment blocks, PSSMs incorporating position-based pseudo-counts were the best performers among available pseudo-count methods and strongly out-performed those constructed using the traditional average score method.

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

[1] *To whom correspondence should be addressed*

E-mail: henikoff@howard.fhcrc.org

Theory

Constructing PSSMs from blocks

Blocks are aligned ungapped arrays of amino acid sequence segments that represent the most highly conserved regions of proteins. A PSSM calculated from a block is as wide as the block it derives from and has 20 rows, one for each amino acid. A PSSM is used to score alignments of the block with a protein or translated DNA sequence. The PSSM is slid along the sequence and at each alignment the score for every amino acid in the sequence is looked up in the column of the PSSM with which it is aligned. Then the scores for all the columns are added to arrive at the alignment score. The basic task when computing a PSSM is to estimate the probabilities of each amino acid appearing at each position of the block. The scores in each column of a PSSM can be derived in a number of ways, but are naturally based on the frequency distribution of the amino acids observed in that position of the block, so that an amino acid that occurs more frequently receives a higher score. The solid theoretical basis of log-odds scoring for alignments (Stormo, 1990; Altschul, 1991) motivates constructing PSSMs of log-odds scores. Adding log-odds scores is equivalent to multiplying the corresponding probability ratios.

In this work, sequences in a block are weighted to compensate for redundancy. The sequence-weighted amino acid frequencies at a position are called the 'counts'. Below, we consider different methods for representing counts in columns of a PSSM.

The odds ratio method

Counts can be converted to odds ratios of expected to observed probabilities. Let p_{ca} be the unknown probability that amino acid a appears in column c of the block, and let p_a be the expected frequency of amino acid a in a random sequence, which can be estimated from the overall occurrence in a large database of protein sequences. Let n_{ca} be the count of the number of times amino acid a appears in column c of the block, and let N_c be the total number of counts in column c. Then n_{ca}/N_c is an estimate of p_{ca} and the odds of amino acid a appearing in column c can be estimated as:

$$\frac{n_{ca}}{N_c}/p_a \tag{1}$$

The odds ratio is simple, and it maximizes selectivity for observed residues, but it has serious drawbacks. Because odds ratios make no allowances for conservative replacements, the PSSM may be insensitive to distantly related members of the family. In addition, block columns consist mostly of zero counts, which convert to odds ratios of zero. A zero count might indicate that the amino acid cannot occur in the position represented by the column, or that not enough related sequences are included in the block to observe the amino acid. In either case, the logarithm of zero is negative infinity, preventing it from being used to make a log-odds PSSM. If these odds ratios are used as PSSM scores without taking logs (Henikoff *et al.*, 1990), then adding them to get an alignment score is not mathematically defensible.

Averaging methods

The most popular PSSM derivation is incorporated into the profile method (Gribskov *et al.*, 1987). An average is taken of all pairwise scores obtained from a substitution matrix for an aligned residue and each of the residues seen in the block column. An unobserved amino acid receives a score based on its presumed association with the observed residues. This average score method both allows for substitutions and deals with zero counts. The PSSM score w_{ca} for amino acid a in column c using the average score method is:

$$w_{ca} = \sum_{i=1}^{20} \frac{n_{ci}}{N_c} * s_{ia} \tag{2}$$

where s_{ia} is a score taken from a 20 × 20 amino acid substitution matrix. The basic odds ratio PSSM score in equation (1) can be interpreted as an average score which uses a diagonal substitution matrix where $s_{aa} = 1/p_a$ and $s_{ia} = 0$ otherwise.

Another formulation for averaging can be used to obtain PSSM entries. Altschul (1991) has demonstrated that scores obtained from any substitution matrix have a log-odds interpretation with an implicit set of amino acid pair substitution probabilities, q_{ia}:

$$s_{ia} = (1/\lambda)\ \log\left(\frac{q_{ia}}{p_i * p_a}\right) \tag{3}$$

where λ is a scaling factor. Substituting (3) into (2), we see that the average score is a weighted average of log-odds ratios. We might alternatively consider weighting each odds ratio before taking the log, thus explicitly retaining a log-odds interpretation of a PSSM entry:

$$w_{ca} = \log\left(\sum_{i=1}^{20} \frac{n_{ci}}{N_c} * \frac{q_{ia}}{p_i * p_a}\right) \tag{4}$$

Equation (4) is especially attractive because of its intuitive interpretation in terms of amino acid pair counts, such as those used to construct the BLOSUM series of substitution matrices (Henikoff and Henikoff, 1992). For example, if the counts for a column aligned with a serine consist of two alanines (A) and one serine (S), and in presumably correct pairwise alignments there are as many AS pairs and four times as many SS pairs as expected for chance alignments (respectively, $q_{AS}/(p_A p_S) = 1$ and $q_{SS}/(p_S p_S) = 4$), then the weighted average of odds ratios is 2/3(1) + 1/3(4) = 2 for this position. Because each position is considered to be independent, this value can be multiplied by the values for each of the positions in the alignment, or equivalently, their logarithms can be added.

A potential drawback of averaging methods is that they do not take into account the number of sequences in the block. They should make sensitive PSSMs when there are few sequences and the actual distributions are uncertain based on the observed data. However, when a sequence sample begins to approximate the actual sequence distribution, averaging substitution values should reduce PSSM specificity.

