Column Scores

Stephen F. Altschul

National Center for Biotechnology Information
National Library of Medicine
National Institutes of Health
The Score for Aligning a Letter to a Column

**Basic question:** What should be the score $s_i$ for aligning a letter $i$ to a particular column $C$ (of aligned letters)?

From many algorithmic perspectives, this question is irrelevant, but it is very important for constructing biologically accurate multiple alignments.

**Preliminary nomenclature and notation**

Having dealt with weights, we are left with the “effective observed counts” of amino acids or nucleotides for $C$, which need not be integral. We will call these simply the *observed counts* $\tilde{c}$, with total number $c$. We will also sometimes have use for the *observed frequencies* $\tilde{f} \equiv \tilde{c}/c$. Note that some, indeed many, of the $c_i$ and $f_i$ may be zero. Null characters may or may not be counted; we will ignore them for now.

Another vector of interest are the *background frequencies* $\tilde{p}$ with which the various letters occur in typical sequences.

Also, we will refer to the scores of a standard substitution matrix, such as BLOSUM-62, as $S_{i,j}$. 
Sum Scores and Average Scores

Perhaps the simplest way to construct a score for aligning a letter to a column $\mathbf{C}$ is as the sum, or average, of scores for individual letters in $\mathbf{C}$. In other words, one may define:

$$s_i \equiv \text{SIM}_{\text{sum}}(i, \mathbf{C}) \equiv \sum_j c_j S_{i,j},$$

or

$$s_i \equiv \text{SIM}_{\text{av}}(i, \mathbf{C}) \equiv \sum_j f_j S_{i,j}.$$

Although these scores are widely used, they have certain clear disadvantages. Even if $\mathbf{C}$ contains many observations, and consists exclusively of a particular letter, that letter is essentially no more favored than it would be if $\mathbf{C}$ consisted of a single sequence.

Log-Odds Scores

A generalization of the log-odds approach to pairwise alignment scores suggests we should define:

\[ s_i \equiv \text{SIM}_{\text{log-odds}}(i, \mathcal{C}) \equiv \log \frac{q_i}{p_i}, \]

where \( q_i \) is the predicted frequency or target frequency for letter \( i \) in column \( \mathcal{C} \).

This leaves open the question of how to infer \( \tilde{q} \) from \( \tilde{c} \). In other words, given a set of observed letters for a column, how do we predict probabilities for new letters?

The simple approach of setting \( \tilde{q} = \tilde{f} \) runs into an immediate problem. The score for any letter \( i \) with \( f_i = 0 \) become \( -\infty \). This indeed makes sense if it is impossible to observe letter \( i \) in the given alignment position, but it is unwise to make such an inference from a small, or indeed a finite set of data.
Pseudocounts

One way around this difficulty is to assume add a vector $\vec{b}$ of $b$ total pseudocounts to the observed counts $\vec{c}$, and infer target frequencies using the formula:

$$\vec{q} = \frac{\vec{c} + \vec{b}}{c + b}$$

The pseudocounts can be chosen to be proportional to $\vec{p}$, so we have only the one free parameter $b$, and our formula becomes:

$$\vec{q} = \frac{\vec{c} + b\vec{p}}{c + b}$$

This approach has two nice properties. First, for $c = 0$ (i.e. no data), it implies log-odds scores $\vec{s}$ that are uniformly 0. (Why?) Second, it implies that that as $c$ grows large, with $b$ fixed, the target frequencies $\vec{q}$ approach the observed frequencies $\vec{f}$.

As we will see, this approach has a rigorous justification in terms of Bayesian statistics.
Problems with Simple Pseudocounts

Simple pseudocounts, defined by $\tilde{b} = bp$, usually work quite well in the DNA context. However, they have major deficiencies when applied to proteins.

Simple pseudocounts carry no information about relationships among the amino acids.

For a column with a single observation (i.e. $c = 1$), simple pseudocounts imply that all mismatch scores are $-\log(1 + \frac{1}{b})$. They clearly underperform the standard substitution scores $S_{i,j}$.

**Question**: How can one construct target frequencies $\tilde{q}$ so that they converge to $\tilde{f}$ when $c$ is large, but so that they reconstitute the standard substitution scores when $c = 1$?
Data-Dependent Pseudocounts

Tatusov et al. proposed letting the pseudocounts depend upon the observed data. For example, if mainly hydrophobic residues are observed in $C$, a preference should be given to hydrophobic pseudocounts. One can engineer this approach to reduce to a standard substitution matrix when $C$ consists of a single amino acid.

Specifically, let $b_i$ equal $bp_i$ times $\sum_j f_j e^{\lambda S_{i,j}}$. Note that if the observed amino acids tend to be similar to amino acid $i$, this factor tends to be greater than 1. The total number of pseudocounts $\sum_i b_i$ remains equal to $b$. (Why?)

Finally, modify the previous approach so that only the pseudocounts are used to infer $\hat{q}$ when $c = 1$:

$$\hat{q} = \frac{(c-1)f + \hat{b}}{c-1+b}.$$  

The column scores then reduce, when $c = 1$, to those implied by the standard matrix. Assuming the observed amino acid is $a$, we have:

$$S_i = \frac{1}{\lambda} \ln \frac{q_i}{p_i} = \frac{1}{\lambda} \ln \frac{bp_i}{bp_i} \sum_j f_j e^{\lambda S_{i,j}} = \frac{1}{\lambda} \ln e^{\lambda S_{i,a}} = S_{i,a}.$$  