Pseudo-count methods

An alternative way to construct a log-odds PSSM is to add hypothetical sequences to the sample. For each column, this involves adding 'pseudo-counts' to the counts based on some belief about the actual, incompletely observed, distribution of amino acids in that column. Let b_{ca} be the number of pseudo-counts for amino acid a in column c and B_c be the total number of pseudo-counts in the column. Both n_{ca}/N_c and b_{ca}/B_c are estimates of p_{ca}, and a weighted average estimate is:

$$p_{ca} = \frac{N_c}{N_c + B_c} * \frac{n_{ca}}{N_c} + \frac{B_c}{N_c + B_c} * \frac{b_{ca}}{B_c} \qquad (5)$$

The relative sizes of N_c and B_c reflect how strongly each estimate contributes. If N_c is large with respect to B_c, then the observed counts dominate, whereas the pseudo-counts dominate when the opposite is true. Pseudo-counts should be constructed to ensure that equation (5) will converge to n_{ca}/N_c with addition of more and more family members, as pointed out by Claverie (1994).

When pseudo-counts are used, p_{ca} is never zero, so the PSSM score for amino acid a in column c is computed as the logarithm of the odds ratio of p_{ca} to expected probabilities based on the background frequency p_a:

$$w_{ca} = \log\left(\frac{p_{ca}}{p_a}\right) \qquad (6)$$

Knowledge about amino acid distributions external to an observed block can be used as *a priori* information to generate pseudo-counts, and several methods have been proposed. In the 'background' method (Lawrence *et al.*, 1993), the pseudo-count (b_{ca}) is proportional to the overall frequency of the amino acid in a protein sequence database, and the total number of pseudo-counts in a column (B_c) is selected in some way:

$$b_{ca} = B_c * \mathit{Probability}\ (a) = B_c * p_a \qquad (7)$$

This method does not take into account possible constraints imposed by amino acids observed in a column. For example, if a tryptophan is observed, then the pseudo-count for a phenylalanine, which often substitutes for tryptophan, should be higher than its background frequency would imply, and that for a proline, which rarely substitutes for tryptophan, should be less.

Pseudo-counts based on substitution probabilities

Pseudo-counts can be improved by basing them on the frequencies with which different amino acids substitute for one another (Tatusov *et al.*, 1994). Let q_{ia} be the probability that amino acid a is substituted by amino acid i as estimated from sequence alignments (Dayhoff, 1978, Henikoff and Henikoff, 1992); this is the same quantity used in equation (3). Then pseudo-counts can be generated by adding the substitution probabilities:

$$b_{ca} = B_c * \sum_{i=1}^{20} \mathit{Probability}\ (i \text{ and } a) = B_c * \sum_{i=1}^{20} q_{ia} \qquad (8)$$

Here amino acid substitutions are considered, but still crudely since no account is taken of the amino acids actually observed in the column. This problem is remedied by conditioning the probabilities on the counts observed in column c (Tatusov *et al.*, 1994):

$$b_{ca} = B_c * \sum_{i=1}^{20} \mathit{Probability}\ (i|\text{column } c) * \mathit{Probability}\ (a|i)$$

$$= B_c * \sum_{i=1}^{20} \frac{n_{ci}}{N_c} * \frac{q_{ia}}{Q_i}, \quad Q_i = \sum_{a=1}^{20} q_{ia} \qquad (9)$$

Claverie (1994) used a method similar in form to equation (9), but the term q_{ia}/Q_i was replaced by scores s_{ia} from a substitution matrix. In either case, the total number of pseudo-counts (B_c) must still be chosen.

Selecting the number of pseudo-counts

Several authors have chosen B_c to be some function of the number of sequences N in a PSSM, such as $B_c = \sqrt{N}$ (Lawrence *et al.*, 1993; Tatusov *et al.*, 1994; Claverie, 1994). However, this choice is not ideal. Consider PSSM properties at extreme values for the number of counts in a column, N_c. When there is only one sequence, the PSSM should resemble a substitution matrix. In this case, $N_c = 1$, $n_{cA} = 1$ (column c contains the single amino acid A) and $n_{ca} = 0$ for amino acids other than A. Inserting these values in (9) and then substituting b_{ca} in (5), equation (6) becomes:

$$w_{ca} = log\left[\frac{B_c}{(1+B_c)} * \frac{q_{Aa}}{Q_A * p_a}\right], a \neq A \qquad (10)$$

Since the substitution matrix score s_{Aa} is $log\ [q_{Aa}/(Q_A * p_a)]$, the larger B_c is, the closer $B_c/(1+B_c)$ is to 1 and the closer w_{ca} is to s_{Aa}. This argues for large values of B_c when N_c is small. However, if $B_c = \sqrt{}N$, then the number of pseudo-counts can never exceed the number of counts, suggesting that $B_c = \sqrt{}N$ is not a good choice for small numbers of sequences. At the other extreme, when there are many sequences, the PSSM should reflect the observed frequencies. From equation (5), this will be the case whenever B_c grows slower than N_c.

The simplest choice for the total number of pseudo-counts in a column is to let B_c be a constant for all PSSMs. The constant must be large enough to allow pseudo-counts to dominate counts for small numbers of sequences and should be determined empirically. If we allow B_c to get extremely large relative to N_c in equation (5), it reduces to a purely pseudo-count estimate of p_{ca}. Then, if pseudo-counts based on substitution probabilities are used [equation (9)]:

$$\lim_{B_c \to \infty} w_{ca} = \log\left(\frac{b_{ca}}{B_c}/p_a\right) = \log\left(\sum_{i=1}^{20} \frac{n_{ci}}{N_c} * \frac{q_{ia}}{Q_i * p_a}\right) \qquad (11)$$

Here the PSSM score is the logarithm of a weighted average of odds ratios. This is identical to equation (4) using marginal expected probabilities, so that equation (4) can be interpreted as a purely pseudo-count method based on substitution probabilities.

Position-based pseudo-counts

A drawback of substitution probability methods discussed so far is that they use the same number of pseudo-counts (B_c) for all columns in a PSSM. We conjectured that computing B_c independently for each column might improve performance. Although it is possible to compute an optimal theoretical value for B_c in equation (5) based on the observed counts, the result has the unacceptable property for PSSMs of being zero for conserved columns regardless of the number of sequences (Bishop *et al.*, 1975). Instead, we looked for properties on which B_c could be based so that it would behave well for both small and large numbers of counts using position-specific information.

A reasonable basis for computing position-specific values for B_c is to take into account residue diversity. A conserved column requires fewer total pseudo-counts than a diverse column. We can use the number of different amino acids in the column, R_c, as a simple indicator of diversity. This is the same measure of position diversity used successfully to compute position-based sequence weights (Henikoff and Henikoff, 1994a). Accordingly, we set the total number of pseudo-counts in column c equal to:

$$B_c = m * R_c \qquad (12)$$

where m is an empirically determined positive number. Since there is always at least one residue in a block column and the number of different residues in a column can never exceed the smaller of 20 or the number of sequences in the column, it follows that:

$$m \leq m * R_c \leq \min(m * N_c,\ m * 20) \qquad (13)$$

From equation (5), pseudo-counts dominate counts when $N_c < B_c$, which always happens when $N_c \leq m * 20$ for position-based pseudo-counts. When $N_c > m * 20$, the counts always dominate regardless of the value of R_c, so that equation (5) tends to n_{ca}/N_c as the number of sequences gets large, as required. For a conserved column ($R_c = 1$), counts dominate when $N_c > m$.

Pseudo-counts based on Dirichlet mixtures

Another method that uses observed amino acids at a position for generating pseudo-counts is similar to the substitution probability method in general form, but the probabilities are derived in a different way. Rather than using pairwise amino acid substitution data, mixtures of Dirichlet densities are computed from columns of multiple sequence alignments (Brown *et al.*, 1993, K. Sjölander, personal communication). A feature of this method is that probability estimates take into account the number of sequences observed in a position-specific manner, without requiring that the total number of pseudo-counts in a column be set arbitrarily. First a number, D, of Dirichlet probability densities is selected. Each density has 20 parameters which sum to α_i, $i = 1, \ldots, D$. In addition there are ($d - 1$) parameters that specify the weight of each Dirichlet density in the mixture. All of these parameters must be estimated from multiple alignment data. Once the parameters are estimated, the pseudo-counts are computed as:

$$b_{ca} = \sum_{i=1}^{D} \alpha_i * \ Probability\ (i|column\ c) * Probability\ (a|i) \qquad (14)$$

Although the sum is over the mixture components rather than amino acids, the general form is parallel to that for substitution probability methods [equation (9)] in the use of conditional probabilities. The total number of pseudo-counts in a column is:

$$B_c = \sum_{i=1}^{D} \alpha_i * \ Probability\ (i|column\ c) \leq \max_{i=1}^{D}(\alpha_i) \quad (15)$$

The Dirichlet mixture method is attractive because of the mathematically elegant way in which the probability that the observed column is an example of each of the D distributions is estimated. However, the number of distributions must be selected in some way, and there is no general agreement about whether any set of column probability distributions is natural. These distributions might assume regularities in the underlying alignment data that are unrealistic. Substitution probability methods for generating pseudo-counts use pairwise substitution data models (e. g., PAM and Blosum) that are more widely accepted and more easily comprehended than the Dirichlet mixture model.

Methods

PSSMs were made from 1673 blocks in Blocks 5.0 (Henikoff and Henikoff, 1991), based on Prosite v.9 (Bairoch, 1992) and Swiss-Prot v.22 (Bairoch and Boeckmann, 1992), for the 465 Prosite groups that had more sequences in Prosite v.12, coordinated with Swiss-Prot v.29, than in Prosite v.9. The PSSMs were then searched against Swiss-Prot v.29 using the BLIMPS searching program (Henikoff *et al.*, 1995) and the full-length sequences from Prosite v.12 were used as the list of true positive sequences for each group (fragment sequences were ignored). Therefore, only a portion of the true positive sequences were used to make the test blocks. The 465 groups averaged 59% (median 44%) more sequences in Prosite v.12 than in Prosite v.9. The raw counts were always sequence-weighted using position-based sequence weights (Henikoff and Henikoff, 1994a). For pseudo-count methods, the PSSM score for amino acid a in column c was computed as in equation (6). To ensure a range of scores when searching with BLIMPS, the scores were scaled to lie between 0 and 99 for each PSSM. For methods based on substitution probabilities, we utilized those that underlie Blosum substitution matrices (Henikoff and Henikoff, 1992).

The evaluation approach was similar to that of Tatusov *et al.*, (1994), except that many more protein groups were included. Odds ratio, averaging and a variety of pseudo-count methods were tested. For each method, all 1673 PSSMs were computed and searched against Swiss-Prot v.29. BLIMPS output consists of a list of the database sequences ranked by PSSM score. Each list was compared against the list from the corresponding odds ratio PSSM [equation (1)]. The number of searches for which the tested method performed better or worse than the odds ratio method was tabulated. Three measures were utilized. The *percentile measure* (Pearson, 1991; Henikoff and Henikoff, 1994a) simply counts the number of known true positive sequences that score above 99.5% of the other sequences in the database, which are assumed to be true negatives. Swiss-Prot v.29 has 38 303 sequences and the average number of true positives for a block was 25, so on average the percentile measure counts the number of true positives above the first 191 true negatives. The *equivalence number* (Pearson, 1995) is the point at which the true negative rank exceeds the true positive rank, determined by counting the true positives from the bottom of the results list and the true negatives from the top. This point is approximately where the number of false positives equals the number of false negatives, and is zero when all the true positive sequences are ranked above any true negative sequences. The *receiver operating characteristic* (*ROC*) *area* (Metz, 1978) also counts the true negatives from the top of the results list, but computes the area under a curve where the true negative number is plotted on the x-axis and the number of true positives ranking above that true negative is plotted on the y-axis. The axes are normalized so the maximum area under the curve is 1.0.

The different evaluation measures may give different results for a particular block. The percentile measure is especially sensitive to detection of true positives, but not to a few high-scoring false positives. This feature is useful in the present tests, because there are true positive sequences that are not cataloged in Prosite (Bairoch, 1992), and these can be erroneously scored as true negatives. The equivalence number more precisely quantifies the separation of true positive and true negative scores than the percentile measure. Because it finds the single point where the true positive and negative score distributions cross, it is sensitive to incorrectly cataloged sequences. The ROC area shares features of the equivalence number, but it attempts to characterize the entire true positive and negative distributions, not just the single point where they cross. However, interpretation is problematic if the two curves cross when superimposed. Because the area under the ROC curve is non-integral, a tolerance value must be selected to decide when one area is 'greater' than another. Since we saved 400 scores for each search, we chose a tolerance of 0.0025. We allowed for the different strengths and weaknesses of the three evaluation measures by requiring that all three measures show improved overall performance before one method was judged better than another.

All programs were written in standard C on Sun workstations using the SunOS 4.1.3 operating system. BLIMPS is available by anonymous ftp to ncbi.nlm.nih.-gov, cd repository/blocks. Other programs and test results are available by request to the authors.

Results

PSSM evaluations based on sequence database searching

PSSMs were constructed using various methods; these were tested by searching them against a protein sequence database and scoring the ability of the PSSMs to separate known true positive and true negative sequences. We wanted to approximate the realistic situation in which a new sequence is compared to a database for possible classification into a known family of proteins. Since Prosite and Swiss-Prot are maintained in tandem, we accomplished this by searching blocks made from an older version of Swiss-Prot against a newer version of Swiss-Prot that contains more sequences, and using the corresponding newer version of Prosite to provide the lists of new true positive sequences. Sequences present in the block were included in evaluation of results, although qualitatively similar results were obtained when these were excluded (data not shown).

Our results are consistent with those of Tatusov et al. (1994) for the methods they tested (see their Table I). There, PSSMs using pseudo-counts from a 9-component Dirichlet mixture performed best. PSSMs using pseudo-counts from substitution probabilities for BLOSUM 62 and the total number of pseudo-counts in each column equal to the square root of the number of sequences ($B_c = \sqrt{N}$), performed about as well as the average score method with BLOSUM 62. Somewhat poorer performance was obtained with PSSMs using background frequencies, again with $B_c = \sqrt{N}$. In our more comprehensive tests using the same methods, very similar results were obtained using 3 different evaluation criteria (Figure 1). This suggests that the many procedural differences between our study and theirs did not seriously influence overall PSSM performance.

In this work, we introduce two extensions of the substitution probability method for computing pseudo-counts [equation (9)]. One keeps the total number of pseudo-counts in each column constant but large, and the other takes into account the diversity of an aligned position by making the total number of pseudo-counts proportional to the number of different residues represented in the column [equation (12)]. We tested a range of values for the constant method, finding that performance leveled off around $B_c = 50$ (data not shown).

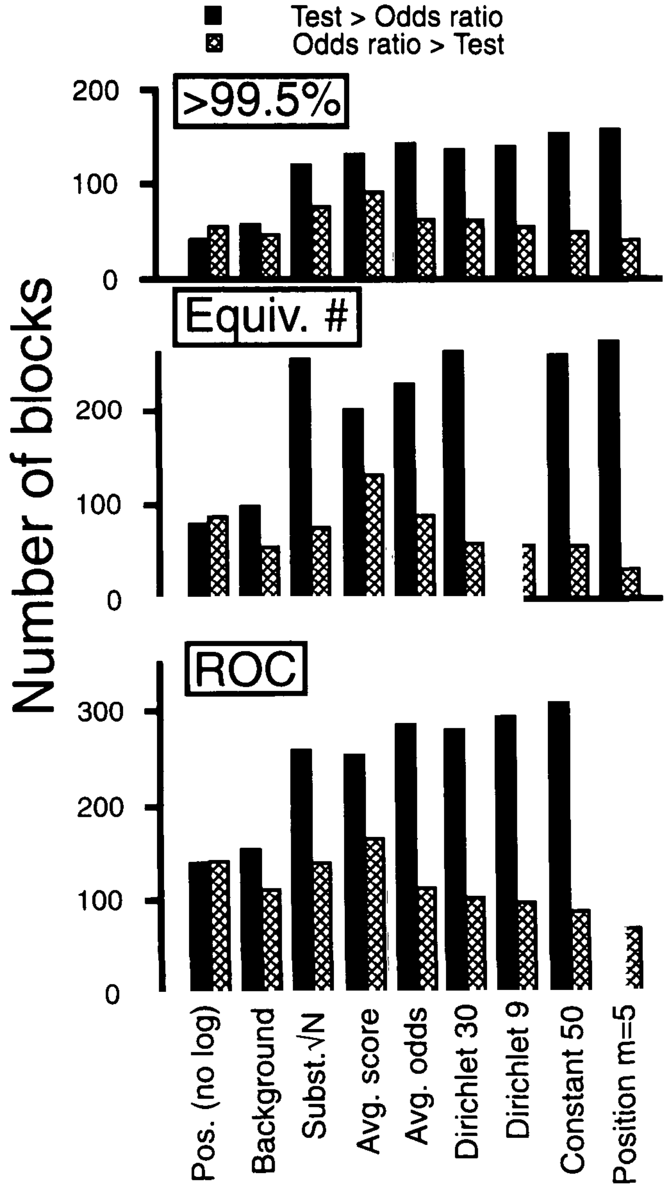

Fig. 1: Tests of different methods for PSSM column representation. A set of 1673 blocks from Blocks v.5.0 (based on Prosite v.9 keyed to Swiss-Prot v.22) was converted to PSSMs using position-based sequence weights and the indicated column representation method. BLOSUM 62 or its underlying substitution probabilities [derived from Blocks v.5.0 (Henikoff and Henikoff, 1991)] were used wherever these values were required. Dirichlet mixtures (Brown *et al.*, 1993) were constructed by K. Sjölander (personal communication) using columns from Blocks v.5.0. Each PSSM was then used to search Swiss-Prot v.29 which corresponds to Prosite v.12, from which lists of true positive sequences were obtained. The methods tested were: Pos. (no log), position-based [equations (9) and (12)] with $m = 5$ but scores are added without taking logarithms; Background [equation (7)] where $B_c = \sqrt{N_c}$; Subst. $\sqrt{N}$, pseudo-counts based on substitution probabilities where $B_c = \sqrt{N_c}$ [equation (9)]; Avg. score, average score method [equation (2)]; Avg. odds, log of average odds ratio [equation (11)]; Dirichlet 30 and Dirichlet 9, pseudo-counts based on Dirichlet mixtures of respectively 30 and 9 components [equation (14)]; Constant 50, $B_c = 50$; Position $m = 5$, position-based where $m = 5$ [equations (9) and (12)]. All test PSSMs were compared against a PSSM made using the odds ratio method [equation (1)]. The solid bars represent the number of test PSSMs for which performance was better than performance of the corresponding odds ratio PSSM. The hatched bars represent the number of odds ratio PSSMs for which performance was better than for the corresponding test PSSM. Three performance measures were used: the number of true positives scoring above the 99.5 percentile level of true negatives, the equivalence number, and the ROC area.

In the position-based method, diverse columns receive more pseudo-counts than conserved columns, reflecting more uncertainty about their composition. To allow the number of pseudo-counts to dominate counts when there are few sequences or when the column is diverse, we multiply the number of different residues in the column R_c by a positive integer m so that $B_c = m * R_c$ in equation (9). To choose the value of m, we tested a range of values until performance leveled off, as expected from equation (11). Figure 2 shows the performance of these position-based pseudo-counts. For all three different evaluation criteria, performance improved with increasing m, leveling off when $m = 5$ to 6. This performance exceeds that of all other methods tested, including the Dirichlet mixture method (Figure 1). Position-based pseudo-counts from both BLOSUM 62 (Figure 2) and BLOSUM 100 (data not shown) performed best when $m = 5$.

Allowing m to become infinitely large corresponds to a PSSM made purely with substitution probability pseudo-counts [equation (11)], which resulted in reduced performance (Figure 2). Nevertheless, Figure 1 reveals that this method, which is essentially the same as the average odds method [equation (4)], out-performed the average score method [equation (2)] implemented in profiles (Gribskov *et al.*, 1987), which it closely resembles.

We also varied the source of substitution probabilities over a wide range, from those for BLOSUM 45, with relative entropy $H = 0.38$ to BLOSUM 100, with $H = 1.5$. Only minor differences were seen (Figure 2). Similarly, using a Dirichlet mixture composed of 30 components rather than 9 had only minor effects on overall performance (Figure 1). Therefore, performance differences appear to depend more on the basic method used for generating pseudo-counts than on the parameters used.

PSSM evaluations based on Blocks Database searching

We also tested the performance of PSSMs when they comprise the scores in a database and protein sequences are used as queries. For this, we repeated tests originally carried out using odds-ratio PSSMs (Henikoff and Henikoff, 1994b) with PSSMs constructed using pseudo-counts computed from 9-component Dirichlet mixtures and from the best position-based method. Databases of PSSMs were made from version 6.0 of the Blocks Database using each method, and 7082 sequences from Swiss-Prot v.24 not represented in the Blocks Database were searched against these databases. All of the high-scoring hits were evaluated. Each sequence was also shuffled by randomly permuting individual residues, and each shuffled sequence was used to query these databases. As before, we searched 7082 non-redundant full-length sequences not present in Prosite against a Blocks Database. For both pseudo-count methods, improved performance was obtained, as reflected in the increases in detection of true positives and decreases in detection of false positives (Table I). For Dirichlet pseudo-counts, the net improvement (true–false positives) was 30 hits and for position-based pseudo-counts, the net increase was 53 hits.

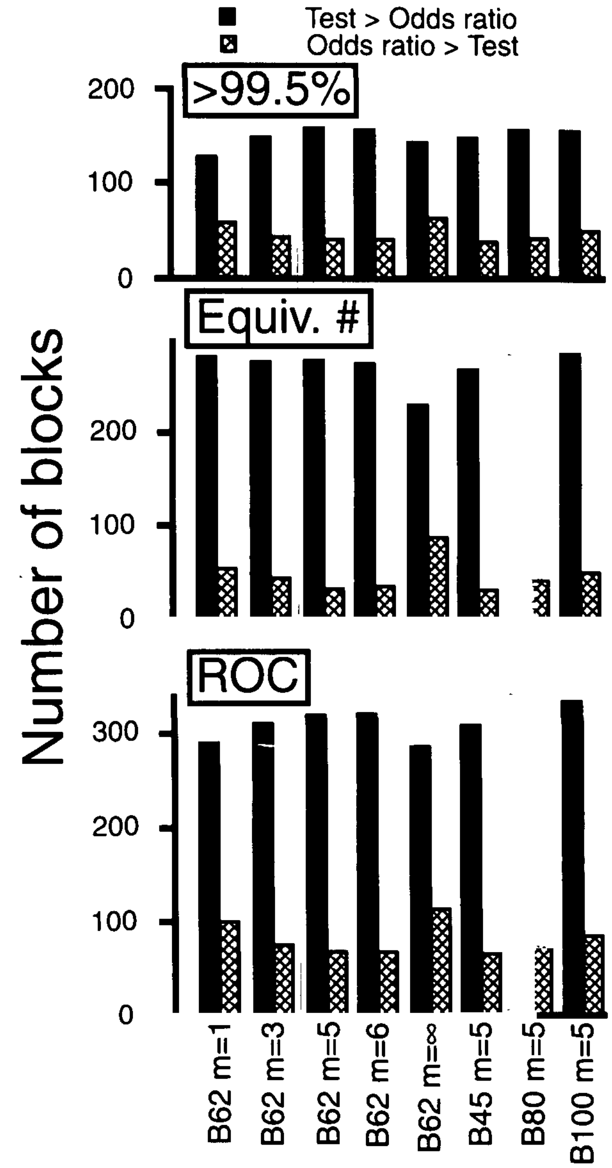

Fig. 2. Effect of changing parameters on performance of PSSMs incorporating position-based pseudo-counts. For substitution probabilities underlying BLOSUM 62 (B62) the value of m was varied, and for $m = 5$, the Blosum parameter was varied from 45 to 100 (Henikoff and Henikoff, 1992). Position-based BLOSUM 62 and m = ∞ is the same as the average odds method for BLOSUM 62 [equation (11)]. See Fig. 1 legend for details.

Discussion

Improvements in PSSM performance were seen when three features were incorporated: consideration of

Table I. Performance differences in searches of the Blocks Database

Found by	TPs* found	FPs found	Total
Position-based, not odds ratio$^+$	51	4	
Odds ratio, not position-based	4	10	
Difference	+47	+6	+53
Dirichlet 9, not odds-ratio	43	16	
Odds ratio, not Dirichlet 9	0	3	
Difference	+43	−13	+30

* High scoring hits were examined manually for occurrences of uncatalogued true positives and new discoveries as previously described (Henikoff and Henikoff, 1994b). If there was uncertainty in classification, the entry was excluded from the analysis.

$^+$ From Blocks v.6.0, in which PSSMs were calculated using odds ratios for counts weighted by the 80% clustering method (Henikoff and Henikoff, 1994a).

substitutions, sensitivity to number of sequences and position specificity (summarized in Table II). The best performing method (position-based pseudo-counts) included all three features, whereas the worst performer (odds ratio) included none. In general, the more of these features, the better the performance. For example, the average score method, which includes one feature, was outperformed by all methods that include two features.

Pseudo-count methods that include at least two features performed especially well. An exception is the substitution probability method in which the total number of pseudo-counts is $\sqrt{N}$, which is always less than the number of counts. The best pseudo-count methods are those that allow pseudo-counts to dominate counts when there are few sequences, or when a variety of amino acids occurs at a position, even one represented by as many as dozens of sequences. Excellent performance was obtained just by using a large constant number of these pseudo-counts, allowing them to sometimes dominate observed counts. PSSM performance using a constant value of 50 pseudo-counts per position based on substitution probabilities was as good as that using Dirichlet mixtures. Taking into account position-specific information further improved performance. Best overall performance was found for the position-based method when the number of pseudo-counts in a column was set equal to about five times the number of different amino acids representing that position. For an invariant position, counts dominate pseudo-counts only when there are more than five sequences, and counts always dominate when there are more than 100 sequences, regardless of the number of different amino acids. This balance between counts and pseudo-counts appears to be the best compromise given our substitution probability model.

Position-based pseudo-counts provide substantially improved performance over the average score method incorporated into profiles (Gribskov *et al.*, 1987), which has been the standard for PSSM construction for the past several years. Based on the equivalence number measure, the average score method was a 50% improvement over the odds ratio control, whereas the position-based pseudo-count method was an eight-fold improvement (Figure 1). Furthermore, position-based pseudo-counts are rather insensitive to the particular choice of the substitution matrix on which the pseudo-counts are modelled (Figure 2), in contrast to the average score method (Lüthy *et al.*, 1994).

Our results also demonstrate the superiority of methods based explicitly on log odds scores, as theory predicts (Stormo, 1990; Altschul, 1991). Non-logarithmic methods in which the odds ratios are summed over all positions performed poorly. This was the case whether or not pseudo-counts were added: the best method for generating pseudo-counts, position-based with $m = 5$, performed similarly to the odds ratio control without logarithms. Moreover, the average score method was out-performed by a simple reformulation that retains an explicit log odds interpretation. Earlier improvements in the profile

Table II. Features of PSSM column representation methods

	Considers substitutions	Sensitive to no. of sequences	Position specific	Relative performance*
Odds ratio	No	No	No	−
Average score	Yes	No	No	+
Average odds	Yes	No	No	++
Pseudo-counts$^+$				
Background $\sqrt{N}$	No	Yes	No	+
Substitution $\sqrt{N}$	Yes	Yes	No	++
Substitution constant	Yes	Yes	No	+ + +
Dirichlet 9	Yes	Yes	Yes	+ + +
Position-based	Yes	Yes	Yes	+ + ++

* Data of Figure 1 were converted to ratios of wins to losses. A ++ method obtained higher ratios for all three evaluation criteria than a + method, and so on.

$^+$ '$\sqrt{N}$' is the total number of pseudo-counts, $B_c = \sqrt{N}$, 'constant' is $B_c = 50$, and Dirichlet 9 uses the 9 component mixture.

method, such as the provision of sequence weights and better amino acid substitution matrices (Thompson *et al.*, 1994; Luthy *et al.*, 1994), retained the original formulation. These improvements apply as well to our reformulation.

We expect that weight matrices incorporating position-based pseudo-counts will also prove useful for representing gapped alignments, such as in profiles, and that their use can be extended to any application in which a multiple alignment column is converted to a count vector (Vingron and Argos, 1989; Smith *et al.*, 1990; Krogh *et al.*, 1994). It is interesting that better performance was obtained for position-based pseudo-counts than for available Dirichlet mixtures (Brown *et al.*, 1993), which represent a more complex mathematical model for determining pseudo-counts. It is possible that the position-based method captures the essence of the Dirichlet model, allowing very robust sets of substitution probabilities to be employed effectively.

Acknowledgements

We thank Phil Green for helpful suggestions, including the use of a large constant number of total pseudo-counts, Gary Stormo, David Haussler and Chip Lawrence for stimulating discussions, and Kimmen Sjölander for providing Dirichlet mixtures. Part of this work was carried out at the 1994 Workshop on Sequence Analysis at Los Alamos National Laboratory. This work was supported by a grant from NIH (GM29009).

References

Adams,M.D., Kelley,J.M., Gocayne,J.D., Dubnick,M., Polymeropoulos,M.H., Xiao,H., Merril,C.R., Wu,A., Olde,B., Moreno,R.F., Kerlavage,A.R., McCombie,W.R. and Venter,J.C. (1991) Complementary DNA sequencing: Expressed sequence tags and human genome project. *Science*, **252**, 1651–1656.

Altschul,S.F. (1991) Amino acid substitution matrices from an information theoretic perspective. *J. Mol. Biol.*, **219**, 555–565.

Attwood,T.K. and Beck,M.E. (1994) PRINTS-a protein motif fingerprint database. *Protein Eng.*, **7**, 841–848.

Bairoch,A. (1992) PROSITE: A dictionary of sites and patterns in proteins. *Nucleic Acids Res.*, **20**, 2013–2018.

Bairoch,A. and Boeckmann,B. (1992) The SWISS-PROT protein sequence data bank. *Nucleic Acids Res.*, **20**, 2019–2022.

Barton,G.J. and Sternberg,M.J. (1990) Flexible protein sequence patterns. A sensitive method to detect weak structural similarities. *J. Mol. Biol.*, **212**, 389–402.

Bishop,Y.M.M., Fienberg,S.E. and Holland,P.W. (1975) *Discrete multivariate analysis: theory and practice.* MIT Press, Cambridge, MA.

Brown,M.P., Hughey,R., Krogh,A., Mian,I.S., Sjölander,K. and Haussler,D. (1993) Using Dirichlet mixture priors to derive hidden Markov models for protein families. In *Proc. First Int. Conf. on Intelligent Systems for Molecular Biology.* Washington D. C., AAAI Press, pp. 47–55.

Claverie,J.-M. (1994) Some useful statistical properties of position-weight matrices. *Comput. Chem.*, **18**, 287–293.

Dayhoff,M. (1978) *Atlas of protein sequence and structure.*, vol. 5, suppl. 3. National Biomedical Research Foundation, Washington, DC, pp. 345–358.

Dodd,I.B. and Egan,J.B. (1987) Systematic method for the detection of potential lambda Cro-like DNA-binding regions in proteins. *J. Mol. Biol.*, **194**, 557–564.

Green,P. (1994) Ancient conserved regions in gene sequences. *Curr. Opin. Struct. Biol.*, **4**, 404–412.

Gribskov,M., McLachlan,A.D. and Eisenberg,D. (1987) Profile analysis: detection of distantly related proteins. *Proc. Natl Acad. Sci. USA*, **84**, 4355–4358.

Henikoff,S. and Henikoff,J.G. (1991) Automated assembly of protein blocks for database searching. *Nucleic Acids Res.*, **19**, 6565–6572.

Henikoff,S. and Henikoff,J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc. Natl Acad. Sci. USA*, **89**, 10915–10919.

Henikoff,S. and Henikoff,J.G. (1994a) Position-based sequence weights. *J. Mol. Biol.*, **243**, 574–578.

Henikoff,S. and Henikoff,J.G. (1994b) Protein family classification based on searching a database of blocks. *Genomics*, **19**, 97–107.

Henikoff,S., Wallace,J.C. and Brown,J.P. (1990) Finding protein similarities with nucleotide sequence databases. *Methods Enzymol.*, **183**, 111–132.

Henikoff,S., Henikoff,J.G., Alford,W.J. and Pietrokovski,S. (1995) Automated construction and graphical presentation of protein blocks from unaligned sequences. *Gene*, **163**, GC17–GC26.

Koonin,E.V., Bork,P. and Sander,C. (1994) Yeast chromosome III: new gene families. *EMBO J.*, **13**, 493–503.

Krogh,A., Brown,M., Mian,I.S., Sjölander,K. and Haussler,D. (1994) Hidden Markov models in computational biology. *J. Mol. Biol.*, **235**, 1501–1531.

Lawrence,C.E., Altschul,S.F., Boguski,M.S., Liu,J.S. Neuwald,A.F., and Wootton,J.C. (1993) Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science*, **262**, 208–214.

Lüthy,R., Xenarios,I. and Bucher,P. (1994) Improving the sensitivity of the sequence profile method. *Prot. Sci.*, **3**, 139–146.

Metz,C.E. (1978) Basic principles of ROC analysis. *Sem. Nuclear Med.*, **8**, 283–298.

Oliver,S.G., van der Aart,Q.J.M., Agostoni-Carbone,M.L., Aigle,M., Alberghina,L., Alexandraki,D., Antoine,G., Anwar,R. and Ballesta,J.P.G. (1992) The complete DNA sequence of yeast chromosome III. *Nature*, **357**, 38–46.

Pearson,W.R. (1991) Searching protein sequence libraries: comparison of the sensitivity and selectivity of the Smith-Waterman and FASTA algorithms. *Genomics*, **11**, 635–650.

Pearson,W.R. (1995) Comparison of methods for searching protein sequence databases. *Prot. Sci.*, **4**, 1145–1160.

Smith,H.O., Annau,T.M. and Chandrasegaran,S. (1990) Finding sequence motifs in groups of functionally related proteins. *Proc. Natl Acad. Sci. USA*, **87**, 826–830.

Sonnhammer,E.L.L. and Kahn,D. (1994) Modular arrangement of proteins as inferred from analysis of homology. *Prot. Sci.*, **3**, 482–492.

Stormo,G.D. (1990) Consensus patterns in DNA. *Methods Enzymol.*, **183**, 211–221.

Tatusov,R.L., Altschul,S.F. and Koonin,E.V. (1994) Detection of conserved segments in proteins: iterative scanning of sequence databases with alignment blocks. *Proc. Natl Acad. Sci. USA*, **91**, 12091–12095.

Thompson,J.D., Higgins,D.G. and Gibson,T.J. (1994) Improved sensitivity of profile searches through the use of sequence weights and gap excision. *Comput. Applic. Biosci.*, **10**, 19–29.

Vingron,M. and Argos,P. (1989) A fast and sensitive multiple sequence alignment algorithm. *Comput. Applic. Biosci.*, **5**, 115–121.

Received on November 11, 1995; revised and accepted on January 4, 1